SNESL formalization

Dandan Xue

September 10, 2017

0 Level-0

Draft version 0.0.4: added the proof of the main correctness theorem (in process)

0.1 Source language syntax

(Ignore empty sequence for now)
Expressions:

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

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

Values:

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

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

0.2 Type system

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

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

- Expression typing rules:

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

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

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

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

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

- Value typing rules:

  Judgment $\boxed{\vdash v : \tau}$

$$\frac{}{\vdash n : \textbf{int}} \qquad \frac{(\vdash v_i : \tau)_{i=1}^k}{\vdash \{v_1, ..., v_k\} : \{\tau\}}$$

0.3 Source language semantics

$\rho = [x_1 \mapsto v_1, ..., x_i \mapsto v_i]$

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

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

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

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

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

0.4 SVCODE syntax

(1) Stream id:

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

(2) Stream tree:

$$\textbf{STree} \ni st ::= s \mid (st_1, s)$$

(3) SVCODE operations:

$$\psi ::= \texttt{Ctrl} \mid \texttt{Const}_\texttt{a} \mid \texttt{ToFlags} \mid \texttt{Usum} \mid \texttt{MapTwo}_\oplus \mid \texttt{ScanPlus}$$

where $\oplus$ stands for some binary operation on **int**.

(4) SVCODE program:

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

(5) Target language values:

$$\begin{gathered} b \in \{\texttt{T}, \texttt{F}\} \\ a ::= n \mid b \mid () \\ \vec{b} = \langle b_1, ..., b_i \rangle \\ \vec{a} = \langle a_1, ..., a_i \rangle \\ \textbf{SVal} \ni w ::= \vec{a} \mid (w, \vec{b}) \end{gathered}$$

(6) Some notations and operations:

- For some a_0 and $\vec{a} = \langle a_1, ..., a_i \rangle$, let $\langle a_0 | \vec{a} \rangle = \langle a_0, a_1, ..., a_i \rangle$.
- $++ : \textbf{SVal} \to \textbf{SVal} \to \textbf{SVal}$
  $\langle a_1, ..., a_i \rangle ++ \langle a_1', ..., a_i' \rangle = \langle a_1, ..., a_i, a_1', ..., a_i' \rangle$
  $(w_1, \vec{b}_1) ++ (w_2, \vec{b}_2) = (w_1 ++ w_2, \vec{b}_1 ++ \vec{b}_2)$
- $\texttt{sids}$ is a function that converts a $st \in \textbf{STree}$ to a set of $s \in \textbf{SId}$:
  $\texttt{sids}(s) = \{s\}$
  $\texttt{sids}((st, s)) = \texttt{sids}(st) \cup \{s\}$
- For some set of **SId**, t, and some $s \in \textbf{SId}$, let $t \lessdot s$ denote $\forall s' \in t. s' < s$.

0.5 SVCODE semantics

SVCODE runtime environment $\sigma = [s_1 \mapsto \vec{a}_1, ..., s_i \mapsto \vec{a}_i]$.
We define some notations and operations related to σ:

(1) Let $\sigma_1 \overset{<s}{=\!=} \sigma_2$ denote $\forall s' < s.\sigma_1(s') = \sigma_2(s')$.

(2) Judgment $\boxed{\sigma(st) = w}$

$$\frac{}{\sigma(s) = \vec{a}} \qquad \frac{\sigma(st) = w \qquad \overline{\sigma(s) = \vec{a}}}{\sigma((st, s)) = (w, \vec{a})}$$

Definition 0.1. $\sigma_1 \overset{st}{\sim} \sigma_2$ *iff*
(1) $dom(\sigma_1) = dom(\sigma_2)$
(2) $\forall s \in (dom(\sigma_1) - \texttt{sids}(st)).\sigma_1(s) = \sigma_2(s)$

Definition 0.2. *For some* $\sigma_1 \overset{st}{\sim} \sigma_2$, $\sigma_1 \overset{st}{\bowtie} \sigma_2 = \sigma$ *where*

$$\sigma(s) = \begin{cases} \sigma_1(s) +\!\!+ \sigma_2(s), & s \in \texttt{sids}(st) \\ \sigma_1(s), & otherwise \end{cases}$$

SVCODE operational semantics:

- Judgment $\boxed{\langle p, \sigma\rangle \downarrow^{\vec{a}_c} \sigma'}$

  $\vec{a}_c$ is the control stream.

$$\frac{}{\langle \epsilon, \sigma\rangle \downarrow^{\vec{a}_c} \sigma} \qquad \frac{\psi(\vec{a}_1, ..., \vec{a}_k) \downarrow^{\vec{a}_c} \vec{a}}{\langle s := \psi(s_1, ..., s_k), \sigma\rangle \downarrow^{\vec{a}_c} \sigma[s \mapsto \vec{a}]} \; ((\sigma(s_i) = \vec{a}_i)_{i=1}^k)$$

$$\frac{}{\langle st := \texttt{WithCtrl}(s, p), \sigma\rangle \downarrow^{\vec{a}_c} \sigma[s_1 \mapsto \langle\rangle, ..., s_i \mapsto \langle\rangle]} \; (\sigma(s) = \langle\rangle, \texttt{sids}(st) = \{s_1, ..., s_i\})$$

$$\frac{\langle p, \sigma\rangle \downarrow^{\vec{a}_s} \sigma''}{\langle st := \texttt{WithCtrl}(s, p), \sigma\rangle \downarrow^{\vec{a}_c} \sigma[s_1 \mapsto \sigma''(s_1), ..., s_i \mapsto \sigma''(s_i)]} \; \begin{pmatrix} \sigma(s) = \vec{a}_s = \langle a_0 | \vec{a}\rangle \\ \texttt{sids}(st) = \{s_1, ..., s_i\} \end{pmatrix}$$

$$\frac{\langle p_1, \sigma\rangle \downarrow^{\vec{a}_c} \sigma'' \qquad \langle p_2, \sigma''\rangle \downarrow^{\vec{a}_c} \sigma'}{\langle p_1; p_2, \sigma\rangle \downarrow^{\vec{a}_c} \sigma'}$$

- *Transducer* semantics:

  Judgment $\boxed{\psi(\vec{a}_1, ..., \vec{a}_k) \downarrow^{\vec{a}_c} \vec{a}}$

$$\frac{\psi(\vec{a}_{11}, ..., \vec{a}_{k1}) \Downarrow \vec{a}_1 \qquad \psi(\vec{a}_{12}, ..., \vec{a}_{k2}) \downarrow^{\vec{a}_c} \vec{a}_2}{\psi(\vec{a}_{11} +\!\!+ \vec{a}_{12}, ..., \vec{a}_{k1} +\!\!+ \vec{a}_{k2}) \downarrow^{\langle a_0 | \vec{a}_c\rangle} \vec{a}} \; (\vec{a} = \vec{a}_1 +\!\!+ \vec{a}_2)$$

$$\frac{}{\psi(\vec{a}_1, ..., \vec{a}_k) \downarrow^{\langle\rangle} \langle\rangle}$$

- Transducer *block* semantics:

  Judgment $\boxed{\psi(\vec{a}_1, ..., \vec{a}_k) \Downarrow \vec{a}}$

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

$$\frac{\psi(\langle \texttt{F} \rangle, ..., \vec{a}_{k1}) \downdownarrows \vec{a}_1 \qquad \psi(\vec{a}_{12}, ..., \vec{a}_{k2}) \Downarrow \vec{a}_2}{\psi(\langle \texttt{F} \rangle +\!\!+ \vec{a}_{12}, ..., \vec{a}_{k1} +\!\!+ \vec{a}_{k2}) \Downarrow \vec{a}} \; (\vec{a} = \vec{a}_1 +\!\!+ \vec{a}_2)$$

$$\frac{\psi(\langle \texttt{T} \rangle, ..., \vec{a}_k) \downdownarrows \vec{a}}{\psi(\langle \texttt{T} \rangle, ..., \vec{a}_k) \Downarrow \vec{a}}$$

- Transducer *unary* semantics:

Judgment $\boxed{\psi(\langle b \rangle, ..., \vec{a}_k) \downdownarrows \vec{a}}$

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

- Semantics of transducer block with *accumulator*:

Judgment $\boxed{\psi_n(\vec{a}_1, ..., \vec{a}_k) \Downarrow \vec{a}}$

$$\frac{\psi_{n_0}(\langle \texttt{F} \rangle, ..., \vec{a}_{k1}) \downdownarrows^{n_0'} \langle n_1 \rangle \qquad \psi_{n_0'}(\vec{a}_{12}, ..., \vec{a}_{k2}) \Downarrow \vec{a}_2}{\psi_{n_0}(\langle \texttt{F} \rangle +\!\!+ \vec{a}_{12}, ..., \vec{a}_{k1} +\!\!+ \vec{a}_{k2}) \Downarrow \langle n_1 \rangle +\!\!+ \vec{a}_2}$$

$$\frac{\psi_{n_0}(\langle \texttt{T} \rangle, ..., \vec{a}_k) \downdownarrows \langle n_1 \rangle}{\psi_{n_0}(\langle \texttt{T} \rangle, ..., \vec{a}_k) \Downarrow \langle n_1 \rangle}$$

- Semantics of transducer unary with *accumulator*:

Judgment $\boxed{\psi_n(\langle \texttt{F} \rangle, ..., \vec{a}_k) \downdownarrows^{n'} \vec{a}}$

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

Judgment $\boxed{\psi_n(\langle \texttt{T} \rangle, ..., \vec{a}_k) \downdownarrows \vec{a}}$

$$\frac{}{\texttt{ScanPlus}_{n_0}(\langle \texttt{T} \rangle, \langle \rangle) \Downarrow \langle n_0 \rangle}$$

Theorem 0.1 (deterministic ??)**.** *If $\langle p, \sigma \rangle \downarrow^{\vec{a}_c} \sigma'$ and $\langle p, \sigma \rangle \downarrow^{\vec{a}_c} \sigma''$, then $\sigma' = \sigma''$.*

Lemma 0.1 (??)**.** *If $\sigma_1 \overset{st}{\sim} \sigma_2$, (!!should have: import(p) = st) $\langle p, \sigma_1 \rangle \downarrow^{\vec{a}_1} \sigma_1'$, $\langle p, \sigma_2 \rangle \downarrow^{\vec{a}_2} \sigma_2'$, then $\langle p, \sigma_1 \bowtie \sigma_2 \rangle \downarrow^{\vec{a}_1 +\!\!+ \vec{a}_2} \sigma_1' \bowtie \sigma_2'$*

Definition 0.3. *$\vec{a}$ is a prefix of $\vec{a}'$ if $\vec{a} \sqsubseteq \vec{a}'$.*

Judgment $\boxed{\vec{a} \sqsubseteq \vec{a}'}$

$$\frac{}{\langle \rangle \sqsubseteq \vec{a}} \qquad \frac{\vec{a} \sqsubseteq \vec{a}'}{\langle a_0 | \vec{a} \rangle \sqsubseteq \langle a_0 | \vec{a}' \rangle}$$

Lemma 0.2. *If*

(i) $(\vec{a}_i' \sqsubseteq \vec{a}_i)_{i=1}^k$ *and* $\psi(\vec{a}_1', ..., \vec{a}_k') \Downarrow \vec{a}'$,

(ii) $(\vec{a}_i'' \sqsubseteq \vec{a}_i)_{i=1}^k$ *and* $\psi(\vec{a}_1'', ..., \vec{a}_k'') \Downarrow \vec{a}''$

then

(i) $(\vec{a}_i' = \vec{a}_i'')_{i=1}^k$

(ii) $\vec{a}' = \vec{a}''$.

0.6 Translation

$\delta = [x_1 \mapsto st_1, ..., x_i \mapsto st_i]$

- Judgment $\boxed{\delta \vdash e \overset{s_0}{\underset{s_1}{\Longrightarrow}} (p, st)}$

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

$$\frac{\vdash \phi(st_1, ..., st_k) \overset{s_0}{\underset{s_1}{\Longrightarrow}} (p, st)}{\delta \vdash \phi(x_1, ..., x_k) \overset{s_0}{\underset{s_1}{\Longrightarrow}} (p, st)} ((\delta(x_i) = st_i)_{i=1}^k)$$

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

- Auxiliary Judgment $\boxed{\vdash \phi(st_1, ..., st_k) \overset{s_0}{\underset{s_1}{\Longrightarrow}} (p, st)}$

$$\frac{}{\textbf{const}_a() \overset{s_0+1}{\underset{s_0}{\Longrightarrow}} (s_0 := \mathtt{Const_a}, s_0)}$$

$$\frac{}{\textbf{iota}(s) \overset{s_4}{\underset{s_0}{\Longrightarrow}} (p, (s_3, s_0))} \left(\begin{array}{l} s_{i+1} = s_i + 1 \\ p = s_0 := \mathtt{ToFlags}(s); \\ \quad s_1 := \mathtt{Usum}(s_0); \\ \quad s_2 := \mathtt{WithCtrl}(s_1, s_2 := \mathtt{Const_1}); \\ \quad s_3 := \mathtt{ScanPlus}(s_0, s_2) \end{array} \right)$$

$$\frac{}{\textbf{plus}(s_1, s_2) \overset{s_0+1}{\underset{s_0}{\Longrightarrow}} (s_0 := \mathtt{MapTwo}_+(s_1, s_2), s_0)}$$

0.7 Value representation

- Judgment $\boxed{v \rhd_\tau w}$

$$\frac{}{n \rhd_{\textbf{int}} \langle n \rangle} \qquad \frac{(v_i \rhd_\tau w_i)_{i=1}^k}{\{v_1, ..., v_k\} \rhd_{\{\tau\}} (w, \langle \mathtt{F}_1, ..., \mathtt{F}_k, \mathtt{T} \rangle)} (w = w_1 +\!\!+ w_2 +\!\!+ ... +\!\!+ w_k)$$

Lemma 0.3. *If $v \rhd_\tau w$, $v' \rhd_\tau w$, then $v = v'$.*

0.8 Correctness proof

Lemma 0.4. *If*

(i) $\vdash \phi : (\tau_1, ..., \tau_k) \to \tau$

(ii) $\vdash \phi(v_1, ..., v_k) \downarrow v$

(iii) $\vdash \phi(st_1, ..., st_k) \overset{s_0}{\underset{s_1}{\Longrightarrow}} (p, st)$

(iv) $(v_i \rhd_{\tau_i} st_i)_{i=1}^k$

(v) $\bigcup_{i=1}^k \mathtt{sids}(st_i) \lessdot s_0$

then

(i) $\langle p, \sigma \rangle \downarrow^{\vec{a}_c} \sigma'$ *(by $\mathcal{P}$)*

(ii) $v \triangleright_\tau \sigma'(st)$ *(by $\mathcal{V}$)*

(iii) $\sigma' \xlongequal{\lessdot s_0} \sigma$

(iv) $\mathtt{sids}(st) \lessdot s_1$

(v) $s_0 \leq s_1$

Theorem 0.2. ***If***

(i) $\Gamma \vdash e : \tau$ *(by some derivation $\mathcal{T}$)*

(ii) $\rho \vdash e \downarrow v$ *(by $\mathcal{E}$)*

(iii) $\delta \vdash e \overset{s_0}{\underset{s_1}{\Rightarrow}} (p, st)$ *(by $\mathcal{C}$)*

(iv) $\forall x \in dom(\Gamma).\ \vdash \rho(x) : \Gamma(x) \wedge \mathtt{sids}(\delta(x)) \lessdot s_0 \wedge \rho(x) \triangleright_{\Gamma(x)} \sigma(\delta(x))$
then

(v) $\langle p, \sigma \rangle \downarrow^{\langle () \rangle} \sigma'$ *(by $\mathcal{P}$)*

(vi) $v \triangleright_\tau \sigma'(st)$ *(by $\mathcal{V}$)*

(vii) $\sigma' \xlongequal{\lessdot s_0} \sigma$

(viii) $\mathtt{sids}(st) \lessdot s_1$

(ix) $s_0 \leq s_1$

Proof. By induction on the syntax of e.

- Case $e = \{e_1 : x \textbf{ in } y \textbf{ using } \cdot\}$.

  We first must have:

  (i) $$\mathcal{T} = \cfrac{\overset{\mathcal{T}_1}{[x \mapsto \tau_1] \vdash e_1 : \tau_2}}{\Gamma \vdash \{e_1 : x \textbf{ in } y \textbf{ using } \cdot\} : \{\tau_2\}} \ (\Gamma(y) = \{\tau_1\})$$

  (ii) $$\mathcal{E} = \cfrac{\overset{\mathcal{E}_i}{([x \mapsto v_i] \vdash e_1 \downarrow v_i')_{i=1}^k}}{\rho \vdash \{e_1 : x \textbf{ in } y \textbf{ using } \cdot\} \downarrow \{v_1', ..., v_k'\}} \ (\rho(y) = \{v_1, ..., v_k\})$$

  (iii) $$\mathcal{C} = \cfrac{\overset{\mathcal{C}_1}{[x \mapsto st_1] \vdash e_1 \overset{s_0+1}{\underset{s_1}{\Longrightarrow}} (p_1, st_2)}}{\delta \vdash \{e_1 : x \textbf{ in } y \textbf{ using } \cdot\} \overset{s_0}{\underset{s_1}{\Rightarrow}} (s_0 := \mathtt{Usum}(s_2); st_2 := \mathtt{WithCtrl}(s_0, p_1), (st_2, s_2))} \ (\delta(y) = (st_1, s_2))$$

  So $p = (s_0 := \mathtt{Usum}(s_2); st_2 := \mathtt{WithCtrl}(s_0, p_1)), \tau = \{\tau_2\}, v = \{v_1', ..., v_k'\}, st = (st_2, s_2)$.

  (iv) $\vdash \rho(y) : \Gamma(y)$ gives us $\vdash \{v_1, ..., v_k\} : \{\tau_1\}$, which must have the derivation:

  $$\frac{(\vdash v_i : \tau_1)_{i=1}^k}{\vdash \{v_1, ..., v_k\} : \{\tau_1\}} \tag{1}$$

  $\mathtt{sids}(\delta(y)) \lessdot s_0$ gives us

  $$\mathtt{sids}(\delta(y)) = \mathtt{sids}((st_1, s_2)) = \mathtt{sids}(st_1) \cup \{s_2\} \lessdot s_0 \tag{2}$$

$\rho(y) \triangleright_{\Gamma(y)} \sigma(\delta(y)) = \{v_1, ..., v_k\} \triangleright_{\{\tau_1\}} \sigma((st_1, s_2))$ must have the derivation:

$$\frac{\begin{array}{c}\mathcal{V}_i \\ (v_i \triangleright_{\tau_1} w_i)_{i=1}^k\end{array}}{\{v_1, ..., v_k\} \triangleright_{\{\tau_1\}} \sigma((st_1, s_2))} \tag{3}$$

where

$$\sigma(st_1) = w = w_1 + \!\!+\, w_2 + \!\!+ ... + \!\!+\, w_k \tag{4}$$

and

$$\sigma(s_2) = \langle \mathtt{F}_1, ..., \mathtt{F}_k, \mathtt{T} \rangle \tag{5}$$

Now we shall show:

(v) $\langle s_0 := \mathtt{Usum}(s_2); st_2 := \mathtt{WithCtrl}(s_0, p_1), \sigma \rangle \downarrow^{\langle () \rangle} \sigma'$

?? proof of MP0

Assume we have $\mathcal{P}_0$ of $\langle s_0 := \mathtt{Usum}(s_2), \sigma \rangle \downarrow^{\langle () \rangle} \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle]$

Then $\mathcal{P}$ must have the shape:

$$\frac{\begin{array}{c}\mathcal{P}_0 \\ \langle s_0 := \mathtt{Usum}(s_2), \sigma \rangle \downarrow^{\langle () \rangle} \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle]\end{array} \qquad \begin{array}{c}\mathcal{P}_1 \\ \langle st_2 := \mathtt{WithCtrl}(s_0, p_1), \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle] \rangle \downarrow^{\langle () \rangle} \sigma'\end{array}}{\langle s_0 := \mathtt{Usum}(s_2); st_2 := \mathtt{WithCtrl}(s_0, p_1), \sigma \rangle \downarrow^{\langle () \rangle} \sigma'}$$

There are two possibilities for $\mathcal{P}_1$:

(a) Subcase $k = 0$, that is $\sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle](s_0) = \langle \rangle$.
So

$$\mathcal{P}_1 = \frac{}{\langle st_2 := \mathtt{WithCtrl}(s_0, p_1), \sigma \rangle \downarrow^{\langle () \rangle} \sigma[s_0 \mapsto \langle \rangle, s'_1 \mapsto \langle \rangle, ..., s'_i \mapsto \langle \rangle]} \; (\mathtt{sids}(st_2) = \{s'_1, ..., s'_i\}) \quad ,$$

thus $\sigma' = \sigma[s_0 \mapsto \langle \rangle, s'_1 \mapsto \langle \rangle, ..., s'_i \mapsto \langle \rangle]$.

(b) Subcase $k > 0$, that is $\sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle] = \langle () | \vec{a} \rangle$.
Then

$$\mathcal{P}_1 = \frac{\langle p_1, \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle] \rangle \downarrow^{\langle ()_1, ..., ()_k \rangle} \sigma''}{\langle st_2 := \mathtt{WithCtrl}(s_0, p_1), \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle] \rangle \downarrow^{\langle () \rangle} \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle]}$$

in which $\sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle](st_2) = \sigma''(st_2)$.

(vi) $\{v'_1, ..., v'_k\} \triangleright_{\{\tau_2\}} \sigma'((st_2, s_2))$ by $\mathcal{V}$.
We still have two subcases based on the two in the proof of (v) respectively.

- Subcase continuing (va)
  Since $k = 0$, then $v = \{\}$, $\sigma(s_2) = \langle \mathtt{T} \rangle$ (from (2)), and
  $\sigma'(s_2) = \sigma[s_0 \mapsto \langle \rangle, s'_1 \mapsto \langle \rangle, ..., s'_i \mapsto \langle \rangle](s_2) = \sigma(s_2) = \langle \mathtt{T} \rangle$,
  $\sigma'(st_2) = \sigma[s_0 \mapsto \langle \rangle, s'_1 \mapsto \langle \rangle, ..., s'_i \mapsto \langle \rangle](st_2) = (...((\langle \rangle, \langle \rangle)_1, \langle \rangle)_2, ...)_{i-1}$.

  Therefore $\sigma'((st_2, s_2)) = (\sigma'(st_2), \sigma'(s_2))$ and we construct

  $$\mathcal{V} = \frac{}{\{\} \triangleright_{\{\tau_2\}} ((...(\langle \rangle, \langle \rangle)_1, ...)_{i-1}, \langle \mathtt{T} \rangle)}$$

  as required.
- Subcase continuing (vb)
  Since we have

  $$\mathcal{T}_1 = \; [x \mapsto \tau_1] \; \vdash \; e_1 : \tau_2$$

  $$\mathcal{E}_i = \; [x \mapsto v_i] \; \vdash \; e_1 \downarrow v'_i,$$

  $$\mathcal{P}_1 = \; [x \mapsto st_1] \; \vdash \; e_1 \xRightarrow[s_1]{s_0+1} (p_1, st_2)$$

Let $\Gamma_1 = [x \mapsto \tau_1], \rho_i = [x \mapsto v_i]$ and $\delta_1 = [x \mapsto st_1]$.
From (1) and (2) it is clear that

$$\forall z \in dom(\Gamma_1).\ \vdash\ \rho_i(z) : \Gamma_1(z) \wedge \mathtt{sids}(\delta_1(z)) \lessdot s_0.$$

We take $\sigma_i \overset{st_1}{\sim} \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle]$ such that $\sigma_i(st_1) = w_i$.
From $\mathcal{V}_i$ in (3) we know that

$$\forall z \in dom(\Gamma_1).\rho_i(z) \rhd_{\Gamma_1(z)} \sigma_i(\delta_1(z)).$$

Then let i range from 1 to k: by IH on $\mathcal{T}_1$ with $\mathcal{E}_i$, $\mathcal{P}_1$ we obtain the following result:

$$(\langle p_1, \sigma_i \rangle \downarrow^{\langle () \rangle} \sigma_i')_{i=1}^k \tag{6}$$

$$(v_i' \rhd_{\tau_2} \sigma_i'(st_2))_{i=1}^k \tag{7}$$

$$(\sigma_i' \xlongequal{< s_0+1} \sigma_i)_{i=1}^k \tag{8}$$

$$\mathtt{sids}(st_2) \lessdot s_1 \tag{9}$$

$$s_0 + 1 \leq s_1 \tag{10}$$

Using Lemma 0.1 (k-1) times on (6) gives us

$$\langle p_1, \sigma_1 \overset{st_1}{\bowtie} ... \overset{st_1}{\bowtie} \sigma_k \rangle \downarrow^{\langle ()_1, ..., ()_k \rangle} \sigma_1' \overset{st_2}{\bowtie} ... \overset{st_2}{\bowtie} \sigma_k'$$

By Lemma **??**

$$\sigma_1 \overset{st_1}{\bowtie} ... \overset{st_1}{\bowtie} \sigma_k = \sigma[s_0 \mapsto \langle ()_1, ..., ()_k \rangle]$$

and

$$\sigma_1' \overset{st_2}{\bowtie} ... \overset{st_2}{\bowtie} \sigma_k' = \sigma''$$

in which $\sigma''(st_2) = \sigma'(st_2) +\!\!+ ... +\!\!+ \sigma'(st_2)$.
Let $\sigma_i'(st_2) = w_i'$ and $\sigma''(st_2) = w'$, then $w' = w_1' +\!\!+ ... +\!\!+ w_k'$.

Since $\sigma'(st_2) = \sigma''(st_2) = w$, and $\sigma'(s_2) = \sigma(s_2) = \langle \mathtt{F}_1, ..., \mathtt{F}_k, \mathtt{T} \rangle$, therefore $\sigma'((st_2, s_2)) = (\sigma'(st_2), \sigma'(s_2)) = (w, \langle \mathtt{F}_1, ..., \mathtt{F}_k, \mathtt{T} \rangle)$, and now we can construct

$$\mathcal{V} = \frac{(v_i' \rhd_{\tau_2} w_i')_{i=1}^k}{\{v_1', ..., v_k'\} \rhd_{\{\tau_2\}} (w', \langle \mathtt{F}_1, ..., \mathtt{F}_k, \mathtt{T} \rangle)}$$

as required.

(vii) $\sigma' \xlongequal{< s_0} \sigma$

(viii) $\mathtt{sids}((st_2, s_2)) \lessdot s_1$

(ix) $s_0 \leq s_1$

- Case $e = x$.
- Case $e = \textbf{let } x = e_1 \textbf{ in } e_2$
- Case $e = \phi(x_1, ..., x_k)$

$\square$