1 Second simulation (completeness)

1.1 Syntax

1.1.1 Index, indices and tables

%datatype *index*
%name *index* n α

$$n \quad ::= \quad 0 \quad | \quad n+1$$

%datatype *vector*
%name *vector* $\mathcal{I}$

$$\mathcal{I} \quad ::= \quad [] \quad | \quad n :: \mathcal{I}$$

%datatype *table*
%name *table* $\mathcal{I}_\mu$

$$\mathcal{I}_\mu \quad ::= \quad [] \quad | \quad \mathcal{I} :: \mathcal{I}_\mu$$

1.1.2 Term

%datatype *term*
%name *term* t

$$t \quad ::= \quad n \quad | \quad t_1\, t_2 \quad | \quad \lambda t \quad | \quad \textbf{get-context}\, t \quad | \quad \textbf{set-context}\, \alpha\, t$$

1.2 Subtraction

%judgment $n_1 \dot{-} n_2 = n_3$

$n_1 \dot{-} 0 = n_1$ [minus₁]

$(n_1 + 1) \dot{-} (n_2 + 1) = n_3$ [minus₂] *when* $n_1 \dot{-} n_2 = n_3$

%mode $+n_1 \dot{-} +n_2 = -n_3$
%worlds $()$ $n_1 \dot{-} n_2 = n_3$
%terminates (n_1) $n_1 \dot{-} n_2 = n_3$
%unique $+n_1 \dot{-} +n_2 = -1n_3$

%lemma $\forall n \cdot n \dot{-} n = 0$ [minus·equals]

Proof.

$$\frac{}{0 : index \cdot \quad 0 \dot{-} 0 = 0^{\,[\mathrm{minus}_1]} \qquad [\text{minus·equals}]}\;[\&1]$$

$$\frac{n\colon index \;\cdot\; \begin{array}{c}\mathcal{D}\\ n \mathbin{\dot{-}} n = 0\end{array}\ ^{[\text{minus}\cdot\text{equals}]}}{n+1\colon index \;\cdot\; \begin{array}{c}\mathcal{D}\\ \dfrac{n \mathbin{\dot{-}} n = 0}{(n+1) \mathbin{\dot{-}} (n+1) = 0}\,[\text{minus}_2]\end{array}\ ^{[\text{minus}\cdot\text{equals}]}}\,[\&2]$$

%mode $+n \;\cdot\; -\mathcal{D}$ $^{[\text{minus}\cdot\text{equals}]}$
%worlds $()$ $n \;\cdot\; \mathcal{D}$ $^{[\text{minus}\cdot\text{equals}]}$
%total n $n \;\cdot\; \mathcal{D}$ $^{[\text{minus}\cdot\text{equals}]}$

1.3 Equality

%judgment $n_1 = n_2$

$$n = n\ ^{[\text{refl}]}$$

%mode $+n = +m$
%worlds $()$ $n = m$

%lemma $n_1 \mathbin{\dot{-}} n_2 = n_3 \quad \wedge \quad n_1 \mathbin{\dot{-}} n_2 = n_3' \quad \Rightarrow \quad n_3 = n_3'$ $^{[\text{minus}\cdot\text{unique}]}$

Proof.

$$\frac{}{\begin{array}{c}\vdots\\ n_1 \mathbin{\dot{-}} n_2 = n_3\end{array} \quad \wedge \quad \begin{array}{c}\vdots\\ n_1 \mathbin{\dot{-}} n_2 = n_3\end{array} \quad \Rightarrow \quad n_3 = n_3\ ^{[\text{refl}]}\ ^{[\text{minus}\cdot\text{unique}]}}\,[\&1]$$

%mode $+\mathcal{D}_1 \quad \wedge \quad +\mathcal{D}_2 \quad \Rightarrow \quad -\mathcal{D}_3$ $^{[\text{minus}\cdot\text{unique}]}$
%worlds $()$ $\mathcal{D}_1 \quad \wedge \quad \mathcal{D}_2 \quad \Rightarrow \quad \mathcal{D}_3$ $^{[\text{minus}\cdot\text{unique}]}$
%terminates $\{\}$ $\mathcal{D}_1 \quad \wedge \quad \mathcal{D}_2 \quad \Rightarrow \quad \mathcal{D}_3$ $^{[\text{minus}\cdot\text{unique}]}$
%total $\{\}$ $\mathcal{D}_1 \quad \wedge \quad \mathcal{D}_2 \quad \Rightarrow \quad \mathcal{D}_3$ $^{[\text{minus}\cdot\text{unique}]}$

1.3.1 Fetch (indices)

%judgment $\mathcal{I}(n_1) = n_2$

$$(n{::}\mathcal{I})(0) = n\ ^{[\text{fetch}_1^{\mathcal{I}}]}$$
$$(n{::}\mathcal{I})(n_1+1) = n_2\ ^{[\text{fetch}_2^{\mathcal{I}}]} \quad when \quad \mathcal{I}(n_1) = n_2$$

%mode $+\mathcal{I}(+n_1) = -n_2$
%worlds $()$ $\mathcal{I}(n_1) = n_2$
%terminates n_1 $\mathcal{I}(n_1) = n_2$
%unique $+\mathcal{I}(+n_1) = -1 n_2$

1.3.2 Fetch (table)

%judgment $\mathcal{I}_\mu(n) = \mathcal{I}$

$$(\mathcal{I} :: \mathcal{I}_\mu)(0) = \mathcal{I}\ ^{\left[\text{fetch}_1^{\mathcal{I}\mu}\right]}$$
$$(\mathcal{I}' :: \mathcal{I}_\mu)(\alpha+1) = \mathcal{I}\ ^{\left[\text{fetch}_2^{\mathcal{I}\mu}\right]} \quad when \quad \mathcal{I}_\mu(\alpha) = \mathcal{I}$$

%mode $+\mathcal{I}_\mu(+\alpha) = -\mathcal{I}$
%worlds $()$ $\mathcal{I}_\mu(\alpha) = \mathcal{I}$
%terminates α $\mathcal{I}_\mu(\alpha) = \mathcal{I}$
%unique $+\mathcal{I}_\mu(+\alpha) = -1\mathcal{I}$

1.3.3 Compute

%judgment $n_1 \dot{-} \mathcal{I}(n_2) = n_3$

$$n \dot{-} \mathcal{I}(l) = g^{\,[\text{compute}_1]} \quad \textit{when} \quad \mathcal{I}(l) = k, \quad n \dot{-} k = g$$

%mode $+n_1 \dot{-} +\mathcal{I}(+n_2) = -n_3$
%worlds $()$ $n_1 \dot{-} \mathcal{I}(n_2) = n_3$
%terminates $\{\}$ $n_1 \dot{-} \mathcal{I}(n_2) = n_3$
%unique $+n_1 \dot{-} +\mathcal{I}(+n_2) = -1n_3$

1.3.4 Closure, environment and stack

%datatype *clos* **%name** *clos* c
%datatype *l-env* **%name** *l-env* $\mathcal{L}$
%datatype *l-table* **%name** *l-table* $\mathcal{L}_\mu$
%datatype *k-env* **%name** *k-env* $\mathcal{E}_\mu$
%datatype *stack* **%name** *stack* $\mathcal{S}$

$$c \quad ::= \quad (t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu)$$

$$\begin{array}{lll} \mathcal{L} & ::= & () \\ & | & (c; \mathcal{L}) \end{array}$$

$$\begin{array}{lll} \mathcal{L}_\mu & ::= & () \\ & | & \mathcal{L} \colon \mathcal{L}_\mu \end{array}$$

$$\begin{array}{lll} \mathcal{E}_\mu & ::= & () \\ & | & (\mathcal{S}; \mathcal{E}_\mu) \end{array}$$

$$\begin{array}{lll} \mathcal{S} & ::= & [] \\ & | & c :: \mathcal{S} \end{array}$$

%datatype *state*
%name *state* σ

$$\sigma \quad ::= \quad \langle t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S} \rangle$$

1.4 Judgments

1.4.1 Fetch a local closure

%judgment $\mathcal{L}(n) = c$

$$(c; \mathcal{L})(0) = c^{\,[\text{fetch}_1]}$$
$$(c'; \mathcal{L})(n+1) = c^{\,[\text{fetch}_2]} \quad \textit{when} \quad \mathcal{L}(n) = c$$

%mode $+\mathcal{L}(+n) = -c$
%worlds $()$ $\mathcal{L}(n) = c$
%terminates $\mathcal{L}$ $\mathcal{L}(n) = c$

%unique $+\mathcal{L}(+n) = -1c$

1.4.2 Fetch a local environment

%judgment $\mathcal{L}_\mu(n) = \mathcal{L}$

$$(\mathcal{L}{:}\, \mathcal{L}_\mu)(0) = \mathcal{L}\ ^{[\text{l}\cdot\text{fetch}_1]}$$
$$(\mathcal{L}'{:}\, \mathcal{L}_\mu)(n+1) = \mathcal{L}\ ^{[\text{l}\cdot\text{fetch}_2]} \quad \textit{when} \quad \mathcal{L}_\mu(n) = \mathcal{L}$$

%mode $+\mathcal{L}_\mu(+n) = -\mathcal{L}$
%worlds $()$ $\mathcal{L}_\mu(n) = \mathcal{L}$
%terminates $\mathcal{L}_\mu$ $\mathcal{L}_\mu(n) = \mathcal{L}$
%unique $+\mathcal{L}_\mu(+n) = -1\mathcal{L}$

1.4.3 Fetch a stack

%judgment $\mathcal{E}_\mu(n) = \mathcal{S}$

$$(\mathcal{S}; \mathcal{E}_\mu)(0) = \mathcal{S}\ ^{[\text{fetch}_1^\mu]}$$
$$(\mathcal{S}'; \mathcal{E}_\mu)(n+1) = \mathcal{S}\ ^{[\text{fetch}_2^\mu]} \quad \textit{when} \quad \mathcal{E}_\mu(n) = \mathcal{S}$$

%mode $+\mathcal{E}_\mu(+n) = -\mathcal{S}$
%worlds $()$ $\mathcal{E}_\mu(n) = \mathcal{S}$
%terminates $\mathcal{E}_\mu$ $\mathcal{E}_\mu(n) = \mathcal{S}$
%unique $+\mathcal{E}_\mu(+n) = -1\mathcal{S}$

1.4.4 Evaluation rules

%judgment $\sigma_1 \rightsquigarrow \sigma_2$

$$\langle k, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle \rightsquigarrow \langle t, \mathcal{L}', \mathcal{L}'_\mu, \mathcal{E}'_\mu, \mathcal{S}\rangle\ ^{[\text{k}\cdot\text{var}]} \quad \textit{when} \quad \mathcal{L}(k) = (t, \mathcal{L}', \mathcal{L}'_\mu, \mathcal{E}'_\mu)$$
$$\langle (t\, u), \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle \rightsquigarrow \langle t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, (u, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu) :: \mathcal{S}\rangle\ ^{[\text{k}\cdot\text{app}]}$$
$$\langle \lambda t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, c :: \mathcal{S}\rangle \rightsquigarrow \langle t, (c; \mathcal{L}), \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle\ ^{[\text{k}\cdot\text{abs}]}$$
$$\langle \textbf{get-context}\, t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle \rightsquigarrow \langle t, \mathcal{L}, (\mathcal{L}{:}\, \mathcal{L}_\mu), (\mathcal{S}; \mathcal{E}_\mu), \mathcal{S}\rangle\ ^{[\text{k}\cdot\text{catch}]}$$
$$\langle \textbf{set-context}\, \alpha\, t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle \rightsquigarrow \langle t, \mathcal{L}', \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}'\rangle\ ^{[\text{k}\cdot\text{throw}]} \quad \textit{when} \quad \mathcal{L}_\mu(\alpha) = \mathcal{L}', \quad \mathcal{E}_\mu(\alpha) = \mathcal{S}'$$

%mode $+\sigma_1 \rightsquigarrow -\sigma_2$
%worlds $()$ $\sigma_1 \rightsquigarrow \sigma_2$
%unique $+\sigma_1 \rightsquigarrow -1\sigma_2$

2 Abstract machine for safe $\boldsymbol{\lambda}_{\mathbf{ct}}$-terms

2.0.5 Syntax

%datatype *clos*
%datatype *c-env*
%datatype *k-env*
%datatype *stack*
%name *clos* $\tilde{c}$
%name *c-env* $\tilde{\mathcal{E}}$
%name *k-env* $\tilde{\mathcal{E}}_\mu$
%name *stack* $\tilde{\mathcal{S}}$

$$\tilde{c} \quad ::= \quad \left(t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu\right)$$

$$\begin{array}{rcl}
\tilde{\mathcal{E}} & ::= & () \\
 & \mid & (\tilde{c};\tilde{\mathcal{E}})
\end{array}$$

$$\begin{array}{rcl}
\tilde{\mathcal{E}}_\mu & ::= & () \\
 & \mid & (\tilde{\mathcal{S}};\tilde{\mathcal{E}}_\mu)
\end{array}$$

$$\begin{array}{rcl}
\tilde{\mathcal{S}} & ::= & [] \\
 & \mid & \tilde{c} :: \tilde{\mathcal{S}}
\end{array}$$

%datatype *state*
%name *state* $\tilde{\sigma}$

$$\tilde{\sigma} \quad ::= \quad \langle t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}} \rangle$$

2.0.6 Fetch a closure

%judgment $\tilde{\mathcal{E}}(n) = \tilde{c}$

$$(\tilde{c};\tilde{\mathcal{E}})(0) = \tilde{c}\ ^{[\mathrm{i{\cdot}fetch_1}]}$$
$$(\tilde{c}';\tilde{\mathcal{E}})(n+1) = \tilde{c}\ ^{[\mathrm{i{\cdot}fetch_2}]} \quad \textit{when} \quad \tilde{\mathcal{E}}(n) = \tilde{c}$$

%mode $+\tilde{\mathcal{E}}(+n) = -\tilde{c}$
%worlds $()$ $\tilde{\mathcal{E}}(n) = \tilde{c}$
%terminates $\tilde{\mathcal{E}}$ $\tilde{\mathcal{E}}(n) = \tilde{c}$
%unique $+\tilde{\mathcal{E}}(+n) = -1\tilde{c}$

2.0.7 Compute

%judgment $\tilde{\mathcal{E}}(n_1 \dot{-} n_2) = \tilde{c}$

$$\tilde{\mathcal{E}}(n \dot{-} k) = \tilde{c}\ ^{[\mathrm{i{\cdot}compute_1}]} \quad \textit{when} \quad n \dot{-} k = g, \quad \tilde{\mathcal{E}}(g) = \tilde{c}$$

%mode $+\tilde{\mathcal{E}}(+n \dot{-} +k) = -\tilde{c}$
%worlds $()$ $\tilde{\mathcal{E}}(n \dot{-} k) = \tilde{c}$
%terminates $\{\}$ $\tilde{\mathcal{E}}(n \dot{-} k) = \tilde{c}$
%unique $+\tilde{\mathcal{E}}(+n \dot{-} +k) = -1\tilde{c}$

2.0.8 Fetch a stack

%judgment $\tilde{\mathcal{E}}_\mu(n) = \tilde{\mathcal{S}}$

$$(\tilde{\mathcal{S}};\tilde{\mathcal{E}}_\mu)(0) = \tilde{\mathcal{S}}\ ^{[\mathrm{i{\cdot}fetch_1^\mu}]}$$
$$(\tilde{\mathcal{S}}';\tilde{\mathcal{E}}_\mu)(n+1) = \tilde{\mathcal{S}}\ ^{[\mathrm{i{\cdot}fetch_2^\mu}]} \quad \textit{when} \quad \tilde{\mathcal{E}}_\mu(n) = \tilde{\mathcal{S}}$$

%mode $+\tilde{\mathcal{E}}_\mu(+n) = -\tilde{\mathcal{S}}$
%worlds $()$ $\tilde{\mathcal{E}}_\mu(n) = \tilde{\mathcal{S}}$
%terminates $\tilde{\mathcal{E}}_\mu$ $\tilde{\mathcal{E}}_\mu(n) = \tilde{\mathcal{S}}$
%unique $+\tilde{\mathcal{E}}_\mu(+n) = -1\tilde{\mathcal{S}}$

2.1 Evaluation rules

%judgment $\tilde{\sigma}_1 \rightsquigarrow \tilde{\sigma}_2$

$$\langle l, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle \rightsquigarrow \langle t, n', \mathcal{I}', \mathcal{I}'_\mu, \tilde{\mathcal{E}}', \tilde{\mathcal{E}}'_\mu, \tilde{\mathcal{S}}\rangle^{[\text{i·var}]}$$
$$\textit{when} \quad n \mathbin{\dot-} \mathcal{I}(l) = g, \quad \tilde{\mathcal{E}}(g) = \big(t, n', \mathcal{I}', \mathcal{I}'_\mu, \tilde{\mathcal{E}}', \tilde{\mathcal{E}}'_\mu\big)$$

$$\langle (t\, u), n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle \rightsquigarrow \langle t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \big(u, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu\big) :: \tilde{\mathcal{S}}\rangle^{[\text{i·app}]}$$

$$\langle \lambda t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{c} :: \tilde{\mathcal{S}}\rangle \rightsquigarrow \langle t, n+1, (n+1 :: \mathcal{I}), \mathcal{I}_\mu, (\tilde{c}; \tilde{\mathcal{E}}), \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle^{[\text{i·abs}]}$$

$$\langle \textbf{get-context}\, t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle \rightsquigarrow \langle t, n, \mathcal{I}, (\mathcal{I} :: \mathcal{I}_\mu), \tilde{\mathcal{E}}, (\tilde{\mathcal{S}}; \tilde{\mathcal{E}}_\mu), \tilde{\mathcal{S}}\rangle^{[\text{i·catch}]}$$

$$\langle \textbf{set-context}\, \alpha\, t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle \rightsquigarrow \langle t, n, \mathcal{I}', \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}'\rangle^{[\text{i·throw}]}$$
$$\textit{when} \quad \mathcal{I}_\mu(\alpha) = \mathcal{I}', \quad \tilde{\mathcal{E}}_\mu(\alpha) = \tilde{\mathcal{S}}'$$

%mode $+\sigma_1 \rightsquigarrow -\sigma_2$
%worlds $()$ $\sigma_1 \rightsquigarrow \sigma_2$
%unique $+\sigma_1 \rightsquigarrow -1\sigma_2$

3 Translation

%judgment $\tilde{c}^\diamond = c$
%judgment $\tilde{\mathcal{S}}^\diamond = \mathcal{S}$
%judgment $\tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu$
%judgment $\textit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}$
%judgment $\textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ \mathcal{I}_\mu = \mathcal{L}_\mu$

$$\big(t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu\big)^\diamond = (t, \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu)^{[\text{clos}^\diamond]} \quad \textit{when} \quad \textit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}, \quad \textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ \mathcal{I}_\mu = \mathcal{L}_\mu, \quad \tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu$$

$$[]^\diamond = []^{[\text{stack}_1^\diamond]}$$
$$\big(\tilde{c} :: \tilde{\mathcal{S}}\big)^\diamond = c :: \mathcal{S}^{[\text{stack}_2^\diamond]} \quad \textit{when} \quad \tilde{c}^\diamond = c, \quad \tilde{\mathcal{S}}^\diamond = \mathcal{S}$$

$$()^\diamond = ()^{[\text{k·env}_1^\diamond]}$$
$$(\tilde{\mathcal{S}}; \tilde{\mathcal{E}}_\mu)^\diamond = (\mathcal{S}; \mathcal{E}_\mu)^{[\text{k·env}_2^\diamond]} \quad \textit{when} \quad \tilde{\mathcal{S}}^\diamond = \mathcal{S}, \quad \tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu$$

$$\textit{flatten}\ n\ \tilde{\mathcal{E}}\ [] = ()^{[\text{flatten}_1]}$$
$$\textit{flatten}\ n\ \tilde{\mathcal{E}}\ (k :: \mathcal{I}) = (c; \mathcal{L})^{[\text{flatten}_2]} \quad \textit{when} \quad \tilde{\mathcal{E}}(n \mathbin{\dot-} k) = \tilde{c}, \quad \tilde{c}^\diamond = c, \quad \textit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}$$

$$\textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ [] = ()^{[\text{map}_1]}$$
$$\textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ (\mathcal{I} :: \mathcal{I}_\mu) = \mathcal{L} \colon \mathcal{L}_\mu{}^{[\text{map}_2]} \quad \textit{when} \quad \textit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}, \quad \textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ \mathcal{I}_\mu = \mathcal{L}_\mu$$

%mode
$+\tilde{c}^\diamond = -c$
$+\tilde{\mathcal{S}}^\diamond = -\mathcal{S}$
$+\tilde{\mathcal{E}}_\mu{}^\diamond = -\mathcal{E}_\mu$
$\textit{flatten}\ +n\ +\tilde{\mathcal{E}}\ +\mathcal{I} = -\mathcal{L}$
$\textit{map}\ \big(\textit{flatten}\ +n\ +\tilde{\mathcal{E}}\big)\ +\mathcal{I}_\mu = -\mathcal{L}_\mu$

%worlds $()$
$\tilde{c}^\diamond = c$
$\tilde{\mathcal{S}}^\diamond = \mathcal{S}$
$\tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu$
$\textit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}$
$\textit{map}\ \big(\textit{flatten}\ n\ \tilde{\mathcal{E}}\big)\ \mathcal{I}_\mu = \mathcal{L}_\mu$

Remark. To do.

%terminates $\left(\tilde{c}\ \tilde{\mathcal{S}}\ \tilde{\mathcal{E}}_\mu\ \mathcal{I}\ \mathcal{I}_\mu\right)$

$\tilde{c}^\diamond = c$

$\tilde{\mathcal{S}}^\diamond = \mathcal{S}$

$\tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu$

$\mathit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L}$

$\mathit{map}\ \left(\mathit{flatten}\ n\ \tilde{\mathcal{E}}\right)\ \mathcal{I}_\mu = \mathcal{L}_\mu$

%unique

$+\tilde{c}^\diamond = -1c$

$+\tilde{\mathcal{S}}^\diamond = -1\mathcal{S}$

$+\tilde{\mathcal{E}}_\mu{}^\diamond = -1\mathcal{E}_\mu$

$\mathit{flatten}\ \ +n\ \ +\tilde{\mathcal{E}}\ \ +\mathcal{I} = -1\mathcal{L}$

$\mathit{map}\ \left(\mathit{flatten}\ \ +n\ \ +\tilde{\mathcal{E}}\right)\ \ +\mathcal{I}_\mu = -1\mathcal{L}_\mu$

%judgment $\tilde{\sigma}^\diamond = \sigma$

$\langle t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu, \tilde{\mathcal{S}}\rangle^\diamond = \langle t', \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu, \mathcal{S}\rangle$ [state$^\diamond$] $\quad \mathit{when} \quad \left(t, n, \mathcal{I}, \mathcal{I}_\mu, \tilde{\mathcal{E}}, \tilde{\mathcal{E}}_\mu\right)^\diamond = (t', \mathcal{L}, \mathcal{L}_\mu, \mathcal{E}_\mu), \quad \tilde{\mathcal{S}}^\diamond = \mathcal{S}$

%mode $+\tilde{\sigma}^\diamond = -\sigma$
%worlds $()\quad \tilde{\sigma}^\diamond = \sigma$
%unique $+\tilde{\sigma}^\diamond = -1\sigma$

4 Completeness

%lemma $n \dot{-} k = g \quad \Rightarrow \quad (n+1) \dot{-} k = g+1$ [minus·succ]

%lemma $\forall \tilde{c}' \cdot \mathit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L} \quad \Rightarrow \quad \mathit{flatten}\ (n+1)\ (\tilde{c}'; \tilde{\mathcal{E}})\ \mathcal{I} = \mathcal{L}$ [weaken·flatten]

%lemma $\forall \tilde{c}' \cdot \mathit{map}\ \left(\mathit{flatten}\ n\ \tilde{\mathcal{E}}\right)\ \mathcal{I}_\mu = \mathcal{L}_\mu \quad \Rightarrow \quad \mathit{map}\ \left(\mathit{flatten}\ (n+1)\ (\tilde{c}';\ \tilde{\mathcal{E}})\right)\ \mathcal{I}_\mu = \mathcal{L}_\mu$ [weaken·map]

%lemma $\mathit{map}\ \left(\mathit{flatten}\ n\ \tilde{\mathcal{E}}\right)\ \mathcal{I}_\mu = \mathcal{L}_\mu \quad \wedge \quad \mathcal{L}_\mu(\alpha) = \mathcal{L}' \quad \Rightarrow \quad \mathit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I}' = \mathcal{L}' \quad \wedge \quad \mathcal{I}_\mu(\alpha) = \mathcal{I}' \quad \mathit{for\ some}\ \mathcal{I}'$ [map·complete]

%lemma $\mathit{flatten}\ n\ \tilde{\mathcal{E}}\ \mathcal{I} = \mathcal{L} \quad \wedge \quad \mathcal{L}(l) = c \quad \Rightarrow \quad \tilde{\mathcal{E}}(n \dot{-} k) = \tilde{c} \quad \wedge \quad \tilde{c}^\diamond = c \quad \wedge \quad \mathcal{I}(l) = k \quad \mathit{for\ some}\ k, \tilde{c}$ [fetch·complete]

%lemma $\tilde{\mathcal{E}}_\mu{}^\diamond = \mathcal{E}_\mu \quad \wedge \quad \mathcal{E}_\mu(\alpha) = \mathcal{S} \quad \Rightarrow \quad \tilde{\mathcal{S}}^\diamond = \mathcal{S} \quad \wedge \quad \tilde{\mathcal{E}}_\mu(\alpha) = \tilde{\mathcal{S}} \quad \mathit{for\ some}\ \tilde{\mathcal{S}}$ [fetch$^\mu$·complete]

%theorem $\sigma_1 \rightsquigarrow \sigma_2 \quad \wedge \quad \tilde{\sigma}_1{}^\diamond = \sigma_1 \quad \Rightarrow \quad \tilde{\sigma}_1 \rightsquigarrow \tilde{\sigma}_2 \quad \wedge \quad \tilde{\sigma}_2{}^\diamond = \sigma_2 \quad \mathit{for\ some}\ \tilde{\sigma}_2$ [completeness]