FunTAL: Reasonably Mixing a Functional Language with Assembly
(Technical Appendix)

Daniel Patterson Jamie Perconti Christos Dimoulas Amal Ahmed

April 16, 2017

Contents

1 Typed assembly language: T
  1.1 Syntax and Semantics ......................................................... 2
  1.2 Contexts and Contextual Equivalence ..................................... 9
  1.3 Logical Relation ............................................................. 11
  1.4 Basic Properties ............................................................. 16
  1.5 Compatibility Lemmas ....................................................... 25

2 Functional language: F ......................................................... 83
  2.1 Syntax and Semantics ....................................................... 83

3 Multi-Language: F+T ............................................................ 84
  3.1 Syntax and Semantics ....................................................... 84
  3.2 General Contexts and Contextual Equivalence ......................... 88
  3.3 Logical Relation ............................................................. 92
  3.4 Basic Properties ........................................................... 100
  3.5 Bridge Lemmas ............................................................... 110
  3.6 Compatibility Lemmas ..................................................... 123
  3.7 Fundamental Property and Soundness ................................... 135
  3.8 Completeness ............................................................... 137
  3.9 Examples ................................................................. 139
1 Typed assembly language: T

NOTE: Throughout this technical appendix, we write \texttt{ret end}\{\tau;\sigma\}\{r_r\} instead of the \texttt{halt} \tau, \sigma \{r_r\} that appears in the accompanying paper. We apologize for any confusion caused by the slight difference in presentation.

1.1 Syntax and Semantics

\begin{verbatim}
\tau ::= \alpha | \text{unit} | \text{int} | \exists \alpha.\tau | \mu\alpha.\tau | \text{ref} (\tau, \ldots, \tau) | \text{box} \psi \\
\psi ::= \forall[\Delta].\{\chi;\sigma\}^q | (\tau, \ldots, \tau) \\
q ::= r | i | \epsilon | \text{end}\{\tau;\sigma\} \\
\omega ::= \tau | \sigma | q \\
e ::= (I, H) \\
v ::= (\text{ret end}\{\tau;\sigma\}\{r\}, \cdot) \\
E ::= (E_1, \cdot) \\
E_1 ::= \cdot \\
r ::= r_1 | r_2 | \ldots | r_7 | ra \\
h ::= \text{code}[\Delta]\{\chi;\sigma\}^q.I | (w, \ldots, w) \\
w ::= () | n | \ell | \text{pack} (\tau, w) \text{ as } \exists\alpha.\tau | \text{fold}_{\mu\alpha.\tau} w | w[\omega] \\
u ::= w | r | \text{pack} (\tau, u) \text{ as } \exists\alpha.\tau | \text{fold}_{\mu\alpha.\tau} u | u[\omega] \\
I ::= \vdash; I | \text{jmp} u | \text{call} u \{\sigma, q\} | \text{ret}\{r\} | \text{ret}\{i\} | \text{ret end}\{\tau;\sigma\}\{r\} \\
i ::= \text{aop} r_d, r_s, u | \text{bnz} r, u | \text{ld} r_d, r_s[i] | \text{str} r_d[i], r_s | \text{ralloc} r_d, n | \text{balloc} r_d, n | \text{mv} r_d, u \\
| \text{unpack} (\alpha, r_d) u | \text{unfold} r_d, u | \text{salloc} n | \text{sfree} n | \text{sld} r_d, i | \text{sst} i, r_s \\
\text{aop} ::= \text{add} | \text{sub} | \text{mult} \\
H ::= \cdot | H, \ell \mapsto h \\
R ::= \cdot | R, r \mapsto w \\
S ::= \text{nil} | w :: S \\
M ::= (H, R, S; \sigma) \\
\Psi ::= \cdot | \Psi, \ell; "\psi \\
\nu ::= \text{ref} | \text{box} \\
\Delta ::= \cdot | \Delta, \alpha | \Delta, \zeta | \Delta, \epsilon \\
\chi ::= \cdot | \chi, r: \tau \\
\sigma ::= \zeta | \bullet | \tau :: \sigma \\
\end{verbatim}

Note that we define \texttt{E[e]}—plugging an evaluation context \texttt{E} with a component \texttt{e}—as follows:

\[(E_1, \cdot)[(I, H)] \overset{\text{def}}{=} (E_1[I], H)\]
1.1.1 Return Marker Metafunctions

The metafunctions ret-type and ret-addr-type follow the return marker in order to look up the type returned by a component, or the type of the return address, respectively.

In the case where the return marker is \( r \) or \( i \), ret-type is somewhat redundant since the result of ret-type is contained in the result of ret-addr-type. But when the return marker is \( \text{end}\{\tau;\sigma\} \), there is no actual return address, so only ret-type is defined. Both metafunctions are used throughout the type system.

\[
\begin{align*}
\text{ret-type}(r, \chi, \sigma) &= \tau;\sigma' & \text{if } \chi(r) = \text{box } \forall \tau . \{r': \tau;\sigma'\}^q \\
\text{ret-type}(i, \chi, \sigma) &= \tau;\sigma' & \text{if } \sigma(i) = \text{box } \forall \tau . \{r': \tau;\sigma'\}^q \\
\text{ret-type}(\text{end}\{\tau;\sigma\}, \chi, \sigma) &= \tau;\sigma' \\
\text{ret-addr-type}(r, \chi, \sigma) &= \forall \tau . \{r': \tau;\sigma'\}^q & \text{if } \chi(r) = \text{box } \forall \tau . \{r': \tau;\sigma'\}^q \\
\text{ret-addr-type}(i, \chi, \sigma) &= \forall \tau . \{r': \tau;\sigma'\}^q & \text{if } \sigma(i) = \text{box } \forall \tau . \{r': \tau;\sigma'\}^q
\end{align*}
\]

1.1.2 Well-Formed Type \( \Delta \vdash \tau \)

\[
\begin{align*}
\alpha &\in \Delta \\
\Delta &\vdash \alpha \\
\Delta &\vdash \text{unit} \\
\Delta &\vdash \text{int} \\
\Delta &\vdash \exists \alpha . \tau \\
\Delta &\vdash \mu \alpha . \tau \\
\Delta &\vdash \text{ref } \langle \tau_0, \ldots, \tau_n \rangle \\
\Delta &\vdash \text{box } \psi
\end{align*}
\]

1.1.3 Well-Formed Heap Value Type \( \Delta \vdash \psi \)

The rule for the code block type uses a judgment for well-formed return markers (presented next). Note that the external type environment \( \Delta \) and the formal type arguments \( \Delta' \) are kept separate in that premise.

\[
\begin{align*}
\Delta, \Delta' &\vdash \chi \\
\Delta, \Delta' &\vdash \sigma \\
\Delta[\Delta'] &\vdash \chi ; \sigma \vdash q \\
\Delta &\vdash \forall \langle \Delta' \rangle . \{\chi ; \sigma\}^q \\
\Delta &\vdash \langle \tau_0, \ldots, \tau_n \rangle \\
\end{align*}
\]

1.1.4 Well-Formed Return Marker \( \Delta[\Delta'] ; \chi ; \sigma \vdash q \)

A code block is not allowed to abstract over its own return marker. Accordingly, this judgment accepts a type variable environment in two pieces, \( \Delta \) and \( \Delta' \). A return marker variable \( \epsilon \) is only well-formed if \( \epsilon \) appears in the outer \( \Delta \). Other return markers are well-formed if ret-type produces a type and stack type that are well formed under the combined type variable environment \( \Delta, \Delta' \).

When this judgment is called by the well-formed heap type judgment (see Section 1.1.3 above), the outer \( \Delta \) is the set of type variables in scope outside the code type, while the inner \( \Delta' \) contains the formal type parameters. The latter may not bind an \( \epsilon \) in this position.

\[
\begin{align*}
\text{ret-type}(q, \chi, \sigma) &= \tau;\sigma' & \Delta, \Delta' &\vdash \tau \\
\Delta[\Delta'] &\vdash \chi ; \sigma \vdash q & \Delta, \Delta' &\vdash \sigma' \\
\epsilon &\in \Delta
\end{align*}
\]

1.1.5 Well-Formed Register File Type \( \Delta \vdash \chi \)

\[
\begin{align*}
\Delta &\vdash \cdot \\
\Delta &\vdash \chi \\
\Delta &\vdash \tau
\end{align*}
\]

1.1.6 Well-Formed Stack Type \( \Delta \vdash \sigma \)

\[
\begin{align*}
\zeta &\in \Delta \\
\Delta &\vdash \zeta \\
\Delta &\vdash \text{nil} \\
\Delta &\vdash \tau \\
\Delta &\vdash \tau :: \sigma
\end{align*}
\]
1.1.7 Register File Subtyping \[ \Delta \vdash \chi_1 \leq \chi_2 \]

\[
\Delta, \chi \vdash \chi, \chi' \\
\Delta \vdash (\chi, \chi') \leq \chi
\]

1.1.8 Well-Typed Heap Fragment \[ \Psi \vdash H : \Psi' \]

\[
\operatorname{dom}(\Psi) \cap \operatorname{dom}(\Psi') = \emptyset \quad \Psi' = \ell_1 : \nu_1 \psi_1, \ldots, \ell_n : \nu_n \psi_n \\
\Psi \vdash \psi_1 \cdots \vdash \psi_n \\
\Psi, \Psi' \vdash h_1 : \nu_1 \psi_1 \cdots \Psi, \Psi' \vdash h_n : \nu_n \psi_n
\]

\[ \Psi \vdash \{ \ell_1 \mapsto h_1, \ldots, \ell_n \mapsto h_n \} : \Psi' \]

1.1.9 Well-Typed Register File \[ \Psi \vdash R : \chi \]

\[ \Psi \vdash \cdots : \quad \Psi \vdash R : \chi \quad \Psi \vdash w : \tau \]

\[ \Psi \vdash R, \ell \mapsto w : \chi, r : \tau \]

1.1.10 Well-Typed Stack \[ \Psi \vdash S : \sigma \]

\[ \Psi \vdash \text{nil} : \cdot \quad \Psi \vdash w : \tau \quad \Psi \vdash S : \sigma \]

\[ \Psi \vdash w :: S :: \sigma \]

1.1.11 Well-Typed Memory \[ \vdash M : (\Psi, \chi, \sigma) \]

\[ \vdash H : \Psi \quad \Psi \vdash R : \chi \quad \Psi \vdash S : \sigma \]

\[ \vdash (H, R, S) : (\Psi, \chi, \sigma) \]

1.1.12 Well-Typed Component \[ \Psi : \Delta ; \chi ; \sigma ; q \vdash e : \tau ; \sigma' \]

\[ \Psi \vdash H : \Psi' \quad \operatorname{boxheap}(\Psi') \quad \operatorname{ret-type}(q, \chi, \sigma) = \tau ; \sigma' \quad (\Psi, \Psi') : \Delta ; \chi ; \sigma ; q \vdash I \]

\[ \Psi : \Delta ; \chi ; \sigma ; q \vdash (I, H) : \tau ; \sigma' \]

\[ \operatorname{boxheap}(\Psi) \overset{\text{def}}{=} \forall (\ell : \nu \psi) \in \Psi, \nu = \operatorname{box} \]
1.1.13 Well-Typed Heap Value \( \Psi \vdash h : \nu \psi \)

\[
\vdash \forall [\Delta]. \{\chi; \sigma\}^q \quad \Psi ; \Delta \vdash \chi; \sigma; q : I \quad \Psi ; \vdash w_0 : \tau_0 \quad \ldots \quad \Psi ; \vdash w_n : \tau_n \\
\Psi \vdash \text{code}[\Delta] \{\chi; \sigma\}^q \cdot I \cdot \boxdot \forall [\Delta]. \{\chi; \sigma\}^q \\
\Psi \vdash \langle w_0, \ldots, w_n \rangle : \nu (\tau_0, \ldots, \tau_n)
\]

1.1.14 Well-Typed Word Value \( \Psi ; \Delta \vdash w : \tau \)

\[
\begin{align*}
\Psi ; \Delta \vdash () : \text{unit} & \quad \Psi ; \Delta \vdash n : \text{int} \\
\ell : \text{ref} \psi \in \Psi & \quad \ell : \text{box} \psi \in \Psi \\
\Psi ; \Delta \vdash w : \tau [\mu \alpha. \tau/\alpha] & \quad \Psi ; \Delta \vdash \text{fold}_{\mu \alpha. \tau} w : \mu \alpha. \tau \\
\Psi ; \Delta \vdash \text{pack}(\tau, w) \text{ as } \exists \alpha. \tau : \exists \alpha. \tau
\end{align*}
\]

1.1.15 Well-Typed Small Value \( \Psi ; \Delta ; \chi \vdash u : \tau \)

\[
\begin{align*}
\Psi ; \Delta \vdash w : \tau & \quad r : \tau \in \chi \\
\Psi ; \Delta \vdash \chi \vdash r : \tau & \quad \Psi ; \Delta \vdash \chi \vdash u : \tau [\mu \alpha. \tau/\alpha] \\
\Psi ; \Delta \vdash \text{fold}_{\mu \alpha. \tau} u : \mu \alpha. \tau & \quad \Psi ; \Delta \vdash \text{pack}(\tau, u) \text{ as } \exists \alpha. \tau : \exists \alpha. \tau \\
\Psi ; \Delta \vdash \chi \vdash u : \tau [\mu \alpha. \tau/\alpha] & \quad \Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \\
\Psi ; \Delta \vdash \text{fold}_{\mu \alpha. \tau} u : \mu \alpha. \tau & \quad \Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I \\
\Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I & \quad \Psi ; \Delta \vdash \text{pack}(\tau, u) \text{ as } \exists \alpha. \tau : \exists \alpha. \tau \\
\Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I & \quad \Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I \\
\Psi ; \Delta \vdash \text{ftv}(q') \subseteq \Delta & \quad \Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I \\
\Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I & \quad \Psi ; \Delta \vdash \chi \vdash u : \text{box} \forall [\alpha, \Delta']. \{\chi; \sigma\}^q \cdot I
\end{align*}
\]
As a side-condition on this judgment, the environment $q$ must not be $\epsilon$ (This can be written $\Psi; \Delta; \chi; \sigma; q \vdash I$).
This argument gives the position of the return address, and a component never abstracts over its own return address.

The rule for sequencing instructions is straightforward: we use the postconditions of the initial instruction $\iota$ as preconditions to type check the remaining instructions.

$$
\Psi; \Delta; \chi; \sigma; q \vdash \iota \Rightarrow \Delta'; \chi'; \sigma'; q' \quad \Psi; \Delta'; \chi'; \sigma'; q' \vdash I
$$

The return instruction typechecks if the indicated point to return to matches the return marker in the environment, and if the register containing the result has the right type. Note that our type system does not permit returns to an address stored on the stack.

$$
\chi(r) = \text{box} \forall \cdot \{r': \tau; \sigma\}^q \quad \chi(r') = \tau \quad \chi(r) = \tau
$$

The jump instruction is used to continue to the next code block within a component, or equivalently, to make a tail call. This requires that the code block we jump to share the current return marker, and that its preconditions are consistent with the current types of the register file and stack.

$$
\Psi; \Delta; \chi \vdash u: \text{box} \forall \cdot \{\chi'; \sigma\}^q \quad \Delta \vdash \chi' \leq \chi' \quad -[\Delta]; \chi; \sigma \vdash q
$$

To call a subroutine, we are required to protect the current return address by storing it in the tail of the stack that is parametrically hidden from the subroutine. The type rule compares the caller’s view of the stack at the time the call is made, $\sigma$, to the subroutine’s view at the time of the call, $\hat{\sigma}$. It also looks at the subroutine’s view at return time, $\hat{\sigma}'$. At both points, the subroutine’s view contains an abstract stack tail $\zeta$ that must be instantiated by the caller’s stack tail type $\sigma_0$. The return marker of the caller must be some $i$ that points far enough into $\sigma$ that the return address is in $\sigma_0$. At return time, the part of the stack that was not hidden from the subroutine could have changed length, so the return marker after the subroutine, which instantiates the subroutine’s type variable $\epsilon$, must accordingly change.

We also have a type rule for the case of calling a subroutine from the top level, when there is no return address and the return marker is $\text{end} \{\tau; \sigma^*\}$. This case works the same way, except that there is no need to worry about lengths of stacks.

We use the shorthand $\Delta \vdash \chi \setminus q$ in the following typing rules to ensure that, in the case that $q$ is some register $r$ in the domain of $\chi$, then $\chi$ without the mapping for $r$ should be well formed under $\Delta$.
As a side-condition on this judgment, the environment $q$ must not be $\epsilon$.

This judgment uses the following two metafunctions:

$$\text{inc}(q, n) = \begin{cases} i + n & q = i \\ q & \text{otherwise} \end{cases}$$

$$\text{dec}(q, n) = \begin{cases} i - n & q = i \geq n \\ \text{undefined} & q = i < n \\ q & \text{otherwise} \end{cases}$$

$$\Psi; \Delta; \chi; \sigma; q \vdash \epsilon \Rightarrow \Delta'; \chi'; \sigma'; q'$$

$$\Psi; \Delta; \chi; r_s ; \text{int} \quad \Psi; \Delta; \chi; u; \text{int} \quad q \neq r_d$$

$$\Psi; \Delta; \chi; \sigma; q \vdash \text{aop } r_d, r_s, u \Rightarrow \Delta; \chi[r_d; \text{int}]; \sigma; q$$

$$\Psi; \Delta; \chi; r_s ; \text{ref } \langle \tau_0, \ldots, \tau_n \rangle$$

$$\Psi; \Delta; \chi; \sigma; q \vdash \text{ld } r_d, r_s[i] \Rightarrow \Delta; \chi[r_d; \tau_i]; \sigma; q$$

$$\Psi; \Delta; \chi; r_d ; \text{ref } \langle \tau_0, \ldots, \tau_n \rangle$$

$$\Psi; \Delta; \chi; \sigma; q \vdash \text{st } r_d[i], r_s \Rightarrow \Delta; \chi[r_d; \text{ref } \langle \tau_0, \ldots, \tau_n \rangle]; \sigma; q$$
1.1.18 Reducton Relation

1.1.19 Instruction Sequence Reducton Relation \((M | I) \rightarrow (M' | I')\)

\[
\begin{align*}
\langle H, R, S \rangle | aop \ r_d, r_s, u; I \quad &\rightarrow \langle H, R[r_d \mapsto \delta(aop, R(r_s), \hat{R}(u))], S \rangle | I \\
\langle H, R, S \rangle | bnz \ r, u; I \quad &\rightarrow \langle H, R, S \rangle | I \quad \text{if } R(r) = 0 \\
\langle H, R, S \rangle | bnz \ r, u; I \quad &\rightarrow \langle H, R, S \rangle | I'[\omega/\Delta] \quad \text{if } R(r) = n, n \neq 0 \\
&\quad \text{where } \hat{R}(u) = \ell[\omega] \text{ and } H(\ell) = \text{code}[\Delta]\{\chi; \sigma\}^q.I' \\
\langle H, R, S \rangle | ld \ r_d, r_s[i]; I \quad &\rightarrow \langle H, R[r_d \mapsto w_i], S \rangle | I \\
&\quad \text{where } R(r_s) = \ell \text{ and } H(\ell) = \langle w_0, \ldots, w_i, \ldots, w_n \rangle \\
\langle H, R, S \rangle | str \ d[i], r_s; I \quad &\rightarrow \langle H[\ell \mapsto \langle w_0, \ldots, w', \ldots, w_n \rangle], R, S \rangle | I \\
&\quad \text{where } R(r_s) = w', R(r_d) = \ell, \text{ and } H(\ell) = \langle w_0, \ldots, w_i, \ldots, w_n \rangle \\
\langle H, R, w :: S \rangle | ralloc \ r_d, n; I \quad &\rightarrow \langle H[\ell \mapsto \langle w \rangle], R[r_d \mapsto \ell], S \rangle | I \quad \text{if } \ell \notin \text{dom}(H), \text{len}(w) = n \\
\langle H, R, w :: S \rangle | balloc \ r_d, n; I \quad &\rightarrow \langle H[\ell \mapsto \langle w \rangle], R[r_d \mapsto \ell], S \rangle | I \quad \text{if } \ell \notin \text{dom}(H), \text{len}(w) = n \\
\langle H, R, S \rangle | mv \ r_d, u; I \quad &\rightarrow \langle H, R[r_d \mapsto \hat{R}(u)], S \rangle | I \\
\langle H, R, S \rangle | unpack \ \langle \alpha, r_d \rangle \ u; I \quad &\rightarrow \langle H, R[r_d \mapsto w], S \rangle | I[\tau'/\alpha] \\
&\quad \text{where } \hat{R}(u) = \text{pack}(\tau', w) \text{ as } \exists \alpha.\tau \\
\langle H, R, S \rangle | unfold \ r_d, u; I \quad &\rightarrow \langle H, R[r_d \mapsto w], S \rangle | I \quad \text{where } \hat{R}(u) = \text{fold}_{\mu, \alpha, \tau} w \\
\langle H, R, S \rangle | salloc \ n; I \quad &\rightarrow \langle H, R, \langle \rangle :: S \rangle | I \\
&\quad \text{len}(\langle \rangle) = n \\
\langle H, R, w :: S \rangle | sfree \ n; I \quad &\rightarrow \langle H, R, S \rangle | I \\
&\quad \text{len}(w) = n \\
\langle H, R, S \rangle | sld \ r_d, i; I \quad &\rightarrow \langle H, R[r_d \mapsto w_i], S \rangle | I \quad \text{where } S = w_0 :: \cdots :: w_i :: S_0 \\
\langle H, R, S \rangle | sst \ i, r_s; I \quad &\rightarrow \langle H, R, S' \rangle | I \\
&\quad \text{where } R(r_s) = w', \\
&\quad S = w_0 :: \cdots :: w_i :: S_0 \text{, and } S' = w_0 :: \cdots :: w' :: S_0, \\
\langle H, R, S \rangle | jmp \ u \quad &\rightarrow \langle H, R, S \rangle | I[\omega/\Delta] \\
&\quad \text{where } \hat{R}(u) = \ell[\omega] \text{ and } H(\ell) = \text{code}[\Delta]\{\chi; \sigma\}^q.I \\
\langle H, R, S \rangle | call \ u \{\sigma, q\} \quad &\rightarrow \langle H, R, S \rangle | I[\omega/\Delta][\sigma/\zeta][q/\epsilon] \\
&\quad \text{where } \hat{R}(u) = \ell[\omega] \text{ and } H(\ell) = \text{code}[\Delta, \zeta, \epsilon]\{\chi; \sigma\}^q.I \\
\langle H, R, S \rangle | ret \ \{r'\} \quad &\rightarrow \langle H, R, S \rangle | I[\omega/\Delta] \\
&\quad \text{where } R(r) = \ell[\omega] \text{ and } H(\ell) = \text{code}[\Delta]\{\chi; \sigma\}^q.I
\end{align*}
\]

where

\[
\begin{align*}
\hat{R}(w) & = w \\
\hat{R}(r) & = R(r) \\
\hat{R}(u[\omega]) & = (\hat{R}(u))[\omega] \\
\hat{R}(\text{pack}(\tau', u) \text{ as } \exists \alpha.\tau) & = \text{pack}(\tau', \hat{R}(u)) \text{ as } \exists \alpha.\tau \\
\hat{R}(\text{fold}_{\mu, \alpha, \tau} u) & = \text{fold}_{\mu, \alpha, \tau}(\hat{R}(u))
\end{align*}
\]
1.1.20 Component Reduction Relation \( \langle M \mid e \rangle \rightarrow \langle M' \mid e' \rangle \)

For purposes of our step-indexed logical relation, we consider the reduction rule that loads heap values from a component’s heap fragment into the main program heap to take 0 reduction steps.

\[
\langle (H, R, S) \mid (I, (\ell \mapsto h, H')) \rangle \rightarrow^0 \langle ((H, \ell' \mapsto h[\ell' / \ell]), R, S) \mid (I[\ell' / \ell], H'[\ell' / \ell]) \rangle \quad \ell' \notin \text{dom}(H)
\]

\[
\frac{\langle M \mid I \rangle \rightarrow \langle M' \mid I' \rangle}{\langle M \mid (I, \cdot) \rangle \rightarrow \langle M' \mid (I', \cdot) \rangle}
\]

1.2 Contexts and Contextual Equivalence

\[
C ::= (C_1, H) \mid (I, C_H)
\]

\[
C_1 ::= [\cdot] \mid \cdot ; C_1
\]

\[
C_H ::= C_H, \ell \mapsto h \mid H, \ell \mapsto \text{code}[\Delta] \{\chi; \sigma\}^q.C_1
\]

1.2.1 Plug Function \( C[e] \)

\[
(C_1, H)[(I, H')] = (C_1[I], (H, H'))
\]

\[
(I, C_H)[(I', H')] = (I, (C_H[I'], H'))
\]

\[
[\cdot][I] = I
\]

\[
(\cdot ; C_1)[I] = \cdot ; C_1[I]
\]

\[
(C_H, \ell \mapsto h)[I] = (C_H[I]), \ell \mapsto h
\]

\[
(H, \ell \mapsto \text{code}[\Delta] \{\chi; \sigma\}^q.C_1)[I] = H, \ell \mapsto \text{code}[\Delta] \{\chi; \sigma\}^q.(C_1[I])
\]

1.2.2 Well-Typed Context \( \vdash C : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q' \vdash \tau'; \sigma_1 ) \)

\[
\Psi' \vdash H; \Psi \quad \text{ret-type}(q', \chi', \sigma_0) = \tau; \sigma_1 \quad \vdash C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow ((\Psi', \Psi); \Delta'; \chi'; \sigma_0; q')
\]

\[
\vdash C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q' \vdash \tau' ; \sigma_1 )
\]

\[
\vdash (I, C_H) : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q' \vdash \tau' ; \sigma_1 )
\]

\[
\Psi \subseteq \Psi' \quad \Delta \subseteq \Delta' \quad \Delta \vdash \chi' \leq \chi \quad \text{ret-type}(q, \chi, \sigma) = \tau; \sigma'
\]

\[
\vdash [\cdot ; C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q')
\]

\[
\vdash C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q')
\]

\[
H = \ell_1 \mapsto h_1, \ldots, \ell_n \mapsto h_n \quad H' = \ell'_1 \mapsto h'_1, \ldots, \ell'_m \mapsto h'_m
\]

\[
\Psi = \{\ell_1 : \nu_1 \psi_1, \ldots, \ell_n : \nu_n \psi_n, \ell : \text{box}[\Delta], \{\chi'; \sigma_0\}^q', \ell'_1 : \nu'_1 \psi'_1, \ldots, \ell'_m : \nu'_m \psi'_m\} \quad \vdash \psi_1 \cdot \cdots \cdot \psi_n
\]

\[
\Psi' ; \Psi \vdash h_1 : \nu_1 \psi_1 \cdots \Psi' ; \Psi \vdash h_n : \nu_n \psi_n \quad \vdash C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow ((\Psi', \Psi); \Delta'; \chi'; \sigma_0; q')
\]

\[
\vdash C_1 : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi'; \Delta'; \chi'; \sigma_0; q')
\]

\[
\vdash H, \ell \mapsto \text{code}[\Delta'] \{\chi'; \sigma_0\}^q.C_1.H' : (\Psi; \Delta; \chi; \sigma ; q \vdash \tau; \sigma' ) \rightarrow (\Psi' \vdash \Psi
\]

9
1.2.3 Contextual Equivalence

\[ \Psi; \Delta; \chi; \sigma; q \vdash e_1 \approx_{ctx} e_2 : \tau; \hat{\sigma} \]

\[ \forall C, M, \Psi', \chi', \sigma', q', \tau', \hat{\sigma}'. \]

\[ \vdash C : (\Psi; \Delta; \chi; \sigma; q \vdash \tau; \hat{\sigma}) \leadsto (\Psi'; \chi'; \sigma'; q' \vdash \tau'; \hat{\sigma}') \land \vdash M : (\Psi', \chi', \sigma') \]

\[ \implies (\langle M \mid C[e_1] \rangle \downarrow \iff \langle M \mid C[e_2] \rangle \downarrow) \]

1.2.4 CIU Equivalence

\[ \Psi; \Delta; \chi; \sigma; q \vdash e_1 \approx_{ciu} e_2 : \tau; \hat{\sigma} \]

\[ \forall \delta, E, M, \Psi', q', \tau', \hat{\sigma}'. \]

\[ \vdash \delta : \Delta \land \vdash E : (\Psi; \chi; \sigma; q \vdash \tau; \hat{\sigma}) \leadsto (\Psi'; \chi'; \sigma'; q' \vdash \tau'; \hat{\sigma}') \land \vdash M : (\Psi', \chi, \sigma) \]

\[ \implies (\langle M \mid E[\delta(e_1)] \rangle \downarrow \iff \langle M \mid E[\delta(e_2)] \rangle \downarrow) \]
1.3 Logical Relation

Worlds and Auxiliary Definitions  Worlds consist of a sequence of islands that describe the current state of the memories (and how they are related) of the two computations we wish to relate. The essential idea here is that the islands $\theta$ in the sequence $\Theta$ will specify constraints on disjoint parts of memory. We obtain constraints on the entire memory via a disjoint union of the memories specified by the islands.

Therefore, we begin with some simple definitions for memory objects that we will make use of in islands. We need to be able to lift various pieces of memory to a full program memory $M = (H, R, S)$. In many cases, we may not want to impose a constraint on the register file and stack, so we allow $\perp$ to appear in those positions. Since disjoint heap fragments can be merged, the heap can be left unconstrained just by using an empty heap.

$$\{\} \overset{\text{def}}{=} (\{\}, \perp, \perp)$$

$$\text{Regs}_\perp = \{R\} \cup \{\perp\} \quad H \overset{\text{def}}{=} (H, \perp, \perp)$$

$$\text{Stack}_\perp = \{S\} \cup \{\perp\} \quad R \overset{\text{def}}{=} (\{\}, R, \perp)$$

$$\quad S \overset{\text{def}}{=} (\{\}, \perp, S)$$

A world $W$ consists of a step index $k$, a pair of heap types $\Psi_1$ and $\Psi_2$, and a sequence $\Theta$ of islands $\theta$. Each island expresses invariants on certain parts of memory by encoding a state transition system and a memory relation $MR$ that establishes which pairs of memories are acceptable in each state. (See Dreyer et al. [2] for details.)

The first three islands in $\Theta$ are distinguished: they track the register file, the stack, and the immutable contents of the heap, respectively. We assign these islands the indices $i_{\text{reg}}$, $i_{\text{stk}}$, and $i_{\text{box}}$, respectively. Further islands can be added to a world to encode invariants about mutable data.

$$\text{World}_n \overset{\text{def}}{=} \{ W = (k, \Psi_1, \Psi_2, \Theta) \mid k < n \land \exists m \geq 3. \Theta \in \text{Island}^m_k \land \left( \exists s_{\text{reg}}. \Theta(i_{\text{reg}}) = \text{island}_{\text{reg}}(s_{\text{reg}}, k) \land \Psi_1 \vdash s_{\text{reg}}. R_1 : s_{\text{reg}}. \chi_1 \land \Psi_2 \vdash s_{\text{reg}}. R_2 : s_{\text{reg}}. \chi_2 \right) \land \left( \exists s_{\text{stk}}. \Theta(i_{\text{stk}}) = \text{island}_{\text{stk}}(s_{\text{stk}}, k) \land \Psi_1 \vdash s_{\text{stk}}. S_1 : s_{\text{stk}}. \sigma_1 \land \Psi_2 \vdash s_{\text{stk}}. S_2 : s_{\text{stk}}. \sigma_2 \right) \land \left( \exists s_{\text{box}}. \Theta(i_{\text{box}}) = \text{island}_{\text{box}}(s_{\text{box}}, k) \land \Psi_1 \vdash s_{\text{box}}. H_1 : \Psi_1^{\text{box}} \land \Psi_2 \vdash s_{\text{box}}. H_2 : \Psi_2^{\text{box}}\right) \}$$

$$\text{Island}_n \overset{\text{def}}{=} \{ \theta = (s, S, \delta, \pi, MR, bij) \mid s \in S \land S \in \text{Set} \land \delta \subseteq S \times S \land \pi \subseteq \delta \land \delta, \pi \text{ reflexive } \land \delta, \pi \text{ transitive } \land MR \in S \rightarrow \text{MemRel}_n \land bij \in S \rightarrow \mathbb{P}(\text{Val} \times \text{Val}) \}$$

$$\text{MemAtom}_n \overset{\text{def}}{=} \{ (W, M_1, M_2) \mid W \in \text{World}_n \land M_1, M_2 \in \text{Heap} \times \text{Regs}_\perp \times \text{Stack}_\perp \}$$

$$\text{MemRel}_n \overset{\text{def}}{=} \{ \varphi_M \subseteq \text{MemAtom}_n \mid \forall (W, M_1, M_2) \in \varphi_M. \forall W' \supseteq W. (W', M_1, M_2) \in \varphi_M \}$$

The transition systems for $\theta_{\text{reg}}$ and $\theta_{\text{stk}}$ encode the current contents of each side’s register file and stack, respectively. They may transition freely between states, since the register file and stack are fairly free in how they can change during program execution. The states of $\theta_{\text{box}}$ encode the contents of the immutable part of the heap on each side. This island is allowed to transition only by adding more immutable data to the heap.

$$i_{\text{reg}} = 1$$

$$S_{\text{reg}} = \{(R_1, \chi_1, R_2, \chi_2)\}$$

$$\text{island}_{\text{reg}}(s, k) = (s, S_{\text{reg}} \times S_{\text{reg}} \times S_{\text{reg}} \times S_{\text{reg}}, \lambda s.\{(W, s.R_1, s.R_2)\} \mid W \in \text{World}_k, \lambda s.\emptyset)$$

$$i_{\text{stk}} = 2$$

$$S_{\text{stk}} = \{(S_1, \sigma_1, S_2, \sigma_2)\}$$

$$\text{island}_{\text{stk}}(s, k) = (s, S_{\text{stk}} \times S_{\text{stk}} \times S_{\text{stk}} \times S_{\text{stk}}, \lambda s.\{(W, s.S_1, s.S_2)\} \mid W \in \text{World}_k, \lambda s.\emptyset)$$
where a given number of steps with object via the ∗W

in existing islands. Public future worlds ways: they may have expended steps, allocated additional memory, added new islands, or taken transitions W immutable memory has been allocated. We use this notation to capture this:

\[
(\underbrace{W, (s, W, \pi, \delta, \lambda)}_{W(k, W)}, \ldots, \underbrace{\lambda}_{W(k, W)})
\]

Two memory objects that describe disjoint parts of memory can be merged into one compound memory object via the \(\otimes\) operator.

\[
M_1 \otimes M_2 \text{ where } M_1, M_2 \in (\{H\} \times \text{Regs} \times \text{Stack})
\]

\[
(H_1, R_1, S_1) \otimes (H_2, R_2, S_2) = \begin{cases} 
(H_1 \cup R_2, R, S) & \text{where } R = R_1 \text{ if } R_2 = \perp; R = R_2 \text{ if } R_1 = \perp \\
S = S_1 & \text{if } S_2 = \perp; S = S_2 & \text{if } S_1 = \perp \\
\text{undefined} & \text{otherwise}
\end{cases}
\]

\[
\varphi_M \otimes \varphi_M' = \{ (W, M_1 \otimes M_1', M_2 \otimes M_2') \mid (W, M_1, M_2) \in \varphi_M \land (W, M_1', M_2') \in \varphi_M' \}
\]

These are standard operations for dealing with step indexing: we can approximate a world or relation to a given number of steps with \([-k, k]\), and we can expend a step using the \(\triangleright\) operator (read “later”).

\[
\triangleright k, \Psi_1, \Psi_2, \Theta = (k, \Psi_1, \Psi_2, \Theta) \quad \text{def} \quad \{(\theta_1, \ldots, \theta_m)_{k} \}
\]

\[
\triangleright e_1, e_2 = \{ (W, e_1, e_2) \mid W.k > 0 \implies (\triangleright W, e_1, e_2) \in \varphi_e \}
\]

\[
\triangleright v_1, v_2 = \{ (W, v_1, v_2) \mid W.k > 0 \implies (\triangleright W, v_1, v_2) \in \varphi_v \}
\]

Future worlds \(W'\) of a given world \(W\), written \(W' \supseteq W\), may differ from \(W\) in any or all of the following ways: they may have expended steps, allocated additional memory, added new islands, or taken transitions in existing islands. Public future worlds \(W' \supseteq_{pub} W\) are similar, but must have taken public transitions from the island states in \(W\).

\[
(k', \Psi_1', \Psi_2', \Theta') \supseteq (k, \Psi_1, \Psi_2, \Theta) \quad \text{def} \quad k' \leq k \land \Psi_1' \supseteq \Psi_1 \land \Psi_2' \supseteq \Psi_2 \land \Theta' \supseteq \Theta \land (k, \Psi_1, \Psi_2, \Theta) \in \text{World} \land (k', \Psi_1', \Psi_2', \Theta') \in \text{World}
\]

\[
(\theta_1', \ldots, \theta_m') \supseteq (\theta_1, \ldots, \theta_m) \quad \text{def} \quad m' \geq m \land \forall j \in \{1, \ldots, m\}. \theta_j' \supseteq \theta_j
\]

\[
(s', S', \delta', \pi', MR', bij') \supseteq (s, S, \delta, \pi, MR, bij) \quad \text{def} \quad (S', \delta', \pi', MR', bij') = (S, \delta, \pi, MR, bij) \land (s', s') \in \delta
\]

\[
W' \supseteq W
\]

\[
(k', \Psi_1', \Psi_2', \Theta') \supseteq_{pub} (k, \Psi_1, \Psi_2, \Theta) \quad \text{def} \quad k' \leq k \land \Psi_1' \supseteq \Psi_1 \land \Psi_2' \supseteq \Psi_2 \land \Theta' \supseteq_{pub} \Theta \land (k, \Psi_1, \Psi_2, \Theta) \in \text{World} \land (k', \Psi_1', \Psi_2', \Theta') \in \text{World}
\]

\[
(\theta_1', \ldots, \theta_m') \supseteq_{pub} (\theta_1, \ldots, \theta_m) \quad \text{def} \quad m' \geq m \land \forall j \in \{1, \ldots, m\}. \theta_j' \supseteq_{pub} \theta_j
\]

\[
(s', S', \delta', \pi', MR', bij') \supseteq_{pub} (s, S, \delta, \pi, MR, bij) \quad \text{def} \quad (S', \delta', \pi', MR', bij') = (S, \delta, \pi, MR, bij) \land (s', s') \in \pi
\]

Given a world \(W\), we often need to talk about future worlds of \(W\) where the only change is that new immutable memory has been allocated. We use this notation to capture this:

\[
W \oplus (H_1, H_2) \text{ def } (W.k, W, \Psi_1 \uplus \Psi_1, W, \Psi_2 \uplus \Psi_2, W, \Theta[i_{box} \mapsto \text{island}_{box}(W(i_{box}), s \uplus (H_1, H_2), W.k)])
\]

if \(W, \Psi_1 \vdash H_1 : \Psi_1 \land W, \Psi_2 \vdash H_2 : \Psi_2 \land \text{boxheap}(\Psi_1) \land \text{boxheap}(\Psi_2).
\]
The following are convenient shorthands for frequently-used pieces of a world:

\[
\begin{align*}
\text{currentMR}(\theta) & \overset{\text{def}}{=} \theta.\text{MR}(\theta.s) & W(i) & \overset{\text{def}}{=} W.\Theta(i)
\end{align*}
\]

\[
\begin{align*}
W.R_1 & \overset{\text{def}}{=} W.\Theta(_{\text{reg}}).s.R_1 & W.S_1 & \overset{\text{def}}{=} W.\Theta(_{\text{stk}}).s.S_1 & W.\chi_1 & \overset{\text{def}}{=} W.\Theta(_{\text{reg}}).s.\chi_1 & W.\sigma_1 & \overset{\text{def}}{=} W.\Theta(_{\text{stk}}).s.\sigma_1 \\
W.R_2 & \overset{\text{def}}{=} W.\Theta(_{\text{reg}}).s.R_2 & W.S_2 & \overset{\text{def}}{=} W.\Theta(_{\text{stk}}).s.S_2 & W.\chi_2 & \overset{\text{def}}{=} W.\Theta(_{\text{reg}}).s.\chi_2 & W.\sigma_2 & \overset{\text{def}}{=} W.\Theta(_{\text{stk}}).s.\sigma_2
\end{align*}
\]

\[
W.\Phi_1 \overset{\text{def}}{=} (W.\psi_1, W.\chi_1, W.\sigma_1) & W.\Phi_2 \overset{\text{def}}{=} (W.\psi_2, W.\chi_2, W.\sigma_2)
\]

Atoms are well-formed worlds together with a pair of components or values that are well-typed at the indicated type under the appropriate memory type of the world.

TermAtom_n([q_1 \vdash \tau_1; \sigma_1], (q_2 \vdash \tau_2; \sigma_2)) \overset{\text{def}}{=} \{(W, e_1, e_2) | W \in \text{World}_n \land W.\psi_1; : W.\chi_1; W.\sigma_1; q_1 \vdash e_1; \tau_1; \sigma_1 \land W.\psi_2; : W.\chi_2; W.\sigma_2; q_2 \vdash e_2; \tau_2; \sigma_2 \}

WvalAtom_n[\tau_1, \tau_2] \overset{\text{def}}{=} \{(W, w_1, w_2) | W \in \text{World}_n \land W.\psi_1; : w_1; \tau_1 \land W.\psi_2; : w_2; \tau_2 \}

StackAtom_n[\sigma_1, \sigma_2] \overset{\text{def}}{=} \{(W, s_1 | s_2) | W \in \text{World}_n \land W.\psi_1 \vdash s_1; \sigma_1 \land W.\psi_2 \vdash s_2; \sigma_2 \}

HvalAtom_n[\psi_1, \psi_2] \overset{\text{def}}{=} \{(W, h_1, h_2) | W \in \text{World}_n \land W.\psi_1 \vdash h_1; \psi_1 \land W.\psi_2 \vdash h_2; \psi_2 \}

ContAtom_n[(q_1 \vdash \tau_1; \sigma_1), (q_2 \vdash \tau_2; \sigma_2)] \overset{\text{def}}{=} \{(W, e_1, e_2) | W \in \text{World}_n \land E_1; \psi_1; : W.\chi_1; W.\sigma_1; q_1 \vdash e_1; \tau_1; \sigma_1 \land E_2; \psi_2; : W.\chi_2; W.\sigma_2; q_2 \vdash e_2; \tau_2; \sigma_2 \}

WvalRel[\tau_1, \tau_2] \overset{\text{def}}{=} \{\varphi_w \subseteq \text{WvalAtom}[\tau_1, \tau_2] | \forall(W, w_1, w_2) \in \varphi_w. \forall W'. \ni W. (W', w_1, w_2) \in \varphi_w \}

TValRel \overset{\text{def}}{=} \{VR = (\tau_1, \tau_2, \varphi_w) | \varphi_w \in \text{WvalRel}[\tau_1, \tau_2] \}

StackRel[\sigma_1, \sigma_2] \overset{\text{def}}{=} \{\varphi_s \subseteq \text{StackAtom}[\sigma_1, \sigma_2] \}

TStackRel \overset{\text{def}}{=} \{SR = (\sigma_1, \sigma_2, \varphi_s) | \varphi_s \in \text{StackRel}[\sigma_1, \sigma_2] \}

The set \(D[\Delta]\) ensures that an environment \(\rho\) mapping type variables to value relations is well-formed.

\[
\begin{align*}
D[\emptyset] & \overset{\text{def}}{=} \{\emptyset\} \\
D[\Delta, \alpha] & \overset{\text{def}}{=} \{\rho[\alpha \rightarrow \text{VR}] | \rho \in D[\Delta] \land \text{VR} \in \text{TValRel}\} \\
D[\Delta, \zeta] & \overset{\text{def}}{=} \{\rho[\zeta \rightarrow \text{SR}] | \rho \in D[\Delta] \land \text{SR} \in \text{TStackRel}\} \\
D[\Delta, \epsilon] & \overset{\text{def}}{=} \{\rho[\epsilon \rightarrow (q_1, q_2)] | \rho \in D[\Delta] \land \text{ftv}(q_1) = \emptyset \land \text{ftv}(q_2) = \emptyset\}
\end{align*}
\]

We use \(\rho_1\) and \(\rho_2\) to denote the substitutions formed by mapping variables in \(\text{dom } \rho\) to the first and second components, respectively, of the tuples they map to.

We also use some shorthands for referring to atoms of a particular type in terms of an environment \(\rho\):

\[
\begin{align*}
\text{TermAtom}_n[q \vdash \tau; \sigma]_{\rho} & \overset{\text{def}}{=} \text{TermAtom}_{\rho_1}(q_1 \vdash \rho_1(\tau); \rho_1(\sigma)), (q_2 \vdash \rho_2(\tau); \rho_2(\sigma)) \] \\
\text{WvalAtom}_n[\tau]_{\rho} & \overset{\text{def}}{=} \text{WvalAtom}_{\rho}(\rho_1(\tau), \rho_2(\tau)) \] \\
\text{HvalAtom}_n[\psi]_{\rho} & \overset{\text{def}}{=} \text{HvalAtom}_{\rho_1}(\psi), \rho_2(\psi) \] \\
\text{ContAtom}_n[q \vdash \tau; \sigma]_{\rho} & \overset{\text{def}}{=} \text{ContAtom}_{\rho_1}(q_1 \vdash \rho_1(\tau); \rho_1(\sigma)), (q_2 \vdash \rho_2(\tau); \rho_2(\sigma)) \] \\
& \overset{\text{def}}{=} \text{ContAtom}_{\rho_1}(q_1 \vdash \rho_1(\tau); \rho_1(\sigma)), (q_2 \vdash \rho_2(\tau); \rho_2(\sigma)) \]
\]
The following relation says that a memory relation \( \varphi_M \) satisfies the constraints imposed by a memory relation \( \varphi'_M \) in all worlds accessible from \( W \).

\[
\varphi_M \in_W \varphi'_M \overset{\text{def}}{=} \forall (\widehat{W}, M_1, M_2) \in \varphi_M. \widehat{W} \supseteq W \implies (\widehat{W}, M_1, M_2) \in \varphi'_M
\]

\[
\begin{align*}
\mathcal{W}[\alpha] & = \rho(\alpha), \varphi_w \\
\mathcal{W}[\text{unit}] & = \{(W, (), ()) \in \text{WvalAtom}[\text{unit}]\rho\} \\
\mathcal{W}[\text{int}] & = \{(W, n, n) \in \text{WvalAtom}[\text{int}]\rho\} \\
\mathcal{W}[\exists \alpha. \tau] & = \{(W, \text{pack}(\tau_1, w_1) \text{ as } \rho_1(\exists \alpha. \tau), \text{pack}(\tau_2, w_2) \text{ as } \rho_2(\exists \alpha. \tau)) \in \text{WvalAtom}[\exists \alpha. \tau]\rho | \\
& \quad \exists \varphi_w \in \text{WvalRel}[\tau_1, \tau_2]. (W, w_1, w_2) \in \mathcal{W}[\tau]\rho \mapsto (\tau_1, \tau_2, \varphi_w)\}\} \\
\mathcal{W}[\mu \alpha. \tau] & = \{(W, \text{fold}_1(\mu \alpha. \tau) w_1, \text{fold}_2(\mu \alpha. \tau) w_2) \in \text{WvalAtom}[\mu \alpha. \tau]\rho | \\
& \quad (w_1, w_2) \in \mathcal{W}[\tau[\mu \alpha. \tau/\alpha]]\rho\} \\
\mathcal{W}[\text{ref } \psi] & = \{(W, \ell_1, \ell_2) \in \text{WvalAtom}[\text{ref } \psi] | \exists i. \forall W' \supseteq W. \\
& \quad (\ell_1, \ell_2) \in W'(i).\text{bij}(W'(i), s) \land \\
& \quad \exists \varphi_M. \text{currentMR}(W'(i)) = \varphi_M \otimes \\
& \quad \{(\widehat{W}, \{\ell \mapsto h_1\}, \{\ell_2 \mapsto h_2\}) \in \text{MemAtom} | (\widehat{W}, h_1, h_2) \in \mathcal{W}[\psi]\}\}\} \\
\mathcal{W}[\text{box } \langle \tau_1, \ldots, \tau_n \rangle] & = \{(W, \ell_1, \ell_2) \in \text{WvalAtom}[\text{box } \langle \tau_1, \ldots, \tau_n \rangle] | \\
& \quad \forall (\widehat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})). \widehat{W} \supseteq W \implies \\
& \quad (\widehat{W}, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{W}[\langle \tau_1, \ldots, \tau_n \rangle]\rho\} \\
\mathcal{W}[\forall [\Delta], \{\chi; \sigma\}^q] & = \{(W, \ell_1[\omega_1], \ell_2[\omega_2]) \in \text{WvalAtom}[\forall [\Delta], \{\chi; \sigma\}^q] | \\
& \quad \forall (\widehat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})). \widehat{W} \supseteq W \implies \\
& \quad (M_1(\ell_1) = \text{code}[\beta_1, \Delta] \{\chi_1; \sigma_1\}^q \cdot I_1 \land \\
& \quad \rho_1(\chi) = \chi_1[\omega_1/\beta_1] \land \rho_1(\sigma) = \sigma_1[\omega_1/\beta_1] \land \rho_1(q) = q_1[\omega_1/\beta_1] \land \\
& \quad M_2(\ell_2) = \text{code}[\beta_2, \Delta] \{\chi_2; \sigma_2\}^q \cdot I_2 \land \\
& \quad \rho_2(\chi) = \chi_2[\omega_2/\beta_2] \land \rho_2(\sigma) = \sigma_2[\omega_2/\beta_2] \land \rho_2(q) = q_2[\omega_2/\beta_2] \land \\
& \quad (\widehat{W}, (\text{code}[\Delta] \{\chi_1; \sigma_1\}^q \cdot I_1)[\omega_1/\beta_1], \\
& \quad (\text{code}[\Delta] \{\chi_2; \sigma_2\}^q \cdot I_2)[\omega_2/\beta_2]) \in \mathcal{W}[\forall [\Delta], \{\chi; \sigma\}^q]\}\}\} \\
\mathcal{W}[\forall [\Delta], \{\chi; \sigma\}^q] & = \\
& \{(W, \text{code}[\Delta] \{\rho_1(\chi) ; \rho_1(\sigma)\}^q \cdot I_1, \text{code}[\Delta] \{\rho_2(\chi) ; \rho_2(\sigma)\}^q \cdot I_2) \in \text{HvalAtom}[\forall [\Delta], \{\chi; \sigma\}^q]\rho | \\
& \quad \forall W' \supseteq W. \forall \rho^* \in \mathcal{D}[\Delta]. \forall t, t'. \text{ let } \rho = \rho \cup \rho^* \text{ in } t; t' = \rho \text{ ret-type}(q, \chi, \sigma) \land \\
& \quad \text{currentMR}(W'(i_\text{reg})) \in W'. [\chi]^{\rho} \land \text{currentMR}(W'(i_\text{stk})) \in W'. [\sigma]^{\rho'} \implies \\
& \quad (W', (\rho_1^*(I_1), \cdot), (\rho_2^*(I_2), \cdot)) \in E[q \vdash t; t']^{\rho'}\}\} \\
\tau; \sigma' =_\rho \text{ ret-type}(q, \chi, \sigma) & \overset{\text{def}}{=} \rho_1(\tau); \rho_1(\sigma') = \text{ret-type}(\rho_1(q), \rho_1(\chi), \rho_1(\sigma)) \land \\
& \quad \rho_2(\tau); \rho_2(\sigma') = \text{ret-type}(\rho_2(q), \rho_2(\chi), \rho_2(\sigma)) \\
\mathcal{W}[\langle \tau_1, \ldots, \tau_n \rangle] & = \{(W, w_{11}, \ldots, w_{1m}), (w_{21}, \ldots, w_{2n}) \in \text{HvalAtom}[\langle \tau_1, \ldots, \tau_n \rangle] | \\
& \quad \forall j \in \{1, \ldots, n\}. (W, w_{1j}, w_{2j}) \in \mathcal{W}[\tau_j]\rho\} \}
\[(M_1, M_2) : W \ \overset{\text{def}}{\Rightarrow} M_1 : W.\Phi_1 \land M_2 : W.\Phi_2 \land \]
\[\overset{\text{running}(k, \langle M | e \rangle)}{\Rightarrow} \exists M', e'. \langle M | e \rangle \mapsto_i k \langle M' | e' \rangle \]
\[O = \{(W, e_1, e_2) \mid \forall(M_1, M_2) : W. ((M_1 | e_1) \downarrow \land \langle M_2 | e_2 \rangle \downarrow) \lor \]
\[\overset{\text{running}(W, k, \langle M_1 | e_1 \rangle)}{\Rightarrow} \overset{\text{running}(W, k, \langle M_2 | e_2 \rangle)}{\Rightarrow} \}
\[\mathcal{K}[q \vdash \tau; \sigma]_\rho = \{(W, E_1, E_2) \mid \forall W', q', r_1, r_2.
\]
\[W' \not\models_{\text{pub}} W \land (q =_\rho q' =_\rho \text{end}\{\tau; \sigma\} \lor \]
\[\exists r.q' = r \land \overline{\text{ret-addr}}(W, \rho_1(q)) = W'.R_1(r) \land \overline{\text{ret-addr}}(W, \rho_2(q)) = W'.R_2(r) \land \]
\[\text{ret-addr}(W, r) = W.\overline{R_1(r)} \ 
\text{ret-addr}(W, i) = W.S_1(i) \]
\[\text{ret-addr}(W, r) = W.\overline{R_2(r)} \ 
\text{ret-addr}(W, i) = W.S_2(i) \]
\[\text{ret-reg}_1(W, r) = r' \ \text{if} \ W.\overline{\chi_1(r)} = \text{box}\forall[\cdot].\{r': \tau; \sigma'\}^q \]
\[\text{ret-reg}_2(W, r) = r' \ \text{if} \ W.\overline{\chi_2(r)} = \text{box}\forall[\cdot].\{r': \tau; \sigma'\}^q \]
\[\mathcal{E}[q \vdash \tau; \sigma]_\rho = \{(W, e_1, e_2) \mid \text{TermAtom}[q \vdash \tau; \sigma]_\rho \mid \]
\[\forall E_1, E_2. (W, E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma]_\rho \Rightarrow (W, E_1[e_1], E_2[e_2]) \in O \}
\[\mathcal{H}[\{\cdot\}] = \text{World} \]
\[\mathcal{H}[\Psi, \ell; \overline{\text{ref}}_\psi] = \mathcal{H}[\Psi] \cap \{ W \in \text{World} \mid (W, \ell, \ell) \in \mathcal{W}[\overline{\text{ref}}_\psi]_{\emptyset} \}
\[\mathcal{H}[\Psi, \ell; \overline{\text{box}}_\psi] = \mathcal{H}[\Psi] \cap \{ W \in \text{World} \mid (W, \ell, \ell) \in \mathcal{W}[\overline{\text{box}}_\psi]_{\emptyset} \}
\[\mathcal{R}[\chi]_\rho = \{(W, R_1[\cdot], R_2[\cdot]) \mid \forall(r : \tau) \in \chi. (W, R_1(r), R_2(r)) \in \mathcal{W}[\tau]_\rho \}
\[\mathcal{S}[\zeta]_\rho = \rho(\zeta).\varphi_s \]
\[\mathcal{S}[\bullet]_\rho = \{(W, \text{nil}[\cdot], \text{nil}[\cdot]) \mid W \in \text{World} \}
\[\mathcal{S}[	au :: \sigma]_\rho = \{(W, (w_1 :: S_1[\cdot], w_2 :: S_2[\cdot])) \mid (W, w_1, w_2) \in \mathcal{W}[\tau]_\rho \land (W, S_1[\cdot], S_2[\cdot]) \in \mathcal{S}[\sigma]_\rho \}
\[\mathcal{H}[\Psi, \Delta; \chi; \sigma]_\rho = \{ W, \rho \in \mathcal{H}[\Psi] \land \rho \in \mathcal{D}[\Delta] \land \]
\[\overset{\text{currentMR}(W)(i_{\text{reg}})) \in W \mathcal{R}[\chi]_\rho \land \]
\[\overset{\text{currentMR}(W)(i_{\text{stk}})) \in W \mathcal{S}[\sigma]_\rho \Rightarrow \}
\[\overset{(W, \rho_1(e_1), \rho_2(e_2)) \in \mathcal{E}[q \vdash \tau; \sigma']_\rho}{(W, \rho_1(e_1), \rho_2(e_2)) \in \mathcal{E}[q \vdash \tau; \sigma']_\rho} \]
1.3.1 Other Logical Equivalences

To simplify the structure of the Lemmas in subsequent chapters, we define the following notions of logical equivalence for heaps, values, and instruction sequences:

**Definition 1.1 (Logical Equivalence for Heaps)**

\[ \Psi \vdash H_1 \approx H_2 : \Psi' \quad \Psi' \vdash H_2 : \Psi' \wedge \forall (\ell ; \psi) \in \Psi'. \Psi, \Psi' \vdash H_1(\ell) \approx_hv H_2(\ell) : \nu \psi \]

**Lemma 1.4 (World Extension is Reflexive and Transitive)**

1. By definition of \( \sqsubset \) for worlds and islands, and by the reflexivity of transition relations \( \delta \) in the definition of World.
2. By definition of \( \sqsubseteq_{pub} \) for worlds and islands, and by the reflexivity of public transition relations \( \pi \) in the definition of World.
3. By definition of \( \sqsubset \) for worlds and islands, and by the transitivity of transition relations \( \delta \) in the definition of World.
4. By definition of \( \sqsubseteq_{pub} \) for worlds and islands, and by the transitivity of public transition relations \( \pi \) in the definition of World.

**Lemma 1.5 (Properties of \( \sqcup \))**

1. If \((M_1, M_2) : W \) and \( M'_1 = (H_1, \perp, \perp), M'_2 = (H_2, \perp, \perp) \), then \((M_1 \sqcup M'_1, M_2 \sqcup M'_2) : W \sqcup (H_1, H_2) \).
2. \((W \Join (H_1, H_2)) \Join (H_1', H_2') = W \Join (H_1 \uplus H_1', H_2 \uplus H_2')\).

3. If \(W \in \text{World}\) and \((W \Join (H_1, H_2))\) is defined, then \((W \Join (H_1, H_2)) \sqsupseteq W\) and \((W \Join (H_1, H_2)) \sqsupseteq \text{pub} W\).

Proof

1. By definition of \(W(i_{\text{box}})\).
2. By definition of \(W(i_{\text{box}})\).
3. By definition of \(\sqsupseteq\), \(\sqsupseteq\text{pub}\), and island\(_{\text{box}}\).

Lemma 1.6 (Properties of \(\rhd\) and \(\sqsupseteq\))

For any \(W \in \text{World}\), we have

1. \(\rhd W \sqsupseteq W\)
2. \(\rhd W \sqsupseteq \text{pub} W\)
3. If \((M_1, M_2) : W\), then \((M_1, M_2) : \rhd W\).
4. If \(W' \sqsupseteq W\), then \(W' \equiv W\).
5. If \(W' \sqsupseteq W\), then \(W' \equiv \rhd W\).

Proof

1. By definition of \(\rhd\) and \(\sqsupseteq\), it suffices to show that \([\theta]_{W,k-1} \sqsupseteq [\theta]_{W,k-1}\) for each island \(\theta \in W.\Theta\). But this relation is reflexive, so we are done.
2. Similar.
3. Note that if \(W.k = 0\), there is nothing to show. Otherwise, the claim follows from the definitions of MemRel and \([\varphi_M]_k\).
4. Immediate from the definition of \(\sqsupseteq\).
5. From the definition of \(\sqsupseteq\) we have \(W'.k < W.k\) and \(W' \sqsupseteq W\). The latter implies that \(W' \in \text{World}\), which gives us \(0 \leq W'.k\). Hence, \(0 < W.k\).

Let \(W = (k + 1, \Psi_1, \Psi_2, \Theta)\). We have that:

\[
(W'.k, W'.\Psi_1, W'.\Psi_2, W'.\Theta) \sqsupseteq (k + 1, \Psi_1, \Psi_2, \Theta)
\]

We must show that:

\[
(W'.k, W'.\Psi_1, W'.\Psi_2, W'.\Theta) \sqsupseteq (k, \Psi_1, \Psi_2, [\Theta]_k)
\]

It suffices to show the following:

- \(W'.k \leq (\rhd W).k\): this follows from \(W'.k < W.k\) and \((\rhd W).k = W.k - 1\).
- \(W'.\Psi_i \supseteq \Psi_i\): by (4) we have \(W' \equiv W\), from which this fact is immediate.
- \(W'.\Theta \supseteq [\theta]_{W'.k}\): From above we have that \(W'.\theta' \supseteq [\theta]_{W'.k}\). Furthermore, since \(W'.k \leq (\rhd W).k = W.k - 1 = k\), we have that \([\theta]_{W'.k} = [\theta]_{W.k}\) so we are done.
1.4.2 Properties of the Observation Relation

Lemma 1.7 (\(O\) Closed under Anti-Reduction)

Given \(W' \supseteq W\), if \(W, k \leq W', k + k_1, W, k \leq W', k + k_2\), and

\[
\forall(M_1, M_2) : W. \exists(M'_1, M'_2) : W', (M_1 \mid e_1) \mapsto^{k_1} \langle M'_1 \mid e'_1 \rangle \land (M_2 \mid e_2) \mapsto^{k_2} \langle M'_2 \mid e'_2 \rangle,
\]

then

\[
(W', e'_1, e'_2) \in O \implies (W, e_1, e_2) \in O.
\]

Proof

Let \((M_1, M_2) : W\). Then, by our assumption, \((M_1 \mid e_1) \mapsto^{k_1} \langle M'_1 \mid e'_1 \rangle\) and \((M_2 \mid e_2) \mapsto^{k_2} \langle M'_2 \mid e'_2 \rangle\) for some \((M'_1, M'_2) : W'\). Since \((W', e'_1, e'_2) \in O\), we have either \((M'_1 \mid e'_1) \downarrow\) and \((M'_2 \mid e'_2) \downarrow\) or that running \((W', k, \langle M'_1 \mid e'_1 \rangle)\) and running \((W', k, \langle M'_2 \mid e'_2 \rangle)\).

In the former case, we have \((M_1 \mid e_1) \downarrow\) and \((M_2 \mid e_2) \downarrow\) by assumption. In the latter case, we have running \((W', k + k_1, \langle M_1 \mid e_1 \rangle)\) and running \((W', k + k_2, \langle M_2 \mid e_2 \rangle)\). Since we have assumptions that both of these are more steps than needed, we have the result. \(\square\)

1.4.3 Monotonicity and Reduction

Lemma 1.8 (Monotonicity)

Let \(\rho \in D[\Delta]\), where \(\Delta \vdash \tau\) and \(\Delta \vdash \psi\). If \(W' \supseteq W\), then

1. \((W, w_1, w_2) \in W[\tau]_\rho \implies (W', w_1, w_2) \in W[\tau]_\rho\)
2. \((W, h_1, h_2) \in H[\psi]_\rho \implies (W', h_1, h_2) \in H[\tau]_\rho\).

Proof

1. Proved by induction on \(W', k\) and on the structure of \(\tau\), simultaneously with Claim 2.

In each case, we will need to show \((W', w_1, w_2) \in W[\tau]_\rho\). This amounts to showing that \(W', \Psi_i \vdash w_i : \tau\) for \(i \in \{1, 2\}\). We have by assumption that \(W, \Psi_i \vdash w_i : \tau\). By definition of world extension, \(W', \Psi_i \supseteq W \Psi_i\), so this property holds.

To complete the proof, consider the possible cases of \(\tau\):

Case \(\alpha\) Follows from \(\rho(\alpha) \cdot \varphi_i^\rho \in W[\alpha]_{\rho(\alpha), \tau_1, \rho(\alpha), \tau_2}\), which holds by \(\rho(\alpha) \in TValRel\).

Case unit Immediate.

Case int Immediate.

Case \(\exists \alpha \cdot \tau'\) Follows from the induction hypothesis for the type.

Case \(\mu \alpha \cdot \tau'\) Follows from the induction hypothesis for the step index.

Case ref \(\psi\) We need to show that \((W', \ell_1, \ell_2) \in W[ref \psi]_\rho\). Let \(W'' \supseteq W'\). By transitivity of world extension, \(W'' \supseteq W\). Thus everything we need holds by our assumption that \((W', \ell_1, \ell_2) \in W[ref \psi]_\rho\).

Case box \(\langle \tau_1, \ldots, \tau_n \rangle\) We need to show that \((W', \ell_1, \ell_2) \in W[box \langle \tau_1, \ldots, \tau_n \rangle]_\rho\).

Let \((W, M'_1, M'_2) \in \text{currentMR}(W'(i_{\text{box}}))\) such that \(\tilde{W} \supseteq W'.\) By definition of island_{box}, \(M'_1 = (W'(i_{\text{box}}).s.H_1)\) and \(M'_2 = (W'(i_{\text{box}}).s.H_2)\). By our assumption, to show

\[
(W, M'_1(\ell_1), M'_2(\ell_2)) \in H[\langle \tau_1, \ldots, \tau_n \rangle]_\rho
\]

it suffices to find some \(M_1\) and \(M_2\) such that \((\tilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))\), \(M_1(\ell_1) = M'_1(\ell_2)\), and \(M_2(\ell_2) = M'_2(\ell_2)\), noting that \(\tilde{W} \supseteq W\) follows from \(\tilde{W} \supseteq W' \supseteq W\).

We claim that \(M_1 = (W(i_{\text{box}}).s.H_1)\) and \(M_2 = (W(i_{\text{box}}).s.H_2)\) are suitable. The first condition holds immediately by definition of island_{box}. Since \(W' \supseteq W\), we know that \(W'(i_{\text{box}}) \supseteq W(i_{\text{box}})\). Thus \(\langle H_1, H_2 \rangle, (H'_1, H'_2) \in \delta_{\text{box}}\) that is, \(H_1 \subseteq H'_1\) and \(H_2 \subseteq H'_2\). Since \(\ell_1\) and \(\ell_2\) must be in the domain of \(H_1\) and \(H_2\), we have the desired property that \(M_1(\ell_1) = M'_1(\ell_1)\) and \(M_2(\ell_2) = M'_2(\ell_2)\).
Let (\(W, M_1, M_2\)) \(\in\) currentMR(\(W'(i_{\text{box}})\)) such that \(\tilde{W} \sqsupset W'\). It suffices to find some \(M_1\) and \(M_2\) such that (\(\tilde{W}, M_1, M_2\)) \(\in\) currentMR(\(W(i_{\text{box}})\)), \(M_1(\ell_1) = M'_1(\ell_1)\), and \(M_2(\ell_2) = M'_2(\ell_2)\), noting that \(\tilde{W} \sqsupset W\). This can be done exactly as in the previous case.

2. Proved simultaneously with Claim 1.

In both cases, we need to show that \((W', h_1, h_2) \in \), HvalAtom[\(\Psi\)]\(\rho\). This amounts to showing that \(W'.\Psi_i \models h_i : \psi\) for \(i \in \{1, 2\}\). We have by assumption that \(W.\Psi_i \models h_i : \psi\). By definition of world extension, \(W'.\Psi_i \sqsupset W.\Psi_i\), so this property holds.

Consider the possible cases of \(\psi\):

Case \(\forall[\Delta], \{\chi; \sigma\}\): We need to show that \((W', h_1, h_2) \in \mathcal{H}[\forall[\Delta], \{\chi; \sigma\}]\rho\). Let \(W'' \sqsupset W'\). By transitivity of world extension, \(W'' \sqsupset W\). Thus everything we need holds by our assumption that \((W, h_1, h_2) \in \mathcal{H}[\forall[\Delta], \{\chi; \sigma\}]\rho\).

Case \(\langle \tau_1, \ldots, \tau_n \rangle\) Follows from Claim 1 using the induction hypothesis for the type.

\(\square\)

Lemma 1.9 (Monotonicity for Heaps)
If \(W' \sqsupset W\) and \(W \in \mathcal{H}[\Psi]\), then \(W' \in \mathcal{H}[\Psi]\).

Proof

We use induction on the structure of \(\Psi\). If \(\Psi = \{\cdot\}\) then there is nothing to show.

If \(\Psi = \Psi', \ell; \text{ref}\psi\), then by the induction hypothesis, \(W' \in \mathcal{H}[\Psi']\), and it remains to show that \((W', \ell, \ell) \in \mathcal{H}[\text{ref}\psi]\). But this follows from \(W \in \mathcal{H}[\Psi, \ell; \text{ref}\psi]\) and Lemma 1.8.

If \(\Psi = \Psi', \ell; \text{box}\psi\), then by the induction hypothesis, \(W' \in \mathcal{H}[\Psi']\), and it remains to show that \((W', \ell, \ell) \in \mathcal{H}[\text{box}\psi]\). But this follows from \(W \in \mathcal{H}[\Psi', \ell; \text{box}\psi]\) and Lemma 1.8.

\(\square\)

Lemma 1.10 (Monotonicity for Evaluation Contexts)
If \(W' \sqsupset \text{pub}\ W\) and if either \(q = q'\) or \(\text{end}\{\tau; \sigma\}\) or \(\text{ret-addr}_1(W, \rho_1(q)) = \text{ret-addr}_1(W', \rho_1(q'))\) and \(\text{ret-addr}_2(W, \rho_2(q)) = \text{ret-addr}_2(W', \rho_2(q'))\), then

\[(W, E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma] \rho \Rightarrow (W', E_1, E_2) \in \mathcal{K}[q' \vdash \tau; \sigma] \rho.\]

Proof

Follows from the transitivity of \(\sqsupset \text{pub}\) and our hypotheses about the relationship between \(q\) and \(q'\).

\(\square\)

Lemma 1.11 (\(\mathcal{E}[q \vdash \tau; \sigma] \rho\) Closed under Type-Preserving Anti-Reduction)
Let \((W, e_1, e_2) \in \text{TermAtom}[q \vdash \tau; \sigma] \rho\). Given \(W' \sqsupset \text{pub} W, W.k \leq W'.k + k_1, \ W.k \leq W'.k + k_2,\) and if \(q = q'\) or \(\text{end}\{\tau; \sigma\}\) or \(\text{ret-addr}_1(W, \rho_1(q)) = \text{ret-addr}_1(W', \rho_1(q'))\) and \(\text{ret-addr}_2(W, \rho_2(q)) = \text{ret-addr}_2(W', \rho_2(q'))\), then

\[(W', e_1', e_2') \in \mathcal{E}[q' \vdash \tau; \sigma] \rho \Rightarrow (W, e_1, e_2) \in \mathcal{E}[q \vdash \tau; \sigma] \rho.\]

Proof

Let \((W, E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma] \rho\). We need to show that \((W, E_1[e_1], E_2[e_2]) \in \mathcal{O}\). Since \((W', E_1, E_2) \in \mathcal{K}[q' \vdash \tau; \sigma] \rho\) by Lemma 1.10, we have \((W', E_1[e_1'], E_2[e_2']) \in \mathcal{O}\). By inspection of the operational semantics and by assumption, for any \((M_1, M_2) : W\), there is an \((M'_1, M'_2) : W'\) such that

\[\langle M_1 | E_1[e_1] \rangle \longrightarrow^{k_1} \langle M'_1 | E_1[e_1'] \rangle \text{ and } \langle M_2 | E_2[e_2] \rangle \longrightarrow^{k_2} \langle M'_2 | E_2[e_2'] \rangle.\]

The result follows by Lemma 1.7.

\(\square\)
1.4.4 Substitution

The next lemma is a simple property, but its proof shows the induction structure by which properties of the mutually-dependent parts of the logical relation can be proved.

**Definition 1.12**

\[\xi ::= \alpha \mid \zeta \mid \epsilon\]

AR ::= VR | SR | QR

**Lemma 1.13**

[Weakening] If \(\rho[\xi \mapsto AR] \in D[\Delta, \xi]\) and \(\xi \not\in \text{ftv}(\tau), \xi \not\in \text{ftv}(\sigma), \xi \not\in \text{ftv}(\chi), \xi \not\in \text{ftv}(\psi)\), then

1. \(S[\sigma]\rho = S[\sigma][\rho[\xi \mapsto AR]\]
2. \(R[\chi]\rho = R[\chi][\rho[\xi \mapsto AR]\]
3. \(W[\tau]\rho = W[\tau][\rho[\xi \mapsto AR]\]
4. \(H\mathcal{V}[\psi]\rho = H\mathcal{V}[\psi][\rho[\xi \mapsto AR]\]
5. \(E[q \vdash \tau; \sigma]\rho = E[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\]
6. \(K[q \vdash \tau; \sigma]\rho = K[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\]

**Proof**

Assume \(\xi \not\in \text{ftv}(\tau)\) and \(\xi \not\in \text{ftv}(\sigma)\). We will need to prove the following:

1. (a) \((W, S_1, S_2) \in S[\sigma]\rho \implies (W, S_1, S_2) \in S[\sigma][\rho[\xi \mapsto AR]\]
   (b) \((W, S_1, S_2) \in S[\sigma][\rho[\xi \mapsto AR]\] \implies (W, S_1, S_2) \in S[\sigma]\rho\)
2. (a) \((W, R_1, R_2) \in R[\chi]\rho \implies (W, R_1, R_2) \in R[\chi][\rho[\xi \mapsto AR]\]
   (b) \((W, R_1, R_2) \in R[\chi][\rho[\xi \mapsto AR]\] \implies (W, R_1, R_2) \in R[\chi]\rho\)
3. (a) \((W, w_1, w_2) \in W[\tau]\rho \implies (W, w_1, w_2) \in W[\tau][\rho[\xi \mapsto AR]\]
   (b) \((W, w_1, w_2) \in W[\tau][\rho[\xi \mapsto AR]\] \implies (W, w_1, w_2) \in W[\tau]\rho\)
4. (a) \((W, h_1, h_2) \in H\mathcal{V}[\psi]\rho \implies (W, h_1, h_2) \in H\mathcal{V}[\psi][\rho[\xi \mapsto AR]\]
   (b) \((W, h_1, h_2) \in H\mathcal{V}[\psi][\rho[\xi \mapsto AR]\] \implies (W, h_1, h_2) \in H\mathcal{V}[\psi]\rho\)
5. (a) \((W, e_1, e_2) \in E[q \vdash \tau; \sigma]\rho \implies (W, e_1, e_2) \in E[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\]
   (b) \((W, e_1, e_2) \in E[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\] \implies (W, e_1, e_2) \in E[q \vdash \tau; \sigma]\rho\)
6. (a) \((W, E_1, E_2) \in K[q \vdash \tau; \sigma]\rho \implies (W, E_1, E_2) \in K[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\]
   (b) \((W, E_1, E_2) \in K[q \vdash \tau; \sigma][\rho[\xi \mapsto AR]\] \implies (W, E_1, E_2) \in K[q \vdash \tau; \sigma]\rho\).

We will prove all these claims simultaneously, by induction on \(W.k\) and \(\tau\).

1. We use an additional induction and case analysis on the structure of \(\sigma\).
   - **Case** \(\zeta\) Immediate, since \(\xi \neq \zeta\).
   - **Case** \(\alpha\) Immediate.
   - **Case** \(\tau::\sigma\) For part (a), we have \(S_1 = w_1::S'_1, S_2 = w_2::S'_2\), \((W, w_1, w_2) \in W[\tau]\rho\), and \((W, S'_1, S'_2) \in S[\sigma]\rho\). We need to show that \((W, w_1, w_2) \in W[\tau]\rho[\xi \mapsto AR]\) and \((W, S'_1, S'_2) \in S[\sigma][\rho[\xi \mapsto AR]\]. The latter holds by the induction hypothesis for \(\sigma\) and the former holds by claim 3.
     - Part (b) is similar.
2. Follows from claim 3.
3. Consider the possible cases of \(\tau\):
   - **Case** \(\alpha\) Immediate, since \(\alpha \neq \xi\).
Proof

Case \text{unit} Immediate.

Case \text{int} Immediate.

Case \exists \alpha. \tau By the induction hypothesis for \tau.

Case \mu \alpha. \tau By the induction hypothesis for W.k.

Case \text{ref} \langle \tau_1, \ldots, \tau_n \rangle Follows from claim 4.

Case \text{box} \langle \tau_1, \ldots, \tau_n \rangle Follows from claim 4.

Case \text{box} \langle \Delta \rangle. \lbrace \chi; \sigma' \rbrace^q Follows from claim 4.

4. Consider the two possible cases of \psi:

Case \forall [\Delta]. \langle \chi; \sigma' \rangle^q Follows from claims 1 and 2 (using the induction hypothesis for \tau) and from claim 5 (using the induction hypothesis for W.k).

Case \langle \tau_1, \ldots, \tau_n \rangle Follows from claim 3 (using the induction hypothesis for \tau).

5. Follows from claim 6.

6. Follows from claims 1 and 3.

\[\square\]

Lemma 1.14 (Substitution)

Let \rho \in D[\Delta], \alpha \notin \Delta, \Delta \vdash \tau', and \Delta, \alpha \vdash \tau, \Delta, \alpha \vdash \sigma, \Delta, \alpha \vdash \chi, \Delta, \alpha \vdash \psi. Then

1. \[S[\sigma][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = S[\sigma[\tau'/\alpha]]\rho\]

2. \[R[\chi][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = R[\chi[\tau'/\alpha]]\rho\]

3. \[W[\tau][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = W[\tau[\tau'/\alpha]]\rho\]

4. \[\mathcal{H}[\psi][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = \mathcal{H}[\psi[\tau'/\alpha]]\rho\]

5. \[E[q \vdash \tau; \sigma][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = E[q[\tau'/\alpha] \vdash \tau[\tau'/\alpha]; \sigma[\tau'/\alpha]]\rho\]

6. \[\mathcal{K}[q \vdash \tau; \sigma][\alpha \rightarrow (p_1(\tau'), p_2(\tau'), \mathcal{W}[\tau']\rho)] = \mathcal{K}[q[\tau'/\alpha] \vdash \tau[\tau'/\alpha]; \sigma[\tau'/\alpha]]\rho\]

Proof

Follows the structure of the proof of Lemma 1.13. The only case that depends on \rho is in claim 3, in the case where \tau = \beta. But the needed equality is immediate in this case, whether \alpha = \beta or not. \[\square\]

Lemma 1.15 (Substitution for Stack Types)

Let \rho \in D[\Delta], \alpha \notin \Delta, \Delta \vdash \tau', and \Delta, \alpha \vdash \tau, \Delta, \alpha \vdash \sigma, \Delta, \alpha \vdash \chi. Then

1. \[S[\sigma][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = S[\sigma[\sigma'/\alpha]]\rho\]

2. \[R[\chi][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = R[\chi[\sigma'/\alpha]]\rho\]

3. \[W[\tau][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = W[\tau[\sigma'/\alpha]]\rho\]

4. \[\mathcal{H}[\psi][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = \mathcal{H}[\psi[\sigma'/\alpha]]\rho\]

5. \[E[q \vdash \tau; \sigma][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = E[q[\sigma'/\alpha] \vdash \tau[\sigma'/\alpha]; \sigma[\sigma'/\alpha]]\rho\]

6. \[\mathcal{K}[q \vdash \tau; \sigma][\alpha \rightarrow (p_1(\sigma'), p_2(\sigma'), \mathcal{S}[\sigma']\rho)] = \mathcal{K}[q[\sigma'/\alpha] \vdash \tau[\sigma'/\alpha]; \sigma[\sigma'/\alpha]]\rho\]

Proof

Follows the structure of the proof of Lemma 1.13. The only case that depends on \rho(\alpha) is in claim 1, in the case where \sigma = \sigma'. But the needed equality is immediate, whether \alpha = \alpha' or not. \[\square\]

Lemma 1.16 (Substitution for Return Markers)

Let \rho \in D[\Delta], \Delta \vdash \tau', and \Delta, \epsilon \vdash \tau, \Delta, \epsilon \vdash \sigma, \Delta, \epsilon \vdash \chi, \Delta, \epsilon \vdash \psi. Then
Proof

Followed the structure of the proof of Lemma 1.13. There are no interesting cases.

1.4.5 Properties of Semantic Interpretations

Lemma 1.17
If \( \rho \in D[\Delta] \) and \( \Delta \vdash \tau \), then \( W[\tau] \rho \in WvalRel[\rho_1(\tau), \rho_2(\tau)] \).

Proof

Follows from monotonicity.

Lemma 1.18
If \( \rho \in D[\Delta] \) and \( \Delta \vdash \sigma \), then \( S[\sigma] \rho \in StackRel[\rho_1(\sigma), \rho_2(\sigma)] \).

Proof

Proceed by induction on the structure of \( \sigma \).

Case 1: From \( \Delta \vdash \sigma \) we have that \( \zeta \in \Delta \). Since \( \rho \in D[\Delta] \), it follows that \( S[\zeta] \rho = \rho(\zeta) \cdot \varphi \cdot S \in StackRel[\rho_1(\zeta), \rho_2(\zeta)] \).

Case 2: In this case, \( S[\bullet] \rho = \{(W, \text{\small nil}, \text{\small nil}) \mid W \in \text{\small World}\} \in StackRel[\bullet, \bullet] \) is immediate from the definition of StackRel.

Case 3: For any \( (W, w_1 : S_1 \vdash, w_2 : S_2 \vdash) \in S[\tau : \sigma] \rho \), we need that \( W.\Psi_i \vdash w_i : S_i : \rho_i(\tau : \sigma) \).

From the definition of \( S[\tau : \sigma] \rho \) we have \( S[\tau : \sigma] \rho \in W[\tau] \rho \) and \( S[S_1 \vdash, S_2 \vdash] \in S[\sigma] \rho \), from which we have that \( W.\Psi_i \vdash w_i : \rho_i(\tau) \) and \( W.\Psi_i \vdash S_i : \rho_i(\sigma) \), which gives us what we need.

 Lemma 1.19 (Register File Subtyping Implies Inclusion)
Let \( \rho \in D[\Delta] \) and \( \Delta \vdash \chi \leq \chi' \). Then \( R[\chi] \rho \subseteq R[\chi'] \rho \).

Proof

Consider arbitrary \( (W, M_1, M_2) \in R[\chi] \rho \). Note that \( M_1 = R_1 \rho \) and \( M_2 = R_2 \rho \). We must show that \( (W, M_1, M_2) = (W, R_1 \rho, R_2 \rho) \) \( \in R[\chi'] \rho \).

Consider \( \tau : \tau \in \chi' \). We must show \( (W, R_1(\tau), R_2(\tau)) \in W[\tau] \rho \). From the hypothesis \( \Delta \vdash \chi \leq \chi' \), it follows that \( \tau : \tau \in \chi \). We use the latter to instantiate \( (W, R_1 \rho, R_2 \rho) \) \( \in R[\chi] \rho \), which gives us what we needed to show.

 Lemma 1.20 (World Updates that Respect Register-File Relation)
Let \( \text{currentMR}(W(i_{\text{reg}})) \in W \vdash R[\chi] \rho \) and \( W' \sqsupseteq W \).

1. If \( W'(i_{\text{reg}}) = W(i_{\text{reg}}) \), then \( \text{currentMR}(W'(i_{\text{reg}})) \in W' \vdash R[\chi] \rho \).

2. Let \( W' = (W, k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{\text{reg}} \mapsto \text{island}_{\text{reg}}(s, W.k)]) \), where \( s = (W.R_1[r_d \mapsto w_1], W.\chi_1[r_d : \rho_1(\tau)], W.R_2[r_d \mapsto w_2], W.\chi_2[r_d : \rho_2(\tau)]) \). If \( (W', w_1, w_2) \in W[\tau] \rho \), then \( \text{currentMR}(W'(i_{\text{reg}})) \in W' \vdash R[\chi[r_d : \tau]] \rho \).
Lemma 1.21 (World Updates that Respect Stack Relation)

Let currentMR(W′(i_{reg})) ∈ W.S[σ]ρ and W′ ⊇ W.

1. If W′(i_{stk}) = W(i_{stk}), then currentMR(W′(i_{stk})) ∈ W.S[σ][ρ].

2. Let W.S_1 = w_{11} :: ··· :: w_{1n} :: S'_1, W.S_2 = w_{21} :: ··· :: w_{2n} :: S'_2, σ = τ_1 :: ··· :: τ_n :: σ' and W′ = (W.k, W.Ψ_1, W.Ψ_2, W.Θ[i_{stk} ↦ island_{reg}(s, W.k)]), where s = (S'_1, S'_2, σ'). currentMR(W′(i_{stk})) ∈ W.S[σ'][ρ].

3. Let W′ = (W.k, W.Ψ_1, W.Ψ_2, W.Θ[i_{stk} ↦ island_{reg}(s, W.k)]), where s = (w_{11} :: ··· :: w_{1n} :: S_1, τ_1 :: ··· :: τ_n :: σ, w_{21} :: ··· :: w_{2n} :: S_2, τ_1 :: ··· :: τ_n :: σ). If (W′, w_{11}, w_{21}) ∈ W.[σ][ρ], then currentMR(W′(i_{stk})) ∈ W.S[τ_1 :: ··· :: τ_n :: σ][ρ].

4. Let W.S_1 = w_{11} :: ··· :: w_{1n} :: S'_1, W.S_2 = w_{21} :: ··· :: w_{2n} :: S'_2, σ = τ_1 :: ··· :: τ_n :: σ' and W′ = (W.k, W.Ψ_1, W.Ψ_2, W.Θ[i_{stk} ↦ island_{reg}(s, W.k)]), where s = (w_{11} :: ··· :: w_{1n-1} :: w'_1 :: S'_1, τ_1 :: ··· :: τ_{n-1} :: τ'_1 :: σ', w_{21} :: ··· :: w_{2n-1} :: w'_2 :: S'_2, τ_1 :: ··· :: τ_{n-1} :: τ'_1 :: σ'). If (W′, w'_1, w'_2) ∈ W.[σ'][ρ], then currentMR(W′(i_{stk})) ∈ W.S[τ_1 :: ··· :: τ_{n-1} :: τ'_1 :: σ'][ρ].

Proof

1. Consider arbitrary (W̃, M_1, M_2) ∈ currentMR(W′(i_{reg})) such that W̃ ⊇ W′. We must show that (W̃, M_1, M_2) ∈ S[σ][ρ].

   Instantiate the first premise with (W̃, M_1, M_2), noting that the latter is in currentMR(W(i_{reg})) since W′(i_{reg}) = W(i_{reg}), and noting that W̃ ⊇ W by transitivity of ⊇. Hence, (W̃, M_1, M_2) ∈ S[σ][ρ] as we needed to show.

2. Consider arbitrary (W̃, M_1, M_2) ∈ currentMR(W′(i_{reg})) such that W̃ ⊇ W′. We must show that (W̃, M_1, M_2) ∈ S[σ][ρ]. Note that M_i must be of the form w_{j_1} :: ··· :: w_{j_n} :: S_j_i and w_{j_1} :: ··· :: w_{j_n} :: S_j = W′.S_j. The desired result follows directly from monotonicity (Lemma 1.8) after unfolding currentMR(W(i_{stk})) ∈ W.S[σ][ρ] and premise (W′, w_{j_1}, w_{j_2}) ∈ W.[σ][ρ].

3. Consider arbitrary (W̃, M_1, M_2) ∈ currentMR(W′(i_{reg})) such that W̃ ⊇ W′. We must show that (W̃, M_1, M_2) ∈ S[τ_1 :: ··· :: τ_n :: σ][ρ] and w_{j_1} :: ··· :: w_{j_n} :: S_j = W′.S_j. The desired result follows directly from monotonicity (Lemma 1.8) after unfolding currentMR(W(i_{stk})) ∈ W.S[τ_1 :: ··· :: τ_n :: σ][ρ] and premise (W′, w_{j_1}, w_{j_2}) ∈ W.[τ_1 :: ··· :: τ_n :: σ][ρ].
4. Let $\sigma' = \tau_1 :: \cdots :: \tau_n :: \sigma$. Consider arbitrary $1 \leq k \leq |\sigma'|$ such that such that $(\sigma'(k) = \tau_k)$. We must show that $(\tilde{W}, W', S_1(k), W'.S_2(k)) \in \mathcal{W}[\tau] \rho$.

Case $k = n$: Then $\tau' = \tau_n$ and $W'.S_1(k) = w'_1$, which means that it suffices to show $(\tilde{W}, w'_1, w'_2) \in \mathcal{W}[\tau'] \rho$. This latter is immediate from the premise $(W', w_1, w_2) \in \mathcal{W}[\tau] \rho$ using monotonicity (Lemma 1.8).

Case $k \neq n$: Then $W'.S_1(k) = W.S_1(k)$, which means that it suffices to show $(\tilde{W}, W.S_1(k), W.S_2(k)) \in \mathcal{W}[\tau_2 \rho]$. Instantiate the first premise with $(\tilde{W}, W.S_1(k), W.S_2(k))$ noting that the latter is in currentMR$(W(i_{reg}))$ by the definition of island$_{reg}$. Also note that $\tilde{W} \sqsupset W$ by transitivity of $\sqsupset$. Hence, we have that $(W, W.S_1(1), W.S_2(1)) \in \mathcal{R}[\sigma] \rho$. Instantiating the latter with $(\tau_k) = \sigma(k)$ gives us $(\tilde{W}, W.S_1(k), W.S_2(k)) \in \mathcal{W}[\tau_k] \rho$ as we needed to show.

Lemma 1.22 (Heap Interpretation Extension with Boxheap)
If $W \in \mathcal{H}[\Psi], \Psi \vdash H_1 \approx_{H} H_2 : \Psi'$, and boxheap$(\Psi')$, then $W \uplus (H_1, H_2) \in \mathcal{H}[\Psi, \Psi']$.

Proof
By induction on $W.k$.
To show $W \uplus (H_1, H_2) \in \mathcal{H}[\Psi, \Psi']$, it suffices to show
$$\forall (\ell : \nu \psi) \in (\Psi, \Psi'). (W \uplus (H_1, H_2), \ell, \ell) \in \mathcal{W}[\nu \psi] \emptyset$$
where $\nu$ is box or ref.
Consider arbitrary $(\ell : \nu \psi) \in (\Psi, \Psi')$. If $\ell \in \text{dom}(\Psi)$, for any value of $W.k$, it follows from the first premise that $(W, \ell, \ell) \in \mathcal{W}[\nu \psi] \emptyset$. By Lemma 1.5 we have that $W \uplus (H_1, H_2) \sqsupset W$, so by monotonicity we have $(W \uplus (H_1, H_2), \ell, \ell) \in \mathcal{W}[\nu \psi] \emptyset$.

Therefore, it remains for us to show that if $\ell \in \text{dom}(\Psi')$ then $(W \uplus (H_1, H_2), \ell, \ell) \in \mathcal{W}[\text{box} \psi] \emptyset$.

Case $W.k = 0$: Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}((W \uplus (H_1, H_2))(i_{box}))$ such that $\tilde{W} \sqsupset (W \uplus (H_1, H_2))$. But the latter implies $\tilde{W}.k < W.k$ which leads to a contradiction since $\tilde{W} \in \text{World}$ which requires that $\tilde{W}.k \geq 0$. So we are done.

Case $W.k = n + 1$ for $n \geq 0$: By the induction hypothesis we know that the lemma we wish to prove holds for any $W$ such that $W.k = n$. We must prove it for any $W$ such that $W.k = n + 1$.

We have that $\ell \in \text{dom}(\Psi')$ and must show that $(W \uplus (H_1, H_2), \ell, \ell) \in \mathcal{W}[\text{box} \psi] \emptyset$.
Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}((W \uplus (H_1, H_2))(i_{box}))$ such that $\tilde{W} \sqsupset (W \uplus (H_1, H_2))$. Note that it must be the case that $M_1 = (W(i_{box}, s.H_1.H_1))$ and $M_2 = (W(i_{box}, s.H_2.H_2))$. Also, since $\ell \in \text{dom}(\Psi')$, from the second premise it follows that $\ell \in \text{dom}(H_1)$ and $\ell \in \text{dom}(H_2)$. Regardless of whether $\psi$ is a code type or tuple type, it suffices to show:
$$\langle \tilde{W}, M_1(\ell), M_2(\ell) \rangle \in \mathcal{H}[\psi] \emptyset$$
$$\equiv \langle \tilde{W}, H_1(\ell), H_2(\ell) \rangle \in \mathcal{H}[\psi] \emptyset$$

From the second premise, since $(\ell : \text{box} \psi) \in \Psi'$, it follows that:
$$\Psi, \Psi' \vdash H_1(\ell) \approx_{hv} H_2(\ell) : \text{box} \psi.$$  (1)
Since $\triangleright W \in \mathcal{H}[\Psi]$ by heap monotonicity (Lemma 1.9) and since $(\triangleright W).k = n$, by the induction hypothesis we have that $(\triangleright W \uplus (H_1, H_2)) \in \mathcal{H}[\Psi, \Psi']$. Thus, we can instantiate (1) with $\triangleright W \uplus (H_1, H_2)$, which allows us to conclude that
$$(\triangleright W \uplus (H_1, H_2), H_1(\ell), H_2(\ell)) \in \mathcal{H}[\psi] \emptyset$$

Now, since $\tilde{W} \sqsupset (W \uplus (H_1, H_2))$, we can use by Lemma 1.6 to conclude $\tilde{W} \sqsupset (\triangleright W \uplus (H_1, H_2))$. Hence, by monotonicity, we have $(\tilde{W}, H_1(\ell), H_2(\ell)) \in \mathcal{H}[\psi] \emptyset$ as we needed to show.
1.5 Compatibility Lemmas

Lemma 1.23 (Component)
If \( \Psi \vdash H_1 \equiv H_2 \); \( \Psi' \), boxheap(\( \Psi' \)), ret-type(\( q, \chi, \sigma \)) \( = \tau; \sigma' \), and \( (\Psi, \Psi') \vdash \Delta; \chi; \sigma; q \vdash I_1 \equiv I_2 \), then \( \Psi; \Delta; \chi; \sigma; q \vdash (I_1, H_1) \equiv (I_2, H_2) : \tau; \sigma' \).

Proof

From the premises, we have that \( \Psi; \Delta; \chi; \sigma; q \vdash (I_1, H_1) : \tau; \sigma' \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in H[\[\Psi]] \), \( \rho \in D[\[\Delta]] \), currentMR(\( W(i_{reg}) \)) \( \in W \cdot R[\[\chi]]\rho \), and currentMR(\( W(i_{stk}) \)) \( \in W \cdot S[\[\sigma]]\rho \). We need to show that

\[
(W, \rho_1((I_1, H_1)), \rho_2((I_2, H_2))) \in E[q - \tau; \sigma']
\]

or equivalently, noting that the typing rules for heap fragments ensure they have no free type variables,

\[
(W, (\rho_1(I_1), H_1), (\rho_1(I_2), H_2)) \in E[q - \tau; \sigma']
\]

In the following, let \( \tau; \sigma = \text{ret-type}(q, \chi, \sigma) \).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that \( (W, (\rho_1(I_1), H_1), (\rho_1(I_2), H_2)) \in \text{TermAtom}[q - \tau; \sigma] \rho \). To establish this, we must show:

\[
W.\Psi_i;::W.\chi_i;W.\sigma_i;\rho_i(q) \vdash (\rho_i(I_i), H_i); \rho_i(\tau_i); \rho_i(\sigma_i)
\]

Applying the typing rule, we see we need to show that

\[
W.\Psi_i;::H_i;\Psi' \quad \text{and} \quad (W.\Psi_i, \Psi');::W.\chi_i;W.\sigma_i;\rho_i(q) \vdash \rho_i(I_i)
\]

We know that \( \Psi \subseteq W.\Psi_i \) (from \( W \in H[\[\Psi]] \)), \( \chi \subseteq W.\chi_i \) (from currentMR(\( W(i_{reg}) \)) \( \in W \cdot R[\[\chi]]\rho \)), and \( \sigma \subseteq W.\sigma_i \) (from currentMR(\( W(i_{stk}) \)) \( \in W \cdot S[\[\sigma]]\rho \)). Combining these with the premise \( (W, \Psi', \Delta; \chi; \sigma; q \vdash I_i \) yields what we need.

- Let \( H'_1 = H_1[\ell'/\ell] \) and \( H'_2 = H_2[\ell'/\ell] \), where \( \ell' \notin (\text{dom}(W.\Psi_i) \cup \text{dom}(W.\Psi_2)) \) and \( \ell' \) are mutually disjoint. Let \( \Psi'' = \Psi'[\ell'/\ell] \). Note that \( \Psi \vdash H'_i; \Psi'' \) and boxheap(\( \Psi'' \)).

Let \( W' = W \sqcup (H'_1, H'_2) \).

- Let \( H_{b1} = W(i_{box}).s.H_1 \) and \( H_{b2} = W(i_{box}).s.H_2 \). Note that \( W'(i_{box}).s = (H_{b1} \sqcup H'_1, H_{b2} \sqcup H'_2) \).

- Note that \( W' \in \text{World} \).

- By Lemma 1.5, we have that \( W' \sqsupseteq_{pub} W \).

- Consider arbitrary \( (M_1, M_2) : W \). Let \( M_i = (H_{Mi}, R_i, S_i) \). Note that \( \text{dom}(H_{Mi}) = \text{dom}(W.\Psi_i) \), which implies \( \ell' \notin \text{dom}(H_{Mi}) \). Also note that

\[
\langle (H_{Mi}, R_i, S_i) | (\rho_i(I_i), H_i) \rangle \mapsto^0 \ldots \mapsto^0 \langle ((H_{Mi}, H'_1), R_{i}, S_i) | (\rho_i(I_i)[\ell'/\ell], \cdot) \rangle
\]

Let \( M'_i = ((H_{Mi}, H'_1), R_{i}, S_i) \). From our choice of \( W' \), it follows that

\[
(W', H_{b1} \sqcup H'_1, H_{b2} \sqcup H'_2) \in \text{currentMR}(W'(i_{box}))
\]

which together with \( (M_1, M_2) : W \) is sufficient to show \( (M'_1, M'_2) : W' \).

- Note that \( W.k \leq W'.k \).

- Note that \( q \neq \epsilon \), which follows from the first hypothesis since it is a side condition of the component typing rules.
• Next, from our last hypothesis, by alpha-renaming locations in dom(Ψ') to dom(Ψ'')—which we can do since ℓ₁ ∉ Ψ, a fact that follows from ℓ₁ ∉ W.Ψ₁ ∪ W.Ψ₂ and W ∈ H[Ψ]—we have the following:

\((Ψ, Ψ'') \vdash Δ; χ; σ; q \vdash I_1[ℓ₁/ℓ] ≈ I_2[ℓ₁/ℓ]\)

We instantiate the above with W' and ρ. Note that W' ∈ H[Ψ, Ψ''] by Lemma 1.22; ρ ∈ D[Δ] from above; currentMR(W'(i_{reg})) ∈ W', R[Δ]ρ by Lemma 1.20 since W'(i_{reg}) = W(i_{reg}); and currentMR(W'(i_{stk})) ∈ W', S[σ]ρ by Lemma 1.21 since W'(i_{stk}) = W(i_{stk}).

Thus, we can conclude that:

\( (W', (p_1(I_1)[ℓ₁/ℓ], \cdot), (p_2(I_2)[ℓ₁/ℓ], \cdot)) ∈ E[q \vdash \text{ret-type}(q, χ, σ)]ρ. \)

Now, the result follows by Lemma 1.11.

□

Lemma 1.24 (Heap Fragment)
Let \( \cdot \vdash ψ_1, \ldots, \cdot \vdash ψ_n \) and Ψ' = \( ι_1: ψ_1, \ldots, ι_n: ψ_n \) such that dom(Ψ) ∩ dom(Ψ') = ∅. If for each i, we have Ψ, Ψ' \vdash h_{i₁} \approx_{hv} h_{2i₁}: ι_i ψ_i, then Ψ \vdash ι_i \mapsto h_{1i₁} \cdots \mapsto h_{1n} \approx_{hv} ι_i \mapsto h_{2i₁} \cdots \mapsto h_{2n}: Ψ'.

Proof

From Ψ, Ψ' \vdash h_{i₁} \approx_{hv} h_{2i₁}: ι_i ψ_i we have Ψ, Ψ' \vdash h_{i₁}: ι_i ψ_i and Ψ, Ψ' \vdash h_{2i₁}: ι_i ψ_i. This means that Ψ \vdash ι_i \mapsto h_{1i₁} \mapsto \cdots \mapsto h_{1n} \approx_{hv} ι_i \mapsto h_{2i₁} \mapsto \cdots \mapsto h_{2n}: Ψ'. Now choose arbitrary ι_i \mapsto ψ_i' ∈ Ψ'. We must show Ψ, Ψ' \vdash H_i(ι_i) \approx_{hv} H_i(ι_i): ι_i ψ_i', where H_i = ι_i \mapsto h_{i₁}, ι_i \mapsto h_{i₂}, \ldots. But note that H_i(ι_i) is h_{i₁}, and from the hypothesis we know that Ψ, Ψ' \vdash h_{i₁} \approx_{hv} h_{2i₁}: ι_i ψ_i', so the result is immediate.

□

Lemma 1.25 (Code Block)
If \( \cdot \vdash ∀[Δ]\{χ; σ\}^q \) and Ψ; Δ; χ; σ; q \vdash I_1 \approx_{hv} I_2, then Ψ \vdash \text{code}[Δ]\{χ; σ\}^q.I_1 \approx_{hv} \text{code}[Δ]\{χ; σ\}^q.I_2 : \text{box}∀[Δ].\{χ; σ\}^q.

Proof

By our hypothesis, Ψ; Δ; χ; σ; q \vdash I_1 and Ψ; Δ; χ; σ; q \vdash I_2, so we have

Ψ \vdash \text{code}[Δ]\{χ; σ\}^q.I_1 : \text{box}∀[Δ].\{χ; σ\}^q \quad \text{and} \quad Ψ \vdash \text{code}[Δ]\{χ; σ\}^q.I_2 : \text{box}∀[Δ].\{χ; σ\}^q.

Let W ∈ H[Ψ]. We need to show that

\( (W, \text{code}[Δ]\{χ; σ\}^q.I_1, \text{code}[Δ]\{χ; σ\}^q.I_2) ∈ H[∀[Δ].\{χ; σ\}^q[0].\]

Let W' ⊇ W, ρ ∈ D[Δ] such that currentMR(W'(i_{reg})) ∈ W', R[Δ]ρ and currentMR(W'(i_{stk})) ∈ W', S[σ]ρ. We need to show that (W', (p_1(I_1), \cdot), (p_2(I_2), \cdot)) ∈ E[q \vdash \text{ret-type}(q, χ, σ)]ρ.

By Lemma 1.9, W', ρ ∈ H[Ψ]. We can instantiate our hypothesis with W' and ρ to get the result.

□

Lemma 1.26 (Tuple)
If for each i, we have Ψ; \vdash w_{i₁} \approx_{w} w_{2i₁}: τ_i, then Ψ \vdash (w_{1₀}, \ldots, w_{1n}) \approx_{hv} (w_{2₀}, \ldots, w_{2n}): ι'(τ_{0}, \ldots, τ_{n}).

Proof

Clearly, Ψ \vdash (w_{1₀}, \ldots, w_{1n}) : ι'(τ_{0}, \ldots, τ_{n}) and Ψ \vdash (w_{2₀}, \ldots, w_{2n}) : ι'(τ_{0}, \ldots, τ_{n}).

Let W ∈ H[Ψ]. We need to show that (W, (w_{1₀}, \ldots, w_{1n}), (w_{2₀}, \ldots, w_{2n})) ∈ H[∀[τ_{0}, \ldots, τ_{n}][0]. But this only requires that for each i, (W, w_{i₁}, w_{2i₁}) ∈ W[τ_i][0], and this holds by our hypothesis.

□

Lemma 1.27 (Unit)
Ψ; Δ; () \approx_{w} (): \text{unit}.

Proof

Immediate, by definition of W[\text{unit}]ρ.

□
Lemma 1.28 (Integer)
Ψ; Δ ⊨ n ≈_w n : int.

Proof
Immediate, by definition of \( W[\text{int}] \).

Lemma 1.29 (Mutable Location)
If \( ℓ : \text{ref} \psi ∈ Ψ \), then \( Ψ; Δ ⊨ ℓ \approx_w ℓ : \text{ref} \psi \).

Proof
By our hypothesis, \( Ψ; ⊢ ℓ : \text{ref} \psi \). Let \( W ∈ H[Ψ] \) and \( ρ ∈ D[Δ] \). We need to show that \( (W, ℓ, ℓ) ∈ W[\text{ref} \psi] ρ \). We know that \( \text{ftv}(\psi) = \emptyset \), so by Lemma 1.13, \( W[\text{ref} \psi] ρ = W[\text{box} \psi] ρ \). By definition of \( H[Ψ] \), \( (W, ℓ, ℓ) ∈ W[\text{box} \psi] ρ \), so we are done.

Lemma 1.30 (Immutable Location)
If \( ℓ : \text{box} \psi ∈ Ψ \), then \( Ψ; Δ ⊨ ℓ \approx_w ℓ : \text{box} \psi \).

Proof
By our hypothesis, \( Ψ; ⊢ ℓ : \text{box} \psi \). Let \( W ∈ H[Ψ] \) and \( ρ ∈ D[Δ] \). We need to show that \( (W, ℓ, ℓ) ∈ W[\text{box} \psi] ρ \). We know that \( \text{ftv}(\psi) = \emptyset \), so by Lemma 1.13, \( W[\text{box} \psi] ρ = W[\text{box} \psi] ρ \). By definition of \( H[Ψ] \), \( (W, ℓ, ℓ) ∈ W[\text{box} \psi] ρ \), so we are done.

Lemma 1.31 (Pack)
If \( Ψ; Δ ⊨ w_1 \approx_w w_2 : τ' / α \), then \( Ψ; Δ ⊨ \text{pack}(τ', w_1) \) as \( ⇒_w \text{pack}(τ', w_2) \) as \( ⇒_w \).

Proof
We have \( Ψ; Δ ⊨ \text{pack}(τ', w_1) \) as \( ⇒_w \) and \( Ψ; Δ \) as \( ⇒_w \) by our hypothesis and the typing rules.
Let \( W ∈ H[Ψ] \) and \( ρ ∈ D[Δ] \). We need to show that
\[
(W, ρ_1(\text{pack}(τ', w_1)) \) as \( ⇒_w (\text{pack}(τ', w_2)) \) as \( ⇒_w (\text{def} \Psi \alpha τ) \).
\]
By our hypothesis, \( (W, w_1, w_2) ∈ W[τ' / α] ρ \). By Lemma 1.14,
\[
W[τ' / α] ρ = W[τ] ρ ⇒ (ρ_1(τ'), ρ_2(τ'), W[τ' / α] ρ).
\]
Thus we can complete the proof by supplying \( W[τ'] ρ \) as the relation required by \( W[⇒_w] ρ \).

Lemma 1.32 (Fold)
If \( Ψ; Δ ⊨ w_1 \approx_w w_2 : τ[μα.τ / α] \), then \( Ψ; Δ ⊨ \text{fold}_{μα.τ} w_1 \approx_w \text{fold}_{μα.τ} w_2 : μα.τ \).

Proof
We have \( Ψ; Δ \) as \( ⇒_w \) by our hypothesis and the typing rules.
Let \( W ∈ H[Ψ] \) and \( ρ ∈ D[Δ] \). We need to show that
\[
(W, ρ_1(w_1), ρ_2(w_2)) ∈ W[τ[μα.τ / α]] ρ.
\]
By our hypothesis, \( (W, ρ_1(w_1), ρ_2(w_2)) ∈ W[τ[μα.τ / α]] ρ \). The result follows by monotonicity.

Lemma 1.33 (Word Type Application)
If \( Ψ; Δ ⊨ w_1 \approx_w w_2 : \text{box} \\forall[α, Δ'], \{ \chi; σ \}^q \) and \( Δ \) as \( ⇒_w \), then
\[
Ψ; Δ ⊨ w_1[τ] \approx_w w_2[τ] : \text{box} \\forall[Δ'], \{ \chi[τ / α]; σ[τ / α] \}^q[τ / α].
\]

Proof
From the hypotheses, we have $\Psi; \Delta \vdash w_i[\tau] : \text{box} \forall[\Delta'].\{\chi[\tau/\alpha]; \sigma[\tau/\alpha]\}^q[\tau/\alpha]$ for $i = 1, \ 2$.

Let $W \in \mathcal{H}[\Psi]$ and $\rho \in D[\Delta]$. We need to show that $(W, \rho_1(w_1[\tau]), \rho_2(w_2[\tau])) \in W[\text{box} \forall[\Delta'].\{\chi[\tau/\alpha]; \sigma[\tau/\alpha]\}^q[\tau/\alpha]] \rho$.

Let $(\tilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))$ such that $\tilde{W} \supseteq W$. Instantiating the first hypothesis with $W$ and $\rho$, noting that $W \in \mathcal{H}[\Psi]$ and $\rho \in D[\Delta]$, we have that

$$(W, \rho_1(w_1), \rho_2(w_2)) \in \mathcal{W}[\text{box} \forall[\alpha, \Delta'].\{\chi; \sigma\}^q] \rho.$$ 

Hence, there must be some $\ell_i$ and $\overline{w_i}$ such that $\rho_1(w_i) = \ell_i[\overline{w_i}]$. Instantiating the above with $(\tilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))$, noting that $\tilde{W} \supseteq W$, we have that $M_i(\ell_i) = \text{code}[\beta_i, \alpha, \Delta']\{\chi_{1}; \sigma_1\}^{\alpha_i}.I_1$, $\rho_1(\chi) = \chi_{1}[\overline{w_i}/\beta_i]$, $\rho_1(\sigma) = \sigma_{1}[\overline{w_i}/\beta_i]$, $\rho_1(q) = q_{1}[\overline{w_i}/\beta_i]$, and

$$(\tilde{W}, \text{code}[\alpha, \Delta']\{\chi_{1}; \sigma_1\}^{\alpha_i}.I_1)[\overline{w_i}/\beta_i], (\text{code}[\alpha, \Delta']\{\chi_2; \sigma_2\}^{\alpha_2}.I_2)[\overline{w_2}/\beta_2]) 
\in \mathcal{H}[\forall[\alpha, \Delta'].\{\chi; \sigma\}^q] \rho. \ (2)$$

From the above equalities, we can conclude $\rho_i(\chi[\tau/\alpha]) = \chi_{1}[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]$, $\rho_i(\sigma[\tau/\alpha]) = \sigma_{1}[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]$, and $\rho_i(q[\tau/\alpha]) = q_{1}[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]$.

It remains to show that

$$(\tilde{W}, \text{code}[\Delta']\{\chi_{1}; \sigma_1\}^{\alpha_i}.I_1)[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha], (\text{code}[\Delta']\{\chi_2; \sigma_2\}^{\alpha_2}.I_2)[\overline{w_2}/\beta_2][\rho_2(\tau)/\alpha]) 
\in \mathcal{H}[\forall[\tau/\alpha].\{\chi[\tau/\alpha]; \sigma[\tau/\alpha]\}^q[\tau/\alpha]] \rho.$$ 

Let $W' \supseteq \tilde{W}$, $\rho^* \in D[\Delta']$, $\rho^\prime = \rho \cup \rho^*$ such that currentMR($W'(i_{\text{reg}})$) $\in W'$, $\mathcal{R}[\forall[\tau/\alpha]] \rho'$ and currentMR($W'(i_{\text{stk}})$) $\in W'$, $\mathcal{S}[\forall[\tau/\alpha]] \rho'$. We need to show that

$$(W', \rho_1^*(I_1[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]), \rho_2^*(I_2[\overline{w_2}/\beta_2][\rho_2(\tau)/\alpha])) \in \mathcal{E}[q[\tau/\alpha] \vdash \text{ret-type}(q[\tau/\alpha], \chi[\tau/\alpha], \sigma[\tau/\alpha])] \rho'.$$

Next, we instantiate (5) with $W'$, $\rho^1 = \rho^*(\alpha \rightarrow \rho_1(\tau), \rho_2(\tau), W[\tau] \rho)$, and $\tau'$; $\sigma'$ = ret-type(q, $\chi$, $\sigma$), noting that $W' \supseteq \tilde{W}$ and $\rho^1 \in D[\alpha, \Delta']$, the latter since $W[\tau] \rho \in \text{WvalRel}[/\rho_1(\tau), \rho_2(\tau)]$, by Lemma 1.17. Let $\rho'' = \rho \circ \rho^1$. Note that $\rho'' = \rho'_1(\alpha \rightarrow (\rho_1(\tau), \rho_2(\tau), W[\tau] \rho)) = \rho'_1(\alpha \rightarrow (\rho'_1(\tau), \rho'_2(\tau), W[\tau] \rho'))$.

Using Lemma 1.14 we have that $\mathcal{R}[\forall[\tau/\alpha]] \rho'' = \mathcal{R}[\forall[\chi]] \rho''$ and $\mathcal{S}[\forall[\tau/\alpha]] \rho'' = \mathcal{R}[\forall[\sigma]] \rho''$. Therefore, note that currentMR($W'(i_{\text{reg}})$) $\in W'$, $\mathcal{R}[\forall[\tau/\alpha]] \rho''$ and currentMR($W'(i_{\text{stk}})$) $\in W'$, $\mathcal{S}[\forall[\sigma]] \rho''$. Hence, it follows that

$$(W', \rho_1^*(I_1[\overline{w_i}/\beta_i]), \rho_2^*(I_2[\overline{w_2}/\beta_2])) \in \mathcal{E}[q \vdash \tau '; \sigma'] \rho''.$$ 

Note that the above is equivalent to

$$(W', \rho_1^*(I_1[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]), \rho_2^*(I_2[\overline{w_2}/\beta_2][\rho_2(\tau)/\alpha])) \in \mathcal{E}[q \vdash \tau '; \sigma'] \rho''.$$ 

By Lemma 1.14, we have that

$$(W', \rho_1^*(I_1[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]), \rho_2^*(I_2[\overline{w_2}/\beta_2][\rho_2(\tau)/\alpha])) \in \mathcal{E}[q[\tau/\alpha] \vdash \tau'[\tau/\alpha]; \sigma'[\tau/\alpha]] \rho'.$$

Finally, note that by the definition of ret-type, since ret-type(q, $\chi$, $\sigma$) = $\tau'$; $\sigma'$, it follows that ret-type(q[\tau/\alpha], $\chi[\tau/\alpha], $\sigma[\tau/\alpha]$) = $\tau'[\tau/\alpha]; \sigma'[\tau/\alpha]$, which allows us to conclude:

$$(W', \rho_1^*(I_1[\overline{w_i}/\beta_i][\rho_1(\tau)/\alpha]), \rho_2^*(I_2[\overline{w_2}/\beta_2][\rho_2(\tau)/\alpha])) \in \mathcal{E}[q[\tau/\alpha] \vdash \text{ret-type}(q[\tau/\alpha], \chi[\tau/\alpha], \sigma[\tau/\alpha])] \rho'.$$

\begin{lemma}[Stack Type Application]
If $\Psi; \Delta \vdash w_1 \approx w_2 : \text{box} \forall[\Delta'].\{\chi; \sigma\}^q$ and $\Delta' \vdash \sigma'$, then

$$\Psi; \Delta \vdash w_1[\sigma'] \approx w_2[\sigma'] : \text{box} \forall[\Delta'].\{\chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta]\}^q[\sigma'/\zeta].$$
\end{lemma}

\end{document}
Proof

From the hypotheses, we have $\Psi; \Delta \vdash w_i[\sigma'] : \forall \Delta'. \{ \chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta] \}^{q[\sigma'/\zeta]}$ for $i = 1, i = 2$.
Let $W \in H[\Psi]$ and $\rho \in D[\Delta]$. We need to show that

$$(W, \rho_1(w_1[\sigma']), \rho_2(w_2[\sigma'])) \in W[\square[\Delta']. \{ \chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta] \}^{q[\sigma'/\zeta]}]$$

Let $(\overline{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))$ such that $\overline{W} \supseteq W$. Instantiating the first hypothesis with $W$ and $\rho$, noting that $W \in H[\Psi]$ and $\rho \in D[\Delta]$, we have that

$$(W, \rho_1(w_1), \rho_2(w_2)) \in W[\square[\zeta, \Delta']. \{ \chi; \sigma \}^{q}]$$

Hence, there must be some $\ell_i$ and $\overline{w_i}$ such that $\rho_1(w_i) = \ell_i[\overline{w_i}]$. Instantiating the above with $(\overline{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))$, noting that $\overline{W} \supseteq W$, we have that $M_i(\ell_i) = \text{code}[\overline{\beta_i}, \zeta, \Delta'] \{ \chi_i; \sigma_i \} \{ \rho \}$. Let $\kappa = \chi[\overline{w_i}/\beta_i], \rho_1(\sigma) = \sigma[\overline{w_i}/\beta_i], \rho_1(\sigma) = q_i[\overline{w_i}/\beta_i]$, and

$$(\overline{W}, \text{code}[\zeta, \Delta'] \{ \chi_1; \sigma_1 \} \{ \rho \}, \text{code}[\zeta, \Delta'] \{ \chi_2; \sigma_2 \} \{ \rho \}) \in H[\forall[\zeta, \Delta']. \{ \chi; \sigma \}^{q}]$$

From the above equalities, we can conclude $\rho_1(\chi[\sigma'/\zeta]) = \chi[\overline{w_i}/\beta_i][\rho_1(\sigma'/\zeta)], \rho_1(\sigma'/\zeta) = \sigma_1[\overline{w_i}/\beta_i][\rho_1(\sigma'/\zeta)],$ and $\rho_1(q[\sigma'/\zeta]) = q_i[\overline{w_i}/\beta_i][\rho_1(\sigma'/\zeta)]$.

It remains to show that

$$(\overline{W}, \text{code}[\Delta'] \{ \chi_1; \sigma_1 \} \{ \rho \}, \text{code}[\Delta'] \{ \chi_2; \sigma_2 \} \{ \rho \}) \in H[\forall[\Delta']. \{ \chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta] \}^{q[\sigma'/\zeta]}]$$

Let $W' \supseteq \overline{W}, \rho' \in D[\Delta'], \rho'' = \rho \cup \rho''$ such that currentMR($W'(i_{\text{reg}}) \subseteq W', \mathcal{R}[\chi[\sigma'/\zeta]]$$
and currentMR($W'(i_{\text{stk}})) \subseteq W', S[\sigma[\sigma'/\zeta]]$$
\rho'$. We need to show that

$$(W', \rho_1(I_1[\overline{w_i}/\beta_i]), \rho_2(I_2[\overline{w_i}/\beta_i])) \in \varepsilon[[q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta]]]$$

Next, we instantiate (6) with $W', \tau'; \sigma'' = \text{ret-type}(q, \chi, \sigma)$, and $\rho[l] = \rho''[\sigma] \Rightarrow (\rho_1(\sigma'), \rho_2(\sigma'), S[\sigma][\rho])$, noting that $W' \supseteq \overline{W}$ and $\rho' \in D[\zeta, \Delta']$, the latter since $S[\sigma][\rho] \in \text{StackRel}(\rho_1(\sigma'), \rho_2(\sigma'))$ by Lemma 1.18. Let $\rho'' = \rho'' \rho'$. Note that $\rho''[\sigma] = \rho[l] \Rightarrow (\rho_1(\sigma'), \rho_2(\sigma'), S[\sigma][\rho])$. Using Lemma 1.15 we have that $\mathcal{R}[\chi[\sigma'/\zeta]]$$
and $S[\sigma[\sigma'/\zeta]]\rho' = \mathcal{R}[\sigma][\rho''].$ Therefore, note that currentMR($W'(i_{\text{reg}})) \subseteq W', \mathcal{R}[\chi[\sigma'/\zeta]]$$
and currentMR($W'(i_{\text{stk}})) \subseteq W', S[\sigma][\rho']$. Hence, it follows that

$$(W', \rho_1(I_1[\overline{w_i}/\beta_i]), \rho_2(I_2[\overline{w_i}/\beta_i])) \in \varepsilon[[q \vdash \tau'; \sigma'']]$$

Note that the above is equivalent to

$$(W', \rho_1(I_1[\overline{w_i}/\beta_i]), \rho_2(I_2[\overline{w_i}/\beta_i])) \in \varepsilon[[q \vdash \tau'; \sigma'']]$$

By Lemma 1.15, we have that

$$(W', \rho_1(I_1[\overline{w_i}/\beta_i]), \rho_2(I_2[\overline{w_i}/\beta_i])) \in \varepsilon[[q[\sigma'/\zeta] \vdash \tau'; \sigma'[\sigma'/\zeta]]$$

Finally, note that by the definition of ret-type, since ret-type($q, \chi, \sigma = \tau'; \sigma''$, it follows that ret-type($q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta] = \tau'[\sigma'/\zeta]; \sigma'[\sigma'/\zeta]$, which allows us to conclude:

$$(W', \rho_1(I_1[\overline{w_i}/\beta_i]), \rho_2(I_2[\overline{w_i}/\beta_i])) \in \varepsilon[[q[\sigma'/\zeta] \vdash \text{ret-type}(q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta])]]$$

$\square$
Lemma 1.35 (Return Marker Type Application)
If $\Psi; \Delta \vdash w_1 \approx_w w_2 : \mathsf{box} \forall [\epsilon, \Delta']. \{\chi_1; \sigma_1\}^q$, $\mathsf{ftv}(q') \subseteq \Delta$, and $\Delta \vdash \forall [\Delta']. \{\chi[q'/\epsilon], \sigma[q'/\epsilon]\}^q$, then
\[ \Psi; \Delta \vdash w_1[q'] \approx_w w_2[q'] : \mathsf{box} \forall [\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q. \]

Proof
From the hypotheses, we have $\Psi; \Delta \vdash w_1[q']: \mathsf{box} \forall [\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q$ for $i = 1, i = 2$.
Let $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$. We need to show that
\[ (W, \rho_1(w_1[q']), \rho_2(w_2[q'])) \in \mathcal{W}[\mathsf{box} \forall [\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q]. \]
Let $(\tilde{W}, M_1, M_2) \in \mathsf{currentMR}(W(i\mathsf{box}))$ such that $\tilde{W} \supseteq W$. Instantiating the first hypothesis with $W$ and $\rho$, noting that $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$, we have that
\[ (W, \rho_1(w_1), \rho_2(w_2)) \in \mathcal{W}[\mathsf{box} \forall [\epsilon, \Delta']. \{\chi; \sigma\}^q]. \]
Hence, there must be some $\ell_i$ and $\omega_i$ such that $\rho_i(w_i) = \ell_i[\omega_i]$. Instantiating the above with $(\tilde{W}, M_1, M_2) \in \mathsf{currentMR}(W(i\mathsf{box}))$, noting that $\tilde{W} \supseteq W$, we have that $M_i(\ell_i) = \mathsf{code}[\beta_i, \epsilon, \Delta']\{\chi_i; \sigma_i\}^q_1$, $\rho_i(\chi) = \chi[\omega_i/\beta_i]$, $\rho_i(\sigma) = \sigma[\omega_i/\beta_i]$, $\rho_i(q) = q[\omega_i/\beta_i]$, and
\[ (\tilde{W}, (\mathsf{code}[\epsilon, \Delta']\{\chi_1; \sigma_1\}^q_1, \mathsf{I}_1)\{\omega_1/\beta_1\}, (\mathsf{code}[\epsilon, \Delta']\{\chi_2; \sigma_2\}^q_2, \mathsf{I}_2)\{\omega_2/\beta_2\}) \in \mathcal{H}[\forall [\epsilon, \Delta']\{\chi; \sigma\}^q]. \]
From the above equalities, we can conclude $\rho_i(\chi[q'/\epsilon]) = \chi[\omega_i/\beta_i][\rho_i(q')/\epsilon]$, $\rho_i(\sigma[q'/\epsilon]) = \sigma[\omega_i/\beta_i][\rho_i(q')/\epsilon]$, and $\rho_i(q[q'/\epsilon]) = q[\omega_i/\beta_i][\rho_i(q')/\epsilon]$.
It remains to show that
\[ (\tilde{W}, (\mathsf{code}[\Delta']\{\chi_1; \sigma_1\}^q_1, \mathsf{I}_1)\{\omega_1/\beta_1\}[\rho_1(q')/\epsilon], (\mathsf{code}[\Delta']\{\chi_2; \sigma_2\}^q_2, \mathsf{I}_2)\{\omega_2/\beta_2\}[\rho_2(q')/\epsilon]) \in \mathcal{H}[\forall [\epsilon, \Delta']\{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q]. \]
Let $W' \supseteq \tilde{W}$, $\rho^* \in \mathcal{D}[\Delta']$, $\rho' = \rho \cup \rho^*$ such that $\mathsf{currentMR}(W'(i\mathsf{reg})) \in W'$, $\mathcal{R}[\chi[q'/\epsilon]]\rho'$ and $\mathsf{currentMR}(W'(i\mathsf{stk})) \in W'$, $\mathcal{S}[\sigma[q'/\epsilon]]\rho'$. We need to show that
\[ (W', \rho_1^*(I_1[\omega_1/\beta_1][\rho_1(q')/\epsilon]), \rho_2^*(I_2[\omega_2/\beta_2][\rho_2(q')/\epsilon])) \in \mathcal{E}[q[q'/\epsilon] \vdash \mathsf{ret-type}(q[q'/\epsilon], \chi[q'/\epsilon], \sigma[q'/\epsilon])]\rho'. \]
Next, we instantiate (7) with $W'$, $\tau'$: $\sigma'' = \mathsf{ret-type}(q, \chi, \sigma)$, and $\rho^i = \rho^*[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))]$, noting that $W' \supseteq \tilde{W} \supseteq W$ and $\rho^i \in \mathcal{D}[\epsilon, \Delta']$, the latter since the hypothesis specifies $\mathsf{ftv}(q') \subseteq \Delta$ and thus $\mathsf{ftv}(\rho_1(q')) = \emptyset$. Let $\rho'' = \rho \cup \rho^i$. Note that $\rho'' = \rho'[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \rho'[\epsilon \mapsto (\rho_1'(q'), \rho_2'(q'))]$. Using Lemma 1.16 we have that $\mathcal{R}[\chi[q'/\epsilon]]\rho'' = \mathcal{R}[\chi]\rho''$ and $\mathcal{S}[\sigma[q'/\epsilon]]\rho'' = \mathcal{S}[\sigma]\rho''$. Therefore, note that $\mathsf{currentMR}(W'(i\mathsf{reg})) \in W'$, $\mathcal{R}[\chi]\rho''$ and $\mathsf{currentMR}(W'(i\mathsf{stk})) \in W'$, $\mathcal{S}[\sigma]\rho''$. Hence, it follows that
\[ (W', \rho_1^*(I_1[\omega_1/\beta_1][\rho_1(q')/\epsilon]), \rho_2^*(I_2[\omega_2/\beta_2][\rho_2(q')/\epsilon])) \in \mathcal{E}[q[q'/\epsilon] \vdash \tau'; \sigma'']\rho''. \]
Note that the above is equivalent to
\[ (W', \rho_1^*(I_1[\omega_1/\beta_1][\rho_1(q')/\epsilon]), \rho_2^*(I_2[\omega_2/\beta_2][\rho_2(q')/\epsilon])) \in \mathcal{E}[q[q'/\epsilon] \vdash \tau'; \sigma'']\rho''. \]
By Lemma 1.16, we have that
\[ (W', \rho_1^*(I_1[\omega_1/\beta_1][\rho_1(q')/\epsilon]), \rho_2^*(I_2[\omega_2/\beta_2][\rho_2(q')/\epsilon])) \in \mathcal{E}[q[q'/\epsilon] \vdash \tau'[q'/\epsilon]; \sigma'[q'/\epsilon]]\rho'. \]
Finally, note that by the definition of ret-type, since ret-type$(q, \chi, \sigma) = \tau'; \sigma'$, it follows that ret-type$(q[q'/\epsilon], \chi[q'/\epsilon], \sigma[q'/\epsilon]) = \tau'[q'/\epsilon]; \sigma'[q'/\epsilon]$, which allows us to conclude:
\[ (W', \rho_1^*(I_1[\omega_1/\beta_1][\rho_1(q')/\epsilon]), \rho_2^*(I_2[\omega_2/\beta_2][\rho_2(q')/\epsilon])) \in \mathcal{E}[q[q'/\epsilon] \vdash \mathsf{ret-type}(q[q'/\epsilon], \chi[q'/\epsilon], \sigma[q'/\epsilon])\rho'. \]
\[ \square \]
Lemma 1.36 (Word Value)
If $\Psi; \Delta \vdash w_1 \approx_w w_2 : \tau$, then $\Psi; \Delta; \chi \vdash w_1 \approx_u w_2 : \tau$.

Proof
We need to first show that $\Psi; \Delta; \chi \vdash w_1 : \tau$ and $\Psi; \Delta; \chi \vdash w_2 : \tau$.
From the hypothesis, we know that $\Psi; \Delta \vdash w_1 : \tau$ and $\Psi; \Delta \vdash w_2 : \tau$, and the typing rule for small values yields the above.
Next, consider arbitrary $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$ such that currentMR($W(i_{reg})$) \in $W[\chi]\rho$. We need to show that $(W, W.\hat{R}_1(\rho_1(w_1))), W.\hat{R}_2(\rho_2(w_2))) \in W[\tau]\rho$.
Instantiating the hypothesis with $W$ and $\rho$, noting that $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$, we have that $(W, \rho_1(w_1), \rho_2(w_2)) \in W[\tau]\rho$. Note that $w_1$ and $w_2$ are not registers, which means that $\rho_i(w_i)$ are not registers. Hence, $W.\hat{R}_1(\rho_i(w_i)) = \rho_i(w_i)$, which means our proof obligation is immediate from $(W, \rho_1(w_1), \rho_2(w_2)) \in W[\tau]\rho$.

Lemma 1.37 (Register)
If $r : \tau \in \chi$, then $\Psi; \Delta; \chi \vdash r \approx_u r : \tau$.

Proof
Consider arbitrary $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$ such that currentMR($W(i_{reg})$) \in $W[\chi]\rho$.
We know $\rho_i(r) = r$, which means we must show
$$(W, W.\hat{R}_1(r), \hat{R}_2(r)) = (W, W.\hat{R}_1(r), \hat{R}_2(r)) \in W[\tau]\rho$$
Since currentMR($W(i_{reg})$) \in $W[\chi]\rho$ and $r : \tau \in \chi$, the definition of $W[\tau]\rho$ yields the result.

Lemma 1.38 (Pack)
If $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2 : \tau[\tau'/\alpha]$, then $\Psi; \Delta; \chi \vdash \text{pack}(\tau', u_1) \equiv \exists \alpha. \tau \equiv_u \text{pack}(\tau', u_2) \equiv \exists \alpha. \tau$.

Proof
We have $\Psi; \Delta; \chi \vdash \text{pack}(\tau', u_1) \equiv \exists \alpha. \tau$ and $\Psi; \Delta; \chi \vdash \text{pack}(\tau', u_2) \equiv \exists \alpha. \tau$ by our hypothesis and the typing rules.
Let $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$, where currentMR($W(i_{reg})$) \in $W[\chi]\rho$. We need to show that
$$(W, W.\hat{R}_1(\rho_1(\text{pack}(\tau', u_1))) \equiv \exists \alpha. \tau), W.\hat{R}_2(\rho_2(\text{pack}(\tau', u_2))) \equiv \exists \alpha. \tau)) \in W[\exists \alpha. \tau]$$
which is equivalent to
$$(W, \rho_1(\text{pack}(\tau', W.\hat{R}_1(u_1))) \equiv \exists \alpha. \tau), \rho_2(\text{pack}(\tau', W.\hat{R}_2(u_2))) \equiv \exists \alpha. \tau)) \in W[\exists \alpha. \tau]$$
By instantiating our hypothesis with $W$ and $\rho$, we have that $(W, W.\hat{R}_1(u_1), W.\hat{R}_2(u_2)) \in W[\tau[\tau'/\alpha]]\rho$.
By Lemma 1.14,
$$W[\tau[\tau'/\alpha]]\rho = W[\tau][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau')), \rho[\tau']\rho]].$$
By Lemma 1.17, $W[\tau']\rho \in \text{WValRel}[(\rho_1(\tau'), \rho_2(\tau'))].$ Thus we can complete the proof by supplying $W[\tau']\rho$ as the relation required by $W[\exists \alpha. \tau]\rho$.

Lemma 1.39 (Fold)
If $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2 : \tau[\mu \alpha. \tau/\alpha]$, then $\Psi; \Delta; \chi \vdash \text{fold}_{\mu \alpha. \tau} u_1 \approx_u \text{fold}_{\mu \alpha. \tau} u_2 : \mu \alpha. \tau$.

Proof
We have \( \Psi; \Delta; \chi \vdash \text{fold}_{\mu \alpha. \tau} u_1; \mu \alpha. \tau \) and \( \Psi; \Delta; \chi \vdash \text{fold}_{\mu \alpha. \tau} u_2; \mu \alpha. \tau \) by our hypothesis and the typing rules.

Let \( W \in \mathcal{H}[\Psi] \) and \( \rho \in \mathcal{D}[\Delta] \) where currentMR(\( W(i_{\text{reg}}) \)) \( \in W \mathcal{R}[\chi] \rho \).

We need to show that

\[
(W, W \hat{R}_1(\rho_1(\text{fold}_{\mu \alpha. \tau} u_1)), W \hat{R}_2(\rho_2(\text{fold}_{\mu \alpha. \tau} u_2))) \in W[\mu \alpha. \tau] \rho.
\]

This proof goal is equivalent to

\[
(W, \text{fold}_{\rho_1(\mu \alpha. \tau)} W \hat{R}_1(\rho_1(u_1)), \text{fold}_{\rho_2(\mu \alpha. \tau)} W \hat{R}_2(\rho_2(u_2))) \in W[\mu \alpha. \tau] \rho.
\]

To establish this goal, it is sufficient to establish the following subgoal,

\[
(W, W \hat{R}_1(\rho_1(u_1)), W \hat{R}_2(\rho_2(u_2))) \in \triangleright W[\tau[\mu \alpha. \tau/\alpha]] \rho.
\]

Using \( W \) and \( \rho \) from unrolling our hypothesis, \( \Psi; \Delta; \chi \vdash u_1 \approx_{u_2 : \text{box} \forall[\alpha, \Delta'] \chi; \sigma} \) and \( \Delta \vdash \tau \), we obtain

\[
(W, W \hat{R}_1(\rho_1(u_1)), W \hat{R}_2(\rho_2(u_2))) \in W[\tau[\mu \alpha. \tau/\alpha]] \rho.
\]

We derive the desired subgoal from the above conclusion and lemmas 1.6 and 1.8.

\[\square\]

**Lemma 1.40 (Word Type Application)**

If \( \Psi; \Delta; \chi \vdash u_1 \approx_{u_2 : \text{box} \forall[\alpha, \Delta'] \chi; \sigma} \) and \( \Delta \vdash \tau \), then

\[\Psi; \Delta; \chi \vdash u_1[\tau] \approx_{u_2[\tau] : \text{box} \forall[\Delta'] \chi[\tau/\alpha]; \sigma[\tau/\alpha]} \]

**Proof**

From the hypotheses, we have \( \Psi; \Delta; \chi \vdash u_1[\tau] : \text{box} \forall[\Delta'] \chi[\tau/\alpha]; \sigma[\tau/\alpha]} \) for \( i = 1, i = 2 \).

Let \( W \in \mathcal{H}[\Psi] \) and \( \rho \in \mathcal{D}[\Delta] \), where currentMR(\( W(i_{\text{reg}}) \)) \( \in W \mathcal{R}[\chi] \rho \).

We need to show that

\[
(W, \hat{R}_1(\rho_1(u_1[\tau])), \hat{R}_1(\rho_2(u_2[\tau]))) \in W[\text{box} \forall[\alpha, \Delta'] \chi[\tau/\alpha]; \sigma[\tau/\alpha]} \rho.
\]

Let \( (\hat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})) \) such that \( \hat{W} \supseteq W \). Instantiating the first hypothesis with \( W \) and \( \rho \), noting that \( W \in \mathcal{H}[\Psi] \) and \( \rho \in \mathcal{D}[\Delta] \), we have that

\[
(W, \hat{R}_1(\rho_1(u_1))), \hat{R}_1(\rho_2(u_2))) \in W[\text{box} \forall[\alpha, \Delta'] \chi[\tau/\alpha]; \sigma[\tau/\alpha]} \rho.
\]

Hence, there must be some \( \ell_i \) and \( \omega_i \) such that \( \hat{R}_1(\rho_i(u_1)) = \ell_i[\omega_i] \). Instantiating the above with \( (\hat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})) \), noting that \( \hat{W} \supseteq W \), we have that \( M_i(\ell_i) = \text{code}[\hat{\beta}_i, \alpha, \Delta'] \chi_i[\sigma_1] \omega_i, I_i \)

\[
\rho_i(\chi) = \chi_i[\omega_i/\beta_i], \rho_i(\sigma) = \sigma_i[\omega_i/\beta_i], \rho_i(q) = q_i[\omega_i/\beta_i],
\]

\[
(\hat{W}, \text{code}[\alpha, \Delta'][\chi_1; \sigma_1] \omega_1, I_1)[\omega_i/\beta_i], (\text{code}[\alpha, \Delta'][\chi_2; \sigma_2] \omega_2, I_2)[\omega_i/\beta_i] \in \mathcal{H}[\forall[\alpha, \Delta'] \chi[\tau/\alpha]} \rho. \tag{5}
\]

From the above equalities, we can conclude \( \rho_i(\chi[\tau/\alpha]) = \chi_i[\omega_i/\beta_i][\rho_i(\tau/\alpha)], \rho_i(\sigma[\tau/\alpha]) = \sigma_i[\omega_i/\beta_i][\rho_i(\tau/\alpha)], \) and \( \rho_i(q[\tau/\alpha]) = q_i[\omega_i/\beta_i][\rho_i(\tau/\alpha)]. \)

It remains to show that

\[
(\hat{W}, \text{code}[\Delta'][\chi_1; \sigma_1] \omega_1, I_1)[\omega_i/\beta_i][\rho_1(\tau/\alpha)], (\text{code}[\Delta'][\chi_2; \sigma_2] \omega_2, I_2)[\omega_i/\beta_i][\rho_2(\tau/\alpha)] \in \mathcal{H}[\forall[\Delta'] \chi[\tau/\alpha]; \sigma[\tau/\alpha]} \rho.
\]
Let \( W' \supseteq \tilde{W}, \rho^* \in D[\Delta'], \rho' = \rho \cup \rho^* \) such that currentMR\( (W'(i_{reg})) \in W' \\mathcal{R}\[\tau/\alpha]\rho' \) and currentMR\( (W'(i_{stk})) \in W' \mathcal{S}[\sigma[\tau/\alpha]]\rho' \). We need to show that

\[
(W', \rho^*_1(1_1[\omega_1/\beta_1][\rho_1(\tau)/\alpha]), \rho^*_2(1_2[\omega_2/\beta_2][\rho_2(\tau)/\alpha]) \in \mathcal{E}[q[\tau/\alpha] \vdash \text{ret-type}(q[\tau/\alpha], \chi[\tau/\alpha], \sigma[\tau/\alpha])\rho'].
\]

Next, we instantiate (5) with \( W', \tau', \sigma' = \text{ret-type}(q, \chi, \sigma) \) and \( \rho^* = \rho^*[\alpha \mapsto (\rho_1(\tau), \rho_2(\tau), W[\tau]\rho)], \) noting that \( W' \supseteq W \) and that \( \rho^* \in D[\alpha, \Delta'] \), the latter since \( W[\tau]\rho \in \text{VwalRel}(\rho_1(\tau), \rho_2(\tau)] \) by Lemma 1.17. Let \( \rho' = \rho \cup \rho^* \). Note that \( \rho' = \rho'[\alpha \mapsto (\rho'_1(\tau), \rho'_2(\tau), W[\tau]\rho)] \). Using Lemma 1.14 we have that \( \mathcal{R}[\tau/\alpha]\rho' = \mathcal{R}[\chi]\rho'' \) and \( \mathcal{S}[\sigma[\tau/\alpha]]\rho' = \mathcal{S}[\sigma]\rho'' \). Therefore, note that currentMR\( (W'(i_{reg})) \in W' \\mathcal{R}\[\tau/\alpha]\rho'' \) and currentMR\( (W'(i_{stk})) \in W' \mathcal{S}[\sigma]\rho'' \). Hence, it follows that

\[
(W', \rho'_1(1_1[\omega_1/\beta_1]), \rho'_2(1_2[\omega_2/\beta_2]) \in \mathcal{E}[q \vdash \tau'; \sigma']\rho''.
\]

Note that the above is equivalent to

\[
(W', \rho^*_1(1_1[\omega_1/\beta_1][\rho_1(\tau)/\alpha]), \rho^*_2(1_2[\omega_2/\beta_2][\rho_2(\tau)/\alpha]) \in \mathcal{E}[q \vdash \tau'; \sigma']\rho''.
\]

By Lemma 1.14, we have that

\[
(W', \rho^*_1(1_1[\omega_1/\beta_1][\rho_1(\tau)/\alpha]), \rho^*_2(1_2[\omega_2/\beta_2][\rho_2(\tau)/\alpha]) \in \mathcal{E}[q[\tau/\alpha] \vdash \tau'[\tau/\alpha]; \sigma'[\tau/\alpha]]\rho'.
\]

Finally, note that by the definition of ret-type, since \( \text{ret-type}(q, \chi, \sigma) = \tau'; \sigma' \), it follows that

\[
\text{ret-type}(q[\tau/\alpha], \chi[\tau/\alpha], \sigma[\tau/\alpha]) = \tau'[\tau/\alpha]; \sigma'[\tau/\alpha], \]

which allows us to conclude:

\[
(W', \rho^*_1(1_1[\omega_1/\beta_1][\rho_1(\tau)/\alpha]), \rho^*_2(1_2[\omega_2/\beta_2][\rho_2(\tau)/\alpha]) \in \mathcal{E}[q[\tau/\alpha] \vdash \text{ret-type}(q[\tau/\alpha], \chi[\tau/\alpha], \sigma[\tau/\alpha])\rho'].
\]

\[\square\]

Lemma 1.41 (Stack Type Application)

If \( \Psi; \Delta; \chi \vdash u_1 \approx_{u_2} \text{box} \forall[\zeta, \Delta'], \{\chi; \sigma\}^q \) and \( \Delta \vdash \sigma' \), then

\[
\Psi; \Delta; \chi \vdash u_1[\sigma'] \approx_u u_2[\sigma'] : \text{box} \forall[\Delta'], \{\chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta]\}^q[\sigma'/\zeta].
\]

Proof

From the hypotheses, we have \( \Psi; \Delta; \chi \vdash u_i[\sigma'] : \text{box} \forall[\Delta'], \{\chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta]\}^q[\sigma'/\zeta] \) for \( i = 1, 2 \).

Let \( W \in H[\Psi] \) and \( \rho \in D[\Delta] \), where currentMR\( (W(i_{reg})) \in W \mathcal{R}\[\chi/\rho]\). We need to show that

\[
(W, \tilde{W}, M_1, M_2) \in \text{currentMR}\( (W(i_{box})) \) such that \( \tilde{W} \supseteq W \). Instantiating the first hypothesis with \( W \) and \( \rho \), noting that \( W \in H[\Psi] \) and \( \rho \in D[\Delta] \), we have that

\[
(W, \tilde{W}, M_1, M_2) \in \text{currentMR}\( (W(i_{box})) \).
\]

Let \( (W, M_1, M_2) \in \text{currentMR}\( (W(i_{box})) \) noting that \( \tilde{W} \supseteq W \), we have that \( M_i(\ell_i) = \text{code}[\beta_1, \Delta'] \{\chi_i; \sigma_i\}^{q_1}I_1, \rho_i(\chi) = \chi_i[\omega_i/\beta_i], \rho_i(\sigma) = \sigma_i[\omega_i/\beta_i], \rho_i(\psi) = q_i[\omega_i/\beta_i], \) and

\[
(W, (\text{code}[\Delta'] \{\chi_1; \sigma_1\}^{q_1}I_1)[\omega_1/\beta_1], (\text{code}[\Delta'] \{\chi_2; \sigma_2\}^{q_2}I_2)[\omega_2/\beta_2]) \in H[\forall[\zeta, \Delta'], \{\chi; \sigma\}^q].
\]

From the above equalities, we can conclude \( \rho_i(\psi[\sigma'/\zeta]) = q_i[\omega_i/\beta_i][\rho_i(\psi)'/\zeta], \rho_i(\psi[\sigma'/\zeta]) = \sigma_i[\omega_i/\beta_i][\rho_i(\psi)'/\zeta], \) and \( \rho_i(\psi[\sigma'/\zeta]) = q_i[\omega_i/\beta_i][\rho_i(\psi)'/\zeta]. \)
It remains to show that

\[(\hat{W}, (\text{code}[\Delta']; \chi_1; \sigma_1)_{q_1} \cdot I_1) \cdot [\omega_1/\beta_1][\rho_1(\sigma')/\zeta], (\text{code}[\Delta']; \chi_2; \sigma_2)_{q_2} \cdot I_2) \cdot [\omega_2/\beta_2][\rho_2(\sigma')/\zeta]) \in \mathcal{H}[\forall[\Delta']; \chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta]] q[\sigma'/\zeta] \rho.\]

Let \(W' \supseteq \hat{W}, \rho^* \in \mathcal{D}[[\Delta']], \rho' = \rho \cup \rho^*\) such that \(\text{currentMR}(W'(i_{\text{reg}})) \in W, \mathcal{R}[\chi[\sigma'/\zeta]] \rho'\) and \(\text{currentMR}(W'(i_{\text{stk}})) \in W, S[\sigma[\sigma'/\zeta]] \rho'\). We need to show that

\[(W', \rho^*_1(I_1[\omega_1/\beta_1][\rho_1(\sigma')/\zeta]), \rho^*_2(I_2[\omega_2/\beta_2][\rho_2(\sigma')/\zeta])) \in \mathcal{E}[q[\sigma'/\zeta] \vdash \text{ret-type}(q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta])] \rho'.\]

Next, we instantiate (6) with \(W', \tau'; \sigma'' = \text{ret-type}(q, \chi, \sigma)\), and \(\rho^1 = \rho^* \implies (\rho_1(\sigma'), \rho_2(\sigma'), S[\sigma'][\rho])]\), noting that \(W' \supseteq \hat{W}\) and \(\rho^1 \in \mathcal{D}[\Delta, \Delta']\), the latter since \(S[\sigma'] \rho \in \text{StackRel}(\rho_1(\sigma'), \rho_2(\sigma')]\) by Lemma 1.18. Let \(\rho'' = \rho \cup \rho^1\). Note that \(\rho''(\sigma') = (\rho_1(\sigma'), \rho_2(\sigma'), S[\sigma'][\rho]) = (\rho(\sigma'), \rho(\sigma'), S[\sigma'][\rho'])\). Using Lemma 1.15 we have that \(\mathcal{R}[\chi[\sigma'/\zeta]] \rho'' = \mathcal{R}[\chi] \rho''\) and \(S[\sigma[\sigma'/\zeta]] \rho'' = S[\sigma]' \rho''\). Therefore, note that \(\text{currentMR}(W'(i_{\text{reg}})) \in W, \mathcal{R}[\chi] \rho''\) and \(\text{currentMR}(W'(i_{\text{stk}})) \in W, S[\sigma]' \rho''\). Hence, it follows that

\[(W', \rho^1(I_1[\omega_1/\beta_1][\rho_1(\sigma')/\zeta]), \rho^2(I_2[\omega_2/\beta_2][\rho_2(\sigma')/\zeta])) \in \mathcal{E}[q \vdash \tau'; \sigma'' \rho''],\]

where \(\tau'; \sigma'' = \text{ret-type}(q, \chi, \sigma)\).

Note that the above is equivalent to

\[(W', \rho^*_1(I_1[\omega_1/\beta_1][\rho_1(\sigma')/\zeta]), \rho^*_2(I_2[\omega_2/\beta_2][\rho_2(\sigma')/\zeta])) \in \mathcal{E}[q \vdash \tau'; \sigma'' \rho''].\]

By Lemma 1.15, we have that

\[(W', \rho^*_1(I_1[\omega_1/\beta_1][\rho_1(\sigma')/\zeta]), \rho^*_2(I_2[\omega_2/\beta_2][\rho_2(\sigma')/\zeta])) \in \mathcal{E}[q[\sigma'/\zeta] \vdash \tau'[\sigma'/\zeta]; \sigma''[\sigma'/\zeta]] \rho'.\]

Finally, note that by the definition of ret-type, since \(\text{ret-type}(q, \chi, \sigma) = \tau'; \sigma''\), it follows that \(\text{ret-type}(q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta]) = \tau'[\sigma'/\zeta]; \sigma''[\sigma'/\zeta]\), which allows us to conclude:

\[(W', \rho^*_1(I_1[\omega_1/\beta_1][\rho_1(\sigma')/\zeta]), \rho^*_2(I_2[\omega_2/\beta_2][\rho_2(\sigma')/\zeta])) \in \mathcal{E}[q[\sigma'/\zeta] \vdash \text{ret-type}(q[\sigma'/\zeta], \chi[\sigma'/\zeta], \sigma[\sigma'/\zeta])] \rho'.\]

\[\Box\]

Lemma 1.42 (Return Marker Type Application)

If \(\Psi; \Delta; \chi \vdash u_1 \approx u_2 \cdot \text{box}[\epsilon, \Delta']. \{\chi; \sigma\} q, \text{ftv}(q') \subseteq \Delta, \text{and } \Delta \vdash \forall[\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\} q[q'/\epsilon]\), then

\(\Psi; \Delta; \chi \vdash u_1[q'] \approx u_2[q'] \cdot \text{box} \forall[\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\} q[q'/\epsilon]\).

**Proof**

From the hypotheses, we have \(\Psi; \Delta; \chi \vdash u_1[q'] \approx u_2[q'] \cdot \text{box} \forall[\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\} q[q'/\epsilon]\) for \(i = 1, i = 2\).

Let \(W \in \mathcal{H}[\Psi]\) and \(\rho \in \mathcal{D}[\Delta]\), where \(\text{currentMR}(W(i_{\text{reg}})) \in W, \mathcal{R}[\chi] \rho\).

We need to show that

\[(W, \hat{R}_1(\rho_1(u_1[q'])), \hat{R}_1(\rho_2(u_2[q']))) \in \mathcal{W}[\text{box} \forall[\Delta']. \{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\} q[q'/\epsilon]][\rho].\]

Let \((\hat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{reg}}))\) such that \(\hat{W} \supseteq W\). Instantiating the first hypothesis with \(W\) and \(\rho\), noting that \(W \in \mathcal{H}[\Psi]\) and \(\rho \in \mathcal{D}[\Delta]\), we have that

\[(W, \hat{R}_1(\rho_1(u_1)), \hat{R}_1(\rho_2(u_2))) \in \mathcal{W}[\text{box} \forall[\epsilon, \Delta']. \{\chi; \sigma\} q][\rho].\]

Hence, there must be some \(\ell_i\) and \(\omega\) such that \(\hat{R}_1(\rho_1(u_i)) = \ell_i[\omega]\). Instantiating the above with \((\hat{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{reg}}))\), noting that \(\hat{W} \supseteq W\), we have that \(M_i(\ell_i) = \text{code}[\beta_i, \epsilon, \Delta']. \{\chi_i; \sigma_i\}_{\beta_i} \cdot I_i, \rho_i(\chi) = \chi_i[\omega_i/\beta_i], \rho_i(\sigma) = \sigma_i[\omega_i/\beta_i], \rho_i(q) = q_i[\omega_i/\beta_i]\), and

\[((\hat{W}, \text{code}[\epsilon, \Delta']. \{\chi_1; \sigma_1\}_{\beta_1} \cdot I_1)[\omega_1/\beta_1], \text{code}[\epsilon, \Delta']. \{\chi_2; \sigma_2\}_{\beta_2} \cdot I_2)[\omega_2/\beta_2]) \in \mathcal{H}[\forall[\epsilon, \Delta']. \{\chi; \sigma\} q][\rho].\]
From the above equalities, we can conclude $\rho_i(\chi[q'/\varepsilon]) = \chi_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon], \rho_i(\sigma[q'/\varepsilon]) = \sigma_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon], \rho_i(q[q'/\varepsilon]) = q_i[\omega_1/\beta_1][\rho_1(q')/\varepsilon].$

It remains to show that

$$(\widetilde{W}, (\text{code}[\Delta']\{\chi_1; \sigma_1\}^q, I_1)[\omega_1/\beta_1][\rho_1(q')/\varepsilon], (\text{code}[\Delta']\{\chi_2; \sigma_2\}^q, I_2)[\omega_2/\beta_2][\rho_2(q')/\varepsilon])$$

$$\in \mathcal{H}[\forall \Delta', \chi[q'/\varepsilon]; \sigma[q'/\varepsilon]]^q_{\text{ret-type}}(\rho).$$

Let $W' \sqsupseteq \widetilde{W}$, $\rho^* \in \mathcal{D}[\Delta'], \rho' = \rho \cup \rho^*$ such that currentMR($W'(i_{reg})$) $\in W$, $\mathcal{R}[\chi[q'/\varepsilon]]\rho'$ and currentMR($W'(i_{stk})$) $\in W$, $\mathcal{S}[\sigma[q'/\varepsilon]]\rho'$. We need to show that

$$(W', \rho_1^i(I_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon]), \rho_2^i(I_2[\omega_2/\beta_2][\rho_2(q')/\varepsilon])) \in \mathcal{E}[q[q'/\varepsilon] \vdash \text{ret-type}(q[q'/\varepsilon], \chi[q'/\varepsilon], \sigma[q'/\varepsilon])]/\rho'.$$

Next, we instantiate (7) with $W'$, $\tau'$; $\sigma' = \text{ret-type}(q, \chi, \sigma)$, and $\rho^1 = \rho^*[\varepsilon \mapsto (\rho_1(q'), \rho_2(q'))]$, noting that $W' \sqsupseteq \widetilde{W}$ and $\rho^1 \in \mathcal{D}[\rho, \Delta']$, the latter since the hypothesis specifies $fte(q') \subseteq \Delta$ and thus $fte(\rho_i(q')) = \emptyset$. Let $\rho'' = \rho \cup \rho^1$. Note that $\rho'' = \rho'[\varepsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \rho'[\varepsilon \mapsto (\rho_1(q'), \rho_2(q'))].$

Using Lemma 1.16 we have that $\mathcal{R}[\chi[q'/\varepsilon]]\rho'' = \mathcal{R}[\chi]\rho''$ and $\mathcal{S}[\sigma[q'/\varepsilon]]\rho'' = \mathcal{S}[\sigma]\rho''$. Therefore, note that currentMR($W'(i_{reg})$) $\in W$, $\mathcal{R}[\chi]\rho''$ and currentMR($W'(i_{stk})$) $\in W$, $\mathcal{S}[\sigma]\rho''$. Hence, it follows that

$$(W', \rho_1^i(I_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon]), \rho_2^i(I_2[\omega_2/\beta_2][\rho_2(q')/\varepsilon])) \in \mathcal{E}[q[q'/\varepsilon] \vdash \tau'; \sigma']/\rho''$$

Note that the above is equivalent to

$$(W', \rho_1^i(I_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon]), \rho_2^i(I_2[\omega_2/\beta_2][\rho_2(q')/\varepsilon])) \in \mathcal{E}[q[q'/\varepsilon] \vdash \tau'; \sigma'']/\rho''$$

By Lemma 1.16, we have that

$$(W', \rho_1^i(I_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon]), \rho_2^i(I_2[\omega_2/\beta_2][\rho_2(q')/\varepsilon])) \in \mathcal{E}[q[q'/\varepsilon] \vdash \tau'[q'/\varepsilon]; \sigma'[q'/\varepsilon]]\rho'.$$

Finally, note that by the definition of ret-type, since ret-type($q, \chi, \sigma) = \tau'; \sigma'$, it follows that ret-type($q[q'/\varepsilon], \chi[q'/\varepsilon], \sigma[q'/\varepsilon]) = \tau'[q'/\varepsilon]; \sigma'[q'/\varepsilon]$, which allows us to conclude:

$$(W', \rho_1^i(I_1[\omega_1/\beta_1][\rho_1(q')/\varepsilon]), \rho_2^i(I_2[\omega_2/\beta_2][\rho_2(q')/\varepsilon])) \in \mathcal{E}[q[q'/\varepsilon] \vdash \text{ret-type}(q[q'/\varepsilon], \chi[q'/\varepsilon], \sigma[q'/\varepsilon])]/\rho'.$$

\begin{lemma}[Arithmetic Operation]
If $\Psi; \Delta; \chi \models r_1 \approx u_1 : \text{int}, \Psi; \Delta; \chi \models u_1 \approx u_2 : \text{int}, q \neq r_d$, and $\Psi; \Delta; \chi[r_d : \text{int}]; \sigma; q \models I_1 \approx I_2$, then $\Psi; \Delta; \chi; \sigma; q \vdash \text{aop } r_d; r_{si}; u_1; I_1 \approx I_2 \text{ aop } r_d; r_{si}; u_2; I_2$.

\end{lemma}

\begin{proof}
Inspecting $\approx_{i}$, we see that $\Psi; \Delta; \chi; \sigma; q \vdash \text{aop } r_d; r_{si}; u_i; I_i$ for $i \in \{1, 2\}$.

Now choose arbitrary $W \in \mathcal{H}[\Psi]$ and $\rho \in \mathcal{D}[\Delta]$ such that currentMR($W(i_{reg})$) $\in W$, $\mathcal{R}[\chi]\rho$ and currentMR($W(i_{stk})$) $\in W$, $\mathcal{S}[\sigma]\rho$.

We need to show $(W, \rho_1((\text{aop } r_d, r_{si}, u_1; I_1),.), \rho_2((\text{aop } r_d, r_{si}, u_2; I_2),.)) \in \mathcal{E}[q[q'/\varepsilon] \vdash \text{ret-type}(q[q'/\varepsilon], \chi[q'/\varepsilon], \sigma[q'/\varepsilon])]/\rho$.

Noting that $u_i$ is either an integer or a register pointing to one, we can push the substitution in to yield the following obligation:

$$(W, (\text{aop } r_d, r_{si}, u_1; \rho_1(I_1),.), (\text{aop } r_d, r_{si}, u_2; \rho_2(I_2),.)) \in \mathcal{E}[q[q'/\varepsilon] \vdash \text{ret-type}(q[q'/\varepsilon], \chi[q'/\varepsilon], \sigma[q'/\varepsilon])]/\rho$$

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running($0, (M_i | E_i[c_i])$).

In the following, assume $W.k > 0$ and let $\tau_i; \sigma_i = \text{ret-type}(q, \chi, \sigma)$.

We will prove this by appealing to Lemma 1.11 with an empty heap fragment. In order to do so, we show the following:
• First, we must show that the above triple is in TermAtom[q ⊩ ret-type(q, χ, σ)]ρ. Since W is drawn from H[[Ψ]]ρ, we know it is in World, which means what we have left to show is that:
  \[ W.Ψ_1; W.χ_1; W.χ_2; \rho_1(q) ⊢ \rho_1(e_1) : \rho_1(\text{ret-type}(q, χ, σ)) \]
  \[ W.Ψ_2; W.χ_1; W.χ_2; \rho_2(q) ⊢ \rho_2(e_1) : \rho_2(\text{ret-type}(q, χ, σ)) \]
where \( \rho_1(e_1) \) and \( \rho_2(e_2) \) are the two components under consideration.
  
  From the typing judgement for components, we see that we need to show that
  \[ W.Ψ_1; W.χ_1; W.χ_2; \rho_1(q) ⊢ \text{aop}_r_1 r_1, r_2, u_1; \rho_1(I_1) \]
  and similarly for the second case.
  
  From the hypothesis, we know \( Ψ_1; Ψ_2; χ_1; χ_2; ρ_1(q) ⊢ \text{aop}_r_1 r_1, r_2, u_1; ρ_1(I_1) \). From the fact that \( W \in H[[Ψ]] \), we know that \( W.Ψ_1 \) is a superset of \( Ψ \), and similarly from currentMR(\( W(\text{isreg}) \)) \( ⊆ W.ρ[[χ]]ρ \) we know that \( W.ρ_1 \) contains well typed bindings for everything in \( χ \), and since \( W \) is a world, this means \( W.χ_1 \) is a superset of \( χ \). Similarly, from currentMR(\( W(\text{isstk}) \)) \( ⊆ W.Σ[[σ]]ρ \) we know that \( W.σ_1 \) is a superset of \( σ \), and combining all of these, via weakening, we get what we need.

• Next, we must choose a \( W' \), where \( W' \equiv \text{pub} W \) and \( W.k \leq W'.k + k_1, W.k + k_2 \), where \( k_1 \) and \( k_2 \) will be determined later.
  
  Let \( s = W.R_1[r_d → δ(aop, W.R_1(r_s), W.ŚR_1(u_1))], W.χ_1[r_d : \text{int}], W.R_2[r_d → δ(aop, W.R_2(r_s), W.ŚR_2(u_2))], W.χ_2[r_d : \text{int}] \) and \( W' = W.R_1, W.Ψ_2, W.Θ[i_{\text{reg}} → \text{Island}_{\text{reg}}(s, W.k)] \).
  
  Since \( s.R_1 : s.χ_1, W' ∈ World \) and the rest is unchanged, \( W' \equiv \text{pub} W \).

• Further, we must show that the return address (if it is not \( \text{end}\{τ; σ\} \)) is unchanged in \( W' \). Since \( q \) is the same, the only way that it could have changed would be if it were a memory location that changed, but the only change to memory is \( r_d \) and from the hypothesis \( q \neq r_d \).

• Finally, consider arbitrary \( M_1, M_2 : W \), where \( (H_i, R_i, S_i) = M_i \).
  
  From \( (M_1, M_2) : W \), since \( W.k > 0 \), we have \( (\triangleright W, M_1, M_2) ∈ \bigotimes\{ \text{currentMR}(θ) | θ ∈ W.Θ \} \).
  
  From the latter, we have the following fact for island \( i_{\text{reg}} \): \( (\triangleright W, R_1, R_2) ∈ \text{currentMR}(W(\text{isreg})) \).
  
  From the latter it follows that \( R_1 = W.R_1 \).
  
  The reduction relation tells us that
  \[(H_1, R_1, S_1) | \text{aop} r_d, r_s, u_i; ρ_i(I_i)) \rightarrow^1 \langle(H_1, R_1[r_d → δ(aop, R_1(r_s), ŚR_1(u_i))], S_i) | ρ_i(I_i) \rangle \]
  
  Let \( M' = (H_1, R_1[r_d → δ(aop, R_1(r_s), ŚR_1(u_i))], S_i) \). Based on the definition of \( W' \), we can see that \( (M'_1, M'_2) : W' \). Note that \( k_1 = k_2 = 1 \), and thus \( W.k \leq W'.k + k_i \).
  
  We now instantiate the hypothesis with \( W' \) and \( ρ \). By Lemma 1.9, \( W' \) is in \( H[[Ψ]] \) and \( ρ ∈ D[Δ] \). From Lemma 1.21, since \( W(i_{\text{isreg}}) = W'(i_{\text{isreg}}) \), \( \text{currentMR}(W'(i_{\text{isreg}})) \equiv W[Ψ][σ][ρ] \).
  
  Using Lemma 1.20, we claim that \( \text{currentMR}(W'(i_{\text{isreg}})) \equiv W[χ][r_d : \text{int}][ρ] \). In order to show this, we need that \( (W', δ(aop, R_1(r_s), ŚR_1(u_i)), δ(aop, R_2(r_s), ŚR_2(u_2))) ∈ W[\text{int}][ρ] \).

  Note that from the first two hypothesis, \( W, W.ŚR_1(u_1), W.ŚR_2(u_2) ∈ W[\text{int}][ρ] \), which means \( W.ŚR_1(u_i) \) is an integer, and similarly for \( W.R_1(r_s) \). This means that \( δ(aop, W.R_1(r_s), W.ŚR_1(u_i)) \) is also an integer, and thus we can conclude:

  \[(W, δ(aop, W.R_1(r_s), W.ŚR_1(u_i)), δ(aop, W.R_2(r_s), W.ŚR_2(u_2))) ∈ W[\text{int}][ρ] \]

  By monotonicity, it follows that:

  \[(W', δ(aop, W'.R_1(r_s), W'.ŚR_1(u_1)), δ(aop, W'.R_2(r_s), W'.ŚR_2(u_2))) ∈ W[\text{int}][ρ], \]

  With that, we can conclude that \( (W', ρ_1((U_1, .)), ρ_2((U_2, .))) ∈ E[q ⊩ \text{ret-type}(q, χ[r_d : \text{int}], σ)]ρ \).

  Since we know that \( r_d ≠ q \) ret-type(\( q, χ[r_d : \text{int}, σ] \) = ret-type(\( q, χ, σ \) and thus we can use Lemma 1.11 to get the result.

\[\square\]
Lemma 1.44 (Branch)
If $\Psi;\Delta;\chi \vdash r_1 \approx_u r_2; \text{int}$, $\Psi;\Delta;\chi \vdash u_1 \approx_u u_2; \text{box} \forall[]\{\chi';\sigma\}^q$, $\Delta \vdash \chi \leq \chi'$, and $\Psi;\Delta;\chi;\sigma; q \vdash I_1 \approx I_2$, then $\Psi;\Delta;\chi;\sigma; q \vdash \text{bnz} r_1, u_1; I_1 \approx \text{bnz} r_2, u_2; I_2$.

Proof
Clearly, $\Psi;\Delta;\chi;\sigma; q \vdash \text{bnz} r_1, u_1; I_1$ and $\Psi;\Delta;\chi;\sigma; q \vdash \text{bnz} r_2, u_2; I_2$ follow from the premises.
Consider arbitrary $W$ and $\rho$ such that $W \in \mathcal{H}[\Psi]$, $\rho \in D[\Delta]$, currentMR($W(i_{\text{reg}})) \in W \mathcal{R}[\chi]\rho$, and currentMR($W(i_{\text{stk}})) \in W \mathcal{S}[\sigma]\rho$. We need to show that
\[
(W, \rho_1((\text{bnz} r_1, u_1; I_1, \cdot)), \rho_2((\text{bnz} r_2, u_2; I_2, \cdot))) = W, (\text{bnz} r_1, \rho_1(u_1); I_1, \cdot), (\text{bnz} r_2, \rho_2(u_2); I_2, \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)]
\]
Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running($0, (M_i \mid E_i\{e_i\}))$.
In the following, assume $W.k > 0$ and let $\tau_i; \sigma_i = \text{ret-type}(q, \chi, \sigma)$. From our first hypothesis, $\Psi;\Delta;\chi \vdash r_1 \approx_u r_2; \text{int}$. Thus $W.R_1(r_1) = W.R_2(r_2)$. We proceed by cases on $W.R_1(r_1)$

- $W.R_1(r_1) = 0$
  We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.
  
  (W, (bnz r_1, \rho_1(u_1); I_1, \cdot), (bnz r_2, \rho_2(u_2); I_2, \cdot)) \in \text{TermAtom}[q \vdash \tau_i; \sigma_i].$
  To establish this, we must show $W.\Psi_1; \vdash W.\chi; I_i; \sigma_i; q \vdash (\text{bnz} r_1, \rho_1(u_1); I_1, \cdot); (\text{bnz} r_2, \rho_2(u_2); I_2, \cdot).$ The latter follows from $\Psi;\Delta;\chi;\sigma; q \vdash \text{bnz} r_1, u_1; I_1$ using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.

- We define $W' = W$. Note that $W' \in \text{World}$ since $W \in \text{world}$.
- $W \equiv_{\text{pub}} W$ trivially.
- Consider arbitrary $M_1$ and $M_2$ such that $(M_1, M_2) : W$. Let $M_i = (H_i, R_i, S_i).$
  From $(M_1, M_2) : W$, since $W.k > 0$, we have $(\text{run}, W, M_1, M_2) \in \otimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\}$. From the latter, we obtain a fact for island $i_{\text{reg}}$: $(\text{run}, W, R_1, R_2) \in \text{currentMR}(W(i_{\text{reg}}))$. From that it follows that $R_i = W.R_i$ and, hence, $R_i(r_i) = 0$.
  From the above fact and the reduction semantics we derive that
\[
((H_i, R_i, S_i) \mid (\text{bnz} r_1, \rho_1(u_1); I_1, \cdot)) \longrightarrow^1 (H_i, R_i, S_i) \mid (\rho_1(I_1), \cdot))
\]
Let $M'_i = M_i = (H_i, R_i, S_i)$. Note that $(M'_1, M'_2) : W'$ is equivalent to $(M_1, M_2) : W$, which holds trivially by assumption.
- Note that $W.k = W'.k$ since $W'.k = W.k$.
- Note that $q \neq e$, which follows from the third hypothesis. Further, since we haven’t changed the world, we can see that the the return address does not change.

- ret-type($q, \chi, \sigma$) = ret-type($q, \chi, \sigma$) holds trivially.
- currentMR($W'(i_{\text{reg}})) \in W, \mathcal{R}[\chi]\rho$ holds trivially by assumption currentMR($W(i_{\text{reg}})) \in W, \mathcal{R}[\chi]\rho$ and $W' = W$.
- currentMR($W'(i_{\text{stk}})) \in W, \mathcal{S}[\sigma]\rho$ holds trivially by assumption currentMR($W(i_{\text{stk}})) \in W, \mathcal{S}[\sigma]\rho$ and $W' = W$.

Hence, we can conclude that
\[
(W', (\rho_1(I_1), (\rho_2(I_2), \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)]\rho
\]
Now, the result follows by Lemma 1.11.

- $W.R_1(r_1) = n, n \neq 0$
  We instantiate the second hypothesis, $\Psi;\Delta;\chi \vdash u_1 \approx_u u_2; \text{box} \forall[]\{\chi';\sigma\}^q$, with $W$ and $\rho$, noting $W \in \mathcal{H}[\Psi]$, $\rho \in D[\Delta]$, and currentMR($W(i_{\text{reg}})) \in W, \mathcal{R}[\chi]\rho$. Thus we have that $(W, W.R_1(\rho_1(u_1)), W.R_2(\rho_2(u_2))) \in \mathcal{E}[\text{box} \forall[]\{\chi';\sigma\}^q]$ and $\mathcal{R}[\chi]$ and $W.R_2.R_1(\rho_1(u_1)) = \mathcal{E}[\text{box} \forall[]\{\chi';\sigma\}^q]$.
  From the definition of the latter, we have that $W.R_1(R_2(\rho_2(u_2))) = \mathcal{E}[\text{box} \forall[]\{\chi';\sigma\}^q]$.
  We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.
\((W, (\text{bnz } r_1, \rho_1(u_1); \rho_1(I_1), \cdot), (\text{bnz } r_2, \rho_2(u_2); \rho_2(I_2), \cdot)) \in \text{TermAtom}[q \vdash \tau; \sigma, \rho]\). To establish this, we must show \(W : \Psi_i : W : \chi_i : W : \sigma_i : \rho_1(q) \vdash (\text{bnz } r_1, \rho_1(u_1); \rho_1(I_1), \cdot); \rho_1(\tau_r); \rho_1(\sigma_r)\).
The latter follows from \(\Psi : \Delta : \chi : \sigma : q \vdash \text{bnz } r_1, u_1; I_1\) using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.

- We define \(W' = \triangleright W\). Note that \(W' \in \text{World}\) since \(W.k > 0\) and \(W \in \text{World}\).
- \(\triangleright W \equiv \text{pub } W\) by Lemma 1.6.

Consider arbitrary \((M_1, M_2) : W\). Let \(M_i = (H_i, R_i, S_i)\). From \((M_1, M_2) : W\), since \(W.k > 0\), we have \((\triangleright W, M_1, M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W : \Theta\}\).
From the latter, we have two facts, one for island \(i_{\text{reg}}\) and the other for island \(i_{\text{box}}\).
First, we have that \((\triangleright W, R_1 \upharpoonright, R_2) \in \text{currentMR}(W(i_{\text{reg}}))\). From the latter it follows that \(R_i = W : R_i\) and, hence, \(R_i(r_i) = n, n \neq 0\) and \(R_i(\rho_i(u_i)) = \ell_i[\omega_i]\).
Second, we have that there exist some \(H_{b1} \subseteq H_1\) and \(H_{b2} \subseteq H_2\) such that \((\triangleright W, H_{b1} \uparrow, H_{b2} \uparrow) \in \text{currentMR}(W(i_{\text{box}}))\). We use the latter to instantiate \((W, \ell_1[\omega_1], \ell_2[\omega_2]) \in \mathcal{W}[\bigvee . \{\chi'; \sigma\}_q]\), noting that \(\triangleright W \equiv \mathcal{W}\), which allows us to conclude:

\[\begin{align*}
\ast & \quad \text{H}_{bi}(\ell_i) = \text{code}[\ell_i](\chi_i; \sigma_i)_q I'_i, \\
\ast & \quad \rho_i(\chi_i) = \chi_i[\omega_i/\beta_i], \\
\ast & \quad \rho_i(\sigma_i) = \sigma_i[\omega_i/\beta_i], \\
\ast & \quad \rho_i(q) = q[\omega_i/\beta_i], \\
\ast & \quad (\triangleright W, \text{code}[\ell_i](\chi_1; \sigma_1)_q I'_1[\omega_1/\beta_1]), (\text{code}[\ell_2](\chi_2; \sigma_2)_q I'_2[\omega_2/\beta_2]) \in \mathcal{H}[\bigvee . \{\chi'; \sigma\}_q]\rho
\end{align*}\]

Thus, we have that \(\tilde{R}_i(\rho_i(u_i)) = \ell_i[\omega_i]\) and \(H_i(\ell_i) = \text{code}[\ell_i](\chi_i; \sigma_i)_q I'_i\); and, from the reduction semantics and \(R_i(r_i) = n, n \neq 0\), we derive that

\[\langle (H_i, R_i, S_i) \mid (\text{bnz } r_i, \rho_i(u_i); \rho_1(I_1), \cdot) \rangle \vdash_1 \langle (H_i, R_i, S_i) \mid (I'_i[\omega_i/\beta_i], \cdot) \rangle\]

Let \(M'_i = M_i = (H_i, R_i, S_i)\). Note that \((M'_1, M'_2) : W'\) is equivalent to \((M_1, M_2) : \triangleright W\), which follows by Lemma 1.6.

- Note that \(W.k \leq W'.k + 1\) since \(W'.k = (\triangleright W).k = W.k - 1\).
- Note that \(\rho_1(\psi) \neq \psi\), which follows from the third hypothesis. Further, since we haven’t changed the world, we can see that the \(\text{the return} \) address does not change.

- Note that

\[\begin{align*}
(\triangleright W, \text{code}[\ell_i](\chi_1; \sigma_1)_q I'_1[\omega_1/\beta_1]), \\
\text{code}[\ell_2](\chi_2; \sigma_2)_q I'_2[\omega_2/\beta_2] \in \mathcal{H}[\bigvee . \{\chi'; \sigma\}_q]\rho
\end{align*}\]

\[\begin{align*}
&\equiv (\triangleright W, \text{code}[\ell_i](\chi_1; \sigma_1)_q I'_1[\omega_1/\beta_1]), \\
&\text{code}[\ell_2](\chi_2; \sigma_2)_q I'_2[\omega_2/\beta_2]) \in \mathcal{H}[\bigvee . \{\chi'; \sigma\}_q]\rho
\end{align*}\]

\[\begin{align*}
&\equiv (\triangleright W, \text{code}[\rho_1(\chi_i)]; \rho_1(\sigma_i))_q I'_i[\omega_i/\beta_i], \\
&\text{code}[\rho_2(\chi_i); \rho_2(\sigma_i))_q I'_2[\omega_2/\beta_2]) \in \mathcal{H}[\bigvee . \{\chi'; \sigma\}_q]\rho
\end{align*}\]

Instantiate the latter with \(\triangleright W\). We note the following:

* \(\triangleright W \equiv \triangleright W\) by reflexivity.
* \(\text{ret-type}(\rho_1(\chi_i); \rho_2(\sigma_i)) \equiv \text{ret-type}(\rho_1(\chi_i); \rho_2(\sigma_i))\). The latter is immediate except in the case when \(\rho_1(\chi_i); \rho_2(\sigma_i)\) is some register \(r'\), in which case we must show that \(r' \in \text{dom}(\chi_i); \rho_2(\sigma_i)\) would be undefined. But note that from our first premise, it follows that \(\Delta \vdash \text{box } \bigvee . \{\chi'; \sigma\}_q\). By inversion of typing rules, we have that \(\Delta \vdash \bigvee . \{\chi'; \sigma\}_q\), and hence \(\Delta ; \chi'; \sigma \vdash q\). From the latter it follows that ret-type\((\rho_1(\chi_i); \rho_2(\sigma_i))\) is defined. Hence, if \(q\) is some register \(r'\), it must be that \(r' \in \text{dom}(\chi_i); \rho_2(\sigma_i)\). Moreover, from the second premise, it follows that \(\chi_i'(r') = \chi_i'(r')\). This is enough to establish our claim.
* \(\text{currentMR}(\triangleright W)(i_{\text{reg}}) \subseteq \triangleright W: \mathcal{R}[\chi']\rho\). To show this, consider arbitrary \((\tilde{W}, M_1, M_2) \in \text{currentMR}(\triangleright W)(i_{\text{reg}})\) such that \(W \equiv \triangleright W\). We must show that

\[\langle \tilde{W}, M_1, M_2 \rangle \in \mathcal{R}[\chi']\rho\]
Note that $\langle W \rangle.k = W.k-1$ and that $\text{currentMR}(\langle W \rangle(i_{\ell})) = \text{currentMR}(W(i_{\ell})) \downarrow W.k-1$. Thus, $(\widetilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\ell}))$. Using the latter we can instantiate currentMR$(W(i_{\ell})) \subseteq W \mathcal{R}[\chi] \rho$ with $(\widetilde{W}, M_1, M_2)$, noting that $\widetilde{W} \supseteq W$ (by transitivity of $\supseteq$), which gives us $(\widetilde{W}, M_1, M_2) \in \mathcal{R}[\chi] \rho$. Finally, by Lemma 1.19 (register-file subtyping implies inclusion) we have that $\mathcal{R}[\chi] \rho \subseteq \mathcal{R}[\chi] \rho$, which is sufficient to show what we need.

* currentMR$(\langle W \rangle(i_{\ell})) \subseteq W \mathcal{S}[\sigma] \rho$. To show this, consider arbitrary $(\widetilde{W}, M_1, M_2) \in$ currentMR$(\langle W \rangle(i_{\ell}))$ such that $\widetilde{W} \supseteq \langle W \rangle$. We must show that

$$(\widetilde{W}, M_1, M_2) \in \mathcal{S}[\sigma] \rho$$

Note that $\langle W \rangle.k = W.k-1$ and that currentMR$(\langle W \rangle(i_{\ell})) = \text{currentMR}(W(i_{\ell})) \downarrow W.k-1$. Thus, $(\widetilde{W}, M_1, M_2) \in$ currentMR$(W(i_{\ell}))$. Using the latter we can instantiate currentMR$(W(i_{\ell})) \subseteq W \mathcal{S}[\sigma] \rho$ with $(\widetilde{W}, M_1, M_2)$, noting that $\widetilde{W} \supseteq W$ (by transitivity of $\supseteq$), which gives us $(\widetilde{W}, M_1, M_2) \in \mathcal{S}[\sigma] \rho$ as needed.

Hence, we can conclude that

$$(\langle W \rangle, (\rho_1(I_1[\omega_1/\beta_1]), \cdot), (\rho_2(I_2[\omega_2/\beta_2]), \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho$$

Now, the result follows by Lemma 1.11.

\[\square\]

**Lemma 1.45 (Load from Mutable Tuple)**

If $\Psi; \Delta; \chi \vdash r_{s_1} \approx u_r_{s_2}; \text{ref}(\tau_0, \ldots, \tau_n), 0 \leq i \leq n, q \neq r_d$, and $\Psi; \Delta; \chi[r_d; \tau_i]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \chi; \sigma; q \vdash 1d r_d, r_{s_1}[i]; I_1 \approx 1d r_d, r_{s_2}[i]; I_2$.

**Proof**

Clearly, $\Psi; \Delta; \chi; \sigma; q \vdash 1d r_d, r_{s_1}[i]; I_1$ and $\Psi; \Delta; \chi; \sigma; q \vdash 1d r_d, r_{s_2}[i]; I_2$.

Consider arbitrary $W$ and $\rho$ such that $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta]$, currentMR$(W(i_{\ell})) \subseteq W \mathcal{R}[\chi] \rho$, and currentMR$(W(i_{\ell})) \subseteq W \mathcal{S}[\sigma] \rho$. We need to show that

$$(W, \rho_1((1d r_d, r_{s_1}[i]; I_1, \cdot), \rho_2((1d r_d, r_{s_2}[i]; I_2, \cdot)))$$

$$(W, (1d r_d, r_{s_1}[i]; I_1)), (1d r_d, r_{s_2}[i]; I_2), \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho.$$

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running$\mathcal{S}[\sigma] \rho = \text{ret-type}(q, \chi, \sigma)$.

In the following, assume $W.k > 0$ and let $\tau_i; \sigma_r = \text{ret-type}(q, \chi, \sigma)$.

Moreover, note the following preliminary facts:

- By instantiating our first hypothesis with $W$ and $\rho$, we have that $(W, W.\hat{R}_1(r_{s_1}), W.\hat{R}_2(r_{s_2})) \in W[\text{ref}(\tau_0, \ldots, \tau_n)] \rho$. From the definition of the latter, we have that $W.\hat{R}_1(r_{s_1}) = \ell_1$ and $W.\hat{R}_2(r_{s_2}) = \ell_2$ and we know there exists an island $i$ such that

$$\forall W' \supseteq W. (\ell_1, \ell_2) \in W'(i).\text{bij}(W'(i).s) \wedge \exists \varphi_M. \text{currentMR}(W'(i)) = \varphi_M \otimes$$

$$\{(\widetilde{W}, \{\ell_1 \mapsto h_1\}, \{\ell_2 \mapsto h_2\}\} \in \text{MemAtom} \mid (\widetilde{W}, h_1, h_2) \in \mathcal{H}[\tau_0, \ldots, \tau_n] \rho\} \tag{8}$$

We proceed by analogy to Lemma 1.11, but choose a world $W'$ after choosing memories, so we cannot use the Lemma as stated.

- We claim that $(W, (1d r_d, r_{s_1}[i]; \rho_1(I_1), \cdot), (1d r_d, r_{s_2}[i]; \rho_2(I_2), \cdot)) \in \text{TermAtom}[q \vdash \tau_i; \sigma_r] \rho$. To establish this, we must show $W.\Psi_j; \mu; W.\chi_j; W.\sigma_j; \rho_j(q) \vdash (1d r_d, r_{s_1}[i]; \rho_1(I_1), \cdot); \rho_j(\tau_i); \rho_j(\sigma_r).$

The latter follows from $\Psi; \Delta; \chi; \sigma; q \vdash 1d r_d, r_{s_1}[i]; I_j$ using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.
Expanding the definition of $E[\cdot]$, choose arbitrary $E_1$ and $E_2$ such that $(W,E_1,E_2) \in K[q \vdash \tau; \sigma]_\rho$.

We must show that $(W,E_1[(1d.r_d, r_s[1]; \rho_1(I_1), \cdot)], E_2[(1d.r_d, r_s[2]; \rho_2(I_2), \cdot)]) \in \mathcal{O}$.

In order to do that, choose arbitrary $(M_1, M_2) : W$. We need to show that the expressions either both terminate with these memories or are both still running after $W.k$ steps.

We do this by taking one step and then using the resulting memories, finding a world that we can then use with the final hypothesis we are given.

Let $M_j = (H_j, R_j, S_j)$ for $j \in \{1, 2\}$.

From $(M_1, M_2) : W$, since $W.k > 0$, we have that $(\triangleright W, M_1, M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\}$. From the latter, we can have the following two facts, one for island $i$ and one for the island $j$ (which we know exists from the preliminary facts we collected above and keeps track of the references $\ell_1$ and $\ell_2$ that we wish to update).

First, we have that $W.R_i$ and, hence, $\widehat{R}_1(r_1) = \ell_1$, $\widehat{R}_2(r_2) = \ell_2$, (which are the same $\ell_1$ and $\ell_2$ from our preliminary facts above).

Second, we have that there exist some $H_{r_1} \subseteq H_1$ and $H_{r_2} \subseteq H_2$ such that $(\triangleright W, H_{r_1}, H_{r_2}) \in \text{currentMR}(W(i))$. Instantiating (8) with $W$, noting $W \equiv W$ by reflexivity, we have that

$$\text{currentMR}(W(i)) = \varphi_M \otimes \{ (\widehat{W}, \ell_1 \mapsto h_1 \cdot \ell_2 \mapsto h_2) \} \in \text{MemAtom} \mid (\widehat{W}, h_1, h_2) \in \mathcal{H}V[\langle \tau_0, \ldots, \tau_n \rangle]_\rho$$

Hence, we have that $H_1(\ell_1) = H_1(\ell_2) = h_1$, and $H_1(\ell_3) = H_2(\ell_2) = h_2$, as well as $(\triangleright W, h_1, h_2) \in \mathcal{H}V[\langle \tau_0, \ldots, \tau_n \rangle]_\rho$.

From the latter, it follows that $h_1 = \langle w_{10}, \ldots, w_{1n} \rangle$ and $h_2 = \langle w_{20}, \ldots, w_{2n} \rangle$. Moreover, since we have as our second hypothesis $0 \leq i \leq n$, it follows that $(\triangleright W, w_{1i}, w_{2i}) \in W[\tau_i]_\rho$.

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:

$$\langle M_j \mid E_j[(1d.r_d, r_s[i]; \rho_j(I_j), \cdot)] \rangle \rightarrow (H_j, R_j[r_d \mapsto w_{ji}], S_j) \mid E_j[(\rho_j(I_j), \cdot)])$$

Let $s = (W.R_i[r_d \mapsto w_{1i}], W.\chi[r_d : \rho_1(\tau_i)], W.R_2[r_d \mapsto w_{2i}], W.\chi_2[r_d : \rho_2(\tau_i)])$. Now let $W' = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{\text{reg}} \mapsto \text{island}_{\text{reg}}(s, W.k)])$.

- We claim that $W' \in \text{World}$.
  - It suffices to show $W'.\Psi_j ; ; w_{ji} ; \rho_j(\tau_i)$.
  - Since $(\triangleright W, w_{1i}, w_{2i}) \in W[\tau_i]_\rho$ and $\triangleright W.\Psi_j = W'.\Psi_j$, WvalAtom gets us what we need.

- We claim that $(M'_j, M''_j) : W'$.
  - Instantiating the last hypothesis with $W'$, noting that Lemmas 1.20 and 1.21 yield the required obligations on memory relations, we know that $(W', E_1[(\rho_1(I_1), \cdot)], E_2[(\rho_2(I_2), \cdot)]) \in \mathcal{O}$.
  - This means that either $\langle M'_j \mid E_j[(\rho_j(I_j), \cdot)] \rangle \downarrow$ or running$(W'.k, \langle M'_j \mid E_j[(\rho_j(I_j), \cdot)] \rangle)$.
  - In the former case, that clearly means that $\langle M_j \mid E_j[(1d.r_d, r_s[i]; \rho_j(I_j), \cdot)] \rangle \downarrow$. In the latter case, note that $W'.k \leq W.k + 1$, which means that running$(W.k, \langle M_j \mid E_j[(1d.r_d, r_s[i]; \rho_j(I_j), \cdot)] \rangle)$ holds, and so we are done.

\[ \square \]

**Lemma 1.46 (Load from Immutable Tuple)**

If $\Psi; \Delta; \chi ; r_s \approx u \Rightarrow r_s : \text{box}(\tau_0, \ldots, \tau_n)$, $0 \leq i \leq n$, $q \neq r_d$, and $\Psi; \Delta; \chi \approx r_d : \tau_s; \sigma ; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \chi; \sigma ; q \vdash 1d.r_d, r_s[i]; I_1 \approx 1d.r_d, r_s[i]; I_2$.

**Proof**

Clearly, $\Psi; \Delta; \chi; \sigma ; q \vdash 1d.r_d, r_s[i]; I_1$ and $\Psi; \Delta; \chi; \sigma ; q \vdash 1d.r_d, r_s[i]; I_2$. 

40
Consider arbitrary $W$ and $\rho$ such that $W \in \mathcal{H}[\Psi]$, $\rho \in \mathcal{D}[\Delta]$, currentMR($W(i_{\text{reg}})$) $\in W$, $\mathcal{R}[\chi]\rho$, and currentMR($W(i_{\text{stk}})$) $\in W$, $S[\sigma]\rho$. We need to show that

$$(W, \rho_1((1d r_d, r_{s_1}[i]; I_1, \cdot)), \rho_2((1d r_d, r_{s_2}[i]; I_2, \cdot)))$$

$s_1 = 1, s_2 = 1, r_1 = 1, r_2 = 1$.

Note that if $W, k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_t$, we can immediately show that running$(0, (M_t \mid E_i[e_i]))$.

In the following, assume $W, k > 0$ and let $\tau_r; \sigma_r = \text{ret-type}(q, \chi, \sigma)$.

Moreover, note the following preliminary fact:

- By instantiating our first hypothesis with $W$ and $\rho$, we have that $(W, W, \tilde{R}_1(r_{s_1}), W, \tilde{R}_2(r_{s_2})) \in W[\text{box}, (\tau_0, \ldots, \tau_n)]\rho$. From the definition of the latter, we have that $W, \tilde{R}_1(r_{s_1}) = \ell_1$ and $W, \tilde{R}_2(r_{s_2}) = \ell_2$ and

$$\forall(W, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})). \quad \tilde{W} \equiv W \implies (\tilde{W}, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{H}V[(\tau_0, \ldots, \tau_n)]\rho \}$$

With the above fact in hand, we now prove (9).

- We claim that $(W, (1d r_d, r_{s_1}[i]; \rho_1(I_1), \cdot), (1d r_d, r_{s_2}[i]; \rho_2(I_2), \cdot)) \in \text{TermAtom}(q \vdash \tau_r; \sigma_r)\rho$. To establish this we must show $W, \Psi_1; : W, \chi_1; W, \sigma_1; \rho_1(q) \vdash (1d r_d, r_{s_1}[i]; \rho_1(I_1), \cdot); \rho_1(\tau_r); \rho_1(\sigma_r)$ and $W, \Psi_2; : W, \chi_2; W, \sigma_2; \rho_2(q) \vdash (1d r_d, r_{s_2}[i]; \rho_2(I_2), \cdot); \rho_2(\tau_r); \rho_2(\sigma_r)$. Each follows from $\Psi_1; \Delta; \chi; \sigma; q \vdash (1d r_d, r_{s_1}[i]; I_1)$ and $\Psi_2; : W, \chi_2; W, \sigma_2; \rho_2(q) \vdash (1d r_d, r_{s_2}[i]; \rho_2(I_2), \cdot)$ respectively using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.

- Consider arbitrary $E_1$ and $E_2$ such that $(W, E_1, E_2) \in \mathcal{K}[q \vdash \tau_r; \sigma_r]\rho$. We must show

$$(W, E_1[(1d r_d, r_{s_1}[i]; \rho_1(I_1), \cdot)], E_2[(1d r_d, r_{s_2}[i]; \rho_2(I_2), \cdot)]) \in \mathcal{O}.$$
and
\[ \langle M_2 \mid E_2[(1 \cdot r_d, r_{\text{sd}}[1]; \rho_2(I_2), \cdot)] \rangle \rightarrow \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle \]
where \( M'_1 = (H_1, R_1[r_d \mapsto w_{11}], S_1) \) with \( R_1(r_{\text{sd}}) = \ell_1 \) and \( H_1(\ell_1) = (w_{10}, \ldots, w_{11}, \ldots, w_{1n}) \), and \( M'_2 = (H_2, R_2[r_d \mapsto w_{11}], S_2) \) with \( R_2(r_{\text{sd}}) = \ell_2 \) and \( H_2(\ell_2) = (w_{20}, \ldots, w_{21}, \ldots, w_{2n}) \).
Note that in order to complete our proof, it suffices to show:
\[
\left( \langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle \downarrow \wedge \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle \downarrow \right) \vee \running(W.k - 1, \langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle) \wedge \running(W.k - 1, \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle)
\]
Let \( W' = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{\text{reg}} \mapsto \text{island}_{\text{reg}}(s, W.k)]) \) where
\[ s = (W.R_1[r_d \mapsto w_{11}], W.\chi_1[r_d \mapsto \rho_1(\tau_1)], W.R_2[r_d \mapsto w_{21}], W.\xi_2[r_d \mapsto \rho_2(\tau_1)]). \]
We claim that \( W' \in \text{World} \). Recall that we have \((W, w_{11}, w_{21}) \in W.\Psi_1.\rho\). Thus we have that \( W.\Psi_i; \cdot \vdash w_{j_i} : \rho_i(\tau_i) \). Given our choice of \( W' \) and since \( W \in \text{World} \), it follows that \( W'.\Psi_i; \cdot \vdash w_{j_i} : \rho_i(\tau_i) \) which is sufficient to establish our claim.
Moreover, note that \( W' \sqsupseteq W \) and \( W' \sqsupseteq \text{pub} W \). Both follow immediately given our choice of \( W' \) and the definition of \( \text{island}_{\text{reg}} \).
We proceed by showing that \( (M'_1, M'_2) : W' \) and then instantiating our final premise \( \Psi; \Delta; \chi[r_d : \tau_1]; \sigma; q \vdash I_1 \approx I_2 \).

- We claim that \( (M'_1, M'_2) : W' \). We prove this claim by establishing the following:
  * \( \vdash M'_1 : W'.\Phi_1 \) and \( \vdash M'_2 : W'.\Phi_2 \), both of which easily follow from \( (M_1, M_2) : W \) and the facts that \( W'.\chi_1 = W.\chi_1[r_d \mapsto \rho_1(\tau_1)] \), \( W'.\xi_2 = W.\xi_2[r_d \mapsto \rho_2(\tau_1)] \) and that we have updated registers \( r_d \) in with well-typed words, i.e., with words of the types designated by the \( t \) elements of \( W.\Psi_1(\ell_1) \) and \( W.\Psi_2(\ell_2) \), namely \( \rho_1(\tau_1) \) and \( \rho_2(\tau_1) \).
  * We assume that \( W.k > 0 \) (and thus \( W'.k > 0 \)). We must show that
\[
(\triangleright W', M'_1, M'_2) \in \bigotimes \{ \text{currentMR}(\theta) \mid \theta \in W', \Theta \}
\]
The latter follows from \( (M_1, M_2) : W \) and monotonicity of MemRel given that we establish the following claim for island \( i_{\text{reg}} \):
\[
(\triangleright W', W.R_1[r_d \mapsto w_{11}] \mid W.R_2[r_d \mapsto w_{21}] ) \in \text{currentMR}(W'(i_{\text{reg}}))
\]
Recall that \( W'.R_i = W.R_i[r_d \mapsto w_{j_i}] \). Thus we obtain trivially that \( (\triangleright W', W.R_1[r_d \mapsto w_{11}], W.R_2[r_d \mapsto w_{21}] ) \in \{ (W', W.R_1, W'.R_2) \mid \overline{W} \in \text{World}_{W,k} \} \). From the latter, we establish directly our claim that
\[
(\triangleright W', W.R_1[r_d \mapsto w_{11}] \mid W.R_2[r_d \mapsto w_{21}] ) \in \text{currentMR}(W'(i_{\text{reg}}))
\]
- Next, we instantiate our final hypothesis with \( W' \) and \( \rho \). Note that \( \rho \in \mathcal{D}[\Delta] \) by assumption, \( W' \in \mathcal{H}[\Psi] \) by heap monotonicity (Lemma 1.9) and that \( \text{currentMR}(W'(i_{\text{stk}})) \subseteq W'.S[\sigma]\rho \) by Lemma 1.21 since \( W'(i_{\text{stk}}) = W(i_{\text{stk}}) \). We also claim that \( \text{currentMR}(W'(i_{\text{reg}})) \subseteq W'.R[\chi[r_d : \tau_1]\rho] \). Above, we have established that \( (W', w_{11}, w_{21}) \in W.\Psi_1.\rho \) which, using Lemma 1.20, is sufficient to establish our claim. Hence, we have that
\[
(W', (\rho_1(I_1, \cdot)), (\rho_2(I_2), \cdot)) \in \mathcal{E}[\pi ; r; \sigma] \rho
\]
Instantiate the above with \( E_1 \) and \( E_2 \). Note that we have \( (W', E_1, E_2) \in \mathcal{K}[\pi \vdash \tau; \sigma] \rho \) which follows from \( (W, E_1, E_2) \in \mathcal{K}[\pi \vdash \tau; \sigma] \rho \) by monotonicity for evaluation contexts (Lemma 1.10), since our choice of \( W' \), \( W' \sqsupseteq \text{pub} W \) and our third hypothesis, \( \pi \neq r_d \), let us easily establish all the premises of that lemma.
Hence, we have
\[
(W', E_1[(\rho_1(I_1, \cdot))], E_2[(\rho_2(I_2), \cdot)]) \in \mathcal{O}.
\]
Instantiate the latter with $M'_1$ and $M'_2$, noting that $(M'_1, M'_2) : W'$. Hence, we have

$$((M'_1 \mid E_1[(\rho_1(I_1), \cdot)]) \downarrow \land (M'_2 \mid E_2[(\rho_2(I_2), \cdot)]) \downarrow \lor$$

$$(\text{running}(W,k, (M'_1 \mid E_1[(\rho_1(I_1), \cdot)]) \land \text{running}(W,k, (M'_2 \mid E_2[(\rho_2(I_2), \cdot)])))$$

which implies what we needed to show.

Lemma 1.47 (Store to Mutable Tuple)

If $\Psi; \Delta; \chi; \tau \vdash r_{d1} \approx_{u} r_{d2} : \text{ref}(\tau_0, \ldots, \tau_n)$, $0 \leq i \leq n$, $\Psi; \Delta; \chi; \tau \vdash r_{s1} \approx_{u} r_{s2} : \tau_i$, and $\Psi; \Delta; \chi; \sigma; q \vdash \tau_1 \approx_{I_1} \tau_2$, then $\Psi; \Delta; \chi; \sigma; q \vdash \text{st}_i r_{d1}[i], r_{s1}; I_1 \approx_{I} \text{st}_i r_{d2}[i], r_{s2}; I_2$.

Proof

Clearly, $\Psi; \Delta; \chi; \sigma; q \vdash \text{st}_i r_{d1}[i], r_{s1}; I_1$ and $\Psi; \Delta; \chi; \sigma; q \vdash \text{st}_i r_{d2}[i], r_{s2}; I_2$.

Consider arbitrary $W$ and $\rho$ such that $W \in \mathcal{H}[\Psi, \rho] \in \mathcal{D}[\Delta]$, currentMR($W(i_{\text{reg}})$) $\equiv_W \mathcal{R}[\chi]$, and currentMR($W(i_{\text{stk}})$) $\equiv_W \mathcal{S}[\sigma]$. We need to show that

$$(W, \rho_1((\text{st}_i r_{d1}[i], r_{s1}; I_1, \cdot)), \rho_2((\text{st}_i r_{d2}[i], r_{s2}; I_2, \cdot))) = (W, (\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot), (\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)]\rho.$$

(11)

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that $\text{running}(0, (M_i \mid E_i[e_i]))$.

In the following, assume $W.k > 0$ and let $\tau; \sigma = \text{ret-type}(q, \chi, \sigma)$.

Moreover, note the following preliminary facts:

- By instantiating our first hypothesis with $W$ and $\rho$, we have that $(W, W.\tilde{R}_1(r_{d1}), W.\tilde{R}_2(r_{d2})) \in W[\text{ref}(\tau_0, \ldots, \tau_n)]\rho$. From the definition of the latter, we have that $W.\tilde{R}_1(r_{d1}) = \ell_1$ and $W.\tilde{R}_2(r_{d2}) = \ell_2$ and we know there exists an island $i$ such that

$$\forall W' \ni W.(\ell_1, \ell_2) \in W'(^i).\text{bij}(W'(i), s) \land \exists \varphi_M. \text{currentMR}(W'(i)) = \varphi_M \ast$$

$$\{ (\tilde{W}, \ell_1 \mapsto h_1), (\ell_2 \mapsto h_2) \} \in \text{MemAtom} \mid (\tilde{W}, h_1, h_2) \in \mathcal{H}[\Psi]; I \in \mathcal{H}[\chi; \tau_0, \ldots, \tau_n]\rho.$$  (12)

- By instantiating the third hypothesis with $W$ and $\rho$, we have that $(W, W.\tilde{R}_1(r_{s1}), W.\tilde{R}_2(r_{s2})) \in W[\tau_i]\rho$. From the latter, we have that there exist some $w'_{1i}$ and $w'_{2i}$ such that $W.\tilde{R}_1(r_{s1}) = w'_{1i}$ and $W.\tilde{R}_2(r_{s2}) = w'_{2i}$.

With the above facts in hand, we now prove (11).

- We claim that $(W, (\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot)), (\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot)) \in \text{TermAtom}(q \vdash \tau_i; \sigma_i)\rho$. To establish this we must show $W.\Psi_1; \cdot; \Psi_2; \cdot; W.\chi; \sigma_1; \rho_1(q) \vdash (\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot), \cdot; \rho_1(\tau_i); \rho_1(\sigma_i))$ and $W.\Psi_2; \cdot; W.\chi; \sigma_2; \rho_2(q) \vdash (\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot), \cdot; \rho_2(\tau_i); \rho_2(\sigma_i))$. Each follows from $\Psi; \Delta; \chi; \sigma; q \vdash \text{st}_i r_{d1}[i], r_{s1}; I_1$ and $\Psi; \Delta; \chi; \sigma; q \vdash \text{st}_i r_{d2}[i], r_{s2}; I_2$, respectively, using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.

- Consider arbitrary $E_1$ and $E_2$ such that $(W, E_1, E_2) \in \mathcal{K}[q \vdash \tau_i; \sigma_i]\rho$. We must show that

$$(W, E_1[(\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot)], E_2[(\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot))]) \in \mathcal{O}.$$  

Consider arbitrary $(M_1, M_2) : W$. We must show either that $(M_1 \mid E_1[(\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot))]) \downarrow$ and $(M_2 \mid E_2[(\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot))]) \downarrow$, or running$(W.k, (M_1 \mid E_1[(\text{st}_i r_{d1}[i], r_{s1}; \rho_1(I_1, \cdot))])$ and running$(W.k, (M_2 \mid E_2[(\text{st}_i r_{d2}[i], r_{s2}; \rho_2(I_2, \cdot))])$.

Let $M_1 = (H_1, R_1, S_1)$ and $M_2 = (H_2, R_2, S_2)$.  

43
From \((M_1, M_2) : W\), since \(W.k > 0\), we have \(\triangleright W, M_1, M_2 \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\}\).

From the latter, we can have the following two facts, one for island \(i_{\text{reg}}\) and one for the island \(i\) (which we know exists from the preliminary facts we collected above and keeps track of the references \(\ell_1\) and \(\ell_2\) that we wish to update).

First, we have that \(\triangleright W, R_1 \mid R_2 \in \text{currentMR}(W(i_{\text{reg}}))\). From the latter it follows that \(R_i = W.R_i\) and, hence, \(R_1(r_{a1}) = \ell_1, R_2(r_{d2}) = \ell_2\) (which are the same \(\ell_1, \ell_2, w_{1i}\), and \(w'_{1i}\) from our preliminary facts above).

Second, we have that there exist some \(H_{i1} \subseteq H_1\) and \(H_{i2} \subseteq H_2\) such that \(\triangleright W, H_{i1}, H_{i2} \in \text{currentMR}(W(i))\). Instantiating (12) with \(W\), noting \(W \supseteq W\) by reflexivity, we have that

\[
\text{currentMR}(W(i)) = \varphi_M \otimes \\
\{ (\hat{W}, \{\ell_1 \mapsto h_1\}, \{\ell_2 \mapsto h_2\}) \in \text{MemAtom} \mid (\hat{W}, h_1, h_2) \in H.V[\langle \tau_0, \ldots, \tau_n \rangle] \rho \}
\]

Hence, we have that \(H_1(\ell_1) = H_{i1}(\ell_1) = h_1\), and \(H_1(\ell_2) = H_{i2}(\ell_2) = h_2\), as well as \(\triangleright W, h_1, h_2 \in H.V[\langle \tau_0, \ldots, \tau_n \rangle] \rho\). From the latter, it follows that \(h_1 = \langle w_{10}, \ldots, w_{1n} \rangle\) and \(h_2 = \langle w_{20}, \ldots, w_{2n} \rangle\). Moreover, since we have as our second hypothesis \(0 \leq i \leq n\), it follows that \(\triangleright W, w_{1i}, w_{2i} \in W[\tau_i] \rho\).

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:

\[
((H_1, R_1, S_1) \mid E_1[(\text{st r}_{d1}[i], r_{a1}; \rho_1(1), \cdot)]) \quad \rightarrow^1 \quad ((H_1[\ell_1 \mapsto h'_1], R_1, S_1) \mid E_1[\rho_1(1), \cdot]))
\]

and

\[
((H_2, R_2, S_2) \mid E_2[(\text{st r}_{d2}[i], r_{a2}; \rho_2(1), \cdot)]) \quad \rightarrow^1 \quad ((H_2[\ell_2 \mapsto h'_2], R_2, S_2) \mid E_2[\rho_2(1), \cdot]))
\]

where \(h'_1 = \langle w_{10}, \ldots, w_{1i}', \ldots, w_{1n} \rangle\) and \(h'_2 = \langle w_{20}, \ldots, w_{2i}', \ldots, w_{2n} \rangle\).

Let \(M'_1 = (H_1[\ell_1 \mapsto h'_1], R_1, S_1)\), let \(M'_2 = (H_2[\ell_2 \mapsto h'_2], R_2, S_2)\).

Note that in order to complete our proof, it suffices to show:

\[
(\langle M'_1 \mid E_1[\rho_1(1), \cdot) \rangle) \land (\langle M'_2 \mid E_2[\rho_2(1), \cdot) \rangle) \lor
\]

\[
(\text{running}(W.k-1, \langle M'_1 \mid E_1[\rho_1(1), \cdot) \rangle) \land \text{running}(W.k-1, \langle M'_2 \mid E_2[\rho_2(1), \cdot) \rangle)))
\]

We proceed by showing that \((M'_1, M'_2) : W\) and then instantiating our final hypothesis \(\Psi; \Delta; \chi; \sigma; q \vdash I_1 \approx I_2\).

- We claim that \((M'_1, M'_2) : W\). We prove this claim by establishing the following:

  \* \(\vdash M'_1 : W.\Phi_1\) and \(\vdash M'_2 : W.\Phi_2\), both of which easily follow from \((M_1, M_2) : W\) and the fact that we have updated locations \(\ell_1\) and \(\ell_2\) in a type-preserving manner, i.e., with tuples of the types designated by \(W.\Psi_1(\ell_1)\) and \(W.\Psi_2(\ell_2)\), namely \(\rho_1(\langle \tau_0, \ldots, \tau_n \rangle)\) and \(\rho_2(\langle \tau_0, \ldots, \tau_n \rangle)\).

  \* We assume that \(W.k > 0\) and we must show that

\[
(\triangleright W, M'_1, M'_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\}
\]

The latter follows from \((M_1, M_2) : W\), if we establish the following claim for island \(i\):

\[
(\triangleright W, H_{r1}[\ell_1 \mapsto h'_1], H_{r2}[\ell_1 \mapsto h'_2]) \in \text{currentMR}(W(i))
\]

Recall that

\[
\text{currentMR}(W(i)) = \varphi_M \otimes \\
\{ (\hat{W}, \{\ell_1 \mapsto h_1\}, \{\ell_2 \mapsto h_2\}) \in \text{MemAtom} \mid (\hat{W}, h_1, h_2) \in H.V[\langle \tau_0, \ldots, \tau_n \rangle] \rho \}
\]
Note that by monotonicity, we have \((\triangleright W, w'_{1i}, w'_{2i}) \in W[\tau_i] \rho\). From the latter, together with \((\triangleright W, h_{1i}, h_{2i}) \in H[\langle \tau_0, \ldots, \tau_n \rangle] \rho\), it follows that \((\triangleright W, h'_{1i}, h'_{2i}) \in H[\langle \tau_0, \ldots, \tau_n \rangle] \rho\).

Also, from \((\triangleright W, H_{r1}, H_{r2}) \in \text{currentMR}(W(i))\), we have that \((\triangleright W, H_{r1} \setminus \{\ell_1\}, H_{r2} \setminus \{\ell_2\}) \in \times M\).

The latter two facts, are sufficient to establish our claim that \((\triangleright W, H_{r1}[\ell_1 \mapsto h'_{1i}] \cup H_{r1}^1[\ell_1 \mapsto h'_{2i}] \cup H_{r1}^2) \in \text{currentMR}(W(i))\).

Next, we instantiate our final premise with \(W\) and \(\rho\). Note that \(W \in H[\Psi], \rho \in D[\Delta]\), currentMR\((W(i_{reg})) \equiv W R[\chi] \rho\), and currentMR\((W(i_{stk})) \equiv W S[\sigma] \rho\). Hence, we have that \((W, (\rho_1(I_1, \cdot)), (\rho_2(I_2), \cdot)) \in E[q \vdash \tau_i; \sigma_i] \rho\).

Instantiate the above with \(E_1\) and \(E_2\). Note that we have \((W, E_1, E_2) \in K[q \vdash \tau_i; \sigma_i] \rho\).

Hence, we have \((W, E_1[\rho_1(I_1, \cdot)], E_2[\rho_2(I_2, \cdot)]) \in \mathcal{O}\).

Instantiate the latter with \(M'_1\) and \(M'_2\), noting that \((M'_1, M'_2) : W\). Hence, we have \((\langle M'_1, E_1[\rho_1(I_1, \cdot)], E_2[\rho_2(I_2, \cdot)] \rangle) \vee (\text{running}(W, k, \langle M'_1, E_1[\rho_1(I_1, \cdot)] \rangle) \wedge \text{running}(W, k, \langle M'_2, E_2[\rho_2(I_2, \cdot)] \rangle))\)

which implies what we needed to show.

\[\square\]

**Lemma 1.48 (Allocate Mutable Tuple)**

If \(\text{len}(\tau) = n, q \neq r_d\), and \(\Psi; \Delta; \chi[d : \text{ref}(\tau_i)]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2\), then \(\Psi; \Delta; \chi; \tau :: \sigma; q \vdash \text{ralloc}_{rd, n} ; I_1 \approx \text{ralloc}_{rd, n} ; I_2\).

**Proof**

Clearly, \(\Psi; \Delta; \chi; \tau :: \sigma; q \vdash \text{ralloc}_{rd, n} ; I_1\) and \(\Psi; \Delta; \chi; \tau :: \sigma; q \vdash \text{ralloc}_{rd, n} ; I_2\).

Consider arbitrary \(W\) and \(\rho\) such that \(W \in H[\Psi], \rho \in D[\Delta]\), currentMR\((W(i_{reg})) \in W R[\chi] \rho\), and currentMR\((W(i_{stk})) \equiv W S[\sigma] \rho\). We need to show that \((W, \rho_1((\text{ralloc}_{rd, n} ; I_1, \cdot)), \rho_2((\text{ralloc}_{rd, n} ; I_2, \cdot))) \in E[q \vdash \text{ret-type}(q, \chi, \tau :: \sigma)] \rho\).

\[\begin{align*}
(W, \rho_1((\text{ralloc}_{rd, n} ; I_1, \cdot)), & \rho_2((\text{ralloc}_{rd, n} ; I_2, \cdot))) \\
= (W, (\text{ralloc}_{rd, n} ; I_1, \cdot), (\text{ralloc}_{rd, n} ; I_2, \cdot)) \in E[q \vdash \text{ret-type}(q, \chi, \tau :: \sigma)] \rho. & (13)
\end{align*}\]

Note that if \(W.k = 0\) then we are done, since for any evaluation contexts \(E_i\) and memories \(M_i\), we can immediately show that \(\text{running}(0, (M_i | E_i[e_i]))\).

In the following, assume \(W.k > 0\) and let \(\tau_i; \sigma_i = \text{ret-type}(q, \chi, \tau :: \sigma)\) and \(\tau = \tau_1 \cdots \tau_m\).

Moreover, note the following preliminary facts:

- By our first hypothesis, we have that \(m = n\).
- By our third hypothesis and the typing rules, we have that \(, \text{dec}(q, n) \neq \epsilon\) and \(\text{dec}(q, n) \neq \text{undefined}\). A consequence of the latter fact is that if \(q = i\) then \(i \geq n\).
- By our assumption currentMR\((W(i_{stk})) \in W S[\tau :: \sigma] \rho\) and the first fact, we have there exist \(w_{11}, \ldots, w_{1n}, w_{21}, \ldots, w_{2n}, S'_1\) and \(S'_2\) such that \(W.S_1 = w_{11} \cdots \cdot w_{1n} :: S'_1, W.S_2 = w_{21} \cdots \cdot w_{2n} :: S'_2\) and \((W, w_{1i}, w_{2i}) \in W[\tau_i] \rho\).

With the above facts in hand, we now prove (13).

- We claim \((W, (\text{ralloc}_{rd, n} ; \rho_1(I_1, \cdot)), (\text{ralloc}_{rd, n} ; \rho_2(I_2, \cdot))) \in \text{TermAtom}(q \vdash \tau_i; \sigma_i) \rho\). From the definitions of \(\text{TermAtom}(q \vdash \tau_i; \sigma_i) \rho\) and \(\text{TermAtom}(q \vdash \tau_i; \sigma_i) \rho\), it suffices to show \(W.\Psi_1 ; W.\chi ; W.\sigma_1 ; \rho_1(q) \vdash (\text{ralloc}_{rd, n} ; \rho_1(I_1, \cdot)) \rho(\tau_i) ; \rho_1(\sigma_i)\) and \(W.\Psi_2 ; W.\chi ; W.\sigma_2 ; \rho_2(q) \vdash (\text{ralloc}_{rd, n} ; \rho_2(I_2, \cdot)) \rho(\tau_i) ; \rho_2(\sigma_i)\). Each follows from \(\Psi; \Delta; \chi; \tau :: \sigma; q \vdash \text{ralloc}_{rd, n} ; I_1\) and \(\Psi; \Delta; \chi; \tau :: \sigma; q \vdash \text{ralloc}_{rd, n} ; I_2\) respectively using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.

45
Consider arbitrary $E_1$ and $E_2$ such that $(W, E_1, E_2) \in K[q \vdash \tau; \sigma]$. We must show that 

$$(W, E_1[(\text{alloc} \cdot d, n; \rho_1(I_1), \cdot)], E_2[(\text{alloc} \cdot d, n; \rho_2(I_2), \cdot)]) \in O.$$ 

Let $M_1 = (H_1, R_1, S_1)$ and $M_2 = (H_2, R_2, S_2)$.

From $(M_1, M_2) : W$, since $W.k > 0$, we have $(\triangleright W, M_1, M_2) \in \bigotimes\{\text{currentMR(} \theta \text{) } | \theta \in W.\Theta \}$. From the latter, we can have the following facts for island $i_{\text{stk}}$.

We have that $(\triangleright W, S_1[i], S_2[i]) \in \text{currentMR}(W(i_{\text{stk}}))$. From the latter it follows that $S_j = W.S_j = w_{j1} : \cdots : w_{jn} : S_j'$.

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:

$$\langle M_1 | E_1[(\text{alloc} \cdot d, n; \rho_1(I_1), \cdot)] \rangle \rightarrow (M_1' | E_1[(\rho_1(I_1), \cdot)])$$

and

$$\langle M_2 | E_2[(\text{alloc} \cdot d, n; \rho_2(I_2), \cdot)] \rangle \rightarrow (M_2' | E_2[(\rho_2(I_2), \cdot)])$$

where $M_1' = (H_1', R_1', S_1')$ where $R_1' = R_1[r_d \mapsto \ell_1]$ and $H_1' = H_1[\ell_1 \mapsto (w_{11}, \cdots, w_{1n})]$, and $M_2' = (H_2', R_2', S_2')$ where $R_2' = R_2[r_d \mapsto \ell_2]$ and $H_2' = H_2[\ell_2 \mapsto (w_{21}, \cdots, w_{2n})]$.

Note that, by the reduction semantics, $\ell_1 \notin \text{dom}(H_1)$ and $\ell_2 \notin \text{dom}(H_2)$.

Note that in order to complete our proof, it suffices to show:

$$(\langle (M_1' | E_1[(\rho_1(I_1), \cdot)] \rangle \downarrow \wedge (M_2' | E_2[(\rho_2(I_2), \cdot)] \rangle \downarrow \wedge \text{running}(W.k - 1, (M_1' | E_1[(\rho_1(I_1), \cdot)] \rangle \wedge \text{running}(W.k - 1, (M_2' | E_2[(\rho_2(I_2), \cdot)] \rangle)))$$

Let $W' = (W.k, W.\Psi, W.\Theta)$ where

1. $\Psi_j = \Psi_j, \ell_j : \text{ref}(\tau_1, \cdots, \tau_n)$;
2. $|\Theta'| = |\Theta| + 1 = i_\ell$;
3. $\forall i \notin \{i_{\text{reg}}, i_{\text{stk}}, i_\ell\}, \Theta'(i) = W.\Theta(i)$;
4. $\Theta'(i_{\text{reg}}) = \text{island}_{\text{reg}}(s, W.k)$ with $s = (W.R_1[r_d \mapsto \ell_1], W.\chi_1[r_d : \rho_1(\text{ref}(\tau_1, \cdots, \tau_n))]$,
   $W.R_2[r_d \mapsto \ell_2], W.\chi_2[r_d : \rho_2(\text{ref}(\tau_1, \cdots, \tau_n))]$).
5. $\Theta'(i_{\text{stk}}) = \text{island}_{\text{stk}}(s, W.k)$ with $s = (S_1, S_2, \sigma)$;
6. $\Theta'(i_\ell)$ is such that $\forall W'.W' \subseteq W$.

$$(\ell_1, \ell_2) \in W'(i_\ell).\text{bij}(W'(i_\ell).s) \wedge \exists \varphi_M. \text{currentMR}(W'(i_\ell)) = \varphi_M \otimes \{ (W', \ell_1 \mapsto h_1), (W', \ell_2 \mapsto h_2) \} \in \text{MemAtom} | (W', h_1, h_2) \in H|_{\text{ref}(\tau_1, \cdots, \tau_n)}$$

We claim that $W' \in \text{World}$. Recall that we have $(W, w_{11}, w_{21}) \in W.\tau$. Thus we have that $W.\Psi_j, \ell_j \vdash W.\psi_j : \rho_j(\tau_i)$. Given our choice of $W'$ and since $W \in \text{World}$, it follows that $W'.\Psi_j, \ell_j \vdash W'.\psi_j : \rho_j(\tau_i)$ which is sufficient to establish our claim.

Moreover, note that $W' \supseteq W$ and $W' \supseteq \text{pub}, W$. Both follow immediately given our choice of $W'$ and the definition of $\text{island}_{\text{reg}}$ and $\text{island}_{\text{stk}}$.

We proceed by showing that $(M_1', M_2') : W'$ and then instantiating our final hypothesis

$$\Psi; \Delta; \chi[r_d : \text{ref}(\tau)]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$$

– We claim that $(M_1', M_2') : W'$. We prove this claim by establishing the following:

   $* \vdash M_1' : W'.\Phi_1$ and $* \vdash M_2' : W'.\Phi_2$. Note the following facts:

   $W'.\chi_1 = W.\chi_1[r_d : \rho_1(\tau)]$ and $W'.\chi_2 = W.\chi_2[r_d : \rho_2(\tau)]$;

   $W'.\sigma_1 = \sigma$ and $W'.\sigma_2 = \sigma$;

   $W'.\Psi_1 = W.\Psi_1, \ell_1 : \text{ref}(\tau_1, \cdots, \tau_n)$ and $W'.\Psi_2 = W.\Psi_2, \ell_2 : \text{ref}(\tau_1, \cdots, \tau_n)$. 


Both claims follow easily from \((M_1, M_2) : W\), the above facts, the fact that we have initialized location \(l\) in both memories with well-typed words, i.e., with tuples of types \(\rho_1((\tau_0, \ldots, \tau_n))\) and \(\rho_2((\tau_0, \ldots, \tau_n))\), and the fact that we have updated registers \(r_d\) to hold locations that point to well-typed words, i.e., the locations \(\ell_1\) and \(\ell_2\) that point to words of types \(\rho_1((\tau_0, \ldots, \tau_n))\) and \(\rho_2((\tau_0, \ldots, \tau_n))\).

* We assume that \(W.k > 0\) (and thus \(W'.k > 0\)). We must show that

\[ (\langle W', M'_1, M'_2 \rangle) \in \bigotimes \{ \text{currentMR}(\theta) \mid \theta \in W'.\Theta \} \]

The latter follows from \((M_1, M_2) : W\) and monotonicity of MemRel given that we establish the following claims:

1. For island \(i_{\text{reg}}\), \(\langle W', W.R_1[r_d \mapsto w_{1i}] | W.R_2[r_d \mapsto w_{2i}] \rangle \in \text{currentMR}(W'(i_{\text{reg}}))\);
2. For island \(i_{\text{stk}}\), \(\langle W', S'_1, S'_2 \rangle \in \text{currentMR}(W'(i_{\text{stk}}))\);
3. For island \(i_{\ell}\), \(\langle W', \{\ell_1 \mapsto \langle w_{11}, \ldots, w_{1n} \rangle\}, \{\ell_2 \mapsto \langle w_{21}, \ldots, w_{2n} \rangle\} \rangle \in \text{currentMR}(\Theta'(i_{\ell}))\).

For the first claim recall that \(W.R_j = W.R_j[r_d \mapsto w_{ji}]\). Thus we obtain trivially that \(\langle W', W.R_1[r_d \mapsto w_{1i}], W.R_2[r_d \mapsto w_{2i}] \rangle \in \{\langle \tilde{W}, W'.R_1, W'.R_2 \rangle \mid \tilde{W} \in \text{World}_{W.k}\}\).

From the latter, we establish directly our claim that

\[ (\langle W', W.R_1[r_d \mapsto w_{1i}] | W.R_2[r_d \mapsto w_{2i}] \rangle) \in \text{currentMR}(W'(i_{\text{reg}})) \]

For the second claim recall that \(W'.S_i = S'_i\). Thus we obtain trivially that \(\langle W', S'_1, S'_2 \rangle \in \{\langle \tilde{W}, W'.S_1, W'.S_2 \rangle \mid \tilde{W} \in \text{World}_{W.k}\}\). From the latter, we establish directly our claim that

\[ (\langle W', S'_1, S'_2 \rangle) \in \text{currentMR}(W'(i_{\text{stk}})) \]

For the fourth claim recall that we have \((W, w_{1i}, w_{2i}) \in W[\tau_i]\). Thus, from monotonicity (lemma 1.8), we obtain \(\langle W', w_{1i}, w_{2i} \rangle \in W'[\tau_i]\). Moreover, by construction island \(W'(i_{\ell})\) is such that \(\forall W' \supseteq W:\)

\[ (\ell_1, \ell_2) \in W'(i_{\ell}).\text{bij}(W'(i_{\ell}).s) \land \]

\[ \exists \varphi_M. \text{currentMR}(W'(i_{\ell})) = \varphi_M \otimes \]

\[ \{ (\tilde{W}, \{\ell_1 \mapsto h_1\}, \{\ell_2 \mapsto h_2\}) \} \in \text{MemAtom} \mid (\tilde{W}, h_1, h_2) \in \mathcal{H}\mathcal{V}\{\tau_1, \ldots, \tau_n\}\} \]

From the latter, we establish directly our claim that

\[ (\langle W', \{\ell_1 \mapsto \langle w_{11}, \ldots, w_{1n} \rangle\}, \{\ell_2 \mapsto \langle w_{21}, \ldots, w_{2n} \rangle\} \rangle) \in \text{currentMR}(\Theta'(i_{\ell})) \]

Next, we instantiate our final hypothesis with \(W'\) and \(\rho\). Note that \(\rho \in \mathcal{D}[\Delta]\) by assumption, \(\tilde{W} \in \mathcal{H}[\Psi]\) by heap monotonicity (Lemma 1.9). Let \(W''\) that is the same as \(W'\) but \(W''(i_{\text{stk}}) = W(i_{\text{stk}}) W'(i_{\text{stk}}) = W(i_{\text{stk}})\). We claim that \(\text{currentMR}(W')(i_{\text{stk}}) \subseteq W''(i_{\text{stk}}) \subseteq W(i_{\text{stk}})\).

Clearly \(W'' \supseteq W'\) and \(W' \supseteq W''\). We claim that \(\text{currentMR}(W''(i_{\text{stk}})) \subseteq W''(S[\sigma])\rho\) by the second case of lemma 1.21 since \(W''(i_{\text{stk}}) = \text{island}_{\text{stk}}(s, W, k)\) with \(s = (S_1, S_2, S_2)\) where \(W''(i_{\text{stk}}) = \text{island}_{\text{stk}}(s, W, k)\) with \(s = (w_{11} :: \ldots :: w_{1n} :: S_1, \tau_1 :: \ldots :: \tau_n :: \sigma, w_{21} :: \ldots :: w_{2n} :: S_2, \tau_1 :: \ldots :: \tau_n :: \sigma)\). Moreover we claim that \(\text{currentMR}(W''(i_{\text{reg}})) \subseteq W''(S[\sigma])\rho\) and \(W'' \supseteq W''\).

The second follows directly from the second case of lemma 1.20 since, above, we have established that \((W', w_{1i}, w_{2i}) \in W'[\tau_i]\) and \(\text{currentMR}(W''(i_{\text{reg}})) \subseteq W''(S[\sigma])\rho\) and \(W'' \supseteq W''\).

Hence, we have that

\[ (W', (\rho_1(I_1, \cdot)), (\rho_2(I_2), \cdot)) \in \mathcal{E}[q \vdash \tau; \sigma]\rho \]

Instantiate the above with \(E_1\) and \(E_2\). Note that we have \((W', E_1, E_2) \in K[q \vdash \tau; \sigma]\rho\), which follows from \((W, E_1, E_2) \in K[q \vdash \tau; \sigma]\rho\) by monotonicity for evaluation contexts.
(Lemma 1.10), since our choice of $W'$, $W' \equiv_{\text{pub}} W$ and our third hypothesis, $q \neq r_d$, and the preliminary fact that if $q = i$ then $i \geq n$ let us easily establish all the premises of that lemma.

Hence, we have

$$(W', E_1[\rho_1(I_1, \cdot)], E_2[(\rho_2(I_2), \cdot)]) \in O.$$  

Instantiate the latter with $M'_1$ and $M'_2$, noting that $(M'_1, M'_2) : W'$. Hence, we have

$$((M'_1 | E_1[(\rho_1(I_1), \cdot)]) \downarrow \land (M'_2 | E_2[(\rho_2(I_2), \cdot)]) \downarrow \lor$$

$$\text{(running}(W.k, (M'_1 | E_1[(\rho_1(I_1), \cdot)])) \land \text{running}(W.k, (M'_2 | E_2[(\rho_2(I_2), \cdot)])))$$

which implies what we needed to show.

\[\square\]

**Lemma 1.49 (Allocate Immutable Tuple)**

If $\text{len}(\overline{\tau}) = n$, $q \neq r_d$, and $\Psi; \Delta; \chi[r_d : \text{box}(\overline{\tau})]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$, then $\Psi; \Delta; \chi; \overline{\tau} :: \sigma; q \vdash \text{baloc} r_d, n; I_1 \approx \text{baloc} r_d, n; I_2$.

**Proof**

Clearly, $\Psi; \Delta; \chi; \overline{\tau} :: \sigma; q \vdash \text{baloc} r_d, n; I_1$ and $\Psi; \Delta; \chi; \overline{\tau} :: \sigma; q \vdash \text{baloc} r_d, n; I_2$.

Consider arbitrary $W$ and $\rho$ such that $\Psi \in \mathcal{H}[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{\text{reg}})) \in W \Rightarrow \chi[\rho], \text{and currentMR}(W(i_{\text{stk}})) \in W \Rightarrow \sigma[\rho]$. We need to show that

$$(W, \rho_1((\text{baloc} r_d, n; I_1, \cdot)), \rho_2((\text{baloc} r_d, n; I_2, \cdot)))$$

$$= (W, (\text{baloc} r_d, n; \rho_1(I_1, \cdot)), (\text{baloc} r_d, n; \rho_2(I_2, \cdot))) \in \mathcal{E}[q \vdash \text{ret}-\text{type}(q, \chi, \overline{\tau} :: \sigma)]$$

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running($0, \langle M_i | E_i \rangle$).

In the following, assume $W.k > 0$ and let $\tau_r; \sigma_r = \text{ret}-\text{type}(q, \chi, \sigma)$.

We proceed by unfolding the definition of $\mathcal{E}[q \vdash \tau_r; \sigma_r] \rho$ and proving the resulting obligations:

- **For arbitrary $E_1$ and $E_2$, $(W, E_1, E_2) \in K[q \vdash \tau_r; \sigma_r] \rho$ implies**

$$((W, E_1[(\text{baloc} r_d, n; \rho_1(I_1, \cdot))], E_2[(\text{baloc} r_d, n; \rho_2(I_2, \cdot))]) \in O.$$  

From the definition of $O$, this requires to show that given arbitrary $(M_1, M_2) : W$, $(M_1 | E_1[(\text{baloc} r_d, n; \rho_1(I_1, \cdot))]) \downarrow \text{iff } (M_2 | E_2[(\text{baloc} r_d, n; \rho_2(I_2, \cdot))]) \downarrow$. We proceed by running each expression for one step and then using the resulting memories, to construct a world that we can then use with our second hypothesis, $\Psi; \Delta; \chi[r_d : \text{ref}(\overline{\tau})]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$:

- Assume that our two arbitrary memories are of the form $M_1 = (H_1, R_1, S_1)$ and $M_2 = (H_2, R_2, S_2)$.

From $(M_1, M_2) : W$, since $W.k > 0$, we have $(>W, M_1, M_2) \in \otimes \{\text{currentMR}(\theta) \mid \theta \in W.\theta\}$. From the latter, we have the following fact for island $i_{\text{stk}}$: $(>W, S_1 \upharpoonright, S_2 \upharpoonright) \in \text{currentMR}(W(i_{\text{stk}}))$. From the latter it follows that $S_1 = W.S_1$.

From the reduction semantics of our language we obtain that:

$$\langle M_1 | E_1[(\text{baloc} r_d, n; \rho_1(I_1, \cdot))] \rangle \rightarrow \langle M'_1 | E_1[(\rho_1(I_1), \cdot)] \rangle$$

48
and
\[ (M_2 \mid E_2[\text{balloc } r_d, n; \rho_1(I_2), i]) \rightarrow (M'_2 \mid E_2[\rho_2(I_2), i]) \]

where \( M'_1 = (H'_1, R'_1, S'_1) \) with \( R'_1 = R_1[r_d \mapsto \ell_1], H'_1 = H_1[\ell_1 \mapsto w]\) and \( S_1 = w \vdash S'_1 \), and \( M'_2 = (H'_2, R_2, S'_2) \) with \( R'_2 = R_2[r_d \mapsto \ell_1], H'_2 = H_2[\ell_1 \mapsto w] \) and \( S_2 = w \vdash S'_2 \).

Let \( s_{reg} = (W.R_1[r_d \mapsto \ell_1], W.\chi[r_d; \rho_1(\text{box } \langle \tau \rangle)], W.R_2[r_d \mapsto \ell_2], W.\chi[r_d; \rho_2(\text{box } \langle \tau \rangle)]) \), \( s_{box} = (W.H_1[\ell_1 \mapsto w\boxwedge], W.H_2[\ell_2 \mapsto w\boxwedge]) \) and \( W' = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta)[i_{reg} \rightarrow \text{island}_{reg}(s_{reg}, W.k), i_{box} \rightarrow \text{island}_{box}(s_{box}, W.k)] \) where \( W.S_1 = w \vdash S'_1 \).

We claim that \( W' \in \text{World} \). From the restriction currentMR(\( W(i_{stk}) \)) \( \in W.S[\Psi :: \sigma] \rho \), we have that
\[ (W.W.S_1(j), W.S_2(j)) \in W[\tau_j \rho] \]
for \( j \in \{1..n\} \).

And thus \( (W, W_1, W_2) \in W.H[\Psi] \rho \).

Thus we have that \( W.\Psi_i \vdash \ell_i; \rho_i(\text{box } \langle \tau \rangle) \) and \( W.\Psi_i \vdash \ell_i; \rho_i(\text{box } \langle \tau \rangle) \), which is sufficient to establish our claim.

Next, to use our third hypothesis, we note that \( W' \in H[\Psi] \) by heap monotonicity (Lemma 1.9) and that currentMR(\( W'(i_{stk}) \)) \( \in W.\Psi[r_d; \text{box } \langle \tau \rangle] \rho \) by Lemma 1.21 since \( W'(i_{stk}) = W(i_{stk}) \). We also claim that currentMR(\( W'(i_{reg}) \)) \( \in W.\Psi[r_d; \text{box } \langle \tau \rangle] \rho \). We have that
\[ (W', \ell_1, \ell_2) \in W.\text{box } \langle \tau \rangle] \rho \]
which, using Lemma 1.20, is sufficient to establish our claim.

Therefore we can apply our third hypothesis to \( W' \) and \( \rho \), finding that
\[ (W', \rho_1(I_1, \cdot), \rho_2(I_2, \cdot)) \in \Psi[\tau \rightarrow \text{ret-type}(\chi, \sigma)] \rho. \]

By unfolding that definition we derive that given \( (M'_1, M'_2) : W', (W', E_1[\rho_1(I_1), \cdot]), E_2[\rho_2(I_2), \cdot]) \in \mathcal{O} \). By construction \( (M'_1, M'_2) : W' \) and thus by unfolding the definition of \( \mathcal{O} \), we obtain
\[ (M'_1 \mid E_1[\rho_1(I_1), \cdot]) \downarrow \quad \text{and} \quad (M'_2 \mid E_2[\rho_2(I_2), \cdot]) \downarrow, \]

or running \( W'.k, (M'_1 \mid E_1[\rho_1(I_1), \cdot]) \)

and running \( W'.k, (M'_2 \mid E_2[\rho_2(I_2), \cdot]) \).

In the first case, it is straightforward to derive from the reduction semantics that \( (M_1 \mid E_1[\text{balloc } r_d, n; \rho_1(I_1), \cdot]) \downarrow \quad \text{and} \quad (M_2 \mid E_2[\text{balloc } r_d, n; \rho_2(I_2), \cdot]) \downarrow. \)

In the second case, since \( W'.k = W.k \), we derive that
\[ \text{running}(W.k, (M_1 \mid E_1[\text{balloc } r_d, n; \rho_1(I_1), \cdot]) \quad \text{and} \quad \text{running}(W.k, (M_2 \mid E_2[\text{balloc } r_d, n; \rho_2(I_2), \cdot]). \]

Thus we conclude that:
\[ (W, E_1[\text{balloc } r_d, n; \rho_1(I_1), \cdot]), E_2[\text{balloc } r_d, n; \rho_2(I_2), \cdot]) \in \mathcal{O}. \]

\[ \square \]

**Lemma 1.50 (Move)**

If \( \Psi; \Delta; \chi \vdash u_1 \approx_n u_2 : \tau, q \neq r_d \), and \( \Psi; \Delta; \chi[r_d; \tau]; \sigma; q \vdash I_1 \approx _1 I_2 \), then \( \Psi; \Delta; \chi; \sigma; q \vdash mv r_d, u_1; I_1 \approx _1 mv r_d, u_2; I_2 \).

**Proof**

Clearly, \( \Psi; \Delta; \chi; \sigma; q \vdash mv r_d, u_1; I_1 \) and \( \Psi; \Delta; \chi; \sigma; q \vdash mv r_d, u_2; I_2 \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in H[\Psi], \rho \in D[\Delta] \), currentMR(\( W(i_{reg}) \)) \( \in W.\mathcal{R}[\chi]\rho \), and currentMR(\( W(i_{stk}) \)) \( \in W.S[\sigma]\rho \). We need to show that
\[ (W, \rho_1((mv r_d, u_1; I_1, \cdot)), \rho_2((mv r_d, u_2; I_2, \cdot))) \]
\[ = (W. (mv r_d, \rho_1(u_1); \rho_1(I_1), \cdot), (mv r_d, \rho_2(u_2); \rho_2(I_2), \cdot)) \in \Psi[\tau \rightarrow \text{ret-type}(q, \chi, \sigma)] \rho. \]

Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_i \) and memories \( M_i \), we can immediately show that running(0, \( (M_i \mid E_i[e_i]) \)).

In the following, assume \( W.k > 0 \) and let \( \tau; \sigma = \text{ret-type}(q, \chi, \sigma) \).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.
Lemma 1.51 (Move Return Address)

We claim that \((W, (\text{mv } r_d, \rho_1(I_1), \cdot), (\text{mv } r_d, \rho_2(u_2)); \rho_2(I_2), \cdot)) \in \text{TermAtom}[q \vdash \tau_r; \sigma_r]_\rho\).

To establish this, we must show \(W; \Psi_i; \cdot \vdash W; \chi; \sigma; q \vdash (\text{mv } r_d, \rho_1(I_1), \cdot); \rho_1(\tau_r); \rho_1(\sigma_r).\)
The latter follows from \(\Psi; \Delta; \chi; \sigma; q \vdash (\text{mv } r_d, u_1); I_1\) using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.

Let \(s = (W, R_1[r_d \mapsto W, \hat{R}_1(\rho_1(u_1))]), W, \chi_1[w_d; \rho_1(\tau)], W, R_2[r_d \mapsto W, \hat{R}_2(\rho_2(u_2))], W, \chi_2[w_d; \rho_2(\tau)]\)
and \(W' = (W, k, W, \Psi_1, W, \Psi_2, W, \Theta[\text{reg } \mapsto \text{island}_{\text{reg}}(s, W, k)]).\)

We claim that \(W' \in \text{World}\). Instantiating the first hypothesis with \(W\) and \(\rho\), we have that
\[
(W, W, \hat{R}_1(\rho_1(u_1))), W, \hat{R}_2(\rho_2(u_2))) \in W[\tau]_\rho
\]
Thus we have that \(W; \Psi_i; \cdot \vdash W, \hat{R}_1(\rho_1(u_1)); \rho_1(\tau).\) Given our choice of \(W'\) and since \(W \in \text{World}\), it follows that \(W; \Psi_i; \cdot \vdash W, \hat{R}_1(\rho_1(u_1)); \rho_1(\tau)\) which is sufficient to establish our claim.

Note that \(W' \models_{\text{pub}} W\). The latter is immediate given our choice of \(W'\) and the definition of \(\text{island}_{\text{reg}}\).

Consider arbitrary \((M_1, M_2) : W\). Let \(M = (H, R, S)\).

From \((M_1, M_2) : W\), since \(W, k > 0\), we have \((\gg W, M_1, M_2) \in \bigotimes \{\text{currentMR}(\theta) | \theta \in W; \Theta\}\).
From the latter, we have the following fact for island \(\text{island}_{\text{reg}}: (\gg W, R_1, R_2) \in \text{currentMR}(W[\text{reg}]).\)
From the latter it follows that \(R = W, R\) and, hence, \(R(\rho_1(u_1)) = W, \hat{R}_1(\rho_1(u_1))\).

Note that
\[
\langle (H, R, S) | (\text{mv } r_d, \rho_1(u_1); \rho_1(I_1), \cdot) \rangle \mapsto (\langle (H, R, S) | (\rho_1(I_1), \cdot) \rangle
\]
Let \(M' = (H, R, S) \mapsto (\rho_1(u_1)), S)\). Given our choice of \(W'\), since \(R = W, R\), we have that \((M', M_2) : W'').\)

Note that \(W, k \leq W', k + 1\).

Note that \(q \neq e\), which follows from the third hypothesis since it is a side condition of the instruction typing rules. Further, since \(q \neq r_d\), which was the only memory location changed between \(W\) and \(W'\), we can see that the the return address does not change.

Next, to use our third hypothesis, we note that \(W' \in H[\Psi]_W\) by heap monotonicity (Lemma 1.9) and that \(\text{currentMR}(W'(\text{isstk})) \in W[S][\sigma]_\rho\) by Lemma 1.21 since \(W'(\text{isstk}) = (\rho_1(\tau_r))_\rho.\) We also claim that \(\text{currentMR}(W'[\text{reg}]) \in W[\tau]_\rho.\) From above, we have that
\[
(W, W, \hat{R}_1(\rho_1(u_1)), W, \hat{R}_2(\rho_2(u_2))) \in W[\tau]_\rho.
\]
Hence, by monotonicity and the definition of \(W'\) we have
\[
(W', W, \hat{R}_1(\rho_1(u_1)), W, \hat{R}_2(\rho_2(u_2))) = (W', W, R_1(r_d), W', R_2(r_d)) \in W[\tau]_\rho,
\]
which, using Lemma 1.20, is sufficient to establish our claim.

Therefore we can apply our third hypothesis to \(W'\) and \(\rho\), finding that
\[
(W', \rho_1((I_1, \cdot)), \rho_2((I_2, \cdot))) \in E[q \vdash \text{ret-type}(q, \chi, \sigma)]_\rho.
\]

Now, the result follows by Lemma 1.11. \(\square\)

Lemma 1.51.1 (Move Return Address)

If \(\chi(r_s) = \tau\) and \(\Psi; \Delta; \chi[w_d; \tau]; \sigma; r_d \vdash I_1 \approx I_2\), then \(\Psi; \Delta; \chi; \sigma; r_s \vdash \text{mv } r_d, r_s; I_1 \approx I \text{mv } r_d, r_s; I_2\).

\textbf{Proof}

Clearly, \(\Psi; \Delta; \chi; \sigma; r_s \vdash \text{mv } r_d, r_s; I_1\) and \(\Psi; \Delta; \chi; \sigma; r_s \vdash \text{mv } r_d, r_s; I_2\).

Consider arbitrary \(W\) and \(\rho\) such that \(W \in H[\Psi]_W, \rho \in D[\Delta], \text{currentMR}(W[\text{reg}]) = W[S][\sigma]_\rho, \text{and currentMR}(W[\text{isstk}]) = W[S][\sigma]_\rho.\) We need to show that
\[
(W, \rho_1((\text{mv } r_d, r_s; I_1, \cdot)), \rho_2((\text{mv } r_d, r_s; I_2, \cdot)))
= (W, (\text{mv } r_d, r_s; \rho_1(I_1, \cdot)), (\text{mv } r_d, r_s; \rho_2(I_2, \cdot))) \in E[r_s \vdash \text{ret-type}(r_s, \chi, \sigma)]_\rho.
\]
Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_i \) and memories \( M_i \), we can immediately show that running \((\langle \rho_i, E_i, [e_i] \rangle)\) finishes.

In the following, assume \( W.k > 0 \) and let \( \tau; \sigma \vdash \text{ret-type}(r_s, \chi, \sigma) \).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that \((W, (\text{mv} \ r_d, r_s; \rho_1(I_1), \cdot), (\text{mv} \ r_d, r_s; \rho_2(I_2), \cdot)) \in \text{TermAtom}[r_s \vdash \tau; \sigma]\). To establish this, we must show \( W, \Psi_i; \cdot, W, \chi; W, \sigma; r_s \vdash (\text{mv} \ r_d, r_s; \rho_1(I_1), \cdot); \rho_2(I_2); \rho_1(I_1) \).
  - The latter follows from \( \Psi; [\varphi; \cdot] : W, W.\chi; W, \sigma; r_s \vdash (\text{mv} \ r_d, r_s; \rho_1(I_1), \cdot); \rho_2(I_2); \rho_1(I_1) \).
  - The latter follows by Lemma 1.11.

Let \( s = \langle W, R_1[r_d \mapsto W, R_1(r_s)], W, \chi, r_d \mapsto W, R_2(r_s), W, \chi, r_d \mapsto W, R_2(r_s) \rangle \) and \( W' = (W, k, W, \Psi_i, W, \sigma; W, \Theta[i_{reg} \mapsto \text{island}_{reg}(s, W,k)]) \).

We claim that \( W' \in \text{World} \). By our first hypothesis, we have that \( (r_s : \tau) \in \chi \). It then follows from \( \text{currentMR}(W(i_{reg})) \subseteq W \cdot R[\chi] \rho \) and the definition of \( \text{island}_{reg} \) that
\[
(W, W. R_1(r_s), W. R_2(r_s)) \in W[\tau] \rho
\]
Thus we have that \( W, \Psi_i; \cdot \vdash W, R_i(r_s); \rho_i(\tau) \). Given our choice of \( W' \) and since \( W \in \text{World} \), it follows that \( W', \Psi_i; \cdot \vdash W, R_i(r_s); \rho_i(\tau) \) which is sufficient to establish our claim.

- Note that \( W' \models \text{pub} \ W \). The latter is immediate given our choice of \( W' \) and the definition of \( \text{island}_{reg} \).

- Consider arbitrary \((M_1, M_2) : W \). Let \( M_i = (H_i, R_i, S_i) \).

From \((M_1, M_2) : W \), since \( W.k > 0 \), we have \((\vdash W, M_1, M_2) \in \bigotimes\{ \text{currentMR}(\theta) \mid \theta \in W, \Theta \} \).

From the latter, we have the following fact for island \( i_{reg} \): \((\vdash W, R_1(r_s), R_2(r_s)) \in \text{currentMR}(W(i_{reg})) \).

From the latter it follows that \( R_i = W, R_i \).

Note that
\[
\langle (H_i, R_i, S_i), (\text{mv} \ r_d, r_s; \rho_1(I_1), \cdot) \rangle \mapsto (\langle H_i, R_i[r_d \mapsto R_i(r_s)], S_i \rangle, (\rho_1(I_1), \cdot))
\]
Let \( M_i' = (H_i, R_i[r_d \mapsto R_i(r_s)], S_i) \). Given our choice of \( W' \), since \( R_i = W, R_i \), we have that \((M_1', M_2') : W' \).

- Note that \( W.k \leq W'.k + 1 \).

- Note that \( \text{ret-addr}_1(W, r_s) = \text{ret-addr}_1(W', r_d) \) and \( \text{ret-addr}_2(W, r_s) = \text{ret-addr}_2(W', r_d) \).

- Next, to use our second hypothesis, we note that \( W' \in H[\Psi] \) by heap monotonicity (Lemma 1.9) and that \( \text{currentMR}(W'(i_{stk})) \subseteq W' \cdot R[\chi] \rho \) by Lemma 1.21 since \( W'(i_{stk}) = W(i_{stk}) \). We also claim that \( \text{currentMR}(W'(i_{reg})) \subseteq W' \cdot R[\chi] \rho \). From \((W, W, R_1(r_s), W, R_2(r_s)) \in W[\tau] \rho \) by monotonicity and the definition of \( W' \), we have
\[
(W', W, R_1(r_s), W, R_2(r_s)) = (W', W, R_1(r_d), W', R_2(r_d)) \in W[\tau] \rho,
\]
which, using Lemma 1.20, is sufficient to establish our claim.

Therefore we can apply our second hypothesis to \( W' \) and \( \rho \), finding that
\[
(W', \rho_1(I_1, \cdot), \rho_2(I_2, \cdot)) \in E[r_d \vdash \text{ret-type}(r_d, \chi[r_d : \tau], \sigma)] \rho = E[r_d \vdash \text{ret-type}(r_s, \chi, \sigma)] \rho.
\]
Now, the result follows by Lemma 1.11. \( \square \)

Lemma 1.52 (Unpack)
If \( \Psi; \Delta; \chi \vdash u_1 \approx u_2 \cdot \exists \alpha, \tau, q \neq r_d \) and \( \Psi; \Delta; \alpha; \chi[r_d : \tau]; \sigma; q \vdash I_1 \approx I_2 \), then \( \Psi; \Delta; \chi; \sigma; q \vdash \text{unpack}(\alpha, r_d) u_1; I_1 \approx \text{unpack}(\alpha, r_d) u_2; I_2 \).

Proof
Clearly, \( \Psi; \Delta; \chi; \sigma; q \vdash \text{unpack}(\alpha, r_d)\ u_1; I_1 \) and \( \Psi; \Delta; \chi; \sigma; q \vdash \text{unpack}(\alpha, r_d)\ u_2; I_2 \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in H[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{reg})) \equiv W. R[\chi] \rho, \) and \( \text{currentMR}(W(i_{stk})) \equiv W. S[\sigma] \rho \). We need to show that

\[
(W, \rho_1((\text{unpack}(\alpha, r_d)\ u_1; I_1, \cdot)), \rho_2((\text{unpack}(\alpha, r_d)\ u_2; I_2, \cdot))) = (W, (\text{unpack}(\alpha, r_d)\ \rho_1(u_1); \rho_1(1), \cdot), (\text{unpack}(\alpha, r_d)\ \rho_2(u_2); \rho_2(1), \cdot)) \in E[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho.
\]

Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_i \) and memories \( M_i \), we can immediately show that running \((0, (M_i | E_i[e_i]))\).

In the following, assume \( W.k > 0 \) and let \( \tau_i; \sigma_i = \text{ret-type}(q, \chi, \sigma) \).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that \( (W, \tau_i; \sigma_i) \in \text{TermAtom}[q \vdash \tau_i; \sigma_i] \rho. \) To establish this, we must show \( W. \Psi_i; \cdot \vdash W. \chi_i; W. \sigma_i; \rho_i(q) \vdash (\text{unpack}(\alpha, r_d)\ \rho_i(u_1); \rho_i(1), \cdot); \rho_i(\tau_i); \rho_i(\sigma_i) \).

  The latter follows from \( \Psi; [\Delta; \chi; \sigma; q \vdash \text{unpack}(\alpha, r_d)\ u_1; I_1 \) using the component typing rule and the properties of \( W \) and \( \rho \) that we have by assumption.

- Let \( s = (W, R_1[r_d \mapsto w_1], W, \chi_1[r_d \mapsto \rho_1(\tau)], W, R_2[r_d \mapsto w_2], W, \chi_2[r_d \mapsto \rho_2(\tau)]) \) where \( W. \hat{R}_1(u_i) = \text{pack}(\tau', \chi_i) \in E[\exists \alpha. \tau] \) and \( W' = (W, k, W. \Psi_1, W. \Psi_2, W. \Theta[i_{reg} \mapsto \text{island}_{reg}(s, W.k)]) \).

  We claim that \( W' \in \text{World} \). Instantiating the first hypothesis with \( W \) and \( \rho \), we have that

\[
(W, W. \hat{R}_1(\rho_1(u_1)), W. \hat{R}_2(\rho_2(u_2))) \in W[\exists \alpha. \tau] \rho
\]

Thus we have that \( W. \Psi_i; \cdot \vdash W. \hat{R}_1(\rho_1(u_1)); \rho_1(\exists \alpha. \tau) \). Given our choice of \( W' \) and since \( W \in \text{World} \), it follows that \( W'. \Psi_i; \cdot \vdash W'. \hat{R}_1(\rho_1(u_1)); \rho_1(\exists \alpha. \tau) \) which is sufficient to establish our claim.

- Note that \( W' \models_{\text{pub}} W \). The latter is immediate given our choice of \( W' \) and the definition of island_{reg}.

- Consider arbitrary \((M_1, M_2) : W \). Let \( M_i = (H_i, R_i, S_i) \).

  From \((M_1, M_2) : W \), since \( W.k > 0 \), we have \((\mapsto W, M_1, M_2) \in \bigotimes \{ \text{currentMR}(\theta) | \theta \in W. \Theta \}\). From the latter, we have the following fact for island \( i_{reg} \): \((\mapsto W, R_1[1], R_2[2]) \in \text{currentMR}(W(i_{reg})) \).

  From the latter it follows that \( R_1 = W. R_i \).

  Note that

\[
((H_i, R_i[1], S_i) | (\text{unpack}(\alpha, r_d)\ \rho_i(u_1); \rho_i(1), \cdot)) \mapsto (H_i, R_i[r_d \mapsto w_1], S_i) | (\rho_i[\alpha \mapsto \tau'](I_i), \cdot))
\]

Where \( W. \hat{R}_1(\rho_i(u_1)) = \text{pack}(\tau', \alpha) \). Let \( M'_i = (H_i, R_i[r_d \mapsto w_1], S_i) \). Note that \((M'_1, M'_2) : W' \).

- Note that \( W.k \leq W'.k + 1 \).

- Note that \( q \neq e \), which follows from the third hypothesis since it is a side condition of the instruction typing rules. Further, since \( q \neq r_d \), which was the only memory location changed between \( W \) and \( W' \), we can see that the return address does not change.

- Next, to use our third hypothesis, we note that \( W' \in H[\Psi] \) by heap monotonicity (Lemma 1.9). From above, we have that \((W, W. \hat{R}_1(\rho_1(u_1)), W. \hat{R}_2(\rho_2(u_2))) = (W, \text{pack}(\tau', w_1) as \rho_1(\exists \alpha. \tau), \text{pack}(\tau', w_1) as \rho_1(\exists \alpha. \tau) \in W[\exists \alpha. \tau] \rho \).

  From the definition of \( W[\exists \alpha. \tau] \rho \), we have that \((W, w_1, w_2) \in W[\tau] \rho[\alpha \mapsto (\tau', \tau, \varphi_w)] \) for some \( \varphi_w \).

  We choose \( \rho' = \rho[\alpha \mapsto (\tau', \tau, \varphi_w)] \in D[\Delta, \alpha] \).

  We claim that \( \text{currentMR}(W'(i_{stk})) \equiv W. S[\sigma] \rho' \). From Lemma 1.21 we have \( \text{currentMR}(W'(i_{stk})) \equiv W. S[\sigma] \rho' \) since \( W'(i_{stk}) = W'(i_{stk}) \). From substitution, we know that \( S[\sigma] \rho' = S[\sigma[\tau'/\alpha]] \rho' \).

  Since \( \sigma \) is well-formed under \( \Delta, \alpha \) is not free and thus \( \text{currentMR}(W'(i_{reg})) \equiv W. S[\sigma] \rho' \).

  We also claim that \( \text{currentMR}(W'(i_{reg})) \equiv W. R[\chi[r_d : \tau][\tau'/\alpha]] \rho' \). From substitution, since \( \chi \) is well-formed under \( \Delta \), we know that this is equivalent to \( \text{currentMR}(W'(i_{reg})) \equiv W. R[\chi[r_d : \tau'[\tau'/\alpha]] \rho' \).

52
We can use Lemma 1.20 provided we can show that \((W', w_1, w_2) \in W \parallel [\tau' / \alpha] \parallel \rho\). But this, via substitution, is equivalent to showing that \((W', w_1, w_2) \in W \parallel \tau' \parallel \rho'\), which we have from the first hypothesis.

Therefore we can apply our third hypothesis to \(W'\) and \(\rho'\), finding that

\[
(W', \rho'_1((I_1, \cdot)), \rho'_2((I_2, \cdot))) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)]\parallel \rho'.
\]

Now, the result follows by Lemma 1.11.

\[\square\]

**Lemma 1.53 (Unfold)**

If \(\Psi; \Delta; \chi \vdash u_1 \approx u_2; \mu \alpha.\tau, q \neq r_d\), and \(\Psi; \Delta; \chi[r_d: \tau[\mu \alpha.\tau / \alpha]]; \sigma; q \vdash I_1 \approx I_2\), then \(\Psi; \Delta; \chi; q \vdash \text{unfold}_d, u_1; I_1 \approx \text{unfold}_d, u_2; I_2\).

**Proof**

Clearly, \(\Psi; \Delta; \chi; q \vdash \text{unfold}_d, u_1; I_1\) and \(\Psi; \Delta; \chi; q \vdash \text{unfold}_d, u_2; I_2\).

Consider arbitrary \(W\) and \(\rho\) such that \(W \in \mathcal{H}[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{reg})) \subseteq W \parallel \mathcal{R}[\chi] \parallel \rho\), and \(\text{currentMR}(W(i_{stk})) \subseteq W \parallel S[\sigma] \parallel \rho\). We need to show that

\[
(W, \rho_1((\text{unfold}_d, u_1; I_1, \cdot)), \rho_2((\text{unfold}_d, u_2; I_2, \cdot))) = (W, (\text{unfold}_d, \rho_1(u_1); \rho_1(I_1, \cdot)), (\text{unfold}_d, \rho_2(u_2); \rho_2(I_2, \cdot))) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)]\parallel \rho.
\]

In the following, let \(\tau_r; \sigma_r = \text{ret-type}(q, \chi, \sigma)\).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that \((W, (\text{unfold}_d, \rho_1(u_1); \rho_1(I_1, \cdot)), (\text{unfold}_d, \rho_2(u_2); \rho_2(I_2, \cdot))) \in \text{TermAtom}[q \vdash \tau_r; \sigma_r] \parallel \rho\). To establish this, we must show \(W, \Psi; \Delta; \chi, q \vdash \text{unfold}_d, u_1; I_1\) and \(W, \Psi; \Delta; \chi, q \vdash \text{unfold}_d, u_2; I_2\) using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.

- Let \(s = (W, R_1[r_d \mapsto w_1], W, \chi_1[r_d: \rho_1(\tau[\mu \alpha.\tau / \alpha])], W, R_2[r_d \mapsto w_2], W, \chi_2[r_d: \rho_2(\tau[\mu \alpha.\tau / \alpha])]\) where \(W, R_1(u_1) = \text{fold}_{\mu \alpha.\tau} w_1\) and \(W' = (W, k - 1, W, \Psi_1, W, \Psi_2, W, \Theta[i_{reg} \mapsto \text{island}_{reg}(s, W, W)]\).

- We claim that \(W' \in \text{World}\). Instantiating the first hypothesis with \(W\) and \(\rho\), we have that

\[
(W, W, \hat{R}_1(\rho_1(u_1)), W, \hat{R}_2(\rho_2(u_2))) \in W \parallel [\mu \alpha.\tau] \parallel \rho
\]

From the definition of \(W \parallel [\mu \alpha.\tau] \parallel \rho\), we have that \((W, w_1, w_2) \in \Rightarrow W \parallel [\tau[\mu \alpha.\tau / \alpha]] \parallel \rho\).

This means that \(W, \Psi; \cdot \vdash \rho_1(\tau[\mu \alpha.\tau / \alpha])\).

Given our choice of \(W'\) and since \(W \in \text{World}\), it follows that \(W, \Psi_1; \cdot \vdash \rho_1(\tau[\mu \alpha.\tau / \alpha])\) which is sufficient to establish our claim.

- Note that \(W' \supseteq_{pub} W\). The latter is immediate given our choice of \(W'\) and the definition of \(\text{island}_{reg}\).

- Consider arbitrary \((M_1, M_2) : W\). Let \(M_i = (H_i, R_i, S_i)\). Note that

\[
((H_i, R_i, S_i) \mid (\text{unfold}_d, \rho_i(u_1); \rho_i(I_i, \cdot)) \mapsto (H_i, R_i[r_d \mapsto w_i], S_i) \mid (\rho_i(I_i, \cdot), \cdot))
\]

Where \(W, \hat{R}_1(\rho_i(u_1)) = \text{fold}_{\mu \alpha.\tau} w_i\). Let \(M_i' = (H_i, R_i[r_d \mapsto w_i], S_i)\). Note that \((M_1', M_2') : W'\).

- Note that \(W, k \leq W', k + 1\).

- Note that \(q \neq e\), which follows from the third hypothesis since it is a side condition of the instruction typing rules. Further, since \(q \neq r_d\), which was the only memory location changed between \(W\) and \(W'\), we can see that the the return address does not change.
Next, to use our third hypothesis, we note that \( W' \in H[\Psi] \) by heap monotonicity (Lemma 1.9).

From above, we have that \( (W, W.R_1(\rho_1(u_1)), W.R_2(\rho_2(u_2))) = (W, \rho_1(\text{fold}_{\mu_\alpha.\tau} w_1), \rho_2(\text{fold}_{\mu_\alpha.\tau} w_2)) \in W[\mu_\alpha.\tau] \rho \).

From the definition of \( W[\mu_\alpha.\tau] \rho \), we have that \( (W, w_1, w_2) \in \text{MR}[\tau[\mu_\alpha.\tau/\alpha]] \rho \).

We claim that \( \text{currentMR}(W')(i_{\text{stk}}) \in W.S[\sigma] \rho \). This follows from Lemma 1.21.

We also claim that \( \text{currentMR}(W'(i_{\text{reg}})) \in W.R[\chi[4: \tau[\mu_\alpha.\tau/\alpha]]] \rho \). In order to show this, we need to show that \( (W', w_1, w_2) \in W[\tau[\mu_\alpha.\tau/\alpha]] \rho \). From the definition of \( W[\mu_\alpha.\tau] \rho \) we have that \( (\triangleright W, w_1, w_2) \in W[\tau[\mu_\alpha.\tau/\alpha]] \rho \). But \( W' \supseteq \triangleright W \), since \( W' \supseteq W \) and \( W').k \leq W.k - 1 \), which means from monotonicity, the required condition holds.

Therefore we can apply our third hypothesis to \( W' \) and \( \rho \), finding that

\[
(W', \rho_1((I_1, \cdot)), \rho_2((I_2, \cdot))) \in E[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho.
\]

Now, the result follows by Lemma 1.11.

\[ \square \]

**Lemma 1.54 (Allocate Stack Space)**

If \( \Psi; \Delta; \chi; \text{unit} :: \cdots^n :: \text{unit} :: \sigma; \text{inc}(q, n) \vdash I_1 \approx_I I_2 \), then \( \Psi; \Delta; \chi; \sigma; q \vdash \text{alloc n}; I_1 \approx_I \text{alloc n}; I_2 \).

**Proof**

Clearly, \( \Psi; \Delta; \chi; \sigma; q \vdash \text{alloc n}; I_1 \) and \( \Psi; \Delta; \chi; \sigma; q \vdash \text{alloc n}; I_2 \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in H[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{\text{reg}})) \in W.R[\chi] \rho \), and \( \text{currentMR}(W(i_{\text{stk}})) \in W.S[\sigma] \rho \). We need to show that

\[
(W, \rho_1((\text{alloc n}; I_1, \cdot)), \rho_2((\text{alloc n}; I_2, \cdot))) = (W, (\text{alloc n}; \rho_1(I_1), \cdot), (\text{alloc n}; \rho_2(I_2), \cdot)) \in E[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho.
\]  

(14)

Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_i \) and memories \( M_i \), we can immediately show that running \((0, (M_i \mid E_i[e_i]))\) in the following, assume \( W.k > 0 \) and let \( \tau_1; \tau_2 = \text{ret-type}(q, \chi, \bar{\tau} :: \sigma) \).

We now prove (13).

We proceed by unfolding the definition of \( E[q \vdash \tau_1; \tau_2] \rho \) and proving the resulting obligations:

- We claim \( (W, (\text{alloc n}; \rho_1(I_1), \cdot), (\text{alloc n}; \rho_2(I_2), \cdot)) \in \text{TermAtom}[q \vdash \tau_1; \tau_2] \rho \). From the definitions of TermAtom\[q \vdash \tau_1; \tau_2] \rho \) and TermAtom\[q \vdash \tau_1; \tau_2] \rho \), it suffices to show \( W.\Psi_1 ; W.\chi_1 ; W.\sigma_1 ; \rho_1(q) ) \vdash (\text{alloc n}; \rho_1(I_1), \cdot): \rho_1(\tau_1); \rho_1(\sigma_1) \) and \( W.\Psi_2 ; W.\chi_2 ; W.\sigma_2 ; \rho_2(q) ) \vdash (\text{alloc n}; \rho_2(I_2), \cdot): \rho_2(\tau_1); \rho_2(\sigma_1) \).

Each follows from \( \Delta; \chi; \sigma; q \vdash \text{alloc n}; I_1 \) and \( \Psi; \Delta; \chi; \sigma; q \vdash \text{alloc n}; I_2 \) respectively using the component typing rule and the properties of \( W \) and \( \rho \) that we have by assumption.

- Consider arbitrary \( E_1 \) and \( E_2 \) such that \( (W, E_1, E_2) \in K[q \vdash \tau_1; \tau_2] \rho \). We must show that

\[
(W, E_1((\text{alloc n}; \rho_1(I_1), \cdot)), E_2((\text{alloc n}; \rho_2(I_2), \cdot))) \in O.
\]

Let \( M_1 = (H_1, R_1, S_1) \) and \( M_2 = (H_2, R_2, S_2) \).

From \((M_1, M_2) : W, \text{ since } W.k > 0, \) we have \( \triangleright W, M_1, M_2 \in \bigotimes \{ \text{currentMR}(\theta) \mid \theta \in W.\Theta \} \).

From the latter, we can have the following facts for island \( i_{\text{stk}} \).

We have that \( \triangleright W, S_1, 2 \subseteq \text{currentMR}(W(i_{\text{stk}})) \). From the latter it follows that \( S_j = W.S_j \).

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:

\[
E_1[(\rho_1(I_1), \cdot)]
\]

and

\[
\langle M_2 \mid E_2((\text{alloc n}; \rho_2(I_2), \cdot)) \rangle \rightarrow \langle M_2' \mid E_2((\rho_2(I_2), \cdot)) \rangle
\]
where \( M'_1 = (H_1, R_1, S'_1) \) with \( S'_1 = () :: \cdots :: () :: S_1 \), and \( M'_2 = (H_2, R_2, S'_2) \) with \( S'_2 = () :: \cdots :: () :: S_2 \).

Note that in order to complete our proof, it suffices to show:

\[
\begin{align*}
\langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle \downarrow & \land \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle \downarrow \\
\text{(running}(W.k - 1, \langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle) & \land \text{(running}(W.k - 1, \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle))
\end{align*}
\]

Let \( W' = (W, W, \Psi_1, W, \Psi_2, \Theta)' \) where

1. \( \forall i \neq i_{\text{stk}}, \Theta'(i) = W.\Theta(i) \);
2. \( \Theta'(i_{\text{stk}}) = \text{island}_{\text{stk}}(s, W, k) \) with \( s = (S'_1, \text{unit} :: \cdots :: \text{unit} :: W.\sigma_1, S'_2, \text{unit} :: \cdots :: \text{unit} :: W.\sigma_2) \).

We claim that \( W' \in \text{World} \). Note that we have \( (W, \cdot, \cdot) \in W[\text{unit}]\rho \). Thus we have that \( W.\Psi_j; \vdash \cdot () : \rho_j(\text{unit}) \). Given our choice of \( W' \) and since \( W \in \text{World} \), it follows that \( W'.\Psi_j; \vdash \cdot () : \rho_j(\text{unit}) \) which is sufficient to establish our claim.

We proceed by showing that \( (M'_1, M'_2) : W' \) and then instantiating our hypothesis

\[
\Psi; \Delta; \chi; \text{unit} :: \cdots :: \text{unit} :: \sigma; \text{inc}(q, n) \vdash I_1 \approx I_2
\]

- We claim that \( (M'_1, M'_2) : W' \). We prove this claim by establishing the following:

\[
\star \vdash M'_1 : W'.\Phi_1 \text{ and } \vdash M'_2 : W'.\Phi_2.
\]

Note the following facts:

\[
\cdot W'.\chi_1 = W.\chi_1 \text{ and } W'.\chi_2 = W.\chi_2;
\cdot W'.\sigma_1 = \text{unit} :: \cdots :: \text{unit} :: W.\sigma_1 \text{ and } W'.\sigma_2 = \text{unit} :: \cdots :: \text{unit} :: W.\sigma_2;
\cdot W'.\Psi_1 = W.\Psi_1 \text{ and } W'.\Psi_2 = W.\Psi_2.
\]

Both claims follow easily from \( (M_1, M_2) : W \) the above facts, and the fact that we have extended both stacks with same number of well-typed words, i.e., with \( n \) \( () \) words.

- We assume that \( W.k > 0 \) (and thus \( W'.k > 0 \)). We must show that

\[
(\langle W', M'_1, M'_2 \rangle \in \bigotimes \{\text{currentMR}(\theta) \mid \theta \in W'.\Theta\})
\]

The latter follows from \( (M_1, M_2) : W \) and monotonicity of MemRel given that we establish the following claim for island \( i_{\text{stk}} \)

\[
(\langle W', S'_1, S'_2 \rangle \in \text{currentMR}(W'(i_{\text{stk}}))
\]

Recall that \( W'.S_i = S'_i \). Thus we obtain trivially that \( (\langle W', S'_1, S'_2 \rangle \in \{\langle \tilde{W}, W'.S_1, W'.S_2 \rangle \mid \tilde{W} \in \text{World}_{W,k} \}) \) from the latter, we establish directly our claim that

\[
(\langle W', S'_1, S'_2 \rangle \in \text{currentMR}(W'(i_{\text{stk}}))
\]

- Next, we instantiate our final hypothesis with \( W' \) and \( \rho \). Note that \( \rho \in \mathcal{P}[\Delta] \) by assumption, \( W' \in \mathcal{H}[\Psi] \) by heap monotonicity (Lemma 1.9). We claim that \( \text{currentMR}(W'(i_{\text{reg}})) \in W'.\mathcal{R}[\chi]\rho \) by the first case of Lemma 1.20. Moreover we claim that \( \text{currentMR}(W'(i_{\text{stk}})) \in W'.\mathcal{S}[\sigma]\rho \) Our claim follows directly from the third case of Lemma 1.21 since, above, we have established that \( (W', \cdot, \cdot) \in W[\text{unit}] \) and \( \text{currentMR}(W'(i_{\text{stk}})) \) \( \in W.\mathcal{S}[\sigma]\rho \). Hence, we have that

\[
(W', (\rho_1(I_1, \cdot), (\rho_2(I_2), \cdot)) \in \mathcal{E}[q \vdash \tau; \sigma]\rho
\]

Instantiate the above with \( E_1 \) and \( E_2 \). Note that we have \( (W', E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma]\rho \) which follows from \( (W, E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma]\rho \) by monotonicity for evaluation contexts (Lemma 1.10), since our choice of \( W', W' \uplus \text{pub} W \) and the reduction does not affect the return marker.

Hence, we have

\[
(W', E_1[(\rho_1(I_1, \cdot))], E_2[(\rho_2(I_2), \cdot)]) \in \mathcal{O}.
\]
Instantiate the latter with \( M'_1 \) and \( M'_2 \), noting that \((M'_1, M'_2) : W'\). Hence, we have
\[
\begin{align*}
&
\langle \langle M'_1 | E_1((\rho_1(1), \cdot)) \rangle \downarrow \land \langle M'_2 | E_2((\rho_2(2), \cdot)) \rangle \downarrow \lor \\
&\text{(running}(W.k, \langle M'_1 | E_1((\rho_1(1), \cdot)) \rangle) \land \text{running}(W.k, \langle M'_2 | E_2((\rho_2(2), \cdot)) \rangle))
\end{align*}
\]
which implies what we needed to show.

\[\square\]

**Lemma 1.55 (Free Stack Space)**

If \( \Psi; \Delta; \chi; \sigma; \text{dec}(q, n) \vdash I_1 \approx_1 I_2 \), then \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfnee} \ n; I_1 \approx_1 \text{sfnee} \ n ; I_2 \).

**Proof**

Clearly, \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfnee} \ n; I_1 \) and \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfnee} \ n; I_2 \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in \mathcal{H}[\Psi], \rho \in D[\Delta] \), currentMR(\( W(i_{reg}) \)) \( \in W \) \( R[\chi] \rho \), and currentMR(\( W(i_{stk}) \)) \( \in W S[\tau_0 :: \cdots :: \tau_{n-1} :: \sigma] \rho \). We need to show that
\[
(W, \rho_1(\text{sfnee} \ n; I_1, \cdot), \rho_2(\text{sfnee} \ n; I_2, \cdot))
\]
\[
= (W, (\text{sfnee} \ n; \rho_1(1), \cdot), (\text{sfnee} \ n; \rho_2(2), \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \tau_0 :: \cdots :: \tau_{n-1} :: \sigma)]\rho. \quad (15)
\]

Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_1 \) and memories \( M_i \), we can immediately show that \( \text{running}(0, \langle M_i | E_1[i_1] \rangle) \).

In the following, assume \( W.k > 0 \) and let \( \tau_r; \sigma_r = \text{ret-type}(q, \chi, \tau :: \sigma) \) and \( \tau = \tau_1 :: \cdots :: \tau_m \).

Moreover, note the following preliminary fact:

- By our hypothesis and the typing rules, we have that \( \text{dec}(q, n) \neq \epsilon \) and \( \text{dec}(q, n) \neq \text{undefined} \).
  A consequence of the latter fact is that if \( q = i \) then \( i \geq n \).

With the above facts in hand, we now prove (15).

- We claim \( (W, (\text{sfnee} \ n; \rho_1(1), \cdot), (\text{sfnee} \ n; \rho_2(2), \cdot)) \in \text{TermAtom}[q \vdash \tau_r; \sigma_r] \rho \). From the definitions of \( \text{TermAtom}[q \vdash \tau_r; \sigma_r] \rho \) and \( \text{TermAtom}[q \vdash \tau_r] \rho \), it suffices to show \( W; \chi_1: W.\sigma_1; \rho_1(q) \vdash (\text{sfnee} \ n; \rho_1(1), \cdot); \rho_1(\tau_r); \rho_1(\sigma_r) \) and \( W; \chi_2: W.\sigma_2; \rho_2(q) \vdash (\text{sfnee} \ n; \rho_2(2), \cdot); \rho_2(\tau_r); \rho_2(\sigma_r) \).
  Each follows from \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfnee} \ n ; I_1 \) and \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfnee} \ n ; I_2 \) respectively using the component typing rule and the properties of \( W \) and \( \rho \) that we have by assumption.

- Consider arbitrary \( E_1 \) and \( E_2 \) such that \( (W, E_1, E_2) \in \mathcal{K}[q \vdash \tau_r; \sigma_r] \rho \). We must show that
\[
(W, E_1[(\text{sfnee} \ n; \rho_1(1), \cdot)], E_2[(\text{sfnee} \ n; \rho_2(2), \cdot)]) \in \mathcal{O}.
\]

Let \( M_1 = (H_1, R_1, S_1) \) and \( M_2 = (H_2, R_2, S_2) \).

From \((M_1, M_2) : W, \text{since } W.k > 0, \text{we have } (\triangleright W, M_1, M_2) \in \mathcal{O} \{ \text{currentMR}(\theta) \mid \theta \in W.\Theta \}\).

From the latter, we can have the following facts for island \( i_{stk} \).

We have that \((\triangleright W, S_1[1, S_2[1]) \in \text{currentMR}(W(i_{stk})) \). From the latter it follows that \( S_j = W.S_j = w_{j1} :: \cdots :: w_{jn} :: S'_j \).

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:
\[
\langle M_1 | E_1[(\text{sfnee} \ n; \rho_1(1), \cdot)] \rangle \rightarrow \langle M'_1 | E_1[(\rho_1(1), \cdot)] \rangle
\]
and
\[
\langle M_2 | E_2[(\text{sfnee} \ n; \rho_2(2), \cdot)] \rangle \rightarrow \langle M'_2 | E_2[(\rho_2(2), \cdot)] \rangle
\]
where \( M'_1 = (H_1, R_1, S'_1) \) and \( M'_2 = (H_2, R_2, S'_2) \).
Note that in order to complete our proof, it suffices to show:

\[
(\langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle) \downarrow \land \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle \downarrow \lor \\
(\text{running}(W.k - 1, \langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)] \rangle) \land \text{running}(W.k - 1, \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)] \rangle))
\]

Let \( W' = (W.k, W.\Psi_1, W.\Psi_2, \Theta') \) where

1. \( \forall i \neq i_{\text{reg}}, i_{\text{stk}}, \Theta'(i) = W.\Theta(i) \);
2. \( \Theta'(i_{\text{stk}}) = \text{island}_{\text{stk}}(s, W.k) \) with \( s = (S'_1, \sigma, S'_2, \sigma) \).

We claim that \( W' \in \text{World} \). Given our choice of \( W' \) and since \( W \in \text{World} \), it is straightforward to establish our claim.

Moreover, note that \( W' \sqsupseteq W \) and \( W' \sqsupseteq_{\text{pub}} W \). Both follow immediately given our choice of \( W' \) and the definition of \( \text{island}_{\text{stk}} \).

We proceed by showing that \( (M'_1, M'_2) : W' \) and then instantiating our hypothesis

\[ \Psi; \Delta; \chi; \sigma; \text{dec}(q, n) \vdash I_1 \approx_1 I_2 \]

- We claim that \( (M'_1, M'_2) : W' \). We prove this claim by establishing the following:
  * \( \vdash M'_1 : W'.\Phi_1 \) and \( \vdash M'_2 : W'.\Phi_2 \). Note the following facts:
    - \( W'.\chi_1 = W.\chi_1 \) and \( W'.\chi_2 = W.\chi_2 \);
    - \( W'.\sigma_1 = \sigma \) and \( W'.\sigma_2 = \sigma \);
    - \( W'.\Psi_1 = W.\Psi_1 \) and \( W'.\Psi_2 = W.\Psi_2 \).
  Both claims follow easily from \( (M_1, M_2) : W \), the above facts and the fact that we remove \( n \) words from the top of each stack.
  * We assume that \( W.k > 0 \) (and thus \( W'.k > 0 \)). We must show that

\[ \langle \uparrow W', M'_1, M'_2 \rangle \in \bigotimes \{ \text{currentMR}(\theta) \mid \theta \in W'.\Theta \} \]

The latter follows from \( (M_1, M_2) : W \) and monotonicity of \( \text{MemRel} \) given that we establish the following claim for \( \text{island}_{\text{stk}} \):

\[ \langle \uparrow W', S'_1, S'_2 \rangle \in \text{currentMR}(W'(i_{\text{stk}})) \]

Recall that \( W'.S_i = S'_i \). Thus we obtain trivially that \( \langle \uparrow W', S'_1, S'_2 \rangle \in \{ \widehat{W}, W'.S_1, W'.S_2 \} \mid W \in \text{World}_{W.k} \}. \) From the latter, we establish directly our claim that

\[ \langle \uparrow W', S'_1, S'_2 \rangle \in \text{currentMR}(W'(i_{\text{stk}})) \]

- Next, we instantiate our final hypothesis with \( W' \) and \( \rho \). Note that \( \rho \in \mathcal{D}[\Delta] \) by assumption, \( W' \in \mathcal{H}[\Psi] \) by heap monotonicity (Lemma 1.9). We claim that \( \text{currentMR}(W'(i_{\text{reg}})) \in_{W'} \mathcal{S}[\sigma] \rho \) by the second case of lemma 1.21 since \( W'(i_{\text{reg}}) = \text{island}_{\text{reg}}(s, W.k) \) with \( s = (S_1, \sigma, S_2, \sigma) \) where \( W'(i_{\text{reg}}) = \text{island}_{\text{reg}}(s, W.k) \) with \( s = (w_1 : \cdots : w_{n} \mapsto S_1, \tau_1 : \cdots : \tau_n \mapsto \sigma, w_{21} : \cdots : w_{2n} \mapsto S_2, \tau_1 : \cdots : \tau_n \mapsto \sigma) \). Moreover we claim that \( \text{currentMR}(W'(i_{\text{reg}})) \in_{W'} R[\chi] \rho \) by the first case of Lemma 1.20. Hence, we have that

\[ (W', (\rho_1(I_1, \cdot)), (\rho_2(I_2), \cdot)) \in \mathcal{E}[q \vdash \tau_r; \sigma_r] \rho \]

Instantiate the above with \( E_1 \) and \( E_2 \). Note that we have \( (W', E_1, E_2) \in \mathcal{K}[q \vdash \tau_r; \sigma_r] \rho \) which follows from \( (W, E_1, E_2) \in \mathcal{K}[q \vdash \tau_r; \sigma_r] \rho \) by monotonicity for evaluation contexts (Lemma 1.10), since our choice of \( W', W' \sqsubseteq_{\text{pub}} W \) and the preliminary fact that if \( q = i \) then \( i \geq n \) let us easily establish all the premises of that lemma.

Hence, we have

\[ (W', E_1[(\rho_1(I_1, \cdot))], E_2[(\rho_2(I_2), \cdot)]) \in \mathcal{O} \]

57
Lemma 1.56 (Load from Stack)

If $\sigma(i) = \tau$, $q \neq r_d$, and $\Psi; \Delta; \chi; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \chi; \sigma; q \vdash \text{sl}d\ r_d, i; I_1 \approx \text{sl}d\ r_d, i; I_2$.

Proof

Clearly, $\Psi; \Delta; \chi; \sigma; q \vdash \text{sl}d\ r_d, i; I_1$ and $\Psi; \Delta; \chi; \sigma; q \vdash \text{sl}d\ r_d, i; I_2$.

Consider arbitrary $W$ and $\rho$ such that $W \in H[\Psi], \rho \in D[\Delta]$, currentMR($W(i_{reg})$) $\in W$, $R[\chi]\rho$, and currentMR($W(i_{stk})$) $\in W$. $S[\sigma]\rho$. We need to show that

$$(W, \rho_1((\text{sl}d\ r_d, i; I_1,)), \rho_2((\text{sl}d\ r_d, i; I_2,)))$$

from the latter, we have the following fact for island $\text{sl}d\ r_d, i; I_1$ using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.

Let $s = (W.R_1[r_d \mapsto W.S_1(i)], W.X_1[r_d : q](\tau), W.R_2[r_d \mapsto W.S_2(i)], W.X_2[r_d : q](\tau))$ and $W' = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{reg} \mapsto \text{island}_{reg}(s, W.k)])$.

We claim that $W' \in \text{World}$. From the hypothesis, we know that $W, W.S_1 \vdash, W.S_2 |) \in S[\sigma]\rho$. Since $\sigma(i) = \tau$, this means $(W, W.S_1(i), W.S_2(i)) \in W[\tau]\rho$. Given our choice of $W'$ and since $W \in \text{World}$, this is sufficient to establish our claim.

Note that $W' \equiv_{\text{pub}} W$. The latter is immediate given our choice of $W'$ and the definition of $\text{island}_{reg}$.

Consider arbitrary $(M_1, M_2) : W$. Let $M_j = (H_j, R_j, S_j)$. From $(M_1, M_2) : W$, since $W.k > 0$, we have $(\triangleright W, M_1, M_2) \in \otimes\{\text{currentMR}(\theta) | \theta \in W.\Theta\}$. From the latter, we have the following fact for island $i_{stk}$: $(\triangleright W, S_1 |, S_2 |) \in \text{currentMR}(W(i_{stk}))$. From the latter it follows that $S_1 = W.S_i$.

Note that

$$(H_j, R_j, S_j) | (\text{sl}d\ r_d, i; \rho_j(I_j,)), \cdot) \mapsto 1 (H_j, R_j[r_d \mapsto S_j(i)], S_j | (\rho_j(I_j), \cdot))$$

Let $M'_j = (H_j, R_j[r_d \mapsto S_j(i)], S_j)$. Note that $(M'_1, M'_2) : W'$.

- Note that $W.k \leq W'.k + 1$.
- Note that $q \neq e$, which follows from the third hypothesis since it is a side condition of the instruction typing rules. Further, since $q \neq r_d$, which was the only memory location changed between $W$ and $W'$, we can see that the the return address does not change.

58
Next, to use our third hypothesis, we note that $W' \in \mathcal{H}[\Psi]$ by heap monotonicity (Lemma 1.9) and that currentMR($W'(i_{stk})$) $\in W$. $S[\sigma]\rho$ by Lemma 1.21 since $W'(i_{stk}) = W(i_{stk})$. We also claim that currentMR($W'(i_{reg})$) $\in W$. $R[\chi[r_i \cdot \tau]]\rho$. From above, we have that $(W, W.S_1(i), W.S_2(i)) \in W[\tau] \rho$. Hence, by monotonicity and the definition of $W'$ we have

$$(W', W.S_1(i), W.S_2(i)) = (W', W'.R_1(r_d), W'.R_2(r_d)) \in W[\tau] \rho,$$

which, using Lemma 1.20, is sufficient to establish our claim.

Therefore we can apply our third hypothesis to $W'$ and $\rho$, finding that

$$(W', \rho_1((I_1, \cdot)), \rho_2((I_2, \cdot))) \in E[\cdot \vdash \text{ret-type}(q, \chi, \sigma)]\rho.$$

Now, the result follows by Lemma 1.11.

\[\square\]

**Lemma 1.57 (Load Return Address from Stack)**

If $\sigma(i) = \tau_i$ and $\Psi; \Delta; \chi[r_i \cdot \tau_i]; \sigma; r_d \vdash I_1 \approx I_2$, then $\Psi; \Delta; \chi; \sigma; i \vdash \text{sld} r_d, i; I_1 \approx \text{sld} r_d, i; I_2$.

**Proof**

Clearly, $\Psi; \Delta; \chi; \sigma; i \vdash \text{sld} r_d, i; I_1$ and $\Psi; \Delta; \chi; \sigma; i \vdash \text{sld} r_d, i; I_2$.

Consider arbitrary $W$ and $\rho$ such that $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta]$, currentMR($W(i_{reg})$) $\in W$. $R[\chi]\rho$, and currentMR($W(i_{stk})$) $\in W S[\sigma]\rho$. We need to show that

$$((W, \rho_1((\text{sld} r_d, n; I_1, \cdot)), \rho_2((\text{sld} r_d, n; I_2, \cdot)))$$

$$= (W, (\text{sld} r_d, n; \rho_1(1), \cdot), (\text{sld} r_d, n; \rho_2(1), \cdot)) \in E[\cdot \vdash \text{ret-type}(i, \chi, \sigma)]\rho.$$ (16)

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running(0, $(M_i \mid E_i[e_i]))$.

In the following, assume $W.k > 0$ and let $\tau_i; \sigma_r = \text{ret-type}(i, \chi, \sigma)$.

Moreover, note the following preliminary fact:

- By our first and second hypothesis, and the typing rules we obtain that $\sigma = \tau_1 : \cdots : \tau_i : \sigma'$.
- By our assumption currentMR($W(i_{reg})$) $\in W S[\sigma]\rho$ and the previous fact, we have that $w_{11}, ..., w_{1i}, w_{21}, ..., w_{2i}, S'_{1}, S'_{2}$ such that $W.S_1 = w_{11} : \cdots : w_{1i} : S'_{1}, W.S_2 = w_{21} : \cdots : w_{2i} : S'_{2}$ and $(W, w_{1i}, w_{2i}) \in W[\tau_i] \rho$.

With the above facts in hand, we now prove (16).

- We claim $(W, (\text{sld} r_d, i; \rho_1(I_1), \cdot), (\text{sld} r_d, i; \rho_2(I_2), \cdot)) \in \text{TermAtom}[i \vdash \tau_i; \sigma_r]\rho$. From the definitions of \text{TermAtom}[$i \vdash \tau_i; \sigma_r]\rho and \text{TermAtom}[$i \vdash \tau_i; \sigma_r]$, it suffices to show $W.\Psi_1; \chi; W.\sigma_1; \rho_1(i) \vdash (\text{sld} r_d, i; \rho_1(I_1), \cdot) : (\chi[\cdot]) : (\sigma_1) ; W.\Psi_2; \chi; W.\sigma_2; \rho_2(i) \vdash (\text{sld} r_d, i; \rho_2(I_2), \cdot) : (\sigma_2)$. Each follows from $\Psi; \Delta; \chi; \sigma; i \vdash \text{sld} r_d, i; I_1$ and $\Psi; \Delta; \chi; \sigma; i \vdash \text{sld} r_d, i; I_2$ respectively using the component typing rule and the properties of $W$ and $\rho$ that we have by assumption.

- Consider arbitrary $E_1$ and $E_2$ such that $(W, E_1, E_2) \in \mathcal{K}[i \vdash \tau_r; \sigma_r]\rho$. We must show that

$$(W, E_1[(\text{sld} r_d, i; \rho_1(I_1), \cdot)], E_2[(\text{sld} r_d, i; \rho_2(I_2), \cdot)]) \in O.$$
and
\[ \langle M_1' | E_1((\text{sl}d\, r_d, i; \rho_1(I_1), \cdot)) \rangle \rightarrow \langle M_1' | E_1((\rho_1(I_1), \cdot)) \rangle \]

where \( M_1' = (H_1, R_1', S_1) \) with \( H_1 = H_1'(r_d \rightarrow w_{1i}) \), and \( M_2' = (H_2, R_2', S_2) \) with \( H_2' = H_2[r_d \rightarrow w_{2i}] \).

Note that in order to complete our proof, it suffices to show:
\[ (\langle M_1' | E_1((\rho_1(I_1), \cdot)) \rangle \downarrow \wedge \langle M_2' | E_2((\rho_2(I_2), \cdot)) \rangle \downarrow) \vee (\text{running}(W.k - 1, \langle M_1' | E_1((\rho_1(I_1), \cdot)) \rangle) \wedge \text{running}(W.k - 1, \langle M_2' | E_2((\rho_2(I_2), \cdot)) \rangle)) \]

Let \( W' = (W.k, W.\Psi_1, W.\Psi_2, \Theta') \) where

1. \( \forall i \not\in \{i_{\text{reg}}\}. \Theta'(i) = W.\Theta(i); \)
2. \( \Theta'(i_{\text{reg}}) = \text{island}_{\text{reg}}(s, W.k) \) with \( s = (W.R_1[r_d \rightarrow w_{1i}], W.\chi_1[r_d : \rho_1(\tau_i)], W.R_2[r_d \rightarrow w_{2i}], W.\chi_2[r_d : \rho_2(\rho_1(\tau_i))] \).

We claim that \( W' \in \text{World} \). Recall that we have \( (W, w_{1i}, w_{2i}) \in W[\tau_i]. \rho \). Thus we have that \( W.\Psi_j; \vdash w_{ji} : \rho_j(\tau_i) \). Given our choice of \( W' \) and since \( W \in \text{World} \), it follows that \( W'.\Psi_j; \vdash w_{ji} : \rho_j(\tau_i) \) which is sufficient to establish our claim.

Moreover, note that \( W' \equiv W \) and \( W' \equiv \text{pub}, W \). Both follow immediately given our choice of \( W' \) and the definition of \( \text{island}_{\text{reg}} \) and \( \text{island}_{\text{stk}} \).

We proceed by showing that \( (M_1', M_2') : W' \) and then instantiating our final hypothesis
\[ \Psi; \Delta; \chi; \sigma; i \vdash \text{sl}d \, r_d, i; I_1 \approx_1 \text{sl}d \, r_d, i; I_2 \]

- We claim that \( (M_1', M_2') : W' \). We prove this claim by establishing the following:
  \* \( \vdash M_1' : W'.\Phi_1 \) and \( \vdash M_2' : W'.\Phi_2 \). Note the following facts:
    \* \( W'.\chi_1 = W.\chi_1[r_d : \rho_1(\tau_i)] \) and \( W'.\chi_2 = W.\chi_2[r_d : \rho_2(\tau_i)] \);
    \* \( W'.\sigma_1 = \sigma \) and \( W'.\sigma_2 = \sigma \);
    \* \( W'.\Psi_1 = W.\Psi_1 \) and \( W'.\Psi_2 = W.\Psi_2 \).

  Both claims follow easily from \( (M_1, M_2) : W \), the above facts, the fact that we have updater register \( r_d \) in both memories words well-typed words, i.e., with words of types \( \rho_1(\tau_i) \) and \( \rho_2(\tau_i) \).

  \* We assume that \( W.k > 0 \) (and thus \( W'.k > 0 \)). We must show that
    \[ (\langle W'.k, M_1', M_2' \rangle \in \bigotimes \{ \text{currentMR}(\theta) \mid \theta \in W'.\Theta \} \]

    The latter follows from \( (M_1, M_2) : W \) and monotonicity of \( \text{MemRel} \) given that we establish the following claim for island \( i_{\text{reg}} \), \( (\langle W', W.R_1[r_d \rightarrow w_{1i}] \rangle, W.R_2[r_d \rightarrow w_{2i}] \rangle) \in \text{currentMR}(W'(i_{\text{reg}})) \).

    For the claim recall that \( W'.R_i = W.R_i[r_d \rightarrow w_{ji}] \). Thus we obtain trivially that
    \[ (\langle W', W.R_1[r_d \rightarrow w_{1i}] \rangle, W.R_2[r_d \rightarrow w_{2i}] \rangle) \in \{ (\langle W, W'.R_1, W'.R_2 \rangle, \tilde{W} \mid \tilde{W} \in \text{World}_{W.k} \}. \]

    From the latter, we establish directly our claim that
    \[ (\langle W', W.R_1[r_d \rightarrow w_{1i}] \rangle, W.R_2[r_d \rightarrow w_{2i}] \rangle) \in \text{currentMR}(W'(i_{\text{reg}})) \]

- By lemmas 1.20 and 1.21, and by unfolding the definition of our second hypothesis, \( \Psi; \Delta; \chi[r_d : \tau_i]; \sigma; r_d \vdash I_1 \approx_1 I_2 \), we derive that \( (M_1', M_2') : W' \), \( (W', E_1[(\text{sl}d \, r_d, i; \rho_1(I_1), \cdot)], E_2[(\rho_2(I_2), \cdot)]) \in \mathcal{O} \). By construction \( (M_1', M_2') : W' \) and thus by unfolding the definition of \( \mathcal{O} \), we obtain \( (M_1' \mid E_1[(\text{sl}d \, r_d, i; \rho_1(I_1), \cdot)]) \downarrow \) and \( (M_2' \mid E_2[(\rho_2(I_2), \cdot)]) \downarrow \), or running \( (W'.k, M_1' \mid E_1[(\rho_1(I_1), \cdot)]) \) and running \( (W'.k, M_2' \mid E_2[(\rho_2(I_2), \cdot)]) \). In the first case, it is straightforward to derive from the reduction semantics that \( \langle M_1' \mid E_1[(\text{sl}d \, r_d, i; \rho_1(I_1), \cdot)] \rangle \downarrow \) and \( \langle M_2' \mid E_2[(\text{sl}d \, r_d, i; \rho_2(I_2), \cdot)] \rangle \downarrow \).

  In the second case, since \( W'.k = W.k \), we derive that running \( (W.k, M_1' \mid E_1[(\text{sl}d \, r_d, i; \rho_1(I_1), \cdot)]) \) and running \( (W.k, M_2' \mid E_2[(\text{sl}d \, r_d, i; \rho_2(I_2), \cdot)]) \). Thus we conclude that:

\[ (W, E_1[(\text{sl}d \, r_d, i; \rho_1(I_1), \cdot)], E_2[(\text{sl}d \, r_d, i; \rho_2(I_2), \cdot)]) \in \mathcal{O} \].
Lemma 1.58 (Store to Stack)

Ψ; \chi \vdash r_{s1} \approx w_{s2}; \tau' \vdash q \vdash \rho \vdash \Psi; \Delta; \chi; \tau_0 ; \cdots ; \tau_{i-1} ; \vdash \tau' :: \sigma ; q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma ; q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma ; q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma ; q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma ; q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma ; q \vdash \rho

Proof

Consider arbitrary W and \rho such that W \in H[\Psi], \rho \in D[\Delta]. currentMR(W(i_{reg})) \in W S[\tau_0 ; \cdots ; \tau_i :: \sigma] \rho. We need to show that

(W, \rho_1((sst i, r_{s1}; I_1, \cdot)), \rho_2((sst i, r_{s2}; I_2, \cdot)))

= (W, (sst i, \rho_1(r_{s1}); \rho_1(I_1, \cdot)), (sst i, \rho_2(r_{s2}); \rho_2(I_2, \cdot))) \in E[q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma].

Note that if W.k = 0 then we are done, since for any evaluation contexts E_i and memories M_i, we can immediately show that running(0, \{M_i \mid E_i[e_i]\}).

In the following, assume W.k > 0 and let \tau_0 :: \cdots :: \tau_i :: \sigma.

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that (W, (sst i, \rho_1(r_{s1}); \rho_1(I_1, \cdot)), (sst i, \rho_2(r_{s2}); \rho_2(I_2, \cdot))) \in \mathsf{TermAtm}[q \vdash \rho \vdash \Delta; \chi; \tau_0 ; \cdots ; \tau_i :: \sigma].

To establish this, we must show W, \Psi_i; \vdash W, \chi; \rho(q) \vdash (sst i, \rho_1(r_{s1}); \rho_1(I_1, \cdot)), (sst i, \rho_2(r_{s2}); \rho_2(I_2, \cdot)).

The latter follows from \Psi_i; \Delta; \chi; \rho(q) \vdash (sst i, \rho_1(r_{s1}); \rho_1(I_1, \cdot)), (sst i, \rho_2(r_{s2}); \rho_2(I_2, \cdot)) using the component typing rule and the properties of W and \rho that we have by assumption.

- We claim that W' \in \mathsf{World}. Instantiating the first hypothesis with W and \rho, we have that

(W, W, \bar{R}_1(\rho_1(r_{s1}))), W, \bar{R}_2(\rho_2(r_{s2}))) \in W[\tau'] \rho

Thus we have that W, \Psi_i; \vdash W, \bar{R}_1(\rho_1(r_{s1})); \rho_1(\tau'), W, \Psi_i; \vdash W, \bar{R}_2(\rho_2(r_{s2})); \rho_2(\tau') which is sufficient to establish our claim.
Lemma 1.59 (Store Return Address to Stack)

Proof

Consider arbitrary \( (M_1, M_2) : W \). Let \( M_i = (H_i, R_i, S_i) \).

From \( (M_1, M_2) : W \), since \( W.k > 0 \), we have \( \rightarrow W, M_1, M_2 \in \otimes \{ \text{currentMR}(\theta) \mid \theta \in W.\Theta \} \).

From the latter, we have the following fact for island \( i_{\text{reg}} \): \( \rightarrow W, R_{i_1}, R_{i_2} \) \in \text{currentMR}(W(i_{\text{reg}})) \).

From the latter it follows that \( R_i = W.R_i \).

Note that 
\[
\langle \theta \rangle \rightarrow \langle \theta \rangle \]

Let \( M'_i = (H_i, R_i, S_i[i \mapsto \hat{R}_i(r_i)]) \). Note that \( (M'_1, M'_2) : W' \).

- Note that \( W.k = W'.k + 1 \).

- Note that \( q \neq i \), which follows from the third hypothesis since it is a side condition of the instruction typing rules. Further, since \( q \neq i \), which was the only memory location changed between \( W \) and \( W' \), we can see that the the return address does not change.

- Next, to use our third hypothesis, we note that \( W' \in \mathcal{H}[[\Psi]] \) by heap monotonicity (Lemma 1.9) and that \( \text{currentMR}(W'(i_{\text{reg}})) \subseteq W.\text{R}[\theta] \rho \) by Lemma 1.20 since \( W'(i_{\text{reg}}) = W(i_{\text{reg}}) \). We also claim that \( \text{currentMR}(W'(i_{\text{stk}})) \subseteq W'.\mathcal{S}[[\tau_0 : \cdots : \tau' : \sigma] \rho \). From above, we have that \( (W, W.\hat{R}_1(r_1), W.\hat{R}_2(r_2)) \in W.\mathcal{S}[\tau'] \rho \). Hence, by monotonicity and the definition of \( W' \) we have

\[
(W', W.\hat{R}_1(r_1), W.\hat{R}_2(r_2)) = (W', W'.S_1(i), W'.S_2(i)) \in W.\mathcal{S}[\tau'] \rho,
\]

which, using Lemma 1.21, is sufficient to establish our claim.

Therefore we can apply our third hypothesis to \( W' \) and \( \rho \), finding that

\[
(W', \rho_1((I_1, \cdot)), \rho_2((I_2, \cdot))) \in \mathcal{E}[q + \text{ret-type}(q, \chi, \sigma)] \rho.
\]

Now, the result follows by Lemma 1.11.

\[\square\]

**Lemma 1.59 (Store Return Address to Stack)**

If \( \chi(r_s) = \tau' \) and \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_i :: \tau' :: \sigma; i \vdash I_1 \approx I_2 \),

then \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_i :: \tau' :: \sigma; r_s \vdash \text{sst} i, r_s; I_1 \approx I_2 \).

**Proof**

Clearly, \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_i :: \sigma; r_s \vdash \text{sst} i, r_s; I_1 \) and \( \Psi; \Delta; \chi; \tau_0 :: \cdots :: \tau_i :: \sigma; r_s \vdash \text{sst} i, r_s; I_2 \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in \mathcal{H}[[\Psi]], \rho \in \mathcal{D}[\Delta], \text{currentMR}(W(i_{\text{reg}})) \subseteq W.\mathcal{R}[\chi] \rho \), and \( \text{currentMR}(W(i_{\text{stk}})) \subseteq W.\mathcal{S}[\tau_0 :: \cdots :: \tau_i :: \sigma] \rho \). We need to show that

\[
(W, \rho_1((\text{sst} i, r_s; I_1, \cdot)), \rho_2((\text{sst} i, r_s; I_2, \cdot))) = (W, (\text{sst} i, r_s; \rho_1(I_1, \cdot)), (\text{sst} i, r_s; \rho_2(I_2, \cdot))) \in \mathcal{E}[r_s + \text{ret-type}(r_s, \chi, \sigma)] \rho.
\]

(17)

Note that if \( W.k = 0 \) then we are done, since for any evaluation contexts \( E_i \) and memories \( M_i \), we can immediately show that \( \text{running}(0, (M_i \mid E_i[i])) \).

In the following, assume \( W.k > 0 \) and let \( \tau_r; \sigma_r = \text{ret-type}(r_s, \chi, \sigma) \).

Moreover, note the following preliminary fact:

- By our first second hypothesis, and \( \text{currentMR}(W(i_{\text{reg}})) \subseteq W.\mathcal{R}[\chi] \rho \) we have that there exist \( w_1 \), and \( w_2 \) such that \( W.R_1(r_s) = w_1 \) and \( W.R_2(r_s) = w_2 \), \( (W, w_1, w_2) \in W.\mathcal{S}[\tau'] \rho \).

With the above facts in hand, we now prove (17).
• claim that \((sst_i, r_s; \rho_1(I_1), \cdot), (sst_i, r_s; \rho_2(I_2), \cdot)\) ∈ TermAtom\([r_s \vdash \tau_i; \sigma_i]_\rho\) From the definitions of TermAtom\([r_s \vdash \tau_i; \sigma_i]_\rho\) and TermAtom\([r_s \vdash \tau_r; \sigma_r]_\rho\), it suffices to show \(W, \Psi \vdash; W, \chi_1; W, \sigma_1; \rho_1(r_s) \vdash\) \((\text{sld}_r, i; \rho_1(I_1), \cdot), \rho_1(\tau_r); \rho_1(\sigma_r))\) and \(W, \Psi \vdash; W, \chi_2; W, \sigma_2; \rho_2(r_s) \vdash\) \((\text{sld}_r, i; \rho_2(I_2), \cdot), \rho_2(\tau_r); \rho_2(\sigma_r))\). Each follows from \(\Psi; \chi; \tau_0 : \cdots : \tau_i : \sigma_i; r_s \vdash; sst_i, r_s; I_1\) and \(\Psi; \chi; \tau_0 : \cdots : \tau_i : \sigma_i; r_s \vdash; sst_i, r_s; I_2\) respectively using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.

• Consider arbitrary \(E_1\) and \(E_2\) such that \((W, E_1, E_2) \in K\[r_s \vdash \tau_r; \sigma_r\]_\rho\). We must show that \((W, E_1[(sst_i, r_s; \rho_1(I_1), \cdot)], E_2[(sst_i, r_s; \rho_2(I_2), \cdot)]) \in O\).

Consider arbitrary \((M_1, M_2) : W\). We must show either that \(\langle M_1 \mid E_1[(sst_r, d_1[i], r_s; \rho_1(I_1), \cdot)]\rangle \downarrow\) and \(\langle M_2 \mid E_2[(sst_r, d_2[i], r_s; \rho_2(I_2), \cdot)]\rangle \downarrow\), or running \((W, k, \langle M_1 \mid E_1[(sst_r, d_1[i], r_s; \rho_1(I_1), \cdot)]\rangle)\) and running \((W, k, \langle M_2 \mid E_2[(sst_r, d_2[i], r_s; \rho_2(I_2), \cdot)]\rangle)\).

Let \(M_1 = (H_1, R_1, S_1)\) and \(M_2 = (H_2, R_2, S_2)\).

From \((M_1, M_2) : W\), since \(W, k > 0\), we have \((W, M_1, M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W, \Theta\}\).

From the latter, we can have the following fact for island \(i_{\text{reg}}\).

We have that \((W, R_1, R_2) \in \text{currentMR}(W(i_{\text{reg}}))\). From the latter it follows that \(R_1 = W, R_4\) and, hence, \(R_1(r_s) = w_1\) and \(R_2(r_s) = w_2\) (which are the same \(w_1\) and \(w_2\) from our preliminary facts above).

Next, by the reduction semantics of our language, with all of the above facts in hand, we have that:

\[
\langle M_1 \mid E_1[(sst_i, r_s; \rho_1(I_1), \cdot)]\rangle \longrightarrow \langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)]\rangle
\]

and

\[
\langle M_2 \mid E_2[(sst_i, r_s; \rho_2(I_2), \cdot)]\rangle \longrightarrow \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)]\rangle
\]

where \(M'_1 = (H_1, R_1, S'_1)\) with \(S'_1 = S_1[i \mapsto w_1]\), and \(M'_2 = (H_2, R_2, S'_2)\) with \(S'_2 = S_2[i \mapsto w_2]\).

Note that in order to complete our proof, it suffices to show:

\[
\left((\langle M'_1 \mid E_1[(\rho_1(I_1), \cdot)]\rangle) \downarrow \land \langle M'_2 \mid E_2[(\rho_2(I_2), \cdot)]\rangle) \downarrow \right) \lor
\]

\[
\left(\text{running}(W, k - 1, \langle M_1 \mid E_1[(\rho_1(I_1), \cdot)]\rangle) \land \text{running}(W, k - 1, \langle M_2 \mid E_2[(\rho_2(I_2), \cdot)]\rangle)\right)
\]

We proceed by showing that \((M'_1, M'_2) : W\) and then instantiating our final hypothesis \(\Psi; \chi; \tau_0 : \cdots : \tau_i : \sigma_i; r_s \vdash; sst_i, r_s; I_1 \approx_1 sst_i, r_s; I_2\).

− We claim that \((M'_1, M'_2) : W\). We prove this claim by establishing the following:

* \(\vdash M'_1 : W, \Phi_1\) and \(\vdash M'_2 : W, \Phi_2\), both of which easily follow from \((M_1, M_2) : W\) and the fact that we have updated the \(i\)th element of each stack with well-typed words of the same type, i.e., with words with types \(\rho_1(\tau'_i)\) and \(\rho_2(\tau'_i)\) respectively.

* We assume that \(W, k > 0\) and we must show that

\[
(W, M'_1, M'_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W, \Theta\}
\]

The latter follows from \((M_1, M_2) : W\) and monotonicity of MemRel given that we establish the following claim for island \(i_{\text{stk}}\), \((W', S'_1, S'_2) \in \text{currentMR}(W'(i_{\text{stk}}))\).

Recall that \(W'.S_1 = S'_1\). Thus, we obtain trivially that \((W', S'_1, S'_2) \in \{(W', W'.S_1, W'.S_2) \mid W' \in \text{Word}_{W,k}\}\). From the latter, we establish directly our claim that

\[
(W', S'_1, S'_2) \in \text{currentMR}(W'(i_{\text{stk}}))
\]

− Next, we instantiate our final hypothesis with \(W'\) and \(\rho\). Note that \(\rho \in D[\Delta]\) by assumption, \(W' \in H[\Psi]\) by heap monotonicity (Lemma 1.9). We claim that \(\text{currentMR}(W'(i_{\text{reg}})) \approx_{W'} S[\sigma']_\rho\) by the first case of Lemma 1.20. Moreover we claim that \(\text{currentMR}(W'(i_{\text{stk}})) \approx_{W'} S[\sigma']_\rho\) Our claim follows directly from the fourth case of lemma 1.21 since, above, we have
Lemma 1.60 (Return from Call)

If \( \chi(r) = \text{box} \forall [] \end{equation} \{ r'; \tau; \sigma \}^n \) and \( \chi(r') = \tau \), then \( \Psi; \Delta; \chi; \sigma; r \vdash \text{ret} \{ r' \} \approx_1 \text{ret} \{ r \} \).

Proof

Clearly, \( \Psi; \Delta; \chi; \sigma; r \vdash \text{ret} \{ r' \} \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta], \) currentMR\( (W(\text{reg})) \in W \mathcal{S}[\sigma] \rho \), and currentMR\( (W(\text{stk})) \in W \mathcal{S}[\sigma] \rho \). From the first premise, we have that ret-type\( (r, \chi, \sigma) = \tau; \sigma \). We need to show that

\[
(W, \rho_1((\text{ret} \{ r' \}), \cdot), \rho_2((\text{ret} \{ r' \}, \cdot))) = (W, (\text{ret} \{ r' \}), (\text{ret} \{ r' \}, \cdot)) \in \mathcal{E}[r + \tau; \sigma] \rho.
\]

Consider arbitrary \( E_1 \) and \( E_2 \) such that \( (W, E_1, E_2) \in \mathcal{K}[r + \tau; \sigma] \rho \). We need to show that

\[
(W, E_1((\text{ret} \{ r' \}), \cdot), E_2((\text{ret} \{ r' \}, \cdot)) \in \mathcal{O}.
\]

Instantiate \( (W, E_1, E_2) \in \mathcal{K}[r + \tau; \sigma] \rho \) with \( W, r, r' \), and \( r' \). Note that \( W \supseteq \text{pub} \) \( W \) (by reflexivity), the return markers are both \( r \), and currentMR\( (W(\text{reg})) \in W \mathcal{S}[\sigma] \rho \). We also claim that \( (\supseteq \text{pub} \ W, W.R_1(r'), W.R_2(r')) \in W[r] \rho \). Instantiate currentMR\( (W(\text{reg})) \in W \mathcal{R}[\chi] \rho \) with \( (\supseteq \text{pub} \ W, W.R_1 \ W.R_2) \) noting that the latter is in currentMR\( (W(\text{reg})) \) by the definition of island\( \text{reg} \) and that \( \supseteq \text{pub} \ W \supseteq \text{pub} \ W \). Hence, we have \( (\supseteq \text{pub} \ W, W.R_1 \ W.R_2) \in \mathcal{R}[\chi] \rho \). By the second premise, we have that \( (r'; \tau) \in \chi \). Therefore our claim follows from the definition of \( \mathcal{R}[\chi] \rho \). Thus, we have exactly what we needed to show.

Lemma 1.61 (Return at End)

If \( \chi(r) = \tau \), then \( \Psi; \Delta; \chi; \sigma; \text{end} \{ \tau; \sigma \} \vdash \text{ret} \{ \tau; \sigma \} \{ r \} \approx_1 \text{ret} \{ \tau; \sigma \} \{ r \} \).

Proof

Clearly, \( \Psi; \Delta; \chi; \sigma; \text{end} \{ \tau; \sigma \} \vdash \text{ret} \{ \tau; \sigma \} \{ r \} \).

Consider arbitrary \( W \) and \( \rho \) such that \( W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta], \) currentMR\( (W(\text{reg})) \in W \mathcal{S}[\sigma] \rho \), and currentMR\( (W(\text{stk})) \in W \mathcal{S}[\sigma] \rho \). We need to show that

\[
(W, \rho_1((\text{ret} \{ \tau; \sigma \} \{ r \}), \cdot), \rho_2((\text{ret} \{ \tau; \sigma \} \{ r \}, \cdot)))
\]

\[
= (W, (\text{ret} \{ \text{end} \{ \tau; \sigma \} \{ r \} \}, \cdot), (\text{ret} \{ \text{end} \{ \tau; \sigma \} \{ r \}, \cdot)) \in \mathcal{E}[\text{end} \{ \tau; \sigma \} + \tau; \sigma] \rho.
\]
Consider arbitrary \(E_1\) and \(E_2\) such that \((W,E_1,E_2) \in \mathcal{K}[\text{end}\{\tau;\sigma\} \vdash \tau;\sigma]\). We need to show that
\[
(W,E_1[(\text{ret end})(\rho_1(\tau);\rho_1(\sigma))\{r,\cdot\}], E_2[(\text{ret end})(\rho_1(\tau);\rho_1(\sigma))\{r,\cdot\}]) \in \mathcal{O}
\]

Instantiate \((W,E_1,E_2) \in \mathcal{K}[\text{end}\{\tau;\sigma\} \vdash \tau;\sigma]\) with \(W\), \(\text{end}\{\tau;\sigma\}\), \(r\), and \(r\). Note that \(W \supseteq_W W\) (by reflexivity), the return markers are both \(\text{end}\{\tau;\sigma\}\), and \(\text{currentMR}(W(i_{\text{stk}})) \in W S[\sigma] \rho\). We also claim that \(\langle >W;W,R_1(r),W,R_2(r)\rangle \in \mathcal{W}[\tau] \rho\). Instantiate \(\text{currentMR}(W(i_{\text{reg}})) \in W R[\chi] \rho\) with \(\langle >W;W,R_1(i),W,R_2(i)\rangle\) noting that the latter is in \(\text{currentMR}(W(i_{\text{reg}}))\) by the definition of \(\text{island}_{\text{reg}}\) and that \(\triangleright W \subseteq W\). Hence, we have \(\langle >W;W,R_1(i),W,R_2(i)\rangle \in R[\chi] \rho\). By the first premise, we have that \((r;\tau) \in \chi\). Therefore our claim follows from the definition of \(R[\chi] \rho\). Thus, we have exactly what we needed to show.

\[
\text{Lemma 1.62 (Jump)}
\]
If \(\Psi;\Delta;\chi \vdash u_1 \approx u_2:\text{box} \forall \cdot\{\chi';\sigma\} q, \Delta \vdash \chi \leq \chi', \text{and} \cdot[\Delta];\chi;\sigma \vdash q,\)
then \(\Psi;\Delta;\chi;\sigma;q \vdash \text{jmp} u_1 \approx_1 \text{jmp} u_2\).

\[
\text{Proof}
\]
Clearly, \(\Psi;\Delta;\chi;\sigma;q \vdash \text{jmp} u_1\) and \(\Psi;\Delta;\chi;\sigma;q \vdash \text{jmp} u_2\) follow from the premises.

Consider arbitrary \(W\) and \(\rho\) such that \(W \in \mathcal{H}[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{\text{reg}})) \subseteq W R[\chi] \rho,\) and \(\text{currentMR}(W(i_{\text{stk}})) \subseteq W S[\sigma] \rho\). We need to show that
\[
(W,\rho_1((\text{jmp} u_1,\cdot)),\rho_2((\text{jmp} u_2,\cdot))) = (W,\text{jmp} \rho_1(u_1),\cdot), (\text{jmp} \rho_2(u_2),\cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q,\chi,\sigma)] \rho.
\]

Note that if \(W.k = 0\) then we are done, since for any evaluation contexts \(E_i\) and memories \(M_i\), we can immediately show that running \((0, (M_i | E_i[\ell_i]))\).

In the following, assume \(W.k > 0\) and let \(\tau;\sigma = \text{ret-type}(q,\chi,\sigma)\).

Instantiate the first premise with \(W\) and \(\rho\), noting \(W \in \mathcal{H}[\Psi], \rho \in D[\Delta], \text{currentMR}(W(i_{\text{reg}})) \subseteq W R[\chi] \rho.\) Thus we have that \((W,W,R_1(\rho_1(u_1)),W,R_2(\rho_2(u_2))) \in \mathcal{W}[\text{box} \forall \cdot \{\chi';\sigma\} q] \rho\). From the definition of the latter, we have that \(W,R_i(\rho_i(u_i)) = \ell_i[\sigma']\).

We will use Lemma 1.11 to complete the proof, so we start by establishing the premises of that lemma.

- We claim that \((W,\text{jmp} \rho_1(u_1),\cdot), (\text{jmp} \rho_2(u_2),\cdot)) \in \text{TermAtom}[q \vdash \tau;\sigma] \rho.\) To establish this, we must show \(W,\Psi;\cdot;W,\chi;W,\sigma;\rho_1(q) \vdash (\text{jmp} \rho_1(u_1),\cdot); \rho_1(\tau); \rho_1(\sigma)).\) The latter follows from \(\Psi;\Delta;\chi;\sigma;q \vdash \text{jmp} u_1\) using the component typing rule and the properties of \(W\) and \(\rho\) that we have by assumption.
- Let \(W' = \triangleright W\). Note that \(W' \in \text{World} \) since \(W.k > 0\) and \(W \in \text{World}\).
- Note that \(\triangleright W \equiv_{\text{pub}} W\) by Lemma 1.6.
- Consider arbitrary \((M_1,M_2) : W\). Let \(M_i = (H_i,R_i,S_i)\).

  From \((M_1,M_2) : W\), since \(W.k > 0\), we have \((\triangleright W,M_1,M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W,\Theta\}\).

  From the latter, we have two facts, one for island \(i_{\text{reg}}\) and the other for island \(i_{\text{box}}\).

  First, we have that \((\triangleright W,R_1 \downarrow, R_2 \downarrow) \in \text{currentMR}(W(i_{\text{reg}}))\). From the latter it follows that \(R_i = W.R_i\) and, hence, \(\hat{R}_i(\rho_i(u_i)) = \ell_i[\sigma']\).

  Second, we have that there exist some \(H_{b1} \subseteq H_1\) and \(H_{b2} \subseteq H_2\) such that \((\triangleright W,H_{b1} \downarrow, H_{b2} \downarrow) \in \text{currentMR}(W(i_{\text{box}}))\). We use the latter to instantiate \((W,\ell_1[\omega_1],\ell_2[\omega_2]) \in \mathcal{W}[\text{box} \forall \cdot \{\chi';\sigma\} q] \rho,\) noting that \(\triangleright W \subseteq W\), which allows us to conclude:
  - \(H_{b1}(\ell_1) = \text{code}[\beta_1\{\chi_i;\sigma_i\}^n.I_1,\)
  - \(\rho_i(\chi) = \chi_i[\omega_1/\beta_1],\)
  - \(\rho_i(\sigma) = \sigma_i[\omega_1/\beta_1],\)
  - \(\rho_i(q) = q_i[\omega_1/\beta_1],\) and
  - \((\triangleright W, (\text{code}[\{\chi_1;\sigma_1\}^n.I_1]) [\omega_1/\beta_1], (\text{code}[\{\chi_2;\sigma_2\} q^2.I_2]) [\omega_2/\beta_2]) \in \mathcal{H}[\forall \cdot \{\chi';\sigma\} q] \rho\)
Hence, we have that \( \hat{R}_i(\rho_i(u_i)) = \ell_i[\overline{w_i}] \) and \( H_i(\ell_i) = \text{code}[\beta_i(\chi_i; \sigma_i)^q_i].I_i \), and
\[
((H_i, R_i, S_i) \mid (\text{jmp} \rho_i(u_i), \cdot)) \mapsto^1 ((H_i, R_i, S_i) \mid (I_i[\overline{w_i}/\beta_i], \cdot))
\]

Let \( M'_i = M_i = (H_i, R_i, S_i) \). Note that \((M'_1, M'_2) : W' \) is equivalent to \((M_1, M_2) : \triangleright W \), which follows by Lemma 1.6.

- Note that \( W.k \leq W'.k + 1 \) since \( W'.k = (\triangleright W).k = W.k - 1 \).
- Note that \( q \neq \epsilon \), which follows from the third hypothesis. Further, since we haven’t changed the world, we can see that the the return address does not change.
- Note that
\[
(\triangleright W, (\text{code}[\chi_1; \sigma_1]^q_1.I_1)[\overline{w_1}/\beta_1],
(\text{code}[\chi_2; \sigma_2]^q_2.I_2)[\overline{w_2}/\beta_2]) \in \mathcal{H}[\forall[].\{\chi'; \sigma\}^q]\rho
\]

\( \equiv (\triangleright W, (\text{code}[\chi_1[\overline{w_1}/\beta_1]; \sigma_1[\overline{w_1}/\beta_1])^q_1.I_1[\overline{w_1}/\beta_1],
(\text{code}[\chi_2[\overline{w_2}/\beta_2]; \sigma_2[\overline{w_2}/\beta_2])^q_2.I_2[\overline{w_2}/\beta_2]) \in \mathcal{H}[\forall[].\{\chi'; \sigma\}^q]\rho
\]

\( \equiv (\triangleright W, (\text{code}[\rho_1(\chi')]; \rho_1(\sigma))^{\rho_1(q)}_1.I_1[\overline{w_1}/\beta_1],
(\text{code}[\rho_2(\chi'); \rho_2(\sigma)]^{\rho_2(q)}_2.I_2[\overline{w_2}/\beta_2]) \in \mathcal{H}[\forall[].\{\chi'; \sigma\}^q]\rho
\)

Instantiate the latter with \( \triangleright W \) and \( \tau; \sigma' = \text{ret-type}(q, \chi, \sigma) \). We note the following:
- We have \( \triangleright W \sqsupset W \) by reflexivity.
- We claim that \( \text{ret-type}(q, \chi', \sigma) = \rho, \text{ret-type}(q, \chi, \sigma) \). Since \( q \neq \epsilon \), the latter is immediate except in the case when \( q \) is some register \( r' \), in which case we must show that \( r' \in \text{dom}(\chi') \) since otherwise \( \text{ret-type}(q, \chi', \sigma) \) would be undefined. But note that from our first premise, it follows that \( \Delta \vdash \text{box} \forall[].\{\chi'; \sigma\}^q \). By inversion of typing rules, we have that \( \Delta \vdash \forall[].\{\chi'; \sigma\}^q \), and hence \( \Delta \vdash \chi'; \sigma \vdash q \). From the latter it follows that \( \text{ret-type}(q, \chi', \sigma) \) is defined. Hence, if \( q \) is some register \( r' \), it must be that \( r' \in \text{dom}(\chi') \). Moreover, from the second premise, it follows that \( \chi'(r') = \chi(r') \). This is enough to establish our claim.
- We claim that \( \text{currentMR}((\triangleright W)(i_{reg})) \in \triangleright W \mathcal{R}[\chi']^q \rho \). To show this, consider arbitrary \((\widehat{W}, M_1, M_2) \in \text{currentMR}((\triangleright W)(i_{reg})) \) such that \( \widehat{W} \sqsupset \triangleright W \). We must show that
\[
(\widehat{W}, M_1, M_2) \in \mathcal{R}[\chi']^q \rho
\]

Note that \( (\triangleright W).k = W.k - 1 \) and that \( \text{currentMR}((\triangleright W)(i_{reg})) = [\text{currentMR}(W(i_{reg}))]_{W.k-1} \).
Thus, \( (\widehat{W}, M_1, M_2) \in \text{currentMR}(W(i_{reg})) \). Using the latter we can instantiate \( \text{currentMR}(W(i_{reg})) \in \triangleright W \mathcal{R}[\chi']^q \rho \) with \( (W, M_1, M_2) \), noting that \( \widehat{W} \sqsupset W \) (by transitivity of \( \sqsupset \) ), which gives us \( (W, M_1, M_2) \in \mathcal{R}[\chi']^q \rho \). Finally, by Lemma 1.19 (register-file subtyping implies inclusion) we have that \( \mathcal{R}[\chi']^q \rho \subseteq \mathcal{R}[\chi']^q \rho \), which is sufficient to show what we need.
- We note that \( \text{currentMR}(\triangleright W)(i_{stk})) \in \triangleright W \mathcal{S}[\sigma]^q \rho \). To show this, consider arbitrary \((\widehat{W}, M_1, M_2) \in \text{currentMR}((\triangleright W)(i_{stk})) \) such that \( \widehat{W} \sqsupset \triangleright W \). We must show that
\[
(\widehat{W}, M_1, M_2) \in \mathcal{S}[\sigma]^q \rho
\]

Note that \( (\triangleright W).k = W.k - 1 \) and that \( \text{currentMR}((\triangleright W)(i_{stk})) = [\text{currentMR}(W(i_{stk}))]_{W.k-1} \).
Thus, \( (\widehat{W}, M_1, M_2) \in \text{currentMR}(W(i_{stk})) \). Using the latter we can instantiate \( \text{currentMR}(W(i_{stk})) \in \triangleright W \mathcal{S}[\sigma]^q \rho \) with \( (W, M_1, M_2) \), noting that \( \widehat{W} \sqsupset W \) (by transitivity of \( \sqsupset \) ), which gives us \( (W, M_1, M_2) \in \mathcal{S}[\sigma]^q \rho \) as needed.

Hence, we can conclude that
\[
(\triangleright W, (\rho_1(I_1[\overline{w_1}/\beta_1]), \cdot), (\rho_2(I_2[\overline{w_2}/\beta_2]), \cdot)) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi', \sigma)]^q \rho
\]

Now, the result follows by Lemma 1.11. \( \square \)
Lemma 1.63 (Call)  
Given the following:

- $\Psi; \Delta; \chi \vdash u_1 \equiv_{u} u_2$; box $\forall[\zeta, \epsilon].\{\hat{x}; \hat{\sigma}\}^q$,
- $\text{ret-addr-type}(\hat{q}, \hat{x}, \hat{\sigma}) = \forall[\cdot].\{\tau; \hat{\sigma}\}^e$,
- $\Delta \vdash \sigma_0$,
- $\Delta \vdash \forall[\cdot].\{\hat{\chi}[\sigma_0/\zeta][(i + k - j)/\epsilon]; \hat{\sigma}[\sigma_0/\zeta][(i + k - j)/\epsilon]\}^q$,
- $\Delta \vdash \chi \leq \hat{\chi}[\sigma_0/\zeta][(i + k - j)/\epsilon]$,
- $\sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0$,
- $\hat{\sigma} = \tau_0 :: \cdots :: \tau_j :: \zeta$,
- $j < i$, and
- $\sigma' = \tau'_0 :: \cdots :: \tau'_k :: \zeta$.

We have that $\Psi; \Delta; \chi; \sigma; i \vdash \text{call } u_1 \{\sigma_0, i + k - j\} \approx_1 \text{call } u_2 \{\sigma_0, i + k - j\}$.

**Proof**

Clearly, $\Psi; \Delta; \chi; \sigma; i \vdash \text{call } u_1 \{\sigma_0, i + k - j\}$ and $\Psi; \Delta; \chi; \sigma; i \vdash \text{call } u_2 \{\sigma_0, i + k - j\}$ follow from the premises.

Consider arbitrary $W, \gamma$, and $\rho$ such that $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta]$, $(W, \gamma) \in \mathcal{G}[\Gamma] \rho$, currentMR$(W(i_{\text{reg}})) \subseteq W$ $\mathcal{R}[\chi] \rho$, and currentMR$(W(i_{\text{stk}})) \subseteq W$ $\mathcal{S}[\sigma] \rho$. We need to show that

$$(W, \rho_1(\gamma_1((\text{call } u_1 \{\sigma_0, i + k - j\}, \cdot)), \rho_2(\gamma_2((\text{call } u_2 \{\sigma_0, i + k - j\}, \cdot))))$$

$$= (W, (\text{call } \rho_1(u_1) \{\rho_1(\sigma_0), i + k - j\}, \cdot), (\text{call } \rho_2(u_2) \{\rho_2(\sigma_0), i + k - j\}, \cdot)) \in \mathcal{E}[i \vdash \text{ret-type}(i, \chi, \sigma)] \rho.$$

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running$(0, (M_i | E_i[e_i]))$.

In the following, assume $W.k > 0$ and let $\tau_i; \sigma_r = \text{ret-type}(i, \chi, \sigma)$.

Consider arbitrary $(W, E_1, E_2) \in \mathcal{K}[i \vdash \tau_i; \sigma_r] \rho$. We need to show that

$$(W, E_1((\text{call } \rho_1(u_1) \{\rho_1(\sigma_0), i + k - j\}, \cdot), E_2((\text{call } \rho_2(u_2) \{\rho_2(\sigma_0), i + k - j\}, \cdot))) \in \mathcal{O}$$

Consider arbitrary $(M_1, M_2) : W$. We must show that for $i \in \{1, 2\}$, either both configurations $(M_i | E_i)[(\text{call } \rho_i(u_i) \{\rho_i(\sigma_0), i + k - j\})]$ terminate or both are running at $W.k$.

Instantiate the first premise with $W$ and $\rho$, noting $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta]$, and currentMR$(W(i_{\text{reg}})) \subseteq W$ $\mathcal{R}[\chi] \rho$. Thus we have that $(W, W.R_1(\rho_1(u_1)), W.R_2(\rho_2(u_2))) \in W[\text{box } \forall[\zeta, \epsilon].\{\hat{x}; \hat{\sigma}\}^q \rho$. From the definition of the latter, we have that $W.R_i(\rho_i(u_i)) = \ell_i[\omega_i]$.

Next, let $M_i = (H_i, R_i, S_i)$.

From $(M_1, M_2) : W$, since $W.k > 0$, we have $(\triangleright W, M_1, M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\}$. From the latter, we have three facts, one each for islands $i_{\text{reg}}, i_{\text{stk}}$, and $i_{\text{box}}$.

First, we have that $(\triangleright W, R_1[\ell_i], R_2[\ell_i]) \in \text{currentMR}(W(i_{\text{reg}}))$. From the latter it follows that $R_1 = W.R_i$ and, hence, $R_i(\rho_i(u_i)) = \ell_i[\omega_i]$.

Second, we have $(\triangleright W, S_1[\ell_i], S_2[\ell_i]) \in \text{currentMR}(W(i_{\text{stk}}))$. From the latter and currentMR$(W(i_{\text{stk}})) \subseteq W$ $\mathcal{S}[\sigma] \rho$, it follows that $(\triangleright W, S_1[\ell_i], S_2[\ell_i]) \subseteq \mathcal{S}[\sigma] \rho$. From the premise $\sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0$, it follows that $S_1 = w_{10} :: \cdots :: w_{1j} :: S_{10}, S_2 = w_{20} :: \cdots :: w_{2j} :: S_{20}$, $(\triangleright W, w_{1n}, w_{2n}) \in W[\tau_n] \rho$ for $n \in \{0, \ldots, j\}$, and $(\triangleright W, S_{10}, S_{20}) \in \mathcal{S}[\sigma] \rho$.

Third, we have that there exist some $H_{b1} \subseteq H_1$ and $H_{b2} \subseteq H_2$ such that $(\triangleright W, H_{b1}[\ell_i], H_{b2}[\ell_i]) \in \text{currentMR}(W(i_{\text{box}}))$. We use the latter to instantiate $(W, \ell_1[\omega_i], \ell_2[\omega_i]) \in W[\text{box } \forall[\zeta, \epsilon].\{\hat{x}; \hat{\sigma}\}^q \rho$, noting that $\triangleright W \supseteq W$, which allows us to conclude:

67
Further, we note that we have $\hat{\rho}(\hat{\chi}) = \chi_i[\omega_i/\beta_i]$. Therefore, it suffices to show that for $\langle \hat{\rho}(\sigma_0), i + k - j \rangle$, and therefore

$$\langle (H_i, R_i, S_i) \mid E_i([\text{call } \rho_i(u_i) \rightarrow \rho_i(\sigma_0)],[i + k - j]) \rangle$$

Hence, we know that $\hat{\beta}_i(\rho_i(u_i)) = \beta_i[\omega_i]$ and $H_i(\epsilon) = code[\beta_i, \epsilon, \{\chi_i; \sigma_1\}]$. Therefore, it remains to show that for $i \in \{1, 2\}$, either both configurations $\langle M_i \mid E_i([\text{call } \rho_i(u_i) \rightarrow \rho_i(\sigma_0)],[i + k - j]) \rangle$ terminate or both are running at $W.k - 1$.

We proceed by noting that from the premise ret-addr-type($\hat{q}, \hat{\chi}, \hat{\sigma}$) = $\forall[]\{r: \tau; \hat{\sigma}'\}$, by definition of ret-addr-type, it must be that either $\hat{q} = 0_{\text{ra}}$ or $\hat{q} = 1_{\text{ra}}$.

Further, we note that we have $\langle W, \text{code}[\chi_1; \sigma_1]\rangle[\omega_i/\beta_i], \langle \text{code}[\chi_2; \sigma_2]\rangle[\omega_2/\beta_2] \rangle \in \mathcal{H}[\forall[]\{\hat{\chi}; \hat{\sigma}\}]$.

$$\langle (H_i, R_i, S_i) \mid E_i([\text{call } \rho_i(u_i) \rightarrow \rho_i(\sigma_0)],[i + k - j]) \rangle$$

Instantiate the latter with $\triangleright W$, $\rho^*$, $\tau$, and $\hat{\sigma}'$, where

$$\rho^* = \{\hat{\chi} \rightarrow (\rho_1(\sigma_0), \rho_2(\sigma_0), \varphi_S), \epsilon \rightarrow (i + k - j, i + k - j)\}$$

and $\varphi_S = \{(\hat{W}, S_{10}, S_{20}) \mid \hat{W} \sqsupseteq \triangleright W\}$. We note the following:

- We have $\triangleright W \sqsupseteq \triangleright W$ by reflexivity.
- Note that $\rho^* \in D[\chi, \epsilon]$, which follows by the definition of $D[\cdot]$ and by applying Lemma 1.18 to $\langle W, S_{10}, S_{20} \rangle \in S[\sigma_0]$. Let $\rho' = \rho \cup \rho^*$. 
- Note that $\tau; \hat{\sigma}' = \rho^*$ ret-type($\hat{q}, \hat{\chi}, \hat{\sigma}$). This follows by applying the substitutions $\rho_1$ and $\rho_2$ to ret-type($\hat{q}, \hat{\chi}, \hat{\sigma}$) = $\tau; \hat{\sigma}'$, which in turn follows from the premise ret-addr-type($\hat{q}, \hat{\chi}, \hat{\sigma}$) = $\forall[]\{r: \tau; \hat{\sigma}'\}$.
- We claim that currentMR($W(i_{\text{reg}})$) $\in \triangleright W \mathcal{R}[\hat{\chi}]\rho'$.

To show this, consider arbitrary $(\hat{W}, M_1^*, M_2^*) \in \text{currentMR}(W(i_{\text{reg}}))$ such that $\hat{W} \sqsupseteq \triangleright W$. We must show that

$$\langle \hat{W}, M_1^*, M_2^* \rangle \in \mathcal{R}[\hat{\chi}]\rho'$$

Instantiate currentMR($W(i_{\text{reg}})$) $\in \triangleright W \mathcal{R}[\hat{\chi}]\rho$ with $(\hat{W}, M_1^*, M_2^*) \in \text{currentMR}(W(i_{\text{reg}}))$, noting that $\hat{W} \sqsupseteq \triangleright W$, which gives us $(\hat{W}, M_1^*, M_2^*) \in \mathcal{R}[\hat{\chi}]\rho$. Finally, by Lemma 1.19 (register-file subtyping implies inclusion) applied to the premise $\Delta \vdash \chi \leq \hat{\chi}[\sigma_0/\epsilon]((i + k - j)/\epsilon)$ we have that $\mathcal{R}[\hat{\chi}]\rho \subseteq \mathcal{R}[\hat{\chi}[\sigma_0/\epsilon]((i + k - j)/\epsilon)]\rho$. Therefore, it remains to show that $\mathcal{R}[\hat{\chi}[\sigma_0/\epsilon]((i + k - j)/\epsilon)]\rho \subseteq \mathcal{R}[\hat{\chi}]\rho'$.

Next, we examine two cases based on the value of $\hat{q}$:
Case \( \hat{q} = i_{ra} \) Hence, from the premise \( \Delta \vdash \hat{x} \setminus \hat{q} \), we have that \( \Delta \vdash \hat{x} \). Therefore, note that

\[
\mathcal{R}[\hat{x}[\sigma_0/\zeta][(i+k-j)/\epsilon]] \rho \equiv \mathcal{R}[\hat{x}] \rho
\]

since \( \zeta, \epsilon \notin \text{ftv}(\hat{x}) \equiv \mathcal{R}[\hat{x}] \rho' \) by Lemma 1.13

which suffices to show what we needed.

Case \( \hat{q} = r_{ra} \) Hence, from the premise \( \Delta \vdash \hat{x} \setminus \hat{q} \), we actually have that \( \Delta \vdash \hat{x} \setminus r_{ra} \).

Hence, it follows that \( \mathcal{R}[\hat{x}[\sigma_0/\zeta][(i+k-j)/\epsilon]] \setminus r_{ra}] \rho \equiv \mathcal{R}[\hat{x}] \setminus r_{ra}] \rho' \).

Therefore, it remains to show the following for the remaining register \( r_{ra} \), whose type by our premises must be \( \text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\} \epsilon \) where \( \Delta \vdash \tau: \mathcal{R}[r_{ra}: \text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho' \).

Therefore, we must show for arbitrary

\[
(W^*, R_1^*(r_{ra}), R_2^*(r_{ra})) \in W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho'
\]

Note, from our most recent assumption, we have

\[
(W^*, R_1^*(r_{ra}), R_2^*(r_{ra})) \in W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'[\sigma_0/\zeta]\}] \epsilon \rho'
\]

By the substitution (Lemmas 1.15 and 1.16), the latter is equivalent to \( (W^*, R_1^*(r_{ra}), R_2^*(r_{ra})) \in W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho' \)

where \( \rho' = \rho[\zeta \rightarrow (\rho_1(\sigma_0), \rho_2(\sigma_0), S[\sigma_0]) \rho], \epsilon \rightarrow (i+k-j, i+k-j)] \).

To show \( (W^*, R_1^*(r_{ra}), R_2^*(r_{ra})) \in W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho' \), consider arbitrary \( (\hat{W}, M_1, M_2) \in \text{currentMR}(W^*(i_{box})) \) such that \( \hat{W} \equiv W^* \).

Instantiating \( (W^*, R_1^*(r_{ra}), R_2^*(r_{ra})) \) into \( W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho' \), we end up in a position where we have to show that

\[
W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho' \subseteq W[\text{box \ \forall}[r] \{r: \tau; \hat{\sigma}'\}] \epsilon \rho'
\]

The latter follows by expanding the definitions and noting that it suffices to prove the following three facts:

- For any world \( W' \), currentMR\((W'(i_{reg})) \in W', \mathcal{R}[r: \tau] \rho' \) implies currentMR\((W'(i_{reg})) \in W', \mathcal{R}[r: \tau] \rho \) since both are equal to \( \text{currentMR}(W'(i_{reg})) \in W', \mathcal{R}[r: \tau] \rho \) because \( \Delta \vdash \tau \).

- For any world \( W' \), given that currentMR\((W'(i_{stk})) \in W', S[\hat{\sigma}] \rho' \) we claim that currentMR\((W'(i_{reg})) \in W', S[\hat{\sigma}] \rho' \).

Note that \( \hat{\sigma}' = \sigma_0'^n \ldots \sigma_0'^0 \ldots \zeta \), and from the premise \( \Delta \vdash \hat{\sigma}'[\sigma_0/\zeta] \), we know that \( \zeta \notin \text{ftv}(\sigma_0^n) \) for \( n \in \{0, \ldots, k\} \).

Therefore, it suffices to show that

\[
S[\sigma_0^n] \rho' \subseteq S[\sigma_0] \rho' \equiv \varphi_S \subseteq S[\sigma_0] \rho
\]

which is immediate from our choice of \( \varphi_S \) above.

- Finally, we note that \( \tau_r; \hat{\sigma}_r = \rho_\epsilon \) ret-type(\( \epsilon, \{r: \tau_r\}, \hat{\sigma}_r \)). (We show the latter in detail later in the proof so we won’t duplicate it here.) Therefore, note that \( E[\epsilon + \tau_r; \hat{\sigma}_r] \rho' = E[i+k-j+\tau_r; \hat{\sigma}_r] \rho' \).

This completes our proof of the claim marked (1).

- Note that currentMR\((\tau W(i_{reg})) \in W', \mathcal{R}[\hat{\chi}] \rho' \), which follows from currentMR\((W(i_{reg})) \in W', \mathcal{R}[\hat{\chi}] \rho \).

- We claim that currentMR\((W(i_{stk})) \in W, S[\hat{\sigma}] \rho' \) (1)

To show this, consider arbitrary \((\hat{W}, M_1^*, M_2^*) \in \text{currentMR}(W(i_{stk})) \) such that \( \hat{W} \equiv \tau W \).

We must show that

\[
(\hat{W}, M_1^*, M_2^*) \in S[\hat{\sigma}] \rho'.
\]

Recall that above we had \((\tau W, S_1^*, S_2^*) \in \text{currentMR}(W(i_{stk})) \). Therefore, by definition of the \( i_{stk} \) island, it must be that \( M_1^* = S_1^* \) and \( M_2^* = S_2^* \). Hence, it remains for us to show:

\[
(\hat{W}, S_1^*, S_2^*) \in S[\hat{\sigma}] \rho'.
\]
where we recall that
\[
S_1 = w_{10} : \cdots : w_{1j} : S_{10} \\
S_2 = w_{20} : \cdots : w_{2j} : S_{20} \\
\tilde{\sigma} = \tau_0 : \cdots : \tau_j : \zeta
\]
and \( \rho' = \rho[\zeta \mapsto (\rho_1(\sigma_0), \rho_2(\sigma_0), \varphi_S), \epsilon \mapsto (i + k - j, i + k - j)] \)
with \( \varphi_S = \{(\bar{W}, S_{10}, S_{20}) \mid \bar{W} \supseteq W\} \).

Thus, to show \( (\tilde{W}, S_1 \upharpoonright S_2) \in S[\tilde{\sigma}]\rho' \), it suffices to show
\[
- (\tilde{W}, w_{1n}, w_{2n}) \in W[\tau_n]\rho' \text{ for } n \in \{0, \ldots, j\}, \text{ which follows from } (\tilde{W}, w_{1n}, w_{2n}) \in W[\tau_n] \\
- (\tilde{W}, S_{10}, S_{20}) \in S[\zeta]\rho', \text{ which is immediate from } S[\zeta]\rho' = \varphi_S \text{ (by definition) and our choice of } \varphi_S \text{ above.}
\]

- Note that \( \text{currentMR}(\langle \pi > W(i_{\text{stk}}) \rangle) \in_{W} S[\tilde{\sigma}]\rho' \), which follows from \( \text{currentMR}(W(i_{\text{stk}})) \in_{W} S[\tilde{\sigma}]\rho' \).

Hence, we can conclude that
\[
(\pi > W, (I_1[\omega_1/\beta_1][\rho_1(\sigma_0)/\zeta][(i + k - j)/\epsilon], \cdot), (I_2[\omega_2/\beta_2][\rho_2(\sigma_0)/\zeta][(i + k - j)/\epsilon], \cdot)) \in E[\bar{q} \vdash \tau; \tilde{\sigma}']\rho'
\]

Next, we instantiate the latter with \( E_1 \) and \( E_2 \), for which we need to show the following:
\[
(\pi > W, E_1, E_2) \in K[\bar{q} \vdash \tau; \tilde{\sigma}']\rho'
\]

Consider arbitrary \( W', \bar{q}', r_1, r_2 \) such that

1. \( W' \supseteq_{\text{pub}} \pi > W \)
2. \( (\bar{q} =_{\rho} \bar{q}' =_{\rho'} \text{end}\{\hat{\tau}; \bar{\sigma}'\}) \lor \) \( \exists r_0. \bar{q}' = r_0 \land \text{ret-addr}_1(\pi > W, \rho'_1(\bar{q}')) = W'.R_1(r_0) \land \text{ret-addr}_2(\pi > W, \rho'_2(\bar{q}')) = W'.R_2(r_0) \land \) \( \text{ret-reg}_1(W', r_0) = r_1 \land \text{ret-reg}_2(W', r_0) = r_2 \)
3. \( (\pi > W', W'.R_1(r_1), W'.R_2(r_2)) \in W[\tau]\rho' \)
4. \( \text{currentMR}(W'(i_{\text{stk}})) \in_{W'} S[\bar{\sigma}']\rho' \)

We are required to show
\[
(W', (\text{ret} \rho_1(\bar{q}') \{r_1\}, \cdot), (\text{ret} \rho_2(\bar{q}') \{r_2\}, \cdot)) \in O
\]

Next, we collect some facts before we proceed:

- Note that from assumption (4) above, it follows that \( \pi > W' \) is defined, which means that \( W'.k > 0 \).
- Recall from above that either \( \bar{q}' = r_{ra} \) or \( \bar{q}' = i_{ra} \). This fact lets us refine (3) above—that is, since \( \bar{q} \neq_{\rho} \bar{q}' \), we know that
\[
\exists r_0. \bar{q}' = r_0 \land \text{ret-addr}_1(\pi > W, \rho'_1(\bar{q}')) = W'.R_1(r_0) \land \text{ret-addr}_2(\pi > W, \rho'_2(\bar{q}')) = W'.R_2(r_0) \land \) \( \text{ret-reg}_1(W', r_0) = r_1 \land \text{ret-reg}_2(W', r_0) = r_2 \)
Hence, \( \rho_i(q') = \rho_i(r_0) = r_0 \). Therefore, it suffices to show
\[
(W', E_1[(\text{ret } r_0 \{ r_1 \}, \cdot), E_2[(\text{ret } r_0 \{ r_2 \}, \cdot)]) \in \mathcal{O}
\]
Consider arbitrary \((M'_1, M'_2) : W'\). We must show that for \( i \in \{1, 2\} \), either both configurations \((M'_1 | E_1[(\text{ret } r_0 \{ r_1 \}, \cdot)]\) terminate or both are running at \(W'.k\).

Next, we establish that \((>W', W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'\) by considering the two cases of \( \hat{q} \):

**Case \( \hat{q} = r_{ra} \):** Hence, by the definition of \( \text{ret-addr-type} \) we have
\[
\text{ret-addr-type}(q, \hat{x}, \hat{\sigma}) = \text{ret-addr-type}(r_{ra}, \hat{x}, \hat{\sigma}) = \forall .\{ r : \tau; \hat{\sigma}\}^\ast \text{ where } \hat{x}(r_{ra}) = \text{box } \forall .\{ r : \tau; \hat{\sigma}\}^\ast
\]
Since \((>W, R_1 \upharpoonright, R_2 \upharpoonright) \in \text{currentMR}(W_{i_{reg}}) \) (from above) and \( \text{currentMR}(W_{i_{reg}}) \in \mathcal{W} \mathcal{R}[\hat{x}]^\ast \rho' \) (\( \uparrow \) from above), it follows that \((>W, R_1 \upharpoonright, R_2 \upharpoonright) \in \mathcal{R}[\hat{x}]^\ast \rho' \). From the latter, since \( r_{ra} : \text{box } \forall .\{ r : \tau; \hat{\sigma}\}^\ast \in \hat{x} \), we have
\[
 (>W, R_1(r_{ra}), R_2(r_{ra})) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]
Since \( R_i = W.R_i = >W.R_i \), the above is equivalent to
\[
(>W, >W.R_1(r_{ra}), >W.R_2(r_{ra})) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]
Moreover, note that we have the following equalities
\[
\text{ret-addr}_i(>W, \hat{q}) = \text{ret-addr}_i(>W, r_{ra}) \quad \text{since } \hat{q} = r_{ra}
\]
\[
= (>W').R_1(r_{ra}) \quad \text{by definition of } \text{ret-addr}_i
\]
\[
= W'.R_1(r_0) \quad \text{since } \text{ret-addr}_i(>W, \hat{q}) = W'.R_1(r_0)
\]
Hence, we have
\[
(>W, W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]

**Case \( \hat{q} = i_{ra} \):** Hence, by the definition of \( \text{ret-addr-type} \) we have
\[
\text{ret-addr-type}(q, \hat{x}, \hat{\sigma}) = \text{ret-addr-type}(i_{ra}, \hat{x}, \hat{\sigma}) = \forall .\{ r : \tau; \hat{\sigma}\}^\ast \text{ where } \hat{x}(i_{ra}) = \text{box } \forall .\{ r : \tau; \hat{\sigma}\}^\ast
\]
Since \((>W, S_1 \upharpoonright, S_2 \upharpoonright) \in \text{currentMR}(W_{i_{stt}}) \) (from above) and \( \text{currentMR}(W_{i_{stt}}) \in \mathcal{W} \mathcal{S}[\hat{\sigma}]^\ast \rho' \) (\( \uparrow \) from above), it follows that \((>W, S_1 \upharpoonright, S_2 \upharpoonright) \in \mathcal{S}[\hat{\sigma}]^\ast \rho' \). From the latter, since \( i_{ra} : \text{box } \forall .\{ r : \tau; \hat{\sigma}\}^\ast \in \hat{\sigma} \), we have
\[
(>W, S_1(i_{ra}), S_2(i_{ra})) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]
Since \( S_i = W.S_i = >W.S_i \), the above is equivalent to
\[
(>W, >W.S_1(i_{ra}), >W.S_2(i_{ra})) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]
Moreover, note that we have the following equalities
\[
\text{ret-addr}_i(>W, \hat{q}) = \text{ret-addr}_i(>W, i_{ra}) \quad \text{since } \hat{q} = i_{ra}
\]
\[
= (>W).S_i(i_{ra}) \quad \text{by definition of } \text{ret-addr}_i
\]
\[
= W'.R_1(r_0) \quad \text{since } \text{ret-addr}_i(>W, \hat{q}) = W'.R_1(r_0)
\]
Hence, we have
\[
(>W, W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\forall .\{ r : \tau; \hat{\sigma}\}]^\ast \rho'.
\]
Since \( W' \supseteq \triangleright W \), by monotonicity we have
\[
(W',W'.R_1(r_0),W'.R_2(r_0)) \in \mathcal{W}[\text{box} \forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p'.
\]
From the above, we have that \( W', \Psi_1; \triangleright W', R_1(r_0) : \rho'_1(\text{box} \forall \cdot \{r; \tau; \hat{\sigma}'\}^*) \) and \( W', \Psi_2; \triangleright W', R_2(r_0) : \rho'_2(\text{box} \forall \cdot \{r; \tau; \hat{\sigma}'\}^*) \). These facts put together with \( W' \in \text{World} \) and \( \text{ret-reg}_i(W', r_0) \) allow us to conclude \( r = r_1 = r_2 \).

Next, let \( M'_i = (H'_i, R'_i, S'_i) \). From \((M'_1, M'_2) : W' \), since \( W', k > 0 \), we have \((W', M'_1, M'_2) \in \otimes \{ \text{currentMR}(\theta) \mid \theta \in W'. \theta \} \). From the latter, we have three facts, one each for islands \( \text{treg}, \text{tstk}, \) and \( \text{itbox} \).

First, we have that \((\triangleright W', R'_1, R'_2) \in \text{currentMR}(W'(i_{\text{reg}})) \). From the latter it follows that \( R'_i = W'.R_i \) and, hence, we have \((\triangleright W', R'_i(r_0), R'_i(r_0)) \in \mathcal{W}[\text{box} \forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p' \). From the latter, it must be that \( R'_i(r_0) = \ell'_i[\omega'_i] \).

Second, we have \((\triangleright W', S'_1, S'_2) \in \text{currentMR}(W'(i_{\text{tstk}})) \). Hence, we can instantiate assumption (5) from above, i.e., \( \text{currentMR}(W'(i_{\text{tstk}})) \subseteq W', S[\{\hat{\sigma}'\}]p' \), with \((\triangleright W', S'_1, S'_2) \), noting \( \triangleright W' \supseteq W' \) (by Lemma 1.6), which allows us to conclude that \((\triangleright W', S'_1, S'_2) \in S[\{\hat{\sigma}'\}]p' \). From the premise \( \hat{\sigma}' = \tau_{0}' \cdots \tau_{k}' : \zeta \), it follows that \( S'_1 = w_{10}' \cdots w_{1k}' : \tau_0', S'_2 = w_{20}' \cdots w_{2k}' : \tau_0' \), \((\triangleright W', w_{10}'w_{20}') \in W[\tau_{k}']p' \). For \( n \in \{0, \ldots, k\} \), and \((\triangleright W', S'_1, S'_2) \) \( \in S[\{\zeta\}]p' \). From the latter, since \( S[\{\zeta\}]p' = \phi, \) it follows that \( S'_1 = S_{10} \) and \( S'_2 = S_{20} \).

Third, we have that there exist some \( H'_{b1} \subseteq H' \) and \( H'_{b2} \subseteq H' \) such that \((\triangleright W', H'_{b1}, H'_{b2}) \) \( \in \text{currentMR}(W'(i_{\text{itbox}})) \). Instantiate \((W', \ell'_i[\omega'_i], \ell'_i[\omega'_i]) \) \( \in \mathcal{W}[\text{box} \forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p' \) with \((\triangleright W', H'_{b1}, H'_{b2}) \) \( \in \text{currentMR}(W'(i_{\text{itbox}})) \), noting \( \triangleright W' \supseteq W' \) by Lemma 1.6. Hence, we have that
\begin{itemize}
  \item \( H'_{b1}(\ell'_i) = \text{code}[\tau'_i][\chi'_i; \sigma'_1]^q_i.I'_i \),
  \item \( \rho'_1(\tau) = \chi'_i[\omega'_i/\beta'_i] \),
  \item \( \rho'_1(\hat{\sigma}') = \sigma'_1[\omega'_i/\beta'_i] \),
  \item \( \rho'_1(\epsilon) = q'_i[\omega'_i/\beta'_i] \),
  \item \((\triangleright W', \text{code}[\chi'_i; \sigma'_1]^q_i.I'_i) [\omega'_i/\beta'_i], \text{code}[\chi'_2; \sigma'_2]^q_i.I'_2 [\omega'_2/\beta'_2]) \in \mathcal{H}[\forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p' \)
\end{itemize}
Hence, we know that \( \tilde{R}_i'(r_0) = \ell_i'[\omega'_i] \) and \( H'_i(\ell'_i) = \text{code}[\tau'_i][\chi'_i; \sigma'_1]^q_i.I'_i \), and therefore
\[
(\langle H'_1, R'_1, S'_1 \rangle \mid E_i(\{\tau \rightarrow r_1, \ldots \}), \langle H'_2, R'_2, S'_2 \rangle \mid E_i(\{I'_i[\omega'_i/\beta'_i], \cdot \})) \rightarrow (\langle H'_1, R'_1, S'_1 \rangle \mid E_i(\{I'_i[\omega'_i/\beta'_i], \cdot \}))
\]
Therefore, it suffices to show that for \( i \in \{1, 2\} \), either both configurations \((M'_i \mid E_i(\{I'_i[\omega'_i/\beta'_i], \cdot \}))\) terminate or both are running at \( W', k - 1 \).

We proceed by noting that
\[
(\triangleright W', \text{code}[\chi'_1; \sigma'_1]^q_i.I'_1 [\omega'_1/\beta'_1], \text{code}[\chi'_2; \sigma'_2]^q_i.I'_2 [\omega'_2/\beta'_2]) \in \mathcal{H}[\forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p'
\]
\[
\equiv (\triangleright W', \text{code}[\chi'_1[\omega'_1/\beta'_1], \sigma'_1[\omega'_1/\beta'_1], I'_1[\omega'_1/\beta'_1]], \text{code}[\chi'_2[\omega'_2/\beta'_2], \sigma'_2[\omega'_2/\beta'_2], I'_2[\omega'_2/\beta'_2])] \in \mathcal{H}[\forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p'
\]
\[
\equiv (\triangleright W', \text{code}[\rho'_1(\tau); \rho'_1(\hat{\sigma}'), \rho'_1(\epsilon)], \text{code}[\rho'_2(\tau); \rho'_2(\hat{\sigma}'), \rho'_2(\epsilon)], I'_1[\omega'_1/\beta'_1], I'_2[\omega'_2/\beta'_2]) \in \mathcal{H}[\forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p'
\]
\[
\equiv (\triangleright W', \text{code}[\rho'_1(\tau); \rho'_1(\hat{\sigma}'), i + k - j, I'_1[\omega'_1/\beta'_1]], \text{code}[\rho'_2(\tau); \rho'_2(\hat{\sigma}'), i + k - j, I'_2[\omega'_2/\beta'_2]]) \in \mathcal{H}[\forall \cdot \{r; \tau; \hat{\sigma}'\}^*]p'
\]
Instantiate the latter with \( \triangleright W', \emptyset \in \mathcal{D}[\cdot], \tau_r, \) and \( \sigma_r \). We note the following:

\[
72
\]
• We have $\triangleright W' \supseteq \triangleright W'$ by reflexivity.

• We claim that $\tau_i; \sigma_r = \rho'$ ret-type($\epsilon, \{r: \tau\}, \hat{\sigma}'$). It suffices to show for $i \in \{1, 2\}$:

\[
\rho'_i(\tau_i); \rho'_i(\sigma_r) = \text{ret-type}(\rho'_i(\epsilon), \rho'_i(\{r: \tau\}), \rho'_i(\hat{\sigma}'))
\]

\[
\equiv \rho'_i(\tau_i); \rho'_i(\sigma_r) = \text{ret-type}(i + k - j, \rho'_i(\{r: \tau\}), \rho'_i(\tau_0 :: \cdots :: \tau_j :: \zeta))
\]

\[
\equiv \rho'_i(\tau_i); \rho'_i(\sigma_r) = \text{ret-type}(i + k - j, \rho'_i(\{r: \tau\}), \rho'_i(\tau_0 :: \cdots :: \rho'_i(\tau_j) :: \rho_i(\sigma_0))
\]

Recall that $\tau_i; \sigma_r = \text{ret-type}(i, \chi, \sigma)$ where $\sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0$. Hence, $FTV(\tau_i; \sigma_r) \subseteq \Delta$ and $\sigma_0(i - j - 1) = \text{box} \forall \{r': \tau_i; \sigma_r\}^q$ for some $r'$ and $q'$.

Putting these facts together, we have that

\[
\rho'_i(\hat{\sigma}'(i + k - j)) = \rho'_i(\sigma_0(i + k - j - (k + 1)))
\]

\[
= \rho_i(\sigma_0(i - j - 1))
\]

\[
= \rho_i(\text{box} \forall \{r': \tau_i; \sigma_r\}^q)
\]

which means that $\rho_i(\tau_i); \rho_i(\sigma_r) = \text{ret-type}(i + k - j, \rho'_i(\{r: \tau\}), \rho'_i(\hat{\sigma}'))$ as we needed to show.

• We claim that currentMR($\triangleright W'(i_{reg}) \triangleright W'$).

To show this, consider arbitrary $(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(\triangleright W'(i_{reg}))$ such that $\tilde{W} \supseteq \triangleright W'$.

We must show that

\[
(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(\triangleright W'(i_{reg})) \subseteq \text{currentMR}(W'(i_{reg}))
\]

Thus, $(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(W'(i_{reg}))$. Moreover, since we already have $(\triangleright W', R^*_1, R^*_2) \in \text{currentMR}(W'(i_{reg}))$, by the definition of island $i_{reg}$, it must be that $M^*_i = R^*_i$. Therefore, we are required to show

\[
(\tilde{W}, R^*_1, R^*_2) \in \text{currentMR}(\triangleright W'(i_{reg}))
\]

It suffices to show

\[
(\tilde{W}, R^*_1(r), R^*_2(r)) \in \text{currentMR}(\triangleright W'(i_{reg}))
\]

which follows by monotonicity (Lemma 1.8) from our earlier assumption (4), namely $(\triangleright W', R^*_1(r_1), R^*_2(r_2)) \in \text{currentMR}(\triangleright W'(i_{reg}))$ since $r = r_1 = r_2$ (from above) and since $W'.R_1 = R'_1$ and $W'.R_2 = R'_2$.

• We claim that currentMR($\triangleright W'(i_{stk}) \triangleright W'$).

To show this, consider arbitrary $(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(\triangleright W'(i_{stk}))$ such that $\tilde{W} \supseteq \triangleright W'$. We must show that

\[
(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(\triangleright W'(i_{stk}))
\]

Note that since $W'.k > 0$ (from above), $(\triangleright W), k = W'.k - 1$ and therefore

\[
\text{currentMR}(\triangleright W'(i_{reg})) = \text{currentMR}(\triangleright W'(i_{reg})) \subseteq \text{currentMR}(W'(i_{reg}))
\]

Thus, $(\tilde{W}, M^*_1, M^*_2) \in \text{currentMR}(W'(i_{reg}))$. Moreover, since we already have $(\triangleright W', S^*_1, S^*_2) \in \text{currentMR}(W'(i_{stk}))$, by the definition of island $i_{stk}$, it must be that $M^*_i = S^*_i$. Therefore, we are required to show

\[
(\tilde{W}, S^*_1, S^*_2) \in \text{currentMR}(\triangleright W'(i_{reg}))
\]

Since $\hat{\sigma}' = \tau^*_0 :: \cdots :: \tau^*_k :: \zeta$, with $S^*_1 = w^*_1 :: \cdots :: w^*_k :: S^*_1$ and $S^*_2 = w^*_2 :: \cdots :: w^*_2 :: S^*_2$, the above follows from
Lemma 1.64 (Call from Top Level)

- \((\widetilde{W}, w'_{1n}, w'_{2n}) \in W[\tau'_n]\rho'\) for \(n \in \{0, \ldots, k\}\), which follow by monotonicity (Lemma 1.8) from \((\triangleright W', w'_{1n}, w'_{2n}) \in W[\tau'_n]\rho'\), and

- \((\widetilde{W}, S'_{10}, S'_{20}) \in S[\zeta]\rho'\), which follows from \(S[\zeta]\rho' = \varphi_S\) and the definition of \(\varphi_S\) since \(S'_{10} = S_{10}\) and \(S'_{20} = S_{20}\).

Hence, we can conclude that

\[(\triangleright W', (I'_1[\omega'_1/\beta'_1], \cdot), (I'_2[\omega'_2/\beta'_2], \cdot)) \in E[i \vdash \tau_r; \sigma_r]\rho']\]

By Lemma 1.16, and noting that \(FTV(\tau_r; \sigma_r) \subseteq \Delta\) we have

\[(\triangleright W', (I'_1[\omega'_1/\beta'_1], \cdot), (I'_2[\omega'_2/\beta'_2], \cdot)) \in E[i + k - j \vdash \tau_r; \sigma_r]\rho]. \tag{18}\]

Near the beginning of this proof, we had assumed \((W, E_1, E_2) \in \mathcal{K}[i \vdash \tau_r; \sigma_r]\rho\),

We now apply Lemma 1.10 (monotonicity for evaluation contexts) to \((W, E_1, E_2) \in \mathcal{K}[i \vdash \tau_r; \sigma_r]\rho\), noting the following:

- \(\triangleright W' \equiv pub W\), which follows by transitivity of \(\equiv pub\) and the following \(\triangleright W' \equiv pub W' \equiv pub\)
  
- \(\triangleright W \equiv pub W\) (which we have from above and from Lemma 1.6).

- \(ret\text{-}addr_1(W, i) = ret\text{-}addr_1(\triangleright W', i + k - j)\) and \(ret\text{-}addr_2(W, i) = ret\text{-}addr_2(\triangleright W', i + k - j)\),
  
which follow from:

\[
\begin{align*}
ret\text{-}addr_1(W, i) &= W.S_1(i) \\
&= (w_{10} :: \cdots :: w_{1j} :: S_{10})(i) \\
&= S_{10}(i - j + 1)
\end{align*}
\]

and

\[
\begin{align*}
ret\text{-}addr_1(\triangleright W', i + k - j) &= W'.S_1(i + k - j) \\
&= (w'_{10} :: \cdots :: w'_{1k} :: S_{10})(i + k - j) \\
&= S_{10}(i + k - j - (k + 1)) \\
&= S_{10}(i - j - 1)
\end{align*}
\]

and analogously for \(ret\text{-}addr_2\).

Hence, we have that \((\triangleright W', E_1, E_2) \in \mathcal{K}[i + k - j \vdash \tau_r; \sigma_r]\rho\).

Instantiating 18 with the above, we have \((\triangleright W', E_1[(I'_1[\omega'_1/\beta'_1], \cdot), E_2[(I'_2[\omega'_2/\beta'_2], \cdot)]) \in O\).

By instantiating the latter with \(M'_1\) and \(M'_2\), noting \((M'_1, M'_2) : \triangleright W'\), we have that \((M'_1 | E_1[(I'_1[\omega'_1/\beta'_1], \cdot)])\)

and \((M'_2 | E_2[(I'_2[\omega'_2/\beta'_2], \cdot)])\) either both terminate or are both running at \(W'.k - 1\), as we needed to show.

Hence, we conclude that

\[(\triangleright W, E_1[(I_1[\omega_1/\beta_1][\rho_1(\sigma_0)/\zeta][q_1/\epsilon], \cdot), E_2[(I_2[\omega_2/\beta_2][\rho_2(\sigma_0)/\zeta][q_2/\epsilon], \cdot)]) \in O\]

Now instantiate the above with \(M_1\) and \(M_2\), noting that we have \((M_1, M_2) : \triangleright W\) by Lemma 1.6 since

\((M_1, M_2) : W\). Hence, we have that for \(i \in \{1, 2\}\), either both configurations \((M_i | E_1[(I_1[\omega_1/\beta_1][\rho_1(\sigma_0)/\zeta][q_1/\epsilon], \cdot)])\)

terminate or both are running at \(W'.k - 1\), which is exactly what we needed to show!
• ret-addr-type($q, \hat{x}, \hat{\sigma}$) = $\forall[\cdot] \{ \tau; \hat{\sigma}' \} \cdot$.

• $\Delta \vdash \sigma_0$.

• $\Delta \vdash \forall[\cdot] \{ \hat{x}[\sigma_0/\zeta] \{ \{ \tau^*; \sigma^* \} / \epsilon \}; \hat{\sigma}[\sigma_0/\zeta] \{ \{ \tau^*; \sigma^* \} / \epsilon \} \} \cdot$.

• $\Delta \vdash \chi \leq \hat{x}[\sigma_0/\zeta] \{ \{ \tau^*; \sigma^* \} / \epsilon \}$,

• $\sigma = \bar{\tau} :: \sigma_0$,

• $\hat{\sigma} = \tau :: \zeta$, and

• $\hat{\sigma}' = \bar{\tau}' :: \zeta$.

we have that $\Psi; \Delta; \chi; \sigma; \{ \{ \tau^*; \sigma^* \} / \epsilon \} \vdash \text{call } u_1 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \} \approx_1 \text{call } u_2 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \}.

Proof

Clearly, $\Psi; \Delta; \chi; \sigma; \{ \{ \tau^*; \sigma^* \} / \epsilon \} \vdash \text{call } u_1 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \} \text{ and }$ $\Psi; \Delta; \chi; \sigma; \{ \{ \tau^*; \sigma^* \} / \epsilon \} \vdash \text{call } u_2 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \}$ follow from the premises. Consider arbitrary $W, \gamma$, and $\rho$ such that $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta], (W, \gamma) \in \mathcal{G}[\Gamma] \rho$, currentMR$(W(i_{\text{reg}})) \subseteq W \mathcal{R}[\chi] \rho$, and currentMR$(W(i_{\text{stk}})) \subseteq W \mathcal{S}[\sigma] \rho$. We need to show that

$$
(W, \rho_1(\gamma_1((\text{call } u_1 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \})), \rho_2(\gamma_2((\text{call } u_2 \{ \sigma_0, \{ \{ \tau^*; \sigma^* \} / \epsilon \} \}) \}))
$$

$$
= (W, \text{call } \rho_1(u_1) \{ \rho_1(\sigma_0), \{ \{ \tau^*; \sigma^* \} / \epsilon \} \} \}, \text{call } \rho_2(u_2) \{ \rho_2(\sigma_0), \{ \{ \tau^*; \sigma^* \} / \epsilon \} \}) \in \mathcal{E}[\{ \{ \tau^*; \sigma^* \} / \epsilon \} \vdash \tau^*; \sigma^*] \rho.
$$

Note that if $W.k = 0$ then we are done, since for any evaluation contexts $E_i$ and memories $M_i$, we can immediately show that running$(0, (M_i \mid E_i[e_i]))$.

In the following, assume $W.k > 0$. Also, let $q_1 = \text{end } \{ \rho_i(\tau^*); \rho_i(\sigma^*) \}$

Consider arbitrary $(W, E_1, E_2) \in \mathcal{K}[\{ \{ \tau^*; \sigma^* \} / \epsilon \} \vdash \tau^*; \sigma^*] \rho$. We need to show that

$$(W, E_1[\{ \text{call } \rho_1(u_1) \{ \rho_1(\sigma_0), q_1, \cdot \} \}, E_2[\{ \text{call } \rho_2(u_2) \{ \rho_2(\sigma_0), q_2, \cdot \} \}] \in \mathcal{O}$$

Consider arbitrary $(M_1, M_2) : W$. We must show that for $i \in \{1, 2\}$, either both configurations $(M_i \mid E_i[\{ \text{call } \rho_i(u_i) \{ \rho_i(\sigma_0), q_i, \cdot \} \})$ terminate or both are running at $W.k$.

Instantiate the first premise with $W$ and $\rho$, noting $W \in \mathcal{H}[\Psi], \rho \in \mathcal{D}[\Delta]$, and currentMR$(W(i_{\text{reg}})) \subseteq W \mathcal{R}[\chi] \rho$. Thus we have that $(W, W.R_1(\rho_1(u_1)), W.R_2(\rho_2(u_2))) \in W \mathcal{S}[\sigma] \rho$ from the definition of the latter, we have that $W.R_i(\rho_i(u_i)) = \epsilon_i[\omega_i]$. Next, let $M_i = (H_i, R_i, S_i)$.

From $(M_1, M_2) : W$, since $W.k > 0$, we have $(\triangleright W, M_1, M_2) \in \bigotimes\{ \text{currentMR}(\theta) \mid \theta \in W.\Theta \}$. From the latter, we have three facts, one each for islands $i_{\text{reg}}, i_{\text{stk}}$, and $i_{\text{box}}$.

First, we have that $(\triangleright W, R_1[\cdot], R_2[\cdot]) \in \text{currentMR}(W(i_{\text{reg}}))$. From the latter it follows that $R_i = W.R_i$ and, hence, $\hat{R}_i(\rho_i(u_i)) = \hat{\epsilon}_i[\omega_i]$. Second, we have $(\triangleright W, S_1[\cdot], S_2[\cdot]) \in \text{currentMR}(W(i_{\text{stk}}))$. From the latter and currentMR$(W(i_{\text{stk}})) \subseteq W \mathcal{S}[\sigma] \rho$, it follows that $(\triangleright W, S_1[\cdot], S_2[\cdot]) \subseteq \mathcal{S}[\sigma] \rho$. From the premise $\sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0$, it follows that $S_1 = w_{10} :: \cdots :: w_{1j} :: S_{10}, S_2 = w_{20} :: \cdots :: w_{2j} :: S_{20}$. $(\triangleright W, w_{1n}, w_{2n}) \in W[\tau_n] \rho$ for $n \in \{0, \ldots, j\}$, and $(\triangleright W, S_{10}[\cdot], S_{20}[\cdot]) \in \mathcal{S}[\sigma_0] \rho$.

Third, we have that there exist some $H_{b1} \subseteq H_1$ and $H_{b2} \subseteq H_2$ such that $(\triangleright W, H_{b1}[\cdot], H_{b2}[\cdot]) \in \text{currentMR}(W(i_{\text{box}}))$. We use the latter to instantiate $(W, \hat{\epsilon}_1[\omega_{1\cdot}], \hat{\epsilon}_2[\omega_{2\cdot}]) \in W[\text{box } \mathcal{S}[\zeta, \cdot] \{ \hat{x}; \hat{\sigma} \} \rho$, noting that $\triangleright W \supseteq W$, which allows us to conclude:

• $H_{bi}(\hat{\epsilon}_i) = \text{code } [\hat{\tau}_i, \zeta, \cdot] \{ \hat{x}_i; \hat{\sigma}_i \} \cup_{1} I_i$. 75
Further, we note that we have
\[ \rho_i(\check{x}) = \chi_i[\omega_i/\beta_i], \]
\[ \rho_i(\check{\sigma}) = \sigma_i[\omega_i/\beta_i], \]
\[ \rho_i(\check{q}) = q_i[\omega_i/\beta_i], \]
and
\[ (\triangleright W, \text{code}[\zeta, \epsilon]\{\chi_1; \sigma_1\}^{q_1}, I_1)[\omega_1/\beta_1], \]
\[ \text{code}[\zeta, \epsilon]\{\chi_2; \sigma_2\}^{q_2}, I_2)[\omega_2/\beta_2]) \in \mathcal{H}[\forall[\zeta, \epsilon].\{\check{x}; \check{\sigma}\}^{q}]\rho \]
Hence, we know that \( \hat{R}_i(\rho_i(u_i)) = \ell_i[\omega_i] \) and \( H_i(\ell_i) = \text{code}[\beta_i, \zeta, \epsilon]\{\chi_i; \sigma_i\}^{q_i}, I_i \), and therefore
\[ \langle(H_i, R_i, S_i) | E_i[[\text{call } \rho_i(u_i) \{\rho_i(\sigma_0), q_i, \cdot\}]] \rangle \]
\[ \longrightarrow^1 \]
\[ \langle(H_i, R_i, S_i) | E_i[[\text{code}[\beta_i, \zeta, \epsilon]\{\rho_i(\sigma_0)/\zeta], q_i, \cdot\}]] \rangle \]
Therefore, it suffices to show that for \( i \in \{1, 2\} \), either both configurations
\[ (M_i | E_i[[\text{code}[\beta_i, \zeta, \epsilon]\{\rho_i(\sigma_0)/\zeta], q_i, \cdot\}]] \]
terminate or both are running at \( W, k - 1 \).

We proceed by noting that from the premise ret-addr-type(\( \check{q}, \check{x}, \check{\sigma} \) = \( \forall [], \{ r: \tau; \check{\sigma}' \}^r \)), by definition of ret-addr-type, it must be that either \( \check{q} = r_{ra} \) or \( \check{q} = r_{ra} \).

Further, we note that we have
\[ (\triangleright W, \text{code}[\zeta, \epsilon]\{\chi_1; \sigma_1\}^{q_1}, I_1)[\omega_1/\beta_1], \]
\[ \text{code}[\zeta, \epsilon]\{\chi_2; \sigma_2\}^{q_2}, I_2)[\omega_2/\beta_2]) \in \mathcal{H}[\forall[\zeta, \epsilon].\{\check{x}; \check{\sigma}\}^{q}]\rho \]
\[ \equiv (\triangleright W, \text{code}[\zeta, \epsilon]\{\chi_1; \sigma_1\}^{q_1}, I_1)[\omega_1/\beta_1], \]
\[ \text{code}[\zeta, \epsilon]\{\chi_2; \sigma_2\}^{q_2}, I_2)[\omega_2/\beta_2]) \in \mathcal{H}[\forall[\zeta, \epsilon].\{\check{x}; \check{\sigma}\}^{q}]\rho \]

Instantiate the latter with \( \triangleright W, \rho^*, \tau, \text{ and } \check{\sigma}' \), where
\[ \rho^* = \{ \zeta \mapsto (\rho_1(\sigma_0), \rho_2(\sigma_0), \varphi_S), \epsilon \mapsto (q_1, q_2) \} \]
and
\[ \varphi_S = \{ (\hat{W}, S_{110}), (S_{201}) | \hat{W} \triangleright \triangleright W \} \]
We note the following:

- We have \( \triangleright W \triangleright \triangleright W \) by reflexivity.

- Note that \( \rho^* \in \mathcal{D}[\zeta, \epsilon] \), which follows by the definition of \( \mathcal{D}[\cdot] \) and by applying Lemma 1.18 to \( (\triangleright W, S_{110}, S_{201}) \in \mathcal{S}[\sigma_0]\rho \).

- Let \( \rho' = \rho \cup \rho^* \).

- Note that \( \tau; \check{\sigma}' = \rho' \) ret-type(\( \hat{q}, \check{x}, \check{\sigma} \)). This follows by applying the substitutions \( \rho_1' \) and \( \rho_2' \) to ret-type(\( \hat{q}, \check{x}, \check{\sigma} \) = \( \tau; \check{\sigma}' \), which in turn follows from the premise ret-addr-type(\( \hat{q}, \check{x}, \check{\sigma} \) = \( \forall [], \{ r: \tau; \check{\sigma}' \}^r \)).

- We claim that currentMR(\( W(i_{reg}) \)) \( \in_{R \triangleright \triangleright W} \mathcal{R}[\hat{x}]\rho' \).

To show this, consider arbitrary \( (\hat{W}, M^*_1, M^*_2) \in \text{currentMR}(W(i_{reg})) \) such that \( \hat{W} \triangleright \triangleright W \). We must show that
\[ (\hat{W}, M^*_1, M^*_2) \in \mathcal{R}[\hat{x}]\rho' \]
Instantiate currentMR(\( W(i_{reg}) \)) \( \in_{R \triangleright \triangleright W} \mathcal{R}[\hat{x}]\rho \) with \( (\hat{W}, M^*_1, M^*_2) \in \text{currentMR}(W(i_{reg})) \), noting that \( \hat{W} \triangleright \triangleright W \), which gives us \( (\hat{W}, M^*_1, M^*_2) \in \mathcal{R}[\hat{x}]\rho \). Finally, by Lemma 1.19 (register-file subtyping implies inclusion) applied to the premise \( \Delta \vdash \chi \leq \hat{x} | {\sigma_0/\zeta}| \text{end}\{\tau^*; \sigma^*\}/\epsilon \) we have that \( \mathcal{R}[\hat{x}]\rho \subseteq \mathcal{R}[\hat{x} | {\sigma_0/\zeta}| \text{end}\{\tau^*; \sigma^*\}/\epsilon] \rho' \).

Therefore, we remain to show that \( \mathcal{R}[\hat{x} | {\sigma_0/\zeta}| \text{end}\{\tau^*; \sigma^*\}/\epsilon] \rho \subseteq \mathcal{R}[\hat{x}]\rho' \).

Next, we examine two cases based on the value of \( \hat{q} \):
Case $\hat{q} = i_{ra}$ Hence, from the premise $\Delta \vdash \hat{\chi} \setminus \hat{q}$, we have that $\Delta \vdash \hat{\chi}$. Therefore, note that

$$\mathcal{R}[\hat{\chi}[\sigma_0/\zeta][\text{end}\{\tau^*; \sigma^*\}/\epsilon]]\rho = \mathcal{R}[\hat{\chi}]\rho'$$

which suffices to show what we needed.

Case $\hat{q} = r_{ra}$ Hence, from the premise $\Delta \vdash \hat{\chi} \setminus \hat{q}$, we actually have that $\Delta \vdash \hat{\chi} \setminus r_{ra}$. Hence, it follows that $\mathcal{R}[(\hat{\chi}[\sigma_0/\zeta][\text{end}\{\tau^*; \sigma^*\}/\epsilon]) \setminus r_{ra}]\rho = \mathcal{R}[\hat{\chi}]\setminus r_{ra}]\rho'$. Therefore, it remains to show the following for the remaining register $r_{ra}$, whose type by our premises must be $\text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon$ where $\Delta \vdash \tau$:

$$\mathcal{R}[r_{ra}: \text{box} \forall [], \{r: \tau; \hat{\sigma}'[\sigma_0/\zeta]\}^\epsilon[\text{end}\{\tau^*; \sigma^*\}/\epsilon]]\rho\subseteq \mathcal{R}[r_{ra}: \text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho'$$

Therefore, we must show for arbitrary $(W^*, R^1_{ra}, R^2_{ra}) \in \mathcal{R}[r_{ra}: \text{box} \forall [], \{r: \tau; \hat{\sigma}'[\sigma_0/\zeta]\}^\epsilon[\text{end}\{\tau^*; \sigma^*\}/\epsilon]]\rho$ that $(W^*, R^1_{ra}, R^2_{ra}) \in \mathcal{R}[r_{ra}: \text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho'$. Hence, it suffices to show that

$$(W^*, R^1_{ra}(r_{ra}), R^2_{ra}(r_{ra})) \in \mathcal{W}[\text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho'.$$

Note, from our most recent assumption, we have

$$(W^*, R^1_{ra}(r_{ra}), R^2_{ra}(r_{ra})) \in \mathcal{W}[\text{box} \forall [], \{r: \tau; \hat{\sigma}'[\sigma_0/\zeta]\}^\epsilon[\text{end}\{\tau^*; \sigma^*\}/\epsilon]]\rho.\text{ By substitution (Lemmas 1.15 and 1.16), the latter is equivalent to } (W^*, R^1_{ra}(r_{ra}), R^2_{ra}(r_{ra})) \in \mathcal{W}[\text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho^\ast \text{ where } \rho^\ast = \rho[\zeta \mapsto (\rho, 0); \rho_2(\zeta), S[\sigma_0/\zeta], \epsilon \mapsto (q_1, q_2)].$$

To show $(W^*, R^1_{ra}(r_{ra}), R^2_{ra}(r_{ra})) \in \mathcal{W}[\text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho^\ast$, consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}(W^*_{i_{ra}})$ such that $W \sqsubseteq W^*$. Instantiating $(W^*, R^1_{ra}(r_{ra}), R^2_{ra}(r_{ra})) \in \mathcal{W}[\text{box} \forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho^\ast$, we end up in a position where we have to show that

$$\mathcal{H}[\forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho^\ast \subseteq \mathcal{H}[\forall [], \{r: \tau; \hat{\sigma}'\}^\epsilon]\rho'.$$

The latter follows by expanding the definitions and noting that it suffices to prove the following three facts:

- For any world $W'$, currentMR(\(W'(i_{reg})\)) $\subseteq W'\mathcal{R}[r: \tau]\rho' \text{ implies currentMR}(W'(i_{reg})) \subseteq W'\mathcal{R}[r: \tau]\rho$ because $\Delta \vdash \tau$.

- For any world $W'$, given that currentMR(\(W'(i_{stk})\)) $\subseteq W'\mathcal{S}[\hat{\sigma}']\rho'$ we claim that currentMR(\(W'(i_{reg})\)) $\subseteq W'\mathcal{S}[\hat{\sigma}']\rho^\ast$. Note that $\hat{\sigma}' = \tau_0^n : \cdot : \tau_k^n : \zeta$, and from the premise $\Delta \vdash \hat{\sigma}'[\sigma_0/\zeta]$, we know that $\zeta \notin \text{ftv}(\tau_k^n)$ for $n \in \{0, \ldots, k\}$. Therefore, it suffices to show that

$$\mathcal{S}[\zeta]\rho' \subseteq \mathcal{S}[\zeta]\rho^\ast \equiv \varphi_S \subseteq \mathcal{S}[\sigma_0/\zeta]\rho$$

which is immediate from our choice of $\varphi_S$ above.

- Finally, we note that $\tau^*; \sigma^* = \rho' \text{ ret-type}(\epsilon, \{r: \tau, \hat{\sigma}'\})$. Therefore, note that $\mathcal{E}[\epsilon \vdash \tau^*; \sigma^*] = \mathcal{E}[\text{end}\{\tau^*; \sigma^*\}] = \tau^*; \sigma^*\rho = \mathcal{E}[\text{end}\{\tau^*; \sigma^*\}] = \tau^*; \sigma^*\rho$. This completes our proof of the claim marked (5).

- Note that currentMR(\((\triangleright W')(i_{reg})\)) $\subseteq_{D-W} \mathcal{R}[\hat{\chi}]\rho'$, which follows from currentMR(\(W(i_{reg})\)) $\subseteq_{D-W} \mathcal{R}[\hat{\chi}]$.

- We claim that currentMR(\(W(i_{stk})\)) $\subseteq_{D-W} \mathcal{S}[\hat{\sigma}]\rho'$

To show this, consider arbitrary $(\tilde{W}, M^1_i, M^2_i) \in \text{currentMR}(W(i_{stk}))$ such that $\tilde{W} \sqsubseteq D-W$. We must show that

$$(\tilde{W}, M^1_i, M^2_i) \in \mathcal{S}[\hat{\sigma}]\rho'$$

Recall that above we had $(\triangleright W, S^1, S^2) \in \text{currentMR}(W(i_{stk}))$. Therefore, by definition of the $i_{stk}$ island, it must be that $M^1_i = S^1$ and $M^2_i = S^2$. Hence, it remains for us to show:

$$(\tilde{W}, S^1, S^2) \in \mathcal{S}[\hat{\sigma}]\rho'$$
where we recall that
\[
S_1 = w_{10} :: \cdots :: w_{1j} :: S_{10} \\
S_2 = w_{20} :: \cdots :: w_{2j} :: S_{20} \\
\hat{\sigma} = \tau_0 :: \cdots :: \tau_j :: \zeta
\]
and \( \rho' = \rho[I \mapsto (\rho_1(\sigma_0), \rho_2(\sigma_0), \varphi_S), \epsilon \mapsto (q_1, q_2)] \)
with \( \varphi_S = \{ (\tilde{W}, S_{10} |, S_{20} |) | \tilde{W} \supseteq \triangledown W \} \).

Thus, to show \((\tilde{W}, S_{10} |, S_{20} |) \in S[\hat{\sigma}][\rho']\), it suffices to show
- \((\tilde{W}, w_{1n}, w_{2n}) \in S[\tau_n][\rho']\) for \(n \in \{0, \ldots, j\}\), which follows from \((\triangledown W, w_{1n}, w_{2n}) \in S[\tau_n]\)
by monotonicity (Lemma 1.8).
- \((\tilde{W}, S_{10}, S_{20}) \in S[\zeta][\rho']\), which is immediate from \(S[\zeta][\rho'] = \varphi_S\) (by definition) and our choice
of \(\varphi_S\) above.

\begin{itemize}
    \item Note that \text{currentMR}((\triangledown W)(i_{stk})) \(\in_{\triangledown W} S[\hat{\sigma}][\rho']\), which follows from \text{currentMR}(W(i_{stk})) \(\in_{\triangledown W} S[\hat{\sigma}][\rho']\).
\end{itemize}

Hence, we can conclude that
\[
(\triangledown W, (I_1[w_1/\beta_1][\rho_1(\sigma_0)/\zeta][q_1/\epsilon], \cdot), (I_2[w_2/\beta_2][\rho_2(\sigma_0)/\zeta][q_2/\epsilon], \cdot)) \in S[\hat{\sigma}][\triangledown W][\rho']
\]

Next, we instantiate the latter with \(E_1\) and \(E_2\), for which we need to show the following:
\[
(\triangledown W, E_1, E_2) \in K[\hat{\sigma}][\triangledown W][\rho']
\]

Consider arbitrary \(W', \hat{q}', r_1, r_2\) such that
\begin{enumerate}
    \item \(W' \supseteq_{\triangledown pub} \triangledown W\),
    \item \((\hat{q} = \rho[I \mapsto (\hat{\rho}_1(\hat{q}')] = \hat{\rho}_2(\hat{q}')] = \hat{\rho}_1(\hat{q}), \hat{\rho}_2(\hat{q})) = W'.R_1(r_0) \wedge \text{ret-addr}_1(\triangledown W, \rho_1(\hat{q})) = W'.R_2(r_0) \wedge \\
\text{ret-addr}_2(\triangledown W, \rho_2(\hat{q})) = W'.R_2(r_0) \wedge \\
\text{ret-reg}_1(W', r_0) = r_1 \wedge \text{ret-reg}_2(W', r_0) = r_2)
\end{enumerate}

We are required to show
\[
(W', E_1[(\text{ret } \rho_1(\hat{q}') \{r_1\}, \cdot)], E_2[(\text{ret } \rho_2(\hat{q}') \{r_2\}, \cdot)]) \in O
\]

Next, we collect some facts before we proceed:
\begin{itemize}
    \item Note that from assumption (4) above, it follows that \(\triangledown W'\) is defined, which means that \(W'.k > 0\).
    \item Recall from above that either \(\hat{q} = r_{ra}\) or \(\hat{q} = i_{ra}\). This fact lets us refine (3) above—that is, since
\[
\exists_{r_0}. \hat{q}' = r_0 \wedge \text{ret-addr}_1(\triangledown W, \rho'_1(\hat{q})) = W'.R_1(r_0) \wedge \\
\text{ret-addr}_2(\triangledown W, \rho'_2(\hat{q})) = W'.R_2(r_0) \wedge \\
\text{ret-reg}_1(W', r_0) = r_1 \wedge \text{ret-reg}_2(W', r_0) = r_2)
\]
\end{itemize}

Hence, \(\rho_i(\hat{q}') = \rho_i(r_0) = r_0\). Therefore, it suffices to show
\[
(W', E_1[(\text{ret } r_0 \{r_1\}, \cdot)], E_2[(\text{ret } r_0 \{r_2\}, \cdot)]) \in O
\]

Consider arbitrary \((M'_1, M'_2) : W'\). We must show that for \(i \in \{1, 2\}\), either both configurations
\((M'_i, E_1[(\text{ret } r_0 \{r_1\}, \cdot)])\) terminate or both are running at \(W'.k\).

Next, we establish that \((\triangledown W, W'.R_1(r_0), W'.R_2(r_0)) \in W[\boxed{\forall}.(r; \tau; \hat{\sigma}')^\epsilon][\rho']\) by considering the two cases of \(\hat{q}\):
Case \( \hat{q} = r_{ra} \): Hence, by the definition of ret-addr-type we have

\[
\text{ret-addr-type}(\hat{q}, \hat{x}, \hat{\sigma}) = \text{ret-addr-type}(r_{ra}, \hat{x}, \hat{\sigma}) = \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon \text{ where } \hat{x}(r_{ra}) = \text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon
\]

Since \( (\triangleright W, R_1 \uparrow, R_2 \uparrow) \in \text{currentMR}(W(i_{reg})) \) (from above) and \( \text{currentMR}(W(i_{reg})) \subseteq_{\triangleright W} \mathcal{R}[\hat{x}] \rho' \) (\( \uparrow \) from above), it follows that \( (\triangleright W, R_1 \uparrow, R_2 \uparrow) \in \mathcal{R}[\hat{x}] \rho' \). From the latter, since \( r_{ra} : \text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon \in \hat{x} \), we have

\[
(\triangleright W, R_1(r_{ra}), R_2(r_{ra})) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Since \( R_1 = W.R_1 = \triangleright W.R_1 \), the above is equivalent to

\[
(\triangleright W, \triangleright W.R_1(r_{ra}), \triangleright W.R_2(r_{ra})) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Moreover, note that we have the following equalities

\[
\text{ret-addr}_i(\triangleright W, \hat{q}) = \text{ret-addr}_i(\triangleright W, r_{ra}) \text{ since } \hat{q} = r_{ra} = (\triangleright W).S_i(r_{ra}) \text{ by definition of ret-addr}_i = W'.R_i(r_0) \text{ since ret-addr}_i(\triangleright W, \hat{q}) = W'.R_i(r_0)
\]

Hence, we have

\[
(\triangleright W, W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Case \( \hat{q} = i_{ra} \): Hence, by the definition of ret-addr-type we have

\[
\text{ret-addr-type}(\hat{q}, \hat{x}, \hat{\sigma}) = \text{ret-addr-type}(i_{ra}, \hat{x}, \hat{\sigma}) = \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon \text{ where } \hat{\sigma}(i_{ra}) = \text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon
\]

Since \( (\triangleright W, S_1 \uparrow, S_2 \uparrow) \in \text{currentMR}(W(i_{stk})) \) (from above) and \( \text{currentMR}(W(i_{stk})) \subseteq_{\triangleright W} \mathcal{S}[\hat{\sigma}] \rho' \) (\( \uparrow \) from above), it follows that \( (\triangleright W, S_1 \uparrow, S_2 \uparrow) \in \mathcal{S}[\hat{\sigma}] \rho' \). From the latter, since \( i_{ra} : \text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon \in \hat{\sigma} \), we have

\[
(\triangleright W, S_1(i_{ra}), S_2(i_{ra})) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Since \( S_1 = W.S_1 = \triangleright W.S_1 \), the above is equivalent to

\[
(\triangleright W, \triangleright W.S_1(i_{ra}), \triangleright W.S_2(i_{ra})) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Moreover, note that we have the following equalities

\[
\text{ret-addr}_i(\triangleright W, \hat{q}) = \text{ret-addr}_i(\triangleright W, i_{ra}) \text{ since } \hat{q} = i_{ra} = (\triangleright W).S_i(i_{ra}) \text{ by definition of ret-addr}_i = W'.R_i(r_0) \text{ since ret-addr}_i(\triangleright W, \hat{q}) = W'.R_i(r_0)
\]

Hence, we have

\[
(\triangleright W, W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]

Since \( W' \supseteq \triangleright W \), by monotonicity we have

\[
(W', W'.R_1(r_0), W'.R_2(r_0)) \in \mathcal{W}[\text{box } \forall \vec{r}. \{ r : \tau ; \hat{\sigma}' \}^\epsilon] \rho'.
\]
From the above, we have that $W'.\Psi_1: \vdash W'.R_1(r_0) : \rho_1'(\text{box } \forall \cdot \{r: \tau; \hat{\sigma}'\}^r)$ and $W'.\Psi_2: \vdash W'.R_2(r_0) : \rho_2'(\text{box } \forall \cdot \{r: \tau; \hat{\sigma}'\}^r)$. These facts put together with $W' \in \text{World}$ and $\text{ret-reg}_1(W', r_0)$ allow us to conclude $r = r_1 = r_2$.

Next, let $M'_i = (H'_i, R'_i, S'_i)$.

From $(M'_1, M'_2) : W'$, since $W'.k > 0$, we have $(\triangleright W', M'_1, M'_2) \in \bigotimes \{\text{currentMR}(\theta) \mid \theta \in W'.\Theta\}$. From the latter, we have three facts, one each for islands $i_{\text{reg}}, i_{\text{stk}}$, and $i_{\text{box}}$.

First, we have that $(\triangleright W', R'_1 | R'_2) \in \text{currentMR}(W'(i_{\text{reg}}))$. From the latter it follows that $R'_1 = W'.R_1$ and, hence, we have $(\triangleright W', R'_1(r_0), R'_1(r_0)) \in \mathcal{W}[\text{box } \forall \cdot \{r: \tau; \hat{\sigma}'\}^r]$. From the latter, it must be that $R'_1(r_0) = \ell'_1[\omega'_1]$.

Second, we have that $(\triangleright W', S'_1 | S'_2) \in \text{currentMR}(W'(i_{\text{stk}}))$. Hence, we can instantiate assumption (5) from above, i.e., currentMR($W'(i_{\text{stk}})) \subseteq W' \mathcal{S}[\hat{\sigma}]' \rho'$, with $(\triangleright W', S'_1 | S'_2)$, noting $\triangleright W' \sqsubseteq W'$ (by Lemma 1.6), which allows us to conclude that $(\triangleright W', S'_1 | S'_2) \in \mathcal{S}[\hat{\sigma}]' \rho'$. From the premise $\hat{\sigma}' = \tau'_0 \cdots \tau'_k :: \zeta$, it follows that $S'_1 = w'_1 \cdots w'_k :: S'_{10}$, $S'_2 = w'_{20} :: \cdots :: w'_{21} :: S'_{20}$, $(\triangleright W', w'_{10}, w'_{20}) \in \mathcal{W}[\tau'_n] \rho'$ for $n \in \{0, \ldots, k\}$, and $(\triangleright W', S'_{10}, S'_{20}) \in \mathcal{S}[\zeta] \rho'$. From the latter, since $\mathcal{S}[\zeta] \rho' = \varphi_S$, it follows that $S'_{10} = S_{10}$ and $S'_{20} = S_{20}$.

Third, we have that there exist some $H'_{b1} \subseteq H'_1$ and $H'_{b2} \subseteq H'_2$ such that $(\triangleright W', H'_{b1} | H'_{b2}) \in \text{currentMR}(W'(i_{\text{box}}))$.

Instantiate $(\triangleright W', \ell'_1[\omega'_1], \ell'_2[\omega'_2]) = \mathcal{W}[\text{box } \forall \cdot \{r: \tau; \hat{\sigma}'\}^r] \rho'$, with $(\triangleright W', H'_{b1} | H'_{b2}) \in \text{currentMR}(W'(i_{\text{box}}))$, noting that $\triangleright W' \sqsubseteq W'$ by Lemma 1.6. Hence, we have that

- $H'_{b1} = \text{code}[\beta][\{\chi'_1; \sigma'_1\}^q].I^*_1$,
- $\rho_1'(r: \tau) = \chi'_1[\omega'_1/\beta]$,
- $\rho_1'(\hat{\sigma}) = \sigma'_1[\omega'_2/\beta]$,
- $\rho_1'(\epsilon) = \sigma'_1[\omega'_2/\beta]$ , and
- $(\triangleright W', \text{code}[]\{\chi'_1; \sigma'_1\}^q.I^*_1[\omega'_1/\beta]) \in \mathcal{H}[\forall \cdot \{r: \tau; \hat{\sigma}'\}^r] \rho'$

Hence, we know that $\bar{R}'(r_0) = \ell'_1[\omega'_1]$ and $H'_1 = \text{code}[\beta][\{\chi'_1; \sigma'_1\}^q].I^*_1$, and therefore

$$(H'_1, R'_1, S'_1) \mid E_i[(\text{ret } r_0 \{r_1\}, \cdot)] \rightarrow^1 (H'_1, R'_1, S'_1) \mid E_i[I'_1(\omega'_1/\beta), \cdot])$$

Therefore, it suffices to show that for $i \in \{1, 2\}$, either both configurations $(M'_i \mid E_i[\{I'_i[\omega'_i/\beta]\}, \cdot])$ terminate or both are running at $W'.k - 1$.

---

We proceed by noting that

$$(\triangleright W', \text{code}[]\{\chi'_1; \sigma'_1\}^q.I^*_1[\omega'_1/\beta], \text{code}[]\{\chi'_2; \sigma'_2\}^q.I^*_2[\omega'_2/\beta]) \in \mathcal{H}[\forall \cdot \{r: \tau; \hat{\sigma}'\}^r] \rho'$$

$$(\triangleright W', \text{code}[]\{\chi'_1[\omega'_1/\beta]; \sigma'_1[\omega'_1/\beta]\}^q.I^*_1[\omega'_1/\beta], \text{code}[]\{\chi'_2[\omega'_2/\beta]; \sigma'_2[\omega'_2/\beta]\}^q.I^*_2[\omega'_2/\beta]) \in \mathcal{H}[\forall \cdot \{r: \tau; \hat{\sigma}'\}^r] \rho'$$

We claim that $\tau^*; \sigma^* = \rho' \ast \text{ret-type}(\epsilon, \{r: \tau\}, \hat{\sigma}')$. It suffices to show for $i \in \{1, 2\}$:

$$\rho_i'(\tau^*); \rho_i'(\sigma^*) = \text{ret-type}(\rho_i'(\epsilon), \rho_i'((r: \tau)), \rho_i'(\hat{\sigma}'))$$

$$\equiv \rho_i'(\tau^*); \rho_i'(\sigma^*) = \text{ret-type}(\rho_i'(\text{end} \tau^*; \sigma^*), \rho_i'((r: \tau)), \rho_i'(\tau_0 :: \cdots :: \tau_j :: \zeta))$$

Note that $\text{FTV}(\tau^*; \sigma^*) \subseteq \Delta$. 80
• We claim that currentMR((\(\triangleright W'\))_{i_{\text{reg}}})) \in_{P:W} R[\tau: \tau']\rho'.
To show this, consider arbitrary \((W, M'_1, M'_2) \in \text{currentMR}((\triangleright W'\))_{i_{\text{reg}}}))\) such that \(\tilde{W} \triangleright W'\).
We must show that
\[ (\tilde{W}, M'_1, M'_2) \in R[\tau: \tau']\rho' \]

Note that since \(W'.k > 0\) (from above), \((\triangleright W').k = W'.k - 1\) and therefore
\[ \text{currentMR}((\triangleright W'))_{i_{\text{reg}}})) = |\text{currentMR}(W')_{i_{\text{reg}}}))|_{W'.k-1} \subseteq \text{currentMR}(W')_{i_{\text{reg}}})) \]
Thus,\((\tilde{W}, M'_1, M'_2) \in \text{currentMR}(W'(i_{\text{reg}})))\). Moreover, since we already have \((\triangleright W', \tilde{R}_1', \tilde{R}_2') \in \text{currentMR}(W'(i_{\text{reg}}))))\), by the definition of island \(i_{\text{reg}}\), it must be that \(M'_1 = \tilde{R}_1'\). Therefore, we are required to show
\[ (\tilde{W}, R_1'(r), R_2'(r)) \in \tilde{W}[\tau']\rho' \]

It suffices to show
\[ (\tilde{W}, R_1'(r), R_2'(r)) \in \tilde{W}[\tau']\rho' \]
which follows by monotonicity (Lemma 1.8) from our earlier assumption (4), namely
\((\triangleright W', W'.R_1'(r_1), W'.R_2'(r_2)) \in \tilde{W}[\tau']\rho'\) since \(r = r_1 = r_2\) (from above) and since \(W'.R_1 = R_1'\) and \(W'.R_2 = R_2'\).

• We claim that currentMR((\(\triangleright W'\))_{i_{\text{stk}}})) \in_{P:W} S[\sigma']\rho'.
To show this, consider arbitrary \((\tilde{W}, M'_1, M'_2) \in \text{currentMR}((\triangleright W'\))_{i_{\text{stk}}}))\) such that \(\tilde{W} \triangleright W'\).
We must show that
\[ (\tilde{W}, M'_1, M'_2) \in S[\sigma']\rho' \]
Note that since \(W'.k > 0\) (from above), \((\triangleright W').k = W.k - 1\) and therefore
\[ \text{currentMR}((\triangleright W'))_{i_{\text{stk}}})) = |\text{currentMR}(W')_{i_{\text{stk}}}))|_{W'.k-1} \subseteq \text{currentMR}(W')_{i_{\text{stk}}})) \]
Thus,\((\tilde{W}, M'_1, M'_2) \in \text{currentMR}(W'(i_{\text{stk}})))\). Moreover, since we already have \((\triangleright W', S'_1, S'_2) \in \text{currentMR}(W'(i_{\text{stk}}))))\), by the definition of island \(i_{\text{stk}}\), it must be that \(M'_1 = S'_1\). Therefore, we are required to show
\[ (\tilde{W}, S'_1, S'_2) \in \tilde{W}[\sigma']\rho' \]
Since \(\sigma' = \tau'_0 :: \cdots :: \tau'_k :: \zeta\), with \(S'_1 = w'_1 \cdots w'_{1k} :: S'_{10}\) and \(S'_2 = w'_2 \cdots w'_{21} :: S'_{20}\), the above follows from
\[ - (\tilde{W}, w'_{1n}, w'_{2n}) \in \tilde{W}[\tau_n']\rho' \text{ for } n \in \{0, \ldots, k\}, \text{ which follow by monotonicity (Lemma 1.8)} \]
from \((\triangleright W', w'_{1n}, w'_{2n}) \in \tilde{W}[\tau_n']\rho'\), and
\[ - (\tilde{W}, S'_{10}, S'_{20}) \in \tilde{W}[\zeta]\rho', \text{ which follows from } \tilde{W}[\zeta]\rho' = \varphi_S \text{ and the definition of } \varphi_S \text{ since } S'_{10} = S_{10} \text{ and } S'_{20} = S_{20}. \]
Hence, we can conclude that
\[ (\triangleright W', (I_1[\omega'/\beta'_1], \cdot), (I_2[\omega'/\beta'_2], \cdot)) \in E[\epsilon \vdash \tau_r; \sigma_r]\rho' \]
By Lemma 1.16, and noting that \(FTV(\tau^*; \sigma^*) \subseteq \Delta\) we have
\[ (\triangleright W', (I_1[\omega'/\beta'_1], \cdot), (I_2[\omega'/\beta'_2], \cdot)) \in E[\text{end}\{\tau^*; \sigma^*\} \vdash \tau^*; \sigma^*]\rho'. \]
Near the beginning of this proof, we had assumed \((W, E_1, E_2) \in \text{end}\{\tau^*; \sigma^*\} \vdash \tau^*; \sigma^*]\rho. We now apply Lemma 1.10 (monotonicity for evaluation contexts) to \((W, E_1, E_2) \in \text{end}\{\tau^*; \sigma^*\} \vdash \tau^*; \sigma^*]\rho, noting the following:

- \(\triangleright W' \supseteq \text{pub} W\), which follows by transitivity of \(\supseteq \text{pub}\) and the following \(\triangleright W' \supseteq \text{pub} W' \supseteq \text{pub} \triangleright W \supseteq \text{pub} W\) (which we have from above and from Lemma 1.6).
- \(\text{end}\{\tau^*; \sigma^*\} \supseteq \text{end}\{\tau^*; \sigma^*\}\).

81
Hence, we have that \((\Rightarrow W', E_1, E_2) \in \mathcal{K}^\#\{\tau^*; \sigma^*\} \vdash \tau^*; \sigma^*\rho\).

Instantiating 19 with the above, we have \((\Rightarrow W', E_1[(I'_1[\omega'_1/\beta'_1], \cdot)], E_2[(I'_2[\omega'_2/\beta'_2], \cdot)]) \in \mathcal{O}.

By instantiating the latter with \(M'_1\) and \(M'_2\), noting \((M'_1, M'_2) : \Rightarrow W'\), we have that \((M'_1 \mid E_1[(I'_1[\omega'_1/\beta'_1], \cdot)])\)
and \((M'_2 \mid E_2[(I'_2[\omega'_2/\beta'_2], \cdot)])\) either both terminate or both are running at \(W'.k - 1\), which gives us what we needed to show.

Hence, we conclude that
\[
(\Rightarrow W, E_1[(I_1[\omega_1/\beta_1][\rho_1(\sigma_0)/\zeta][q_1/\epsilon], \cdot)], E_2[(I_2[\omega_2/\beta_2][\rho_2(\sigma_0)/\zeta][q_2/\epsilon], \cdot)]) \in \mathcal{O}
\]

Now instantiate the above with \(M_1\) and \(M_2\), noting that we have \((M_1, M_2) : \Rightarrow W\) by Lemma 1.6 since \((M_1, M_2) : W\). Hence, we have that for \(i \in \{1, 2\}\), either both configurations \((M_i \mid E_1[(I_i[\omega_i/\beta_i][\rho_i(\sigma_0)/\zeta][q_i/\epsilon], \cdot)])\) terminate or both are running at \(W.k - 1\), which is exactly what we needed to show!
2 Functional language: F

2.1 Syntax and Semantics

\[ \tau ::= \alpha | \text{unit} | \text{int} | (\tau) \to \tau | \mu \alpha.\tau \mid \langle \tau \rangle \]
\[ e ::= t \]
\[ t ::= x | (\_ ) | n | t \cdot p \cdot t | \text{if0} t \cdot t \cdot t | \lambda (x:\tau).\tau | t \overline{\tau} | \text{fold}_{\mu \alpha.\tau} t | \text{unfold} t | \langle E \rangle | \pi_i(t) \]
\[ p ::= + | - | * \]
\[ v ::= () | n | \lambda (x:\tau).v | \text{fold}_{\mu \alpha.\tau} v | \langle v \rangle \]
\[ E ::= [\_] | E \cdot p \cdot t | v \cdot p \cdot E | \text{if0} E \cdot t \cdot t | E \overline{\tau} | v \cdot v \overline{\tau} | \text{fold}_{\mu \alpha.\tau} E | \text{unfold} E | \langle v, E, \overline{\tau} \rangle | \pi_i(E) \]
\[ \Delta ::= \_ | \Delta, \alpha \]
\[ \Gamma ::= \_ | \Gamma, x: \tau \]

2.1.1 Well-Formed Type \[ \Delta \vdash \tau \]

\[ \alpha \in \Delta \]
\[ \Delta \vdash \alpha \]
\[ \Delta \vdash \text{unit} \]
\[ \Delta \vdash \text{int} \]
\[ \Delta \vdash (\tau) \to \tau' \]
\[ \Delta, \alpha \vdash \tau \]
\[ \Delta \vdash \tau_1 \cdot \tau_n \]

2.1.2 Well-Formed Type Environment \[ \vdash \Gamma \]

\[ \vdash \_ \]
\[ \vdash \_ \vdash \tau \]
\[ \vdash \Gamma, x: \tau \]

2.1.3 Well-Typed Component \[ \Gamma \vdash e: \tau \]

\[ x: \tau \in \Gamma \] \[ \Gamma \vdash x: \tau \]
\[ \Gamma \vdash (): \text{unit} \]
\[ \Gamma \vdash n: \text{int} \]
\[ \Gamma \vdash t_1: \text{int} \]
\[ \Gamma \vdash t_2: \tau \]
\[ \Gamma \vdash t_3: \tau \]
\[ \Gamma, x: \tau \vdash t: \tau' \]
\[ \Gamma \vdash \lambda (x:\tau).t: (\tau) \to \tau' \]
\[ \Gamma \vdash t_0: (\tau_1) \to \tau_2 \]
\[ \Gamma \vdash \overline{\tau_1} \]
\[ \Gamma \vdash \text{fold}_{\mu \alpha.\tau} t: \mu \alpha.\tau \]
\[ \Gamma \vdash \text{unfold} t: (\tau_1, \ldots, \tau_n) \]
\[ \Gamma \vdash t: (\tau_1, \ldots, \tau_n) \]
\[ \Gamma \vdash \pi_i(t): \tau_i \]

2.1.4 Reduction Relation \[ e \rightarrow e' \]

\[ E[n_1 \cdot p \cdot n_2] \rightarrow E[\text{prim}(p, n_1, n_2)] \]
\[ E[\text{if0} \ n_0 \ t_1 \ t_2] \rightarrow E[t_1] \]
\[ E[\text{if0} n \ t_1 \ t_2] \rightarrow E[t_2] \]
\[ E[\lambda (x:\tau).t \overline{\tau}] \rightarrow E[t]\overline{x}] \]
\[ E[\text{unfold} (\text{fold}_{\mu \alpha.\tau} v)] \rightarrow E[v] \]
\[ E[\pi_i(\langle \nu_1, \ldots, \nu_n \rangle)] \rightarrow E[\nu_i] \]
3 Multi-Language: F+T

3.1 Syntax and Semantics

\[ \begin{align*}
\tau & ::= \cdots \mid (\tau) \phi, \phi' \tau' \\
\phi & ::= \cdots \\
\phi & ::= \cdots
\end{align*} \]

\[ \begin{align*}
t & ::= \cdots \mid \tau F T e \mid \lambda \phi (x:\tau).t \mid t \tilde{t}
\end{align*} \]

\[ \begin{align*}
\nu & ::= \cdots \\
E & ::= \cdots \mid \tau F T E
\end{align*} \]

\[ \begin{align*}
q & ::= \cdots \\
\iota & ::= \cdots \mid \text{protect } \phi, \zeta \mid \text{import } r_d, \sigma \mid \tau F T e
\end{align*} \]

\[ \begin{align*}
E_I & ::= \cdots \mid \text{import } r_d, \sigma \mid \tau F T E; I
\end{align*} \]

\[ \begin{align*}
\phi & ::= \cdots \mid \tau :: \phi
\end{align*} \]

\[ \begin{align*}
\sigma & ::= \zeta \mid \bullet \mid \phi :: \sigma
\end{align*} \]

3.1.1 Boundary Type Translation

\[ \begin{align*}
\alpha^T &= \alpha \\
\text{unit}^T &= \text{unit} \\
\text{int}^T &= \text{int} \\
\tau_1, \ldots, \tau_n \to \tau'^T &= \text{box} \forall [\zeta, \epsilon].\{r_a: \forall[], \{r_1: \tau'\,t; \zeta\}^\epsilon; \sigma'^r\}
\end{align*} \]

\[ \begin{align*}
\tau_1, \ldots, \tau_n \phi_1 : \phi_0 \to \tau'^T &= \text{box} \forall [\zeta, \epsilon].\{r_a: \forall[], \{r_1: \tau'\,t; \phi_0 :: \zeta\}^\epsilon; \sigma'^r\}
\end{align*} \]

NOTE: FT inherits any judgments of F or T not mentioned here explicitly.

3.1.2 Well-Typed Heap Value \[ \Psi \vdash h : \nu \psi \]

\[ \begin{align*}
\Psi & \vdash \forall [\Delta].\{\chi; \sigma\}^q \\
\Psi & \vdash \text{code}[\Delta][\{\chi; \sigma\}^q, I] \text{ box } \forall [\Delta].\{\chi; \sigma\}^q \\
\Psi & \vdash (w_0, \ldots, w_n) : \nu (\tau_0, \ldots, \tau_n)
\end{align*} \]

3.1.3 Well-Typed Instruction Sequence \[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I \]

To the corresponding rules of T, amend the side-condition to forbid \( q = \text{out} \) as well as \( q = \epsilon \) and add the environment \( \Gamma \). We also add the following rule, which allows Tcode to protect the tail of a stack and get a name for the abstract tail.

\[ \begin{align*}
\sigma &= \phi :: \sigma_0 \\
\sigma' &= \phi :: \zeta \\
\Psi & \vdash \text{protect } \phi, \zeta; I
\end{align*} \]

3.1.4 Well-Typed Instruction \[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \iota \Rightarrow \Delta'; \chi'; \sigma'; q' \]

To the corresponding rules of T, amend the side-condition to forbid \( q = \text{out} \) as well as \( q = \epsilon \), add the environment \( \Gamma \), and add the following rules:
3.1.5 Well-Typed Component \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e : \tau; \sigma' \)

To the corresponding rules of T, add the environment \( \Gamma \) and the following rules:

\[
\begin{align*}
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t_1 : int; \sigma_1 & \Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t_2 : int; \sigma_2 \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t_1 p t_2 : int; \sigma_2 \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash t_1 : int; \sigma_1 & \Psi; \Delta; \Gamma; \chi; \sigma_1 &; out \vdash t_2 : \tau; \sigma_2 \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t : \tau' ; \zeta \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash \lambda (x : \tau). t : (\tau) \rightarrow \tau' ; \sigma \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash \phi_i : \phi_o \rightarrow \tau' ; \sigma_0 \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash t : (\tau_1 \cdots \tau_n) ; \phi_i : \phi_o \rightarrow \tau' ; \sigma_0 \\
\Psi; \Delta; \Gamma; \chi; \sigma_1 &; out \vdash t_1 \cdots t_n : \tau' ; \sigma_n \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t : \tau [\mu \alpha . \tau / \alpha] ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash fold_{\mu \alpha . \tau} t : \mu \alpha . \tau ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash unfold t : \tau [\mu \alpha . \tau / \alpha] ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t_1 : \tau_1 ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash (t_1, \ldots, t_n) ; \tau_1, \ldots, \tau_n ; \sigma_n \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash t : (\tau_1, \ldots, \tau_n) ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma_0 &; out \vdash \pi_i (t) : \tau_i ; \sigma_1 \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash t : \tau T ; \sigma' \\
\Psi; \Delta; \Gamma; \chi; \sigma &; out \vdash \pi F t : \tau ; \sigma' \\
\end{align*}
\]
3.1.6 Value Translation

\[ \text{TF}^\text{unit}((), M) = ((), M) \]
\[ \text{TF}^\text{int}(n, M) = (n, M) \]
\[ \text{TF}^{\mu\alpha\tau}(\text{fold}_{\mu\alpha\tau} \mathbf{v}, M) = (\text{fold}_{\mu\alpha\tau} \mathbf{v}, M') \]
\[ \text{TF}^\tau(\ell) \rightarrow \tau' (\lambda(x:\tau).t, M) = (\ell, (M, \ell \mapsto h)) \]

where \( h = \text{code}[\zeta, \epsilon]\{\text{ra:} \forall[].\{r_1: \tau'\}; \zeta; \tau' :: \zeta \}^{\epsilon} \).

\[ \text{e} = (\lambda(x:\tau).t) \tau' (\text{fold}_{\mu\alpha\tau} \mathbf{v}, M') \]
\[ \text{TF}^\tau(\ell) \xrightarrow{\phi_i \phi_o} \tau' (\lambda(x:\tau).t, M) = (\ell, (M, \ell \mapsto h)) \]

where \( h = \text{code}[\zeta, \epsilon]\{\text{ra:} \forall[].\{r_1: \tau'\}; \phi_o :: \zeta; \tau' :: \phi_i :: \zeta \}^{\epsilon} \).

\[ \text{M} = \text{M}0, \text{and } \text{TF}^{\tau_0}(\mathbf{v}_0, \mathbf{M}_i) = (\mathbf{w}_i, \mathbf{M}_{i+1}) \]

\[ \text{unit}^\text{FT}((), M) = ((), M) \]
\[ \text{int}^\text{FT}(n, M) = (n, M) \]
\[ \mu\alpha\tau^\text{FT} (\text{fold}_{\mu\alpha\tau} w) = (\text{fold}_{\mu\alpha\tau} \mathbf{v}, M') \]
\[ (\tau) \rightarrow \tau' \text{FT}(w, M) = (\mathbf{v}, (M, \ell_{\text{end}} \mapsto h_{\text{end}})) \]

where \( \mathbf{v} = \lambda(x_{n} : \tau_{n}).\tau'\text{FT} \)

\[ h_{\text{end}} = \text{code}[\zeta]\{r_1: \tau' \}; \zeta \} \text{end}\{\tau' \}; \zeta \} \text{ret}\{\tau' \}; \zeta \} \{r_1 \}

\[ \text{hend} = \text{code}[\zeta]\{r_1: \tau' \}; \zeta \} \text{end}\{\tau' \}; \zeta \} \text{ret}\{\tau' \}; \zeta \} \{r_1 \}

\[ \text{hend} = \text{code}[\zeta]\{r_1: \tau' \}; \zeta \} \text{end}\{\tau' \}; \zeta \} \text{ret}\{\tau' \}; \zeta \} \{r_1 \}

86
3.1.7 Reduction Relation \( \langle M | e \rangle \rightarrow \langle M' | e' \rangle \)

Lift the individual language reduction relations to obtain a multilanguage reduction relation, noting that the contexts are untyped, as for \( T \), we may reach intermediate states with different types and return markers \( q \). This means that a proof of type safety would not be possible using progress and preservation, and we would instead have to use a unary logical relation in the style of [1]. We have not done this, but do not anticipate any problems, since the form of this logical relation would be a special case of the binary logical relation for equivalence that we have used.

\[
\begin{align*}
e & \rightarrow e' \\
\langle M | E[e] \rangle & \rightarrow \langle M | E[e'] \rangle
\end{align*}
\]

Add the reduction rules for boundaries:

\[
\begin{align*}
\langle M | E[\tau FT (\text{ret end} \{ r \} ; \sigma \{ r \} ; \cdot) \} ] \rangle & \rightarrow \langle M' | E[\cdot] \rangle & \text{if } \tau FT(M.R(r), M) = (v, M') \\
\langle M | E[\text{import } r, \sigma \{ \tau \} FT \; v \; I \{ \cdot \} ] \rangle & \rightarrow \langle M' | E[\tau v \; r, w ; I \{ \cdot \} ] \rangle & \text{if } \tau FT(v, M) = (w, M')
\end{align*}
\]

Finally, add beta reduction for the new lambda form, which is identical to normal beta reduction:

\[
\langle M | E[\lambda \phi_i \; (x : \tau) . t] \rangle \rightarrow \langle M | E[t'/x] \rangle
\]

3.1.8 Reduction Relation \( \langle M | I \rangle \rightarrow \langle M' | I' \rangle \)

Add instruction reduction for stack protection, which has no operational consequence:

\[
\langle (H, R, S) | \text{protect } \phi, \zeta ; I \rangle \rightarrow \langle (H, R, S) | I \rangle
\]
3.2 General Contexts and Contextual Equivalence

\[ C ::= \begin{align*}
& \mathbf{[\cdot]} | \mathbf{C} \ p \ t | t \ p \ C | \mathbf{if0} \ C \ t t \ \\
& | \mathbf{if0} \ C \ t t | \mathbf{if0} \ t t C | \lambda(\bar{x}:\tau).C \ \\
& | \lambda^{\phi_i}(\bar{x}:\tau).C | C \overline{t} | t \overline{t} C \overline{t} \ \\
& | \mathbf{fold}_{\mu_\alpha,\tau} C | \mathbf{unfold} C | (\overline{t}, C, \overline{t}) \ \\
& | \pi_1(C) | \tau^{\mathcal{F}T} C
\end{align*} \]

\[ C ::= C | C \]

3.2.1 Plug Function \( C[e] \)

\[
\begin{align*}
\mathbf{[\cdot]}[e] &= e \\
(C \ p \ t)[e] &= (C[e]) \ p \ t \\
(t \ p \ C)[e] &= t \ p \ (C[e]) \\
(\mathbf{if0} \ C \ t_1 \ t_2)[e] &= \mathbf{if0} \ (C[e]) \ t_1 \ t_2 \\
(\mathbf{if0} \ t_0 \ C \ t_2)[e] &= \mathbf{if0} \ t_0 \ (C[e]) \ t_2 \\
(\mathbf{if0} \ t_0 \ C \ t_1)[e] &= \mathbf{if0} \ t_0 \ t_1 \ (C[e]) \\
(\lambda(\bar{x}:\tau).C)[e] &= \lambda^{\phi_i}(\bar{x}:\tau).(C[e]) \\
(\mathbf{fold}_{\mu_\alpha,\tau} C)[e] &= \mathbf{fold}_{\mu_\alpha,\tau} \ (C[e]) \\
(\mathbf{unfold} C)[e] &= \mathbf{unfold} \ (C[e]) \\
((\overline{t}, C, \overline{t}))[e] &= (\overline{t}, (C[e]), \overline{t}) \\
(\pi_1(C))[e] &= \pi_1(C[e]) \\
(\tau^{\mathcal{F}T} C)[e] &= \tau^{\mathcal{F}T} \ (C[e])
\end{align*}
\]

\[
(C_1, H)[e] = \begin{cases} 
(C_1[\mathbf{I}], (H, H')) & e = (I, H') \land C_1 \text{ contains no language boundaries} \\
(C_1[e], H) & \text{otherwise}
\end{cases}
\]

\[
(I, C_H)[e] = \begin{cases} 
(I, (C_H[\mathbf{I}], H')) & e = (I, H') \land C_H \text{ contains no language boundaries} \\
(I, C_H[e]) & \text{otherwise}
\end{cases}
\]

\[
[\cdot][\mathbf{I}] = I \\
(\mathbf{import} \ r_d, ^\alpha \mathcal{F} \mathcal{T}; I)[e] = \mathbf{import} \ r_d, ^\alpha \mathcal{F} \mathcal{T} \ (C[e]); I
\]

\[
(\ell; C_1)[e] = \ell; C_1[e] \\
(C_H, \ell \mapsto h)[e] = (C_H[e], \ell \mapsto h) \\
(H, \ell \mapsto \mathbf{code}[\Delta] \{\chi; \sigma\}^q.C_1)[e] = H, \ell \mapsto \mathbf{code}[\Delta] \{\chi; \sigma\}^q.(C_1[e])
\]

88
3.2.2 Well-Typed Context

\[ C (: \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q' \vdash \tau' ; \rho_1) \]

\[ \Psi \subseteq \Psi' \quad \Delta \subseteq \Delta' \quad \Gamma \subseteq \Gamma' \quad \Delta \vdash \chi' \leq \chi \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma' \vdash \int ; \sigma_1) \quad \Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_1 ; \vdash \tau' \vdash \int ; \sigma_2 \]

\[ \Psi ; \Delta' ; \Gamma' ; \vdash \sigma_0 ; \vdash \tau \vdash \int ; \sigma_1 \quad \vdash C : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]

\[ \vdash C : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma' \vdash \int ; \sigma_1) \quad \Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_1 ; \vdash \tau' \vdash \int ; \sigma_2 \]

\[ \Psi' ; \Delta', \Gamma', \chi', \sigma_0 ; \vdash \tau' \vdash \int ; \sigma_1 \quad \vdash C : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; \vdash \tau ; \sigma_2 \vdash \int ; \sigma_1) \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]

\[ \vdash \left[ \vdash \right] : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \Rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash \tau ; \sigma') \]
⊢ C: (Ψ; Δ; Γ; χ; σ; q ⊢ τ; σ') ⊨ (Ψ'; Δ'; Γ'; χ'; σ₀: out ⊢ τ[μα.τ/α]; σ₁)

⊢ foldμα.τ C: (Ψ; Δ; Γ; χ; σ; q ⊢ τ; σ') ⊨ (Ψ'; Δ'; Γ'; χ'; σ₀: out ⊢ μα.τ; σ₁)

⊢ C: (Ψ; Δ; Γ; χ; σ; q ⊢ τ; σ') ⊨ (Ψ'; Δ'; Γ'; χ'; σ₀: out ⊢ μα.τ; σ₁)

⊢ unfold C: (Ψ; Δ; Γ; χ; σ; q ⊢ τ; σ') ⊨ (Ψ'; Δ'; Γ'; χ'; σ₀: out ⊢ τ[μα.τ/α]; σ₁)

Ψ'; Δ'; Γ'; χ'; σ₀: out ⊨ t₁: τ₁; τ₁ ⊢ t₁: τ₁; ...; τ₁ ⊢ t₁: τ₁; ...; τ₁ ⊢ t₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁: τ₁; ...; τ₁ ⊨ τ₁:
3.2.3 Contextual Equivalence

\[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{ctx} e_2 : \tau; \hat{\sigma} \triangleq \]
\[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 : \tau; \hat{\sigma} \land \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_2 : \tau; \hat{\sigma} \land \]
\[ \forall C, M, \Psi'; \chi', \sigma', q', \tau', \hat{\sigma}' . \]
\[ \vdash C : (\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \tau; \hat{\sigma}) \rightarrow (\Psi'; \chi'; \sigma'; q' \vdash \tau'; \hat{\sigma}') \land \vdash M : (\Psi', \chi', \sigma') \]
\[ \implies (\langle M \mid C[e_1] \rangle \downarrow \iff \langle M \mid C[e_2] \rangle \downarrow) \]

3.2.4 CIU Equivalence

\[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{ciu} e_2 : \tau; \hat{\sigma} \triangleq \]
\[ \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 : \tau; \hat{\sigma} \land \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_2 : \tau; \hat{\sigma} \land \]
\[ \forall \delta, \gamma, E, M, \Psi', q', \tau', \hat{\sigma}'. \]
\[ \vdash \delta : \Delta \land \Psi'; \chi'; \sigma'; q' \vdash \tau'; \hat{\sigma}' \land \]
\[ \vdash E : (\Psi'; \chi'; \sigma; q \vdash \tau; \hat{\sigma}) \rightarrow (\Psi'; \chi'; \sigma; q' \vdash \tau'; \hat{\sigma}') \land \vdash M : (\Psi', \chi, \sigma) \]
\[ \implies (\langle M \mid E[\delta(\gamma)] \rangle \downarrow \iff \langle M \mid E[\delta(\gamma)] \rangle \downarrow) \]
3.3 Logical Relation

NOTE: We used curr-S(W) ∈ X as shorthand for currentMR(W(i_{stk})) ∈ W X and curr-R(W) ∈ X as shorthand for currentMR(W(i_{reg})) ∈ W X in the accompanying paper. This shorthand does not appear in this technical report.

Worlds and Auxiliary Definitions Worlds consist of a sequence of islands that describe the current state of the memories (and how they are related) of the two computations we wish to relate. The essential idea here is that the islands θ in the sequence Θ will specify constraints on disjoint parts of memory. We obtain constraints on the entire memory via a disjoint union of the memories specified by the islands.

Therefore, we begin with some simple definitions for memory objects that we will make use of in islands. We need to be able to lift various pieces of memory to a full program memory M = (H, R, S). In many cases, we may not want to impose a constraint on the register file and stack, so we allow ⊥ to appear in those positions. Since disjoint heap fragments can be merged, the heap can be left unconstrained just by using an empty heap.

\[
\{\cdot\} \overset{\text{def}}{=} (\{\cdot\}, \bot, \bot)
\]

\[
\text{Regs}_\bot = \{R\} \cup \{\bot\}
\]

\[
H \overset{\text{def}}{=} (H, \bot, \bot)
\]

\[
\text{Stack}_\bot = \{S\} \cup \{\bot\}
\]

\[
R \overset{\text{def}}{=} (\{\cdot\}, R, \bot)
\]

\[
S \overset{\text{def}}{=} (\{\cdot\}, \bot, S)
\]

A world W consists of a step index k, a pair of heap types Ψ_1 and Ψ_2, and a sequence Θ of islands θ. Each island expresses invariants on certain parts of memory by encoding a state transition system and a memory relation MR that establishes which pairs of memories are acceptable in each state. (See Dreyer et al. [2] for details.)

The first three islands in Θ are distinguished: they track the register file, the stack, and the immutable contents of the heap, respectively. We assign these islands the indices i_{reg}, i_{stk}, and i_{box}, respectively. Further islands can be added to a world to encode invariants about mutable data.

\[
\text{World}_n \overset{\text{def}}{=} \{ W = (k, \Psi_1, \Psi_2, \Theta) \mid k < n \land \exists m \geq 3. \Theta \in \text{Island}_m^n \land \]

\[
(\exists s_{\text{reg}}. \Theta(i_{\text{reg}}) = \text{island}_{\text{reg}}(s_{\text{reg}}, k) \land \Psi_1 \vdash s_{\text{reg}} : \varnothing_{\text{reg}} : \chi_1 \land \Psi_2 \vdash s_{\text{reg}} : \varnothing_{\text{reg}} : \chi_2) \land \]

\[
(\exists s_{\text{stk}}. \Theta(i_{\text{stk}}) = \text{island}_{\text{stk}}(s_{\text{stk}}, k) \land \Psi_1 \vdash s_{\text{stk}} : \varnothing_{\text{stk}} : \chi_1 \land \Psi_2 \vdash s_{\text{stk}} : \varnothing_{\text{stk}} : \chi_2) \land \]

\[
(\exists s_{\text{box}}. \Theta(i_{\text{box}}) = \text{island}_{\text{box}}(s_{\text{box}}, k) \land \Psi_1^\text{ref} \vdash s_{\text{box}} : H_1 : \Psi_1^\text{box} \land \Psi_2^\text{ref} \vdash s_{\text{box}} : H_2 : \Psi_2^\text{box})\}
\]

\[
\text{Island}_n \overset{\text{def}}{=} \{ \theta = (s, S, \delta, \pi, \text{MR}, \text{bij}) \mid s \in S \land S \in \text{Set} \land \delta \subseteq S \times S \land \pi \subseteq \delta \land \]

\[
\delta, \pi \text{ reflexive} \land \delta, \pi \text{ transitive} \land \text{MR} \in S \rightarrow \text{MemRel}_n \land \text{bij} \in S \rightarrow \mathcal{P}(\text{Val} \times \text{Val}) \}
\]

\[
\text{MemAtom}_n \overset{\text{def}}{=} \{ (W, M_1, M_2) \mid W \in \text{World}_n \land M_1, M_2 \in \text{Heap} \times \text{Regs}_\bot \times \text{Stack}_\bot \}
\]

\[
\text{MemRel}_n \overset{\text{def}}{=} \{ \varphi_M \subseteq \text{MemAtom}_n \mid \forall (W, M_1, M_2) \in \varphi_M. \forall W' \supseteq W. (W', M_1, M_2) \in \varphi_M \}
\]

The transition systems for θ_{reg} and θ_{stk} encode the current contents of each side’s register file and stack, respectively. They may transition freely between states, since the register file and stack are fairly free in how they can change during program execution. The states of θ_{box} encode the contents of the immutable part of
the heap on each side. This island is allowed to transition only by adding more immutable data to the heap.

\[ i_{\text{reg}} = 1 \]
\[ S_{\text{reg}} = \{ (R_1, \chi_1, R_2, \chi_2) \} \]
\[ \text{island}_{\text{reg}}(s, k) = (s, S_{\text{reg}}, S_{\text{reg}} \times S_{\text{reg}}, S_{\text{reg}} \times S_{\text{reg}}, \lambda s. \{ (W, s.R_1], s.R_2] ) \mid W \in \text{World}_k, \lambda s. \emptyset) \]

\[ i_{\text{stk}} = 2 \]
\[ S_{\text{stk}} = \{ (S_1, \sigma_1, S_2, \sigma_2) \} \]
\[ \text{island}_{\text{stk}}(s, k) = (s, S_{\text{stk}}, S_{\text{stk}} \times S_{\text{stk}}, S_{\text{stk}} \times S_{\text{stk}}, \lambda s. \{ (W, s.S_1], s.S_2] ) \mid W \in \text{World}_k, \lambda s. \emptyset) \]

\[ i_{\text{box}} = 3 \]
\[ S_{\text{box}} = \{ (H_1, H_2) \} \]
\[ \delta_{\text{box}} = \{ ((H_1, H_2), (H_1', H_2')) \mid H_1 \subseteq H_1' \land H_2 \subseteq H_2' \} \]
\[ \text{island}_{\text{box}}(s, k) = (s, S_{\text{box}}, \delta_{\text{box}}, \delta_{\text{box}}, \lambda s. \{ (W, (s.H_1), (s.H_2)) ) \mid W \in \text{World}_k, \lambda s. \emptyset) \]

Two memory objects that describe disjoint parts of memory can be merged into one compound memory object via the \( \otimes \) operator.

\[ M_1 \otimes M_2 \text{ where } M_1, M_2 \in (\{H\} \times \text{Regs} \times \text{Stack}) \]

\[ (H_1, R_1, S_1) \otimes (H_2, R_2, S_2) = \begin{cases} (H_1 \uplus H_2, R, S) & \text{where } R = R_1 \text{ if } R_2 = \bot; R = R_2 \text{ if } R_1 = \bot \\ S = S_1 & \text{if } S_2 = \bot; S = S_2 & \text{if } S_1 = \bot \\ \text{undefined} & \text{otherwise} \end{cases} \]

\[ \varphi_M \otimes \varphi'_M = \{ (W, M_1 \otimes M'_1, M_2 \otimes M'_2) \mid (W, M_1, M_2) \in \varphi_M \land (W, M'_1, M'_2) \in \varphi'_M \} \]

These are standard operations for dealing with step indexing: we can approximate a world or relation to a given number of steps with \( [\cdot]_k \), and we can expend a step using the \( \triangleright \) operator (read “later”).

\[ [\theta_1, \ldots, \theta_m]_k \overset{\text{def}}{=} ([\theta_1]_k, \ldots, [\theta_m]_k) \]
\[ ([s, S, \delta, \pi, MR, bij])_k \overset{\text{def}}{=} (s, S, \delta, \pi, [MR]_k, bij) \]
\[ [MR]_k \overset{\text{def}}{=} \lambda s. [MR(s)]_k \]
\[ [\varphi_M]_k \overset{\text{def}}{=} \{ (W, M_1, M_2) \in \varphi_M \mid W.k < k \} \]

\[ \triangleright (k + 1, \Psi_1, \Psi_2, \Theta) \overset{\text{def}}{=} (k, \Psi_1, \Psi_2, [\Theta]_k) \]
\[ \triangleright \varphi_e \overset{\text{def}}{=} \{ (W, e_1, e_2) \mid W.k > 0 \implies (\triangleright W, e_1, e_2) \in \varphi_e \} \]
\[ \triangleright \varphi_v \overset{\text{def}}{=} \{ (W, v_1, v_2) \mid W.k > 0 \implies (\triangleright W, v_1, v_2) \in \varphi_v \} \]
\[ \triangleright \varphi_w \overset{\text{def}}{=} \{ (W, w_1, w_2) \mid W.k > 0 \implies (\triangleright W, w_1, w_2) \in \varphi_w \} \]

Future worlds \( W' \) of a given world \( W \), written \( W' \sqsupseteq W \), may differ from \( W \) in any or all of the following ways: they may have expended steps, allocated additional memory, added new islands, or taken transitions in existing islands. Public future worlds \( W' \sqsupseteq_{\text{pub}} W \) are similar, but must have taken public transitions
from the island states in $W$.

\[(k', \Psi'_1, \Psi'_2, \Theta') \equiv (k, \Psi_1, \Psi_2, \Theta) \]
\[\quad \text{def} \quad k' \leq k \land \Psi'_1 \supseteq \Psi_1 \land \Psi'_2 \supseteq \Psi_2 \land \Theta' \supseteq \Theta \]
\[\quad \land (k, \Psi_1, \Psi_2, \Theta) \in \text{World} \land (k', \Psi'_1, \Psi'_2, \Theta') \in \text{World}\]

\[(\theta'_1, \ldots, \theta'_m) \equiv (\theta_1, \ldots, \theta_m) \]
\[\quad \text{def} \quad m' \geq m \land \forall j \in \{1, \ldots, m\}. \theta'_j \equiv \theta_j\]

\[(s', S', \delta', \pi', MR', \text{bij}') \equiv (s, S, \delta, \pi, MR, \text{bij})\]
\[\quad \text{def} \quad (S', \delta', \pi', MR', \text{bij}') = (S, \delta, \pi, MR, \text{bij}) \land (s, s') \in \delta\]

\[W' \supseteq W\]
\[\quad \text{def} \quad W'.k < W.k \land W' \equiv W\]

\[(k', \Psi'_1, \Psi'_2, \Theta') \equiv_{\text{pub}} (k, \Psi_1, \Psi_2, \Theta) \]
\[\quad \text{def} \quad k' \leq k \land \Psi'_1 \supseteq \Psi_1 \land \Psi'_2 \supseteq \Psi_2 \land \Theta' \equiv_{\text{pub}} \Theta \]
\[\quad \land (k, \Psi_1, \Psi_2, \Theta) \in \text{World} \land (k', \Psi'_1, \Psi'_2, \Theta') \in \text{World}\]

\[(\theta'_1, \ldots, \theta'_m) \equiv_{\text{pub}} (\theta_1, \ldots, \theta_m) \]
\[\quad \text{def} \quad m' \geq m \land \forall j \in \{1, \ldots, m\}. \theta'_j \equiv_{\text{pub}} \theta_j\]

\[(s', S', \delta', \pi', MR', \text{bij}') \equiv_{\text{pub}} (s, S, \delta, \pi, MR, \text{bij}) \]
\[\quad \text{def} \quad (S', \delta', \pi', MR', \text{bij}') = (S, \delta, \pi, MR, \text{bij}) \land (s, s') \in \pi\]

Given a world $W$, we often need to talk about future worlds of $W$ where the only change is that new immutable memory has been allocated. We use this notation to capture this:

\[W \equiv (H_1, H_2) \equiv (W.k, W.\Psi_1 \equiv W.\Psi_2 \equiv W.\Theta \cdot \text{i_{box}} \rightarrow \text{island}_{\text{i_{box}}}(W.(\text{i_{box}}).s \equiv (H_1, H_2), W.k))\]

if $W.\Psi_1 \vdash H_1 \cdot \Psi_1 \land W.\Psi_2 \vdash H_2 \cdot \Psi_2 \land \text{boxheap}(\Psi_1) \land \text{boxheap}(\Psi_2)$.

The following are convenient shorthands for frequently-used pieces of a world:

\[\text{currentMR}(\theta) \equiv \theta.\text{MR}(\theta.s)\]
\[W(i) \equiv W.\Theta(i)\]
\[W.R_1 \equiv W.\Theta(i_{\text{reg}}).s.R_1\]
\[W.S_1 \equiv W.\Theta(i_{\text{stk}}).s.S_1\]
\[W.\chi_1 \equiv W.\Theta(i_{\text{reg}}).s.\chi_1\]
\[W.\sigma_1 \equiv W.\Theta(i_{\text{stk}}).s.\sigma_1\]
\[W.R_2 \equiv W.\Theta(i_{\text{reg}}).s.R_2\]
\[W.S_2 \equiv W.\Theta(i_{\text{stk}}).s.S_2\]
\[W.\chi_2 \equiv W.\Theta(i_{\text{reg}}).s.\chi_2\]
\[W.\sigma_2 \equiv W.\Theta(i_{\text{stk}}).s.\sigma_2\]
\[W.\psi_1 \equiv (W.\Psi_1, W.\chi_1, W.\sigma_1)\]
\[W.\psi_2 \equiv (W.\Psi_2, W.\chi_2, W.\sigma_2)\]

Atoms are well-formed worlds together with a pair of components or values that are well-typed at the indicated type under the appropriate memory type of the world.

\[\text{TermAtom}_n([(q_1 \vdash \tau_1; \sigma_1), (q_2 \vdash \tau_2; \sigma_2)] \equiv \{ (W, e_1, e_2) \mid W \in \text{World}_n \land W.\Psi_1; \cdots; W.\chi_1; W.\sigma_1; q_1 \vdash e_1; \tau_1; \sigma_1 \land W.\Psi_2; \cdots; W.\chi_2; W.\sigma_2; q_2 \vdash e_2; \tau_2; \sigma_2 \}\}\]

\[\text{ValAtom}_n(\tau_1, \tau_2) \equiv \{ (W, v_1, v_2) \in \text{TermAtom}_n(\text{out} \vdash \tau_1; W.\sigma_1), (\text{out} \vdash \tau_2; W.\sigma_2) \}\}\]

\[\text{StackAtom}_n(\sigma_1, \sigma_2) \equiv \{ (W, S_1, S_2) \mid W \in \text{World}_n \land W.\Psi_1; \cdots; W.\chi_1; W.\sigma_1 \land W.\Psi_2; \cdots; W.\chi_2; W.\sigma_2 \}\}\]

\[\text{HValAtom}_n(\psi_1, \psi_2) \equiv \{ (W, h_1, h_2) \mid W \in \text{World}_n \land W.\Psi_1; h_1; \psi_1 \land W.\Psi_2; h_2; \psi_2 \}\}\]

\[\text{ContAtom}(\psi_1 \equiv \chi_1; \chi_1; \psi_2) \equiv \{ (W, E_1, E_2) \mid W \in \text{World} \land W.\chi_1; W.\sigma_1; q_1 \vdash \tau_1; \sigma_1 \land W.\chi_2; W.\sigma_2; q_2 \vdash \tau_2; \sigma_2 \}\]\n
\[\text{ContAtom}(\psi_1 \equiv \chi_1; \chi_1; \psi_2) \equiv \{ (W, E_1, E_2) \mid W.\Psi_1; \cdots; W.\chi_1; W.\sigma_1; q_1 \vdash \tau_1; \sigma_1' \land W.\Psi_2; \cdots; W.\chi_2; W.\sigma_2; q_2 \vdash \tau_2; \sigma_2' \}\}\]

\[\text{ContAtom}(\tau_1; \tau_1; \tau_2; \tau_2) \equiv \{ (W, E_1, E_2) \mid W.\Psi_1; \cdots; W.\chi_1; W.\sigma_1; q_1 \vdash \tau_1; \tau_1; \tau_1; \sigma_1; q_2 \vdash \tau_2; \tau_2; \sigma_2 \}\}\]
WvalRel[τ₁, τ₂] \stackrel{\text{def}}{=} \{ \varphi_w \subseteq WvalAtom[τ₁, τ₂] \mid \forall (W, w₁, w₂) \in \varphi_w . \forall W' \supseteq W . (W', w₁, w₂) \in \varphi_w \}

TValRel \stackrel{\text{def}}{=} \{ VR = (τ₁, τ₂, \varphi_w) \mid \varphi_w \in WvalRel[τ₁, τ₂] \}

StackRel[σ₁, σ₂] \stackrel{\text{def}}{=} \{ \varphi_S \subseteq StackAtom[σ₁, σ₂] \}

TStackRel \stackrel{\text{def}}{=} \{ SR = (σ₁, σ₂, \varphi_S) \mid \varphi_S \in StackRel[σ₁, σ₂] \}

The set D[Δ] ensures that an environment ρ mapping type variables to value relations is well-formed.

```
D[\cdot] \stackrel{\text{def}}{=} \{ \emptyset \}
D[Δ, α] \stackrel{\text{def}}{=} \{ ρ[α \mapsto VR] \mid ρ \in D[Δ] \land VR \in TValRel \}
D[Δ, ζ] \stackrel{\text{def}}{=} \{ ρ[ζ \mapsto SR] \mid ρ \in D[Δ] \land SR \in TStackRel \}
D[Δ, ε] \stackrel{\text{def}}{=} \{ ρ[ε \mapsto (q₁, q₂)] \mid ρ \in D[Δ] \land ftv(q₁) = \emptyset \land ftv(q₂) = \emptyset \}
```

We use ρ₁ and ρ₂ to denote the substitutions formed by mapping variables in dom ρ to the first and second components, respectively, of the tuples they map to.

We also use some shorthands for referring to atoms of a particular type in terms of an environment ρ:

```
TermAtom[q \vdash τ; σ]_ρ \stackrel{\text{def}}{=} \text{TermAtom}[(ρ₁(q) \vdash ρ₁(τ); ρ₁(σ)), (ρ₂(q) \vdash ρ₂(τ); ρ₂(σ))]
ValAtom[τ]_ρ \stackrel{\text{def}}{=} \text{ValAtom}[ρ₁(τ), ρ₂(τ)]
WvalAtom[τ]_ρ \stackrel{\text{def}}{=} \text{WvalAtom}[ρ₁(τ), ρ₂(τ)]
HvalAtom[ψ]_ρ \stackrel{\text{def}}{=} \text{HvalAtom}[ρ₁(ψ), ρ₂(ψ)]
ContAtom[q \vdash τ; σ]_ρ ↦ [q' \vdash τ'; σ']_ρ' \stackrel{\text{def}}{=} \text{ContAtom}[(ρ₁(q) \vdash ρ₁(τ); ρ₁(σ)), (ρ₂(q) \vdash ρ₂(τ); ρ₂(σ))]
\vdash [(ρ₁'(q') \vdash ρ₁'(τ'); ρ₁'(σ')), (ρ₂'(q') \vdash ρ₂'(τ'); ρ₂'(σ'))]
```

The following relation says that a memory relation ϕ_M satisfies the constraints imposed by a memory relation ϕ_M' in all worlds accessible from W.

```
ϕ_M \in_W ϕ_M' \stackrel{\text{def}}{=} \forall (\widetilde{W}, M₁, M₂) \in ϕ_M . \widetilde{W} \supseteq W \implies (\widetilde{W}, M₁, M₂) \in ϕ_M'
```

```
V[\text{unit}]_ρ \ = \ (W, (), ()) \in \text{ValAtom}[\text{unit}]_ρ
V[\text{int}]_ρ \ = \ (W, n, n) \in \text{ValAtom}[\text{int}]_ρ
V[μα.τ]_ρ \ = \ (W, \text{fold}_μα.τ v₁, \text{fold}_μα.τ v₂) \in \text{ValAtom}[μα.τ]_ρ |
                   \quad (W, v₁, v₂) \in \triangleright V[μα.τ/α]_ρ
V[τ₁, ..., τₙ]_ρ \ = \ (W, (v₁₁, ..., vₙ₁), (v₂₁, ..., vₙ₂)) \in \text{ValAtom}[τ₁, ..., τₙ]_ρ |
                   \quad ∀ j \in \{1, ..., n\} . (W, v₁j, v₂j) \in V[τ_j]_ρ
V[τ] \rightarrow τ' \_ρ \ = \ (W, v₁, v₂) \in \text{ValAtom}[τ \rightarrow τ']_ρ |
                   \quad ∀ W' \supseteq W . ∀ SR \in \text{TStackRel} . ∀ \overline{v₁}, \overline{v₂},
                   \quad \text{let } ρ' = ρ[ζ \mapsto SR]\text{ in}
                   \quad \text{currentMR}(W'(i_{stk})) \in W' . S[ζ]_ρ' \land (W', v₁', v₂') \in V[τ]_ρ'
                   \quad \implies (W', v₁ v₁', v₂ v₂') \in E[\text{out} \vdash τ'; ζ]_ρ'
V[ϕ_i; ϕ_o] \rightarrow τ' \_ρ \ = \ (W, v₁, v₂) \in \text{ValAtom}[ϕ_i; ϕ_o] \rightarrow τ' \_ρ |
                   \quad ∀ W' \supseteq W . ∀ SR \in \text{TStackRel} . ∀ \overline{v₁}, \overline{v₂},
                   \quad \text{let } ρ' = ρ[ζ \mapsto SR]\text{ in}
                   \quad \text{currentMR}(W'(i_{stk})) \in W' . S[ϕ_i \vdash ζ]_ρ' \land (W', v₁ v₁', v₂ v₂') \in V[ϕ_o \vdash ζ]_ρ'
                   \quad \implies (W', v₁ v₁', v₂ v₂') \in E[\text{out} \vdash τ'; ϕ_o \vdash ζ]_ρ'
```

\[\mathcal{W}[\alpha]_\rho = \rho(\alpha) \cdot \varphi_w\]
\[\mathcal{W}[\text{unit}]_\rho = \{(W, ()) \in \text{WvalAtom}[\text{unit}]_\rho\}\]
\[\mathcal{W}[\text{int}]_\rho = \{(W, n, n) \in \text{WvalAtom}[\text{int}]_\rho\}\]
\[\mathcal{W}[\exists \alpha. \tau]_\rho = \{(W, \text{pack}(\tau_1, w_1) \text{ as } \rho_1(\exists \alpha. \tau), \text{pack}(\tau_2, w_2) \text{ as } \rho_2(\exists \alpha. \tau)) \in \text{WvalAtom}(\exists \alpha. \tau)_\rho\mid \exists \varphi_w \in \text{WvalRel}(\tau_1, \tau_2). (W, w_1, w_2) \in \mathcal{W}[\tau]_\rho[\alpha \mapsto (\tau_1, \tau_2, \varphi_w)]\}\]
\[\mathcal{W}[\mu \alpha. \tau]_\rho = \{(W, \text{fold}_1(\mu \alpha. \tau) w_1, \text{fold}_2(\mu \alpha. \tau) w_2) \in \text{WvalAtom}(\mu \alpha. \tau)_\rho\mid (W, w_1, w_2) \in \mathcal{W}[\tau]_\rho[\alpha/\mu \alpha. \tau]\}\]
\[\mathcal{W}[\text{ref } \psi]_\rho = \{(W, \ell_1, \ell_2) \in \text{WvalAtom}[\text{ref } \psi]_\rho \mid \exists \psi. \forall W' \supseteq W. (\ell_1, \ell_2) \in W'(i). \text{bij}(W'(i), s) \land \exists \varphi_M. \text{currentMR}(W'(i)) = \varphi_M \otimes \{(\bar{W}, \{\ell_1 \mapsto h_1\}, \{\ell_2 \mapsto h_2\})\} \in \text{MemAtom} \mid (\bar{W}, h_1, h_2) \in \mathcal{H}[\psi]_\rho\}\]
\[\mathcal{W}[\text{box } (\tau_1, \ldots, \tau_n)]_\rho = \{(W, \ell_1, \ell_2) \in \text{WvalAtom}(\text{box } (\tau_1, \ldots, \tau_n))_\rho \mid \forall (\bar{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})). \bar{W} \supseteq W \Rightarrow (\bar{W}, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{H}[\tau_1, \ldots, \tau_n]_\rho\}\]
\[\mathcal{W}[\forall [\Delta]. \{\chi; \sigma\}^q]_\rho = \{(W, \ell_1[\ell_{\tau_1}], \ell_2[\ell_{\tau_2}]) \in \text{WvalAtom}(\forall [\Delta]. \{\chi; \sigma\}^q)_\rho \mid \forall (\bar{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})). \bar{W} \supseteq W \Rightarrow (M_1(\ell_1) = \text{code}(\mathcal{B}_{\tau_1}, [\Delta]_{\{\chi_1; \sigma_1\}^q}, I_1) \land \rho_1(\chi_1) = \chi_1[\omega_1/\beta_1] \land \rho_1(\sigma) = \sigma_1[\omega_1/\beta_1] \land \rho_1(q) = q_1[\omega_1/\beta_1] \land M_2(\ell_2) = \text{code}(\mathcal{B}_{\tau_2}, [\Delta]_{\{\chi_2; \sigma_2\}^q}, I_2) \land \rho_2(\chi_2) = \chi_2[\omega_2/\beta_2] \land \rho_2(\sigma) = \sigma_2[\omega_2/\beta_2] \land \rho_2(q) = q_2[\omega_2/\beta_2] \land (
\bar{W}, \text{code}(\Delta)_{\{\chi_1; \sigma_1\}^q, I_1)[\omega_1/\beta_1],
\text{code}(\Delta)_{\{\chi_2; \sigma_2\}^q, I_2)[\omega_2/\beta_2]) \in \mathcal{H}[\forall [\Delta]. \{\chi; \sigma\}^q]_\rho\}\}\]
\[\mathcal{H}[\forall [\Delta]. \{\chi; \sigma\}^q]_\rho = \{(W, \text{code}(\Delta)_{\rho_1(\chi), \rho_1(\sigma)} \rho_1(q), I_1, \text{code}(\Delta)_{\rho_2(\chi), \rho_2(\sigma)} \rho_2(q), I_2) \in \text{HvalAtom}(\forall [\Delta]. \{\chi; \sigma\}^q)_\rho \mid \forall W' \supseteq W. \forall \rho^* \in \mathcal{D}_{\Delta}. \forall \tau, \tau'. \text{ let } \rho' = \rho \cup \rho^* \text{ in } \tau; \sigma' = \rho' \text{ ret-type}(q, \chi, \sigma) \land \text{currentMR}(W'(i_{\text{reg}})) \in W \cdot \mathcal{R}[\chi]_{\rho'} \land \text{currentMR}(W'(i_{\text{st}})) \in W \cdot \mathcal{S}[\sigma]_{\rho'} \Rightarrow (W', (\rho_1'(I_1, \cdot), (\rho_2'(I_2, \cdot))) \in \mathcal{E}[q \vdash \tau; \sigma']_{\rho'})\}\]
\[\tau; \sigma' = \rho \text{ ret-type}(q, \chi, \sigma) \overset{\text{def}}{=} \rho_1(\tau_1); \rho_1(\sigma') = \text{ret-type}(\rho_1(q), \rho_1(\chi), \rho_1(\sigma)) \land \rho_2(\tau_2); \rho_2(\sigma') = \text{ret-type}(\rho_2(q), \rho_2(\chi), \rho_2(\sigma))\]
\[\mathcal{H}[\tau_1, \ldots, \tau_n]_\rho = \{(W, (w_{11}, \ldots, w_{1m}), (w_{21}, \ldots, w_{2n})) \in \text{HvalAtom}(\tau_1, \ldots, \tau_n)_\rho \mid \forall j \in \{1, \ldots, n\}. (W, w_{1j}, w_{2j}) \in \mathcal{W}[\tau_j]_\rho\}\]
\[(M_1, M_2) : W \overset{\text{def}}{=} \vdash M_1 : W.\Phi_1 \land \vdash M_2 : W.\Phi_2 \land \]
\[(W.k > 0 \implies (\triangleright W, M_1, M_2) \in \bigotimes\{\text{currentMR}(\theta) \mid \theta \in W.\Theta\})\]
\[
\text{running}(k, (M | e)) \overset{\text{def}}{=} \exists M', e'. (M | e) \mapsto^k (M' | e')
\]
\[
\mathcal{O} = \{(W, e_1, e_2) \mid \forall (M_1, M_2) : W. ((M_1 | e_1) \downarrow \land (M_2 | e_2) \downarrow) \lor \]
\[
(\text{running}(W.k, (M_1 | e_1)) \land \text{running}(W.k, (M_2 | e_2)))\}
NOTE: Our continuations are untyped. See the operational semantics (3.1.7) for more detail.

\( K[\text{out} \vdash \tau; \sigma]_\rho = \{ (W, E_1, E_2) \mid \forall W', v_1, v_2. \\
W' \sqsupseteq_{\text{pub}} W \land (W', v_1, v_2) \in \mathcal{V}[\tau]_\rho \land \text{currentMR}(W'(i_{\text{stk}})) \subseteq W'. S[\sigma]_\rho \\
\Rightarrow (W', E_1[v_1], E_2[v_1]) \in \mathcal{O} \} \)

\( K[q \vdash \tau; \sigma]_\rho = \{ (W, E_1, E_2) \mid \forall W', q', r_1, r_2. \\
W' \sqsupseteq_{\text{pub}} W \land (q =_\rho q' =_\rho \text{end}\{\tau; \sigma\} \lor \\
(\exists r. q' = r \land \text{ret-addr}_1(W, \rho_1(q)) = W'.R_1(r) \land \text{ret-addr}_2(W, \rho_2(q)) = W'.R_2(r) \land \\
\text{ret-reg}_1(W', r) = r_1 \land \text{ret-reg}_2(W', r) = r_2) \land \\
(W'.R_1(r_1), W'.R_2(r_2)) \in W[\tau]_\rho \land \text{currentMR}(W'(i_{\text{stk}})) \subseteq W'. S[\sigma]_\rho \\
\Rightarrow (W', E_1[[\text{ret} \rho_1(q') \{r_1\}, \cdot]], E_2[[\text{ret} \rho_2(q') \{r_2\}, \cdot]]) \in \mathcal{O} \} \)

\( q =_\rho q' \overset{\text{def}}{=} \rho_1(q) =_\rho \rho_1(q') \land \rho_2(q) =_\rho \rho_2(q') \)

\( \forall r. q' = r \land \text{ret-addr}_1(W, \rho_1(q)) = W'.R_1(r) \land \text{ret-addr}_2(W, \rho_2(q)) = W'.R_2(r) \land \\
\text{ret-reg}_1(W', r) = r_1 \land \text{ret-reg}_2(W', r) = r_2 \land \\
(W'.R_1(r_1), W'.R_2(r_2)) \in W[\tau]_\rho \land \text{currentMR}(W'(i_{\text{stk}})) \subseteq W'. S[\sigma]_\rho \\
\Rightarrow (W', E_1[[\text{ret} \rho_1(q') \{r_1\}, \cdot]], E_2[[\text{ret} \rho_2(q') \{r_2\}, \cdot]]) \in \mathcal{O} \} \)

\( \mathcal{E}[q \vdash \tau; \sigma]_\rho = \{ (W, e_1, e_2) \in \text{TermAtom}[q \vdash \tau; \sigma]_\rho \mid \\
\forall E_1, E_2. (W, E_1, E_2) \in K[q \vdash \tau; \sigma]_\rho \Rightarrow (W, E_1[e_1], E_2[e_2]) \in \mathcal{O} \} \)

\( G[\cdot]_\rho \overset{\text{def}}{=} \{ (W, \emptyset) \mid W \in \text{World} \} \)

\( G[\Gamma, x : \tau]_\rho \overset{\text{def}}{=} \{ (W, \gamma[x \mapsto (v_1, v_2)]) \mid (W, \gamma) \in G[\Gamma]_\rho \land (W, v_1, v_2) \in \mathcal{V}[\tau]_\rho \} \)

\( \mathcal{H}[\cdot] = \text{World} \)

\( \mathcal{H}[\Psi, \ell : \text{ref}\psi] = \mathcal{H}[\Psi] \cap \{ W \in \text{World} \mid (W, \ell, \ell) \in W[\text{ref}\psi]_\emptyset \} \)

\( \mathcal{H}[\Psi, \ell : \text{box}\psi] = \mathcal{H}[\Psi] \cap \{ W \in \text{World} \mid (W, \ell, \ell) \in W[\text{box}\psi]_\emptyset \} \)

\( \mathcal{R}[\chi]_\rho = \{ (W, R_1, R_2) \mid \forall (x : \tau) \in \chi. (W, R_1(r), R_2(r)) \in W[\tau]_\rho \} \)

\( S[\zeta]_\rho = \rho(\zeta).\varphi_S \)

\( S[\bullet]_\rho = \{ (W, \text{nil}, \text{nil}) \mid W \in \text{World} \} \)

\( S[\tau :: \sigma]_\rho = \{ (W, (w_1 :: S_1), (w_2 :: S_2)) \mid (W, w_1, w_2) \in W[\tau]_\rho \land (W, S_1, S_2) \in S[\sigma]_\rho \} \)
\[\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx e_2; \tau; \sigma' \overset{\text{def}}{=} \Psi; \Delta; \Gamma; \chi'; \sigma'; q \vdash e_1; \tau; \sigma'; \wedge \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_2; \tau; \sigma' \wedge \forall W, \gamma, \rho. W \in \mathcal{H}[\Psi] \land \rho \in \mathcal{D}[\Delta] \land (W, \gamma) \in \mathcal{G}[\Gamma] \rho \land \text{currentMR}(W(i_{\text{reg}})) \in W \mathcal{R}[\chi] \rho \land \text{currentMR}(W(i_{\text{stk}})) \in W \mathcal{S}[\sigma] \rho \implies (W, \rho_1(\gamma_1(e_1)), \rho_2(\gamma_2((e_2)))) \in \mathcal{E}[q \vdash \tau; \sigma'] \rho\]

3.3.1 Other Logical Equivalences

To simplify the structure of the Lemmas in subsequent chapters, we define the following notions of logical equivalence for heaps, values, and instruction sequences:

**Definition 3.1 (Logical Equivalence for Heaps)**

\[\Psi \vdash H_1 \approx_{H} H_2 : \Psi' \overset{\text{def}}{=} \Psi \vdash H_1 : \Psi' \wedge \forall (\ell : \psi) \in \Psi'. \Psi, \Psi' \vdash H_1(\ell) \approx_{hv} H_2(\ell) : \nu \psi\]

**Definition 3.2 (Logical Equivalence for Small Values)**

\[\Psi; \Delta \vdash w_1 \approx_{w} w_2 : \tau \overset{\text{def}}{=} \Psi; \Delta \vdash w_1 : \tau \land \Psi; \Delta \vdash w_2 : \tau \land \forall W \in \mathcal{H}[\Psi]. \forall \rho \in \mathcal{D}[\Delta]. (W, \rho_1(w_1), \rho_2(w_2)) \in W[\tau] \rho\]

**Definition 3.3 (Logical Equivalence for Instruction Sequences)**

\[\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I_1 \approx_{I} I_2 \overset{\text{def}}{=} \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I_1 \land \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I_2 \land \forall W \in \mathcal{H}[\Psi]. \forall \rho \in \mathcal{D}[\Delta]. \forall \gamma \in \mathcal{G}[\Gamma] \rho. \text{currentMR}(W(i_{\text{reg}})) \in W \mathcal{R}[\chi] \rho \land \text{currentMR}(W(i_{\text{stk}})) \in W \mathcal{S}[\sigma] \rho \implies (W, \rho_1(\gamma_1((I_1, \cdot))), \rho_2(\gamma_2((I_2, \cdot)))) \in \mathcal{E}[q \vdash \text{ret-type}(q, \chi, \sigma)] \rho\]
3.4 Basic Properties

3.4.1 Operations on Worlds

Lemma 3.4 (World Extension is Reflexive and Transitive)
For any $W, W', W'' \in \text{World}$, we have

1. $W \sqsupseteq W$
2. $W \sqsupseteq_{pub} W$
3. If $W'' \sqsupseteq W'$ and $W' \sqsupseteq W$, then $W'' \sqsupseteq W$
4. If $W'' \sqsupseteq_{pub} W'$ and $W' \sqsupseteq_{pub} W$, then $W'' \sqsupseteq_{pub} W$

Proof

1. By definition of $\sqsupseteq$ for worlds and islands, and by the reflexivity of transition relations $\delta$ in the definition of World.
2. By definition of $\sqsupseteq_{pub}$ for worlds and islands, and by the reflexivity of public transition relations $\pi$ in the definition of World.
3. By definition of $\sqsupseteq$ for worlds and islands, and by the transitivity of transition relations $\delta$ in the definition of World.
4. By definition of $\sqsupseteq_{pub}$ for worlds and islands, and by the transitivity of public transition relations $\pi$ in the definition of World.

Lemma 3.5 (Properties of $\boxplus$)

1. If $(M_1, M_2) : W$ and $M'_1 = (H_1, \bot, \bot)$, $M'_2 = (H_2, \bot, \bot)$, then
   
   $(M_1 \boxplus M'_1, M_2 \boxplus M'_2) : W \boxplus (H_1, H_2)$.
2. $(W \boxplus (H_1, H_2)) \boxplus (H'_1, H'_2) = W \boxplus (H_1 \boxplus H'_1, H_2 \boxplus H'_2)$.
3. If $W \in \text{World}$ and $(W \boxplus (H_1, H_2))$ is defined, then $(W \boxplus (H_1, H_2)) \sqsupseteq W$ and $(W \boxplus (H_1, H_2)) \sqsupseteq_{pub} W$.

Proof

1. By definition of $W(i_{box})$.
2. By definition of $W(i_{box})$.
3. By definition of $\sqsupseteq$, $\sqsupseteq_{pub}$, and island_{box}.

Lemma 3.6 (Properties of $\triangleright$ and $\sqsupseteq$)

For any $W \in \text{World}$, we have

1. $\triangleright W \sqsupseteq W$
2. $\triangleright W \sqsupseteq_{pub} W$
3. If $(M_1, M_2) : W$, then $(M_1, M_2) : \triangleright W$.
4. If $W' \sqsupseteq W$, then $W' \sqsupseteq W$.
5. If $W' \sqsupseteq W$, then $W' \sqsupseteq \triangleright W$.

Proof

1. By definition of $\triangleright$ and $\sqsupseteq$, it suffices to show that $[\theta]_{W,k-1} \sqsupseteq [\theta]_{W,k-1}$ for each island $\theta \in W, \Theta$. But this relation is reflexive, so we are done.
2. Similar.
3. Note that if $W.k = 0$, there is nothing to show. Otherwise, the claim follows from the definitions of MemRel and $[\varphi_M].k$.
4. Immediate from the definition of $\sqsubseteq$.
5. From the definition of $\sqsubseteq$ we have $W'.k < W.k$ and $W' \sqsupseteq W$. The latter implies that $W' \in \text{World}$, which gives us $0 \leq W'.k$. Hence, $0 < W.k$. Let $W = (k + 1, \Psi_1, \Psi_2, \Theta)$. We have that:

\[ (W'.k, W'.\Psi_1, W'.\Psi_2, W'.\Theta) \sqsupseteq (k + 1, \Psi_1, \Psi_2, \Theta) \]

We must show that:

\[ (W'.k, W'.\Psi_1, W'.\Psi_2, W'.\Theta) \sqsupseteq (k, \Psi_1, \Psi_2, [\Theta]_k) \]

It suffices to show the following:

- $W'.k \leq (\triangleright W).k$: this follows from $W'.k < W.k$ and $(\triangleright W).k = W.k - 1$.
- $W'.\Psi_i \supseteq \Psi_i$: by (4) we have $W' \sqsupseteq W$, from which this fact is immediate.
- $W'.\Theta \sqsupseteq [\Theta]_{W'.k}$: From above we have that $W'.\Theta \sqsupseteq [\Theta]_{W'.k}$. Furthermore, since $W'.k \leq (\triangleright W).k = W.k - 1 = k$, we have that $[\Theta]_{W'.k} = [\Theta]_{W',k}$ so we are done.

\[ \square \]

### 3.4.2 Properties of the Observation Relation

**Lemma 3.7 (O Closed under Anti-Reduction)**

Given $W' \sqsupseteq W$, if $W.k \leq W'.k + k_1$, $W.k \leq W'.k + k_2$, and

\[ \forall (M_1, M_2) : W. \exists (M'_1, M'_2) : W'. \langle M_1 | e_1 \rangle \mapsto^{k_1} \langle M'_1 | e'_1 \rangle \land \langle M_2 | e_2 \rangle \mapsto^{k_2} \langle M'_2 | e'_2 \rangle, \]

then

\[ (W', e_1, e_2) \in \mathcal{O} \implies (W, e_1, e_2) \in \mathcal{O}. \]

**Proof**

Let $(M_1, M_2) : W$. Then, by our assumption, $\langle M_1 | e_1 \rangle \mapsto^{k_1} \langle M'_1 | e'_1 \rangle$ and $\langle M_2 | e_2 \rangle \mapsto^{k_2} \langle M'_2 | e'_2 \rangle$ for some $(M'_1, M'_2) : W'$. Since $(W', e'_1, e'_2) \in \mathcal{O}$, we have either that $(M'_1 | e'_1) \downarrow$ and $(M'_2 | e'_2) \downarrow$ or that running$(W'.k, \langle M'_1 | e'_1 \rangle)$ and running$(W'.k, \langle M'_2 | e'_2 \rangle)$.

In the former case, we have $(M_1 | e_1) \downarrow$ and $(M_2 | e_2) \downarrow$ by assumption. In the latter case, we have running$(W'.k + k_1, \langle M_1 | e_1 \rangle)$ and running$(W'.k + k_2, \langle M_2 | e_2 \rangle)$. Since we have assumptions that both of these are more steps than needed, we have the result.

\[ \square \]

### 3.4.3 Monotonicity and Reduction

**Lemma 3.8 (Monotonicity)**

Let $\rho \in \mathcal{D}[\Delta]$, where $\Delta \vdash \tau$, $\Delta \vdash \psi$, and $\Delta \vdash \tau$. If $W' \sqsupseteq W$, then

1. $\langle W, w_1, w_2 \rangle \in W[\tau] \rho \implies \langle W', w_1, w_2 \rangle \in W[\tau] \rho$
2. $\langle W, h_1, h_2 \rangle \in \mathcal{H}[\psi] \rho \implies \langle W', h_1, h_2 \rangle \in \mathcal{H}[\tau] \rho$.
3. $\langle W, w_1, w_2 \rangle \in \mathcal{V}[\tau] \rho \implies \langle W', w_1, w_2 \rangle \in \mathcal{V}[\tau] \rho$.

**Proof**

1. Proved by induction on $W'.k$ and on the structure of $\tau$, simultaneously with Claim 2.

   In each case, we will need to show $\langle W', w_1, w_2 \rangle \in \text{WvalAtom}[\tau] \rho$. This amounts to showing that $W'.\Psi_i \vdash w_i : \tau$ for $i \in \{1, 2\}$. We have by assumption that $W.\Psi_i \vdash w_i : \tau$. By definition of world extension, $W'.\Psi_i \supseteq W.\Psi_i$, so this property holds.

   To complete the proof, consider the possible cases of $\tau$.
Case \( \alpha \) follows from \( \rho(\alpha).\varphi^T \in W_{\text{ValRel}}[\rho(\alpha), T_1, \rho(\alpha), T_2] \), which holds by \( \rho(\alpha) \in TV_{\text{ValRel}} \).

Case unit Immediate.

Case \( \text{int} \) Immediate.

Case \( \exists \alpha.\tau \) Follows from the induction hypothesis for the type.

Case \( \mu \alpha.\tau \) Follows from the induction hypothesis for the step index.

Case \( \text{ref } \psi \) We need to show that \( (W', \ell_1, \ell_2) \in W[\text{ref } \psi] \). Let \( W'' \supseteq W' \). By transitivity of world extension, \( W'' \supseteq W \). Thus everything we need holds by our assumption that \( (W, \ell_1, \ell_2) \in W[\text{ref } \psi] \).

Case box \( \langle \tau_1, \ldots, \tau_n \rangle \) We need to show that \( (W', \ell_1, \ell_2) \in W[\text{box } \langle \tau_1, \ldots, \tau_n \rangle] \).

Let \( \langle W, M'_1, M'_2 \rangle \in \text{currentMR}(W(i_{\text{box}})) \) such that \( \widetilde{W} \supseteq W' \). By definition of island-box, \( M'_1 = (W(i_{\text{box}}).s.H_1) \) and \( M'_2 = (W(i_{\text{box}}).s.H_2) \). By our assumption, to show

\[
(\langle \widetilde{W}, M'_1(\ell_1), M'_2(\ell_2) \rangle) \in \mathcal{H}[\langle \tau_1, \ldots, \tau_n \rangle]\rho
\]

It suffices to find some \( M_1 \) and \( M_2 \) such that \( (\widetilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})) \), \( M_1(\ell_1) = M'_1(\ell_1) \), and \( M_2(\ell_2) = M'_2(\ell_2) \), noting that \( \widetilde{W} \supseteq W' \) follows from \( \widetilde{W} \supseteq W'' \supseteq W \).

We claim that \( M_1 = (W(i_{\text{box}}).s.H_1) \) and \( M_2 = (W(i_{\text{box}}).s.H_2) \) are suitable. The first condition holds immediately by definition of island-box. Since \( W' \supseteq W \), we know that \( W'(i_{\text{box}}) \supseteq W(i_{\text{box}}) \supseteq W \). Thus \( ((H_1, H_2), (H'_1, H'_2)) \in \delta_{\text{box}} \) that is, \( H_1 \subseteq H'_1 \) and \( H_2 \subseteq H'_2 \).

Since \( \ell_1 \) and \( \ell_2 \) must be in the domain of \( H_1 \) and \( H_2 \), we have the desired property that \( M_1(\ell_1) = M'_1(\ell_1) \) and \( M_2(\ell_2) = M'_2(\ell_2) \).

Case box \( \forall[\Delta].\{\chi: \sigma\}^g \) Let \( \langle W, M'_1, M'_2 \rangle \in \text{currentMR}(W(i_{\text{box}})) \) such that \( \widetilde{W} \supseteq W' \). It suffices to find some \( M_1 \) and \( M_2 \) such that \( (\widetilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}})) \), \( M_1(\ell_1) = M'_1(\ell_1) \), and \( M_2(\ell_2) = M'_2(\ell_2) \), noting that \( \widetilde{W} \supseteq W \). This can be done exactly as in the previous case.

2. Proved simultaneously with Claim 1.

In both cases, we need to show that \( (W', h_1, h_2) \in H_{\text{ValAtom}}[\psi] \). This amounts to showing that \( W'.\Psi i \vdash h_1: \psi \) for \( i \in \{1, 2\} \). We have by assumption that \( W'.\Psi i \supseteq W.\Psi i \), so this property holds.

Consider the possible cases of \( \psi \):

Case \( \forall[\Delta].\{\chi: \sigma\}^g \) We need to show that \( (W', h_1, h_2) \in H[\forall[\Delta].\{\chi: \sigma\}^g] \). Let \( W'' \supseteq W' \). By transitivity of world extension, \( W'' \supseteq W \). Thus everything we need holds by our assumption that \( (W, h_1, h_2) \in H[\forall[\Delta].\{\chi: \sigma\}^g] \).

Case \( \langle \tau_1, \ldots, \tau_n \rangle \) Follows from Claim 1 using the induction hypothesis for the type.

3. Proved by induction on \( W'.k \) and on the structure of \( \tau \).

In each case, we will need to show \( (W', \nu_1, \nu_2) \in V_{\text{Val}}[\tau] \). This amounts to showing that \( W'.\Psi i; \cdots; W'.\chi; W'.\sigma_1; \text{out } \vdash \nu_i: \tau; W'.\sigma_1 \) for \( i \in \{1, 2\} \). We have by assumption that \( W'.\Psi i; \cdots; W'.\chi; W.\sigma_1; \text{out } \vdash \nu_i: \tau; W.\sigma_1 \). By definition of world extension, \( W'.\Psi i \supseteq W.\Psi i \). However, note that \( W'.\chi \) and \( W'.\sigma_1 \) may be arbitrarily different from \( W.\chi \) and \( W.\sigma_1 \). But note that the only place where there are non-trivial dependencies of the typing on either of those are past the \( \text{FT} \)-boundaries. Note first that at the boundary, what is inside is restricted to type under the empty register file typing, so any changes to the typing outside the boundary is irrelevant. Note secondly that since we are dealing with values, the only place an \( \text{FT} \)-boundary can occur is within a lambda. The typing rule for lambdas requires that the body types under a fresh \( \zeta \) variable for the stack typing, both in and out, which means that any changes that are made to the stack typing outside of the lambda are irrelevant.

To complete the proof, consider the possible cases of \( \tau \):

Case unit No proof obligations beyond what was shown above.

Case int No proof obligations beyond what was shown above.

Case \( \mu \alpha.\tau \) This follows from the induction hypothesis.
Case $\langle \tau \rangle$ This follows from the induction hypothesis.

Case $\langle \tau \rangle \rightarrow \tau'$ The obligations in this case follow trivially from the transitivity of world extension, since with any future world $W' \supseteq W$ we can instantiate what we are given.

Case $\langle \tau \rangle \phi_{1} \cdot \phi_{0}$ This case is the same as the other arrow case.

Lemma 3.9 (Monotonicity for Heaps)
If $W' \supseteq W$ and $W \in H[\Psi]$, then $W' \in H[\Psi]$.

Proof
We use induction on the structure of $\Psi$. If $\Psi = \{ \cdot \}$ then there is nothing to show.

If $\Psi = \Psi', \ell; ref \psi$, then by the induction hypothesis, $W' \in H[\Psi']$, and it remains to show that $(W', \ell, \ell) \in W[ref \psi]$. But this follows from $W \in H[\Psi', \ell; ref \psi]$ and Lemma 3.8.

If $\Psi = \Psi', \ell; box \psi$, then by the induction hypothesis, $W' \in H[\Psi']$, and it remains to show that $(W', \ell, \ell) \in W[box \psi]$. But this follows from $W \in H[\Psi', \ell; box \psi]$ and Lemma 3.8.

Lemma 3.10 (Monotonicity for F Evaluation Contexts)
If $W' \supseteq_{pub} W$, then

$$(W, E_{1}, E_{2}) \in \mathcal{K}[out \vdash \tau; \sigma]\rho \implies (W', E_{1}, E_{2}) \in \mathcal{K}[out \vdash \tau; \sigma]\rho.$$ 

Proof
Follows from the transitivity of $\supseteq_{pub}$.

Lemma 3.11 (Monotonicity for T Evaluation Contexts)
If $W' \supseteq_{pub} W$ and if either $q =_{\rho} q' =_{\rho} end\{\tau; \sigma\}$ or $q =_{\rho} q' =_{\rho} out$ or if $\text{ret-addr}(W, \rho_{1}(q)) = \text{ret-addr}(W', \rho_{1}(q'))$ and $\text{ret-addr}(W, \rho_{2}(q)) = \text{ret-addr}(W', \rho_{2}(q'))$, then

$$(W, E_{1}, E_{2}) \in \mathcal{K}[q \vdash \tau; \sigma]\rho \implies (W', E_{1}, E_{2}) \in \mathcal{K}[q' \vdash \tau; \sigma]\rho.$$ 

Proof
Follows from the transitivity of $\supseteq_{pub}$ and our hypotheses about the relationship between $q$ and $q'$.

Lemma 3.12 ($E[q \vdash \tau; \sigma]\rho$ Closed under Type-Preserving Anti-Reduction)
Let $(W, e_{1}, e_{2}) \in \text{TermAtom}[q \vdash \tau; \sigma]\rho$. Given $W' \supseteq_{pub} W$, $W.k \leq W'.k + k_{1}$, $W.k \leq W'.k + k_{2}$, and if $q =_{\rho} q' =_{\rho} end\{\tau; \sigma\}$ or $q =_{\rho} q' =_{\rho} out$ or if $\text{ret-addr}(W, \rho_{1}(q)) = \text{ret-addr}(W', \rho_{1}(q'))$ and $\text{ret-addr}(W, \rho_{2}(q)) = \text{ret-addr}(W', \rho_{2}(q'))$, and if

$$\forall(M_{1}, M_{2}) : W. \exists(M'_{1}, M'_{2}) : W'. \langle M_{1} | e_{1} \rangle \rightarrow^{k_{1}} \langle M'_{1} | e'_{1} \rangle \land \langle M_{2} | e_{2} \rangle \rightarrow^{k_{2}} \langle M'_{2} | e'_{2} \rangle,$$

then

$$(W', e'_{1}, e'_{2}) \in E[q' \vdash \tau; \sigma]\rho \implies (W, e_{1}, e_{1}) \in E[q \vdash \tau; \sigma]\rho.$$ 

Proof
Now let $(W, E_{1}, E_{2}) \in \mathcal{K}[q \vdash \tau; \sigma]\rho$. We need to show that $(W, E_{1}[e_{1}], E_{2}[e_{2}]) \in O$.

We can use Lemma 3.11 and Lemma 3.10 to conclude that $(W', E_{1}, E_{2}) \in \mathcal{K}[q' \vdash \tau; \sigma]\rho$. Instantiating the latter with $(W', e'_{1}, e'_{2}) \in E[q' \vdash \tau; \sigma]\rho$ gives us $(W', E_{1}[e'_{1}], E_{2}[e'_{2}]) \in O$.

By inspection of the operational semantics and by assumption, for any $(M_{1}, M_{2}) : W$, there is an $(M'_{1}, M'_{2}) : W'$ such that

$$\langle M_{1} | E_{1}[e_{1}] \rangle \rightarrow^{k_{1}} \langle M'_{1} | E_{1}[e'_{1}] \rangle \quad \text{and} \quad \langle M_{2} | E_{2}[e_{2}] \rangle \rightarrow^{k_{2}} \langle M'_{2} | E_{2}[e'_{2}] \rangle.$$ 

The rest of this case follows by Lemma 3.7.
Lemma 3.13 \((\mathcal{E}[q \vdash \tau; \sigma] \rho\) Closed under Memory-Preserving Anti-Reduction\)

Let \((W, e_1, e_2) \in \text{TermAtom}[q \vdash \tau; \sigma] \rho\).

If
\[
\forall (M_1, M_2) : W. \langle M_1 | e_1 \rangle \rightarrow^* \langle M_1 | e_1' \rangle \land \langle M_2 | e_2 \rangle \rightarrow^* \langle M_2 | e_2' \rangle,
\]
then
\[
(W; e_1', e_2') \in \mathcal{E}[q \vdash \tau; \sigma] \rho \implies (W, e_1, e_1) \in \mathcal{E}[q \vdash \tau; \sigma] \rho.
\]

Proof

This follows from Lemma 3.12.

3.4.4 Substitution

The next lemma is a simple property, but its proof shows the induction structure by which properties of the mutually-dependent parts of the logical relation can be proved.

Definition 3.14

\[
\xi ::= \alpha \mid \zeta \mid \epsilon
\]

AR ::= VR \mid SR \mid QR

Lemma 3.15

[Weakening] If \(\rho[\xi \mapsto \text{AR}] \in \mathcal{D}[\Delta, \xi]\) and \(\xi \notin \text{ftv}(\tau)\), \(\xi \notin \text{ftv}(\sigma)\), \(\xi \notin \text{ftv}(\chi)\), \(\xi \notin \text{ftv}(\psi)\), then

1. \(S[\sigma] \rho = S[\sigma] \rho[\xi \mapsto \text{AR}]\)
2. \(R[\chi] \rho = R[\chi] \rho[\xi \mapsto \text{AR}]\)
3. \(W[\tau] \rho = W[\tau] \rho[\xi \mapsto \text{AR}]\)
4. \(HV[\psi] \rho = HV[\psi] \rho[\xi \mapsto \text{AR}]\)
5. \(E[q \vdash \tau; \sigma] \rho = E[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\)
6. \(K[q \vdash \tau; \sigma] \rho = K[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\).
7. \(V[\tau] \rho = V[\tau] \rho[\xi \mapsto \text{AR}]\)
8. \(E[q \vdash \tau; \sigma] \rho = E[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\)
9. \(K[q \vdash \tau; \sigma] \rho = K[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\).

Proof

Assume \(\xi \notin \text{ftv}(\tau)\) and \(\xi \notin \text{ftv}(\sigma)\). We will need to prove the following:

1. (a) \((W, S_1, S_2) \in S[\sigma] \rho \implies (W, S_1, S_2) \in S[\sigma] \rho[\xi \mapsto \text{AR}]\)
   (b) \((W, S_1, S_2) \in S[\sigma] \rho[\xi \mapsto \text{AR}] \implies (W, S_1, S_2) \in S[\sigma] \rho\)
2. (a) \((W, R_1, R_2) \in R[\chi] \rho \implies (W, R_1, R_2) \in R[\chi] \rho[\xi \mapsto \text{AR}]\)
   (b) \((W, R_1, R_2) \in R[\chi] \rho[\xi \mapsto \text{AR}] \implies (W, R_1, R_2) \in R[\chi] \rho\)
3. (a) \((W, w_1, w_2) \in W[\tau] \rho \implies (W, w_1, w_2) \in W[\tau] \rho[\xi \mapsto \text{AR}]\)
   (b) \((W, w_1, w_2) \in W[\tau] \rho[\xi \mapsto \text{AR}] \implies (W, w_1, w_2) \in W[\tau] \rho\)
4. (a) \((W, h_1, h_2) \in HV[\psi] \rho \implies (W, h_1, h_2) \in HV[\psi] \rho[\xi \mapsto \text{AR}]\)
   (b) \((W, h_1, h_2) \in HV[\psi] \rho[\xi \mapsto \text{AR}] \implies (W, h_1, h_2) \in HV[\psi] \rho\)
5. (a) \((W, e_1, e_2) \in E[q \vdash \tau; \sigma] \rho \implies (W, e_1, e_2) \in E[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\)
   (b) \((W, e_1, e_2) \in E[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}] \implies (W, e_1, e_2) \in E[q \vdash \tau; \sigma] \rho\)
6. (a) \((W, E_1, E_2) \in K[q \vdash \tau; \sigma] \rho \implies (W, E_1, E_2) \in K[q \vdash \tau; \sigma] \rho[\xi \mapsto \text{AR}]\)
We will prove all these claims simultaneously, by induction on \( W.k \) and \( \tau \).

1. We use an additional induction and case analysis on the structure of \( \sigma \).
   
   Case \( \zeta \) Immediate, since \( \xi \neq \zeta \).
   
   Case \( \bullet \) Immediate.
   
   Case \( \tau : : \sigma \) For part (a), we have \( S_1 = w_1 :: S'_1 \), \( S_2 = w_2 :: S'_2 \), \( (W, w_1, w_2) \in W[\tau] : : \rho \), and \( (S, S'_1, S'_2) \subseteq S[\sigma] : : \rho \). We need to show that \( (W, w_1, w_2) \in W[\tau] : : \rho[\xi \mapsto \text{AR}] \) and \( (W, S'_1, S'_2) \subseteq S[\sigma] : : \rho[\xi \mapsto \text{AR}] \). The latter holds by the induction hypothesis for \( \sigma \) and the former holds by claim 3.

   Part (b) is similar.

2. Follows from claim 3.

3. Consider the possible cases of \( \tau \):
   
   Case \( \alpha \) Immediate, since \( \alpha \neq \xi \).
   
   Case \( \text{unit} \) Immediate.
   
   Case \( \exists \alpha. \tau \) By the induction hypothesis for \( \tau \).
   
   Case \( \mu \alpha . \tau \) By the induction hypothesis for \( W.k \).
   
   Case \( \text{ref } \langle \tau_1, \ldots, \tau_n \rangle \) Follows from claim 4.
   
   Case \( \text{box } \langle \tau_1, \ldots, \tau_n \rangle \) Follows from claim 4.
   
   Case \( \text{box } \forall \Delta . \langle \chi ; \sigma' \rangle^\tau \) Follows from claim 4.

4. Consider the two possible cases of \( \psi \):
   
   Case \( \forall \Delta . \langle \chi ; \sigma' \rangle^\tau \) Follows from claims 1 and 2 (using the induction hypothesis for \( \tau \)) and from claim 5 (using the induction hypothesis for \( W.k \)).
   
   Case \( \langle \tau_1, \ldots, \tau_n \rangle \) Follows from claim 3 (using the induction hypothesis for \( \tau \)).

5. Follows from claim 6.

6. Follows from claims 1 and 3.

7. Consider the possible cases of \( \tau \) that are defined in \( W[\tau] : : \rho \):
   
   Case \( () \) Immediate.
   
   Case \( \text{int} \) Immediate.
   
   Case \( \mu \alpha . \tau \) By the induction hypothesis for \( W.k \).
   
   Case \( \langle \tau \rangle \) By the induction hypothesis for \( \tau \).
   
   Case \( \tau \rightarrow \tau \) Follows from claim 1 (using induction hypothesis for \( \tau \)) and from claim 8 (using induction hypothesis for \( W.k \)).
   
   Case \( \tau \rightarrow \rightarrow \tau \) Same as other arrow case.

8. Follows from claim 9.

9. Follows from claims 1 and 7.
Lemma 3.16 (Substitution)

Let \( \rho \in D[\Delta], \alpha \not\in \Delta, \Delta \vdash \tau', \) and \( \Delta, \alpha \vdash \tau, \Delta, \alpha \vdash \sigma, \Delta, \alpha \vdash \chi, \Delta, \alpha \vdash \psi. \) Then

1. \( S[\tau][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = S[\tau'][\rho]\)
2. \( R[\chi][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = R[\chi'][\rho]\)
3. \( W[\tau'][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = W[\tau][\tau'][\rho]\)
4. \( \mathcal{H}[\psi][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = \mathcal{H}[\psi'][\rho]\)
5. \( \mathcal{E}[q \vdash \tau; \sigma][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = \mathcal{E}[q[\tau'/\alpha] \vdash \tau[\tau'/\alpha]; \sigma[\tau'/\alpha]]\)
6. \( \mathcal{K}[q \vdash \tau; \sigma][\rho[\alpha \mapsto (\rho_1(\tau'), \rho_2(\tau'))] = \mathcal{K}[q[\tau'/\alpha] \vdash \tau[\tau'/\alpha]; \sigma[\tau'/\alpha]]\)

Proof

Follows the structure of the proof of Lemma 3.15, aside from the cases about source types, since they cannot have free type variables. The only case that depends on \( \rho \) is in claim 3, in the case where \( \tau = \beta \). But the needed equality is immediate in this case, whether \( \alpha = \beta \) or not.

\[ \square \]

Lemma 3.17 (Substitution for Stack Types)

Let \( \rho \in D[\Delta], \zeta \not\in \Delta, \Delta \vdash \tau', \) and \( \Delta, \zeta \vdash \tau, \Delta, \zeta \vdash \sigma, \Delta, \zeta \vdash \chi, \Delta, \zeta \vdash \psi. \) Then

1. \( S[\tau][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = S[\sigma'/\zeta][\rho]\)
2. \( R[\chi][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = R[\chi'][\rho]\)
3. \( W[\tau'][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = W[\tau][\sigma'/\zeta][\rho]\)
4. \( \mathcal{H}[\psi][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = \mathcal{H}[\psi'][\rho]\)
5. \( \mathcal{E}[q \vdash \tau; \sigma][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = \mathcal{E}[q[\sigma'/\zeta] \vdash \tau[\sigma'/\zeta]; \sigma[\sigma'/\zeta]]\)
6. \( \mathcal{K}[q \vdash \tau; \sigma][\rho[\zeta \mapsto (\rho_1(\sigma'), \rho_2(\sigma'))] = \mathcal{K}[q[\sigma'/\zeta] \vdash \tau[\tau'/\alpha]; \sigma[\tau'/\alpha]]\)

Proof

Follows the structure of the proof of Lemma 3.15. The only case that depends on \( \rho(\zeta) \) is in claim 1, in the case where \( \sigma = \zeta' \). But the needed equality is immediate, whether \( \zeta = \zeta' \) or not.

\[ \square \]

Lemma 3.18 (Substitution for Return Markers)

Let \( \rho \in D[\Delta], \Delta \vdash q', \) and \( \Delta, \epsilon \vdash \tau, \Delta, \epsilon \vdash \sigma, \Delta, \epsilon \vdash \chi, \Delta, \epsilon \vdash \psi. \) Then

1. \( S[\tau][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = S[q'/\epsilon][\rho]\)
2. \( R[\chi][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = R[q'/\epsilon][\rho]\)
3. \( W[\tau][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = W[q'/\epsilon][\rho]\)
4. \( \mathcal{H}[\psi][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \mathcal{H}[q'/\epsilon][\rho]\)
5. \( \mathcal{E}[q \vdash \tau; \sigma][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \mathcal{E}[q[q'/\epsilon] \vdash \tau[q'/\epsilon]; \sigma[q'/\epsilon]]\)
6. \( \mathcal{K}[q \vdash \tau; \sigma][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \mathcal{K}[q[q'/\epsilon] \vdash \tau[q'/\epsilon]; \sigma[q'/\epsilon]]\)
7. \( \mathcal{E}[q \vdash \tau; \sigma][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \mathcal{E}[q[q'/\epsilon] \vdash \tau; \sigma[q'/\epsilon]]\)
8. \( \mathcal{K}[q \vdash \tau; \sigma][\rho[\epsilon \mapsto (\rho_1(q'), \rho_2(q'))] = \mathcal{K}[q[q'/\epsilon] \vdash \tau; \sigma[q'/\epsilon]]\)

Proof

Followed the structure of the proof of Lemma 3.15. There are no interesting cases.

\[ \square \]
3.4.5 Properties of Semantic Interpretations

**Lemma 3.19**
If $\rho \in D[\Delta]$ and $\Delta \vdash \tau$, then $W[\tau] \rho \in Wval[\rho_1(\tau), \rho_2(\tau)]$.

**Proof**
Follows from monotonicity.

**Lemma 3.20**
If $\rho \in D[\Delta]$ and $\Delta \vdash \sigma$, then $S[\sigma] \rho \in StackRel[\rho_1(\sigma), \rho_2(\sigma)]$.

**Proof**
Proceed by induction on the structure of $\sigma$.

**Case** $\zeta$ From $\Delta \vdash \sigma$ we have that $\zeta \in \Delta$. Since $\rho \in D[\Delta]$, it follows that $S[\zeta] \rho = \rho(\zeta) \cdot \varphi S \in StackRel[\rho_1(\zeta), \rho_2(\zeta)]$.

**Case** $\cdot$ In this case, $S[\cdot] \rho = \{(W, \text{nil} \cdot, \text{nil} \cdot) \mid W \in \text{World}\} \in StackRel[\cdot, \cdot]$ is immediate from the definition of StackRel.

**Case** $\tau :: \sigma$ For any $(W, w_1 :: S_1, w_2 :: S_2) \in S[\tau :: \sigma] \rho$, we need that $W.\Psi_i \vdash w_i :: S_i : \rho_i(\tau :: \sigma)$.
From the definition of $S[\tau :: \sigma] \rho$ we have $(W, w_1, w_2) \in W[\tau] \rho$ and $(W, S_1, S_2) \in S[\sigma] \rho$, from which we have that $W.\Psi_i :: \vdash w_i : \rho_i(\tau)$ and $W.\Psi_i \vdash S_i : \rho_i(\sigma)$, which gives us what we need.

**Lemma 3.21** (Register File Subtyping Implies Inclusion)
Let $\rho \in D[\Delta]$ and $\Delta \vdash \chi \leq \chi'$. Then $R[\chi] \rho \subseteq R[\chi'] \rho$.

**Proof**
Consider arbitrary $(W, M_1, M_2) \in R[\chi] \rho$. Note that $M_1 = R_1 \cdot$ and $M_2 = R_2 \cdot$. We must show that $(W, M_1, M_2) = (W, R_1 \cdot, R_2 \cdot) \in R[\chi'] \rho$.
Consider $(\tau : \tau) \in \chi'$. We must show $(W, R_1(\tau), R_2(\tau)) \in W[\tau] \rho$. From the hypothesis $\Delta \vdash \chi \leq \chi'$, it follows that $\tau : \tau \in \chi$. We use the latter to instantiate $(W, R_1 \cdot, R_2 \cdot) \in R[\chi] \rho$, which gives us what we needed to show.

**Lemma 3.22** (World Updates that Respect Register-File Relation)
Let $\text{currentMR}(W(i_{\text{reg}})) \in W \cdot R[\chi] \rho$ and $W' \sqsupseteq W$.

1. If $W'(i_{\text{reg}}) = W(i_{\text{reg}})$, then $\text{currentMR}(W'(i_{\text{reg}})) \in W' \cdot R[\chi] \rho$.
2. Let $W' = (W, k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{\text{reg}} \mapsto \text{island}_{\text{reg}}(s, W.k)])$, where $s = (W.R_1[r_d \mapsto w_1], W.\chi_1[r_d ; \rho_1(\tau)], W.R_2[r_d \mapsto w_2], W.\chi_2[r_d ; \rho_2(\tau)])$. If $(W', w_1, w_2) \in W[\tau] \rho$, then $\text{currentMR}(W'(i_{\text{reg}})) \in W' \cdot R[\chi[r_d : \tau]] \rho$.

**Proof**
1. Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}(W'(i_{\text{reg}}))$ such that $\tilde{W} \sqsupseteq W'$. We must show that $(\tilde{W}, M_1, M_2) \in R[\chi] \rho$.
   Instantiate the first premise with $(\tilde{W}, M_1, M_2)$, noting that the latter is in $\text{currentMR}(W(i_{\text{reg}}))$ since $W'(i_{\text{reg}}) = W(i_{\text{reg}})$, and noting that $\tilde{W} \sqsupseteq W$ by transitivity of $\sqsupseteq$. Hence, $(\tilde{W}, M_1, M_2) \in R[\chi] \rho$ as we needed to show.
2. Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}(W'(i_{\text{reg}}))$ such that $\tilde{W} \sqsupseteq W'$. We must show that $(\tilde{W}, M_1, M_2) \in R[\chi] \rho$. Note that $M_i$ must be of the form $R_i \cdot$ and $R_i = W'.R_i$.
   Consider arbitrary $(\tau : \tau') \in \chi[r_d : \tau]$. We must show that $(\tilde{W}, R_1(\tau), R_2(\tau)) \in W[\tau'] \rho$. 

107
Lemma 3.23 (World Updates that Respect Stack Relation)

Let currentMR(W(i_{stk})) \in W.S[\sigma]\rho and W' \supseteq W.

1. If W'(i_{stk}) = W(i_{stk}), then currentMR(W'(i_{stk})) \in W'. S[\sigma]\rho.

2. Let W.S_1 = w_{11} : \cdots : w_{1n} : S'_1, W.S_2 = w_{21} : \cdots : w_{2n} : S'_2, \sigma = \tau_1 : \cdots : \tau_n : \sigma' and W = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{stk} \mapsto island_{reg}(s, W.k)]), \ s = (S'_1, \sigma', S'_2, \sigma'). currentMR(W'(i_{stk})) \in W'. S[\sigma']\rho.

3. Let W' = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{stk} \mapsto island_{reg}(s, W.k)]), \ s = (w_{11} : \cdots : w_{1n} : S_1, \tau_1 : \cdots : \tau_n : \sigma, w_{21} : \cdots : w_{2n} : S_2, \tau_1 : \cdots : \tau_n : \sigma). If (W', w_{11}, w_{21}) \in W[\tau_1]\rho, then currentMR(W'(i_{stk})) \in W'. S[\tau_1 : \cdots : \tau_n : \sigma]\rho.

4. Let W.S_1 = w_{11} : \cdots : w_{1n} : S'_1, W.S_2 = w_{21} : \cdots : w_{2n} : S'_2, \sigma = \tau_1 : \cdots : \tau_n : \sigma' and W = (W.k, W.\Psi_1, W.\Psi_2, W.\Theta[i_{stk} \mapsto island_{reg}(s, W.k)]), \ s = (w_{11} : \cdots : w_{1n-1} : w'_1 : S'_1, \tau_1 : \cdots : \tau_{n-1} : \tau' : \sigma', w_{21} : \cdots : w_{2n-1} : w'_2 : S'_2, \tau_1 : \cdots : \tau_{n-1} : \tau' : \sigma'). If (W', w'_1, w'_2) \in W[\tau_1]\rho, then currentMR(W'(i_{stk})) \in W'. S[\tau_1 : \cdots : \tau_n : \tau' : \sigma']\rho.

Proof

1. Consider arbitrary (\tilde{W}, M_1, M_2) \in currentMR(W'(i_{stk})) such that \tilde{W} \supseteq W'. We must show that (\tilde{W}, M_1, M_2) \in S[\sigma]\rho.

   Instantiate the first premise with (\tilde{W}, M_1, M_2), noting that the latter is in currentMR(W(i_{stk})) since W'(i_{stk}) = W(i_{stk}), and noting that \tilde{W} \supseteq W by transitivity of \supseteq. Hence, (\tilde{W}, M_1, M_2) \in S[\sigma]\rho as we needed to show.

2. Consider arbitrary (\tilde{W}, M_1, M_2) \in currentMR(W'(i_{stk})) such that \tilde{W} \supseteq W'. We must show that (\tilde{W}, M_1, M_2) \in S[\sigma]\rho. Note that M_i must be of the form S'_i and S'_i = W'.S_i. The desired result follows directly from monotonicity (Lemma 3.8) after unfolding currentMR(W'(i_{stk})) \in W.S[\sigma]\rho.

3. Consider arbitrary (\tilde{W}, M_1, M_2) \in currentMR(W'(i_{stk})) such that \tilde{W} \supseteq W'. We must show that (\tilde{W}, M_1, M_2) \in S[\tau_1 : \cdots : \tau_n : \sigma]\rho. Note that M_i must be of the form w_{j1} : \cdots : w_{jn} : S_j] and w_{j1} : \cdots : w_{jn} : S_j = W'.S_j. The desired result follows directly from monotonicity (Lemma 3.8) after unfolding currentMR(W(i_{stk})) \in W.S[\tau_1 : \cdots : \tau_n : \sigma]\rho and premise (W', w_{11}, w_{21}) \in W[\tau_1]\rho.

4. Let \sigma' = \tau_1 : \cdots : \tau_n : \sigma. Consider arbitrary 1 \leq k \leq |\sigma'| such that such that (\sigma'(k) = \tau_k. We must show that (\tilde{W}, W'.S_1(k), W'.S_2(k)) \in W[\tau_k]\rho.

   Case k = n: Then \tau' = \tau_k and W'.S_j(k) = w'_j, which means that it suffices to show (\tilde{W}, w'_1, w'_2) \in W[\tau']\rho. This latter is immediate from the premise (W', w_1, w_2) \in W[\tau]\rho using monotonicity (Lemma 3.8).

   Case k \neq n: Then W'.S_j(k) = W.S_j(k), which means that it suffices to show (\tilde{W}, W.S_1(k), W.S_2(k)) \in W[\tau_k]\rho. Instantiate the first premise with (\tilde{W}, W.S_1(k), W.S_2(k)) noting that the latter is in currentMR(W(i_{reg})) by the definition of island_{reg}. Also note that \tilde{W} \supseteq W by transitivity of \supseteq. Hence, we have that (\tilde{W}, W.S_1(k), W.S_2(k)) \in R[\sigma]\rho. Instantiating the latter with (\tau_k) = \sigma(k) gives us (\tilde{W}, W.S_1(k), W.S_2(k)) \in W[\tau_k]\rho as we needed to show.
Lemma 3.24 (Heap Interpretation Extension with Boxheap)
If $W \in \mathcal{H}[\Psi]$, $\Psi \vdash H_1 \cong H_2 : \Psi'$, and boxheap($\Psi'$), then $W \boxplus (H_1, H_2) \in \mathcal{H}[\Psi, \Psi']$.

Proof

By induction on $W.k$.

To show $W \boxplus (H_1, H_2) \in \mathcal{H}[\Psi, \Psi']$, it suffices to show

$$\forall (\ell : \nu \psi) \in (\Psi, \Psi'). (W \boxplus (H_1, H_2), \ell, \ell) \in \mathcal{H}[\nu \psi]$$

where $\nu$ is box or ref.

Consider arbitrary $(\ell : \nu \psi) \in (\Psi, \Psi')$. If $\ell \in \text{dom}(\Psi)$, for any value of $W.k$, it follows from the first premise that $(W, \ell, \ell) \in \mathcal{H}[\nu \psi]$. By Lemma 3.5 we have that $W \boxplus (H_1, H_2) \supseteq W$, so by monotonicity we have $(W \boxplus (H_1, H_2), \ell, \ell) \in \mathcal{H}[\nu \psi]$.

Therefore, it remains for us to show that if $\ell \in \text{dom}(\Psi')$ then $(W \boxplus (H_1, H_2), \ell, \ell) \in \mathcal{H}[\text{box} \psi]$.

Case $W.k = 0$: Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}((W \boxplus (H_1, H_2))(i_{\text{box}}))$ such that $\tilde{W} \supseteq (W \boxplus (H_1, H_2)).$ But the latter implies $\tilde{W}.k < W.k = 0$ which leads to a contradiction since $\tilde{W} \in \text{World}$ which requires that $\tilde{W}.k \geq 0$. So we are done.

Case $W.k = n + 1$ for $n \geq 0$: By the induction hypothesis we know that the lemma we wish to prove holds for any $W$ such that $W.k = n$. We must prove it for any $W$ such that $W.k = n + 1$.

We have that $\ell \in \text{dom}(\Psi')$ and must show that $(W \boxplus (H_1, H_2), \ell, \ell) \in \mathcal{H}[\text{box} \psi]$.

Consider arbitrary $(\tilde{W}, M_1, M_2) \in \text{currentMR}((W \boxplus (H_1, H_2))(i_{\text{box}}))$ such that $\tilde{W} \supseteq (W \boxplus (H_1, H_2)).$ Note that it must be the case that $M_1 = (W(i_{\text{box}}).s.H_1)\mid$ and $M_2 = (W(i_{\text{box}}).s.H_2)\mid$. Also, since $\ell \in \text{dom}(\Psi')$, from the second premise it follows that $\ell \in \text{dom}(H_1)$ and $\ell \in \text{dom}(H_2)$.

Regardless of whether $\psi$ is a code type or tuple type, it suffices to show:

$$(\tilde{W}, M_1(\ell), M_2(\ell)) \in \mathcal{H}[\nu \psi] \subseteq (\tilde{W}, H_1(\ell), H_2(\ell)) \in \mathcal{H}[\nu \psi]$$

From the second premise, since $(\ell : \text{box} \psi) \in \Psi'$, it follows that:

$$\Psi, \Psi' \vdash H_1(\ell) \cong_{hv} H_2(\ell) : \text{box} \psi. \quad (20)$$

Since $\triangleright W \in \mathcal{H}[\Psi]$ by heap monotonicity (Lemma 3.9) and since $(\triangleright W).k = n$, by the induction hypothesis we have that $(\triangleright W \boxplus (H_1, H_2)) \in \mathcal{H}[\Psi, \Psi']$. Thus, we can instantiate (20) with $(\triangleright W \boxplus (H_1, H_2)$, which allows us to conclude that

$$(\triangleright W \boxplus (H_1, H_2), H_1(\ell), H_2(\ell)) \in \mathcal{H}[\nu \psi]$$

Now, since $\tilde{W} \supseteq (W \boxplus (H_1, H_2))$, we can use by Lemma 3.6 to conclude $\tilde{W} \supseteq (\triangleright W \boxplus (H_1, H_2))$. Hence, by monotonicity, we have $(\tilde{W}, H_1(\ell), H_2(\ell)) \in \mathcal{H}[\nu \psi]$ as we needed to show. \qed
3.5 Bridge Lemmas

Lemma 3.25 (F Values are F Expressions)
If \((W, v_1, v_2) \in \mathcal{V}[\tau]_\rho\) and currentMR\((W_i_{stk})\) \(\subseteq W\) \(S[\sigma]_\rho\) then \((W, v_1, v_2) \in \mathcal{E}[out \vdash \tau; \sigma]_\rho\).

Proof
Consider arbitrary \(E_1, E_2\) such that \((W, E_1, E_2) \in \mathcal{K}[out \vdash \tau; \sigma]_\rho\).
We need to show that \((W, E_1[v_1], E_2[v_2]) \in \mathcal{O}\). But we can instantiate the \(\mathcal{K}[\cdot]\) relation with our hypotheses, noting \(W \models_{\text{pub}} W\) by reflexivity, to get the result.

Lemma 3.26 (Monad Bind F To F)
If \((W, e_1, e_2) \in \mathcal{E}[out \vdash \tau; \sigma]_\rho\), \((W, E_1, E_2) \in \text{ContAtom}[out \vdash \tau; \sigma]_\rho \hookrightarrow [out \vdash \tau^*; \sigma^*]_\rho\), and \(\forall W' \models_{\text{pub}} W.\forall v_1, v_2. W'.(W', v_1, v_2) \in \mathcal{V}[\tau]_\rho \land \text{currentMR}(W'(i_{stk})) \subseteq W'. S[\sigma]_\rho\)

\(\Rightarrow (W', E_1[v_1], E_2[v_2]) \in \mathcal{E}[out \vdash \tau^*; \sigma^*]_\rho\), then \((W, E_1[e_1], E_2[e_2]) \in \mathcal{E}[out \vdash \tau^*; \sigma^*]_\rho^*\).

Proof
Consider arbitrary \(E'_1, E'_2\) such that \((W, E'_1, E'_2) \in \mathcal{K}[out \vdash \tau^*; \sigma^*]_\rho^*\).
It suffices to show that \((W, E'_1[E_1], E'_2[E_2]) \in \mathcal{K}[out \vdash \tau; \sigma]_\rho\).
In order to do that, consider arbitrary \(W', v_1, v_2\) where \(W' \models_{\text{pub}} W\), \((W', v_1, v_2) \in \mathcal{V}[\tau]_\rho\) and currentMR\((W'(i_{stk})) \subseteq W'. S[\sigma]_\rho\). We need to show that \((W', E'_1[v_1], E'_2[v_2]) \in \mathcal{O}\).
From the hypothesis, we know that \((W', E'_1[v_1], E'_2[v_2]) \in \mathcal{E}[out \vdash \tau^*; \sigma^*]_\rho^*\). We can then instantiate that with \((W', E'_1, E'_2)\), which, appealing to Lemma 3.10, we know are in \(\mathcal{K}[out \vdash \tau^*; \sigma^*]_\rho^*\), which yields the result that we need.

Lemma 3.27 (Monadic Bind T to F)
If \((W, e_1, e_2) \in \mathcal{E}[\text{end}\{\tau^T; \sigma\} \vdash \tau^T; \sigma]_\rho\), \((W, E_1, E_2) \in \text{ContAtom}[\text{end}\{\tau^T; \sigma\} \vdash \tau^T; \sigma]_\rho \hookrightarrow [out \vdash \tau; \sigma_1]_\rho\),
and \(\forall W' \models_{\text{pub}} W.\forall r_1, r_2. (\langle W', W', R_1(r_1), W'.R_2(r_2) \rangle) \in W[\tau^T]_\rho \land \text{currentMR}(W'(i_{stk})) \subseteq W'. S[\sigma]_\rho\)

\(\Rightarrow (W', E_1[\text{ret end}\{\rho_1(\tau^T); \sigma\} \{r_1\}], E_2[\text{ret end}\{\rho_2(\tau^T); \sigma\} \{r_2\}] \in \mathcal{E}[out \vdash \tau; \sigma_1]_\rho\), then \((W, E_1[e_1], E_2[e_2]) \in \mathcal{E}[out \vdash \tau; \sigma_1]_\rho\).

Proof
Consider arbitrary \(E'_1, E'_2\) such that \((W, E'_1, E'_2) \in \mathcal{K}[out \vdash \tau; \sigma_1]_\rho\).
It suffices to show that \((W, E'_1[E_1], E'_2[E_2]) \in \mathcal{K}[\text{end}\{\tau^T; \sigma\} \vdash \tau^T; \sigma]_\rho\).
In order to do that, consider arbitrary \(W', q', r_1, r_2\) such that the following hold, noting that based on the form of the return marker, \(q' = \rho \text{ end}\{\tau^T; \sigma\}\), rather than possibly being a register.

- \(W' \models_{\text{pub}} W\)
- \((W', E'_1[E_1], E'_2[E_2]) \in \text{ContAtom}[q' \vdash \tau^T; \sigma]_\rho \hookrightarrow [q^* \vdash \tau^*; \sigma^*]_\rho^*\)
- \(q' = \rho \text{ end}\{\tau^T; \sigma\}\)
- \(\langle W', W'.R_1(r_1), W'.R_2(r_2) \rangle) \in W[\tau^T]_\rho\)
- \(\text{currentMR}(W(i_{stk})) \subseteq W. S[\sigma]_\rho\)

Given the above, we must show that \((\langle W, (E_1[\text{ret}\rho_1(q') \{r_1\}], \cdot),(E_2[\text{ret}\rho_2(q') \{r_2\}], \cdot) \rangle \in \mathcal{O}\).
From the hypothesis, we know that \((W', E_1[\text{ret}\rho_1(q') \{r_1\}], E_2[\text{ret}\rho_2(q') \{r_2\}] \in \mathcal{E}[out \vdash \tau; \sigma_1]_\rho\)
We can then instantiate that with \((W',E_1',E_2')\), which, appealing to Lemma 3.10, we know are in \(\mathcal{K}[\out \vdash \tau;\sigma_1]\), which yields the result that we need.

\[\]

Lemma 3.28 (Bridge Lemma)

Let \(\rho \in \mathcal{D}[\Delta]\) and \(\Delta \vdash \tau\).

1. (a) If \((W,e_1,e_2) \in \mathcal{E}[\end{\tau;\sigma}] \vdash \tau;\sigma\rho\) then \((W;\rho_1(\tau)\FT e_1,\rho_2(\tau)\FT e_2) \in \mathcal{E}[\out \vdash \tau;\sigma_1]\rho\).

(b) If the following hold
   - \(\forall SR \in \text{TStackRel}(W,e_1,e_2) \in \mathcal{E}[\out \vdash \tau;\tau' :: \cdots :: \tau'_k :: \zeta] \rho[\zeta \mapsto SR]\)
   - \(\Delta \vdash \sigma\)
   - \(\sigma' = \tau_0' :: \cdots :: \tau'_k :: \sigma_0\)
   - \(q = i \text{ or } q = \text{end}\{\hat{\tau};\hat{\sigma}\}\)
   - \(W \in \mathcal{H}[\Psi]\)
   - \(\text{currentMR}(W(i_{stk})) \in \mathcal{W}.\mathcal{S}[\sigma']\rho\)
   - \(\Psi;\Delta;\Gamma;\chi[x_d : \tau];\sigma';\text{inc}(q,k-j) \vdash I_1 \approx I_2\)
   - \(\text{ret-type}(q,\chi,\sigma) = \hat{\tau};\hat{\sigma}\)

then \((W,\text{import }r_d,\rho_1(\sigma_0)\FT \rho_1(I_1),\text{import }r_d,\rho_2(\sigma_0)\FT \rho_2(I_2)) \in \mathcal{E}[q \vdash \hat{\tau};\hat{\sigma}]\rho\).

2. (a) If the following hold
   - \((W,w_1,w_2) \in \mathcal{W}[\tau]\rho\)
   - \((M_1,M_2) : W\)
   - \(\rho_1(\tau)\FT((w_1,M_1)) = (v_1,M_1 \uplus M'_1)\)
   - \(\rho_2(\tau)\FT((w_2,M_2)) = (v_2,M_2 \uplus M'_2)\)

then \((W \uplus (M'_1,M'_2),w_1,w_2) \in \mathcal{W}[\tau]\rho\)

(b) If the following hold
   - \((W,v_1,v_2) \in \mathcal{V}[\tau]\rho\)
   - \((M_1,M_2) : W\)
   - \(\text{TF}_\rho_1(\tau)((v_1,M_1)) = (w_1,M_1 \uplus M'_1)\)
   - \(\text{TF}_\rho_2(\tau)((v_2,M_2)) = (w_2,M_2 \uplus M'_2)\)

then \((W \uplus (M'_1,M'_2),w_1,w_2) \in \mathcal{W}[\tau]\rho\)

Proof

1. (a) Appeal to Lemma 3.27, letting \(E_i = \rho_i(\tau)\FT [\cdot]\).

From 3.27 we are given registers \(r_1,r_2\), and world \(W' \equiv_{\text{pub}} W\) such that
   - \((>W',W'.R_1(r_1),W'.R_2(r_2)) \in \mathcal{W}[\tau]\rho\)
   - \(\text{currentMR}(W'(i_{stk})) \in \mathcal{W}.\mathcal{S}[\sigma]\rho\)

and need to show that

\((W',\rho_1(\tau)\FT \text{ret end}\{\rho_1(\tau)\sigma\} \{r_1\},\rho_2(\tau)\FT \text{ret end}\{\rho_2(\tau)\sigma\} \{r_2\}) \in \mathcal{E}[\out \vdash \tau;\sigma_1]\rho\)

In order to do that, consider arbitrary \(E_1,E_2\) such that \((W,E_1,E_2) \in \mathcal{K}[\out \vdash \tau;\sigma_1]\). We need to show that

\((W,E_1[\rho_1(\tau)\FT \text{ret end}\{\rho_1(\tau)\sigma\} \{r_1\}],E_2[\rho_2(\tau)\FT \text{ret end}\{\rho_2(\tau)\sigma\} \{r_2\}]) \in \mathcal{O}\)

Consider arbitrary \((M_1,M_2) : W\). We take one step.
\[
(M_i | E_1^{[\rho_1(\tau^T); \sigma]} \{r_1\}) \mapsto (M_i \uplus M'_i | E_1[w_i])
\]

where \(\mathcal{FTR}(M_i, r_i, M_i) = (w_i, M_i \uplus M'_i)\)

From part 2(a) we have that \((\triangleright W' \boxplus (M'_1, M'_2), v_1, v_2) \in \mathcal{V}[\tau^T] \rho\), noting from Lemmas 3.4, 3.5, 3.6, that

\[
(M_1, M_2) : \triangleright W' \text{ and } \triangleright W' \boxplus (M'_1, M'_2) \sqsupseteq_{\text{pub}} W'
\]

This means we can instantiate \(E_1, E_2\) to get that

\[
(\triangleright W' \boxplus (M'_1, M'_2), E_1[v_1], E_2[v_2]) \in O
\]

which is sufficient to prove that

\[
(W', E_1^{[\rho_1(\tau^T); \sigma]} \{r_1\}, E_2^{[\rho_2(\tau^T); \sigma]} \{r_2\}) \in O
\]

since each took exactly one reduction step.

(b) We must show that

\[
(W, \text{import } r_d, \rho_1(\sigma_0) \mathcal{F} \mathcal{R} e_1; \rho_1(I_1), \text{import } r_d, \rho_2(\sigma_0) \mathcal{F} \mathcal{R} e_2; \rho_2(I_2)) \in \mathcal{E}[q \vdash \tau; \sigma^T] \rho
\]

In order to do that, consider arbitrary \(E_1, E_2\), such that \((W, E_1, E_2) \in \mathcal{K}[q \vdash \tau; \sigma^T] \rho\).

We are required to show that

\[
(W, E_1^{[\rho_1(\sigma_0)] \mathcal{F} \mathcal{R} e_1; \rho_1(I_1)}, E_2^{[\rho_2(\sigma_0)] \mathcal{F} \mathcal{R} e_2; \rho_2(I_2)}) \in O
\]

Consider arbitrary \((M_1, M_2) : W\). From the premise, we know that there exist \(S_1\) and \(S_2\) such that \((W, S_1 \uplus S_2) \in \mathcal{S}[\sigma^T] \rho\).

Since \(\sigma' = \sigma_0 : \cdots : \sigma_2 : \sigma_0\), there exist \(S_{10}\) and \(S_{20}\) such that \((W, S_{10}, S_{20}) \in \mathcal{S}[\sigma_0]\).

We can instantiate the first premise with \(SR = (\rho_1(\sigma_0), \rho_2(\sigma_0), \phi_S)\) where \(\phi_S = \{(\tilde{W}, S_{10}, S_{20}) | \tilde{W} \supseteq W\}\).

Let \(\rho^* = \rho[\zeta \mapsto SR]\). Hence, \((W, e_1, e_2) \in \mathcal{E}[\text{out } \vdash \tau; \sigma^T] \rho^*\).

We can instantiate this with evaluation contexts

- \(E'_1 = E_1^{[\text{import } r_d, \rho_1(\sigma_0)] \mathcal{F} \mathcal{R} e_1; I_1}\)
- \(E'_2 = E_2^{[\text{import } r_d, \rho_2(\sigma_0)] \mathcal{F} \mathcal{R} e_2; I_2}\)

It now suffices to show that

\[
(W, E'_1, E'_2) \in \mathcal{K}[\text{out } \vdash \tau; \sigma^T] \rho^*\]

Consider arbitrary \(W', v_1, v_2\) such that \((W', v_1, v_2) \in \mathcal{V}[\tau^T] \rho^*, \text{currentMR}(W'(i_{\text{stk}})) \sqsubseteq_{W', S}[\sigma^T] \rho^*\).

We need to show that

\[
(W', E'_1[v_1], E'_2[v_2]) = (W', E_1^{[\text{import } r_d, \rho_1(\sigma_0)] \mathcal{F} \mathcal{R} v_1; I_1}, E_2^{[\text{import } r_d, \rho_2(\sigma_0)] \mathcal{F} \mathcal{R} v_2; I_2}) \in O
\]

Consider arbitrary \((M_1, M_2) : W'\). We first take one step

\[
(M_i | E_i^{[\text{import } r_d, \rho_1(\sigma_0)] \mathcal{F} \mathcal{R} v_i; I_i}) \mapsto (M_i \uplus M'_i | E_i[w_i, r_d; I_i])
\]

where \(\mathcal{FTR}(v_i, M_i) = (w_i, M_i \uplus M'_i)\) Noting from part 2(b), we have that \((\triangleright W' \boxplus (M'_1, M'_2), w_1, w_2) \in \mathcal{V}[\tau^T] \rho^*\), and from Lemma 3.5, \(W' \boxplus (M'_1, M'_2) \sqsupseteq_{\text{pub}} W'\).

Then we take another step

\[
(M_i \uplus M'_i | E_i[w_i, r_d; (w_1, w_2)] | I_i) \mapsto (M_i \uplus M'_i[w_i; (w_1, w_2)])
\]

112
We now apply our hypothesis about \( \textbf{I}_1 \), noting that the results above and choice of \( \rho^* \) fulfill the requirements for the stack and heap.

We can now instantiate the resulting \( \mathcal{E}[\cdot] \) with \( E_1, E_2 \), which gives us that

\[
(W' \uplus M_i \uplus M'_i[r_d \rightarrow (w_1, w_2)], E_1[I_1], E_2[I_2]) \in \mathcal{O}
\]

from which the result follows, since in

\[
(W', E_1[\text{import } r_d, \rho_1^*(\sigma_0) \mathcal{T} \mathcal{F}_\tau v_1; I_1], E_2[\text{import } r_d, \rho_2^*(\sigma_0) \mathcal{T} \mathcal{F}_\tau v_2; I_2]) \in \mathcal{O}
\]

each took exactly two steps to reach the above.

2. (a) Proceed by induction first on the step index \( W.k \) and then on the structure of \( \tau \).

\textbf{Case} unit Follows trivially from the value translation and \( W[\text{unit}] \rho \) and \( V[\text{unit}] \rho \).

\textbf{Case} int Follows trivially from the value translation and \( W[\text{int}] \rho \) and \( V[\text{int}] \rho \).

\textbf{Case} \( \mu \alpha.\tau \) We're given that

\[
(W, \text{fold} \mu \alpha.\tau \ w_1, \text{fold} \mu \alpha.\tau \ w_2) \in W[\mu \alpha.\tau] \rho
\]

which is translated to \( \text{fold} \mu \alpha.\tau \ v_i, M_i \uplus M'_i \).

From this, we know that

\[
(W, w_1, w_2) \in W[\tau] \rho
\]

Since we are inducting on the step index \( W.k \), we can use this, along with the value translations that we have from the hypothesis to get that

\[
(W \uplus (M'_1, M'_2), v_i, v_2) \in V[\tau] \rho
\]

which, when combined with the typing rules, yields the required result.

\textbf{Case} \( \langle \tau_1, \ldots, \tau_n \rangle \) From the definition of \( W[\text{box} \langle \tau_1, \ldots, \tau_n \rangle] \rho \) we know that

\[
(W, M_1(w_1), M_2(w_2)) \in V[\langle \tau_1, \ldots, \tau_n \rangle] \rho
\]

where \( M_i(w_i) = (w_{i1}, \ldots, w_{in}) \).

From the definition of \( V[\cdot] \rho \) we know that

\[
(W, w_{1j}, w_{2j}) \in W[\tau_j] \rho
\]

By the value translation, \( \rho^*(\tau_j) \mathcal{F}_T((w_{ij}, M_{ij})) = (v_{ij}, M_{ij} \uplus M'_{ij}) \), where

- \( M_{10} = M \)
- \( M_{ij} = M_{i(j-1)} \uplus M'_{ij(j-1)} \)
- \( M'_{ij} = \cup_j M'_{ij} \)

We know that \( (M_{1n}, M_{2n}) : W \) since \( (M_{10}, M_{20}) : W \) and the only changes to the memory are additions.

We can use the inductive hypothesis to conclude that

\[
(W \uplus (M'_1, M'_2), v_i, v_2) \in V[\tau_j] \rho
\]

which combined with the typing rules, yields what we need.

\textbf{Case} \( \tau \rightarrow \tau' \) Given some \( W' \equiv W \uplus (\ell_{\text{end}} \mapsto h_{\text{end}}, \ell_{\text{end}} \mapsto h_{\text{end}}) \), where

- \( h_{\text{end}} = \text{code}[\zeta \{r_1; \tau' \tau \}; \zeta \{r_1\} \text{end} \{\tau' \tau \}; \zeta \} \text{.ret} \text{end} \{\tau' \tau \}; \zeta \} \{r_1\} \}
- \( SR \in \text{TStackRel} \)
- \( \rho' = \rho[\zeta \mapsto SR]\)
- \( \text{currentMR}(W'(i_{\text{stk}})) \subset W', S[\zeta] \rho' \)
- \( (W', v_{1j}, v_{2j}) \in V[\tau_j] \rho' \) for \( j \in \{1..n\} \)
we need to show that

\[ (W', v_1v_{i0} \ldots v_{1n}, v_2v_{20} \ldots v_{2n}) \in E[\text{out} \vdash \tau'; \zeta] \rho \]

where

\[ v_i = \lambda(x_{n_i}; \tau_{n_i}).\tau_i^F T (\text{protect }, \zeta; \text{import } r_1, \zeta T F \tau \tau_1 x_1; \text{salloc } 1; \text{sst } 0, r_1; \ldots; \]

\[ \text{import } r_1, \zeta T F \tau v_{in}; \text{salloc } 1; \text{sst } 0, r_1; \]

\[ \text{mv ra}, \epsilon_{\text{end}[\zeta]}; \text{jmp } w_1[\zeta](\text{end}\{\tau'/T; \zeta\}), \cdots \]

We proceed by appealing to Lemma 3.4.3.

Consider arbitrary \((M_1, M_2) : W'\). We first take one step

\[ \langle M_1 | v_1v_{i0} \ldots v_{in} \rangle \longrightarrow \langle M_1 | \tau_i F T (\text{protect }, \zeta; \text{import } r_1, \zeta T F \tau \tau_1 v_{i1}; \rangle \text{salloc } 1; \text{sst } 0, r_1; \ldots; \]

\[ \text{import } r_1, \zeta T F \tau v_{in}; \text{salloc } 1; \text{sst } 0, r_1; \]

\[ \text{mv ra}, \epsilon_{\text{end}[\zeta]}; \text{jmp } w_{i1}[\zeta](\text{end}\{\tau'/T; \zeta\}), \cdots \]

To show that this is in \(E[\text{out} \vdash \tau'; \zeta]\), it suffices to show that what is within the boundary is in \(\mathcal{E}[\text{end}\{\tau'/T; \zeta\} \vdash \tau'/T; \zeta]\), as part 1(a) will then yield the result.

We now appeal to Lemma 3.4.3 again, noting we can take \(3n + 1\) steps:

\[ \langle M_1 | (\text{protect }, \zeta; \text{import } r_1, \zeta T F \tau \tau_1 v_{i1}; \rangle \longrightarrow 3^n \langle M_i^* | (\text{mv ra}, \epsilon_{\text{end}[\zeta]}; \text{jmp } w_{i1}[\zeta](\text{end}\{\tau'/T; \zeta\}), \cdots \rangle \]

where

- \(M_{i0a} = M_i\)
- \(T_i^T(\nu_j, M_{ijb}) = (w_{ij}, M_{ijb})\)
- \(M_{ijc} = M_{ijb}[r_1 \mapsto w_{ij}]\)
- \(M_{ijd} = M_{ijc}() ::\)
- \(M_{ije} = M_{ijc}[w_{ij} ::]\)
- \(M_i[1, j+1] = M_{ije}\)
- \(M_i^* = M_{in}\)

We take one more step, resulting in:

\[ \langle M_i^* [ra \mapsto \epsilon_{\text{end}[\zeta]}]; (\text{jmp } w_{i1}[\zeta](\text{end}\{\tau'/T; \zeta\}), \cdots) \rangle \]

We know

\[ (W, w_1, w_2) \in \mathcal{W}[\text{box } \forall [\zeta, \epsilon]. [ra : \forall[]. [r_1 : \tau'/T; \zeta]^a; \sigma']^a] \rho \]

where \(\sigma' = \sigma' = \tau_1 T :: \cdots :: \tau_n T :: \zeta\).

This means that

\[ (W, w_1[\zeta](\text{end}\{\tau'/T; \zeta\}), w_{i1}[\zeta](\text{end}\{\tau'/T; \zeta\})) \]

\[ \in \mathcal{W}[\text{box } \forall[]. [ra : \forall[]. [r_1 : \tau'/T; \zeta]^{\text{end}\{\tau'/T; \zeta\}; \sigma']^a] \rho \]

114
And thus \( w_i \) is a location that points to \( \text{code}[\zeta, e]\{\text{ra}: \forall[]; \{r_1 : \tau'/\zeta'\}; \tau\}^{\text{ra}}, I_1 \). This means that the next reduction step results in:

\[
\langle M'_i[^{\text{ra}}]\{\text{ra} \mapsto \ell_{\text{end}}[\zeta] \mid (I_1, \cdot) \rangle.
\]

From \( W\models \Box \forall[], \{\text{ra}: \forall[]; \{r_1 : \tau'/\zeta\}; \tau\}^{\text{ra}}, I_1 \)

we know that

\[
(W, \text{code}[^{\text{ra}}]\{\text{ra}: \forall[]; \{r_1 : \tau'/\zeta\}; \tau\}^{\text{ra}, I_1},
\text{code}[^{\text{ra}}]\{\text{ra}: \forall[]; \{r_1 : \tau'/\zeta\}; \tau\}^{\text{ra}, I_2}) \\
\in \mathcal{H}\forall\forall[], \{\text{ra}: \forall[]; \{r_1 : \tau'/\zeta\}; \tau\}^{\text{ra}}, I_1)\]

This means that for any \( W' \equiv W \), fulfilling the stack and register requirements, in particular, \( W^* = W' \boxplus (M^*_2 \backslash M_1, M^*_1 \backslash M_2) \), we have that \( (W^*, (I_1, \cdot), (I_2, \cdot)) \in \mathcal{E}\forall\Box^{\text{ra}} \tau'/\zeta' ; \tau' \).

Recall that what we need is that the reduced expression is in \( \mathcal{E}\forall\Box^{\text{ra}} \tau'/\zeta' ; \tau' \).

In order to show that, consider arbitrary \( E_1, E_2 \) such that \( (W', E_1, E_2) \in \mathcal{K}\Box^{\text{ra}} \tau'/\zeta' ; \tau' \).

We need to show that \( (W^*, E_1[e_1], E_2[e_2]) \in \mathcal{O} \) where \( e_i \) are the reduced expressions. Note that this is equivalent to showing that \( (W^*, E_1[e_1], E_2[e_2]) \in \mathcal{O} \), since \( W^* \equiv W' \).

Since we know that \( (W^*, e_1, e_2) \in \mathcal{E}\forall\Box^{\text{ra}} \tau'/\zeta' ; \tau' \), we know that for any continuations \( E'_1 \) and \( E'_2 \) in \( \mathcal{K}\Box^{\text{ra}} \tau'/\zeta' ; \tau' \), \( (W^*, E'_1, E'_2) \in \mathcal{O} \).

We argue that \( E_1 \) and \( E_2 \) are suitable choices for \( E'_1 \) and \( E'_2 \). By the definition of the \( \mathcal{K}\Box^{\cdot} \) relation, we need that \( (W^*, E_1[^{\text{ra} \{\text{ret ra} \{r_1\}, \cdot}\}} E_2[^{\text{ra} \{\text{ret ra} \{r_1\}, \cdot}\}}) \in \mathcal{O} \).

Since currentMR(\( W^*(i_{\text{reg}}) \)) \( \in W' \), \( \{\forall[]; \{r_1 : \tau'/\zeta\}; \tau\}^{\text{end}}, \{\tau'/\zeta\} \), this reduces, by the operational semantics, to \( (W^*, E_1[^{\text{I}_1, \cdot}), E_2[^{\text{I}_2, \cdot}) \], \) where \( (W^*, (I_1, \cdot), (I_2, \cdot)) \in \mathcal{E}\forall\Box^{\text{end}} \tau'/\zeta' ; \tau'/\zeta ; \tau' \).

But this exactly means that \( (W^*, E_1[^{\text{I}_1, \cdot}), E_2[^{\text{I}_2, \cdot}) \] \( \in \mathcal{O} \), as \( (W^*, E_1, E_2) \in \mathcal{K}\Box^{\text{end}} \tau'/\zeta' ; \tau'/\zeta ; \tau' \), so we are done.

**Case** \( \overrightarrow{\text{τ}} \overrightarrow{\text{φ}} \overrightarrow{\text{φ}_o} \rightarrow \text{τ}' \)

Given some \( W' \equiv W \boxplus (\ell_{\text{end}} \mapsto h_{\text{end}}, \ell_{\text{end}} \mapsto h_{\text{end}}) \), where

\[
\begin{align*}
\text{h}_{\text{end}} &= \text{code}[\zeta] \{r_1 : \tau'/\zeta; \tau\}^{\text{end}}, \text{ret end} \{\tau'/\zeta \} \{r_1 \} \\
\text{SR} &\in T_{\text{StackRel}} \\
\rho' &= \rho[\zeta \mapsto \text{SR}] \\
\text{currentMR}(W'(i_{\text{stak}})) &\in W'. \text{S}[^{\text{φ}_1 \mapsto \zeta}] \rho' \\
(W', v_{1j}, v_{2j}) &\in \mathcal{V}[\gamma] \rho' \text{ for } j \in \{1..n\}
\end{align*}
\]

we need to show that

\[
(W', v_{1i} v_{1o} .. v_{1n}, v_{2i} v_{2o} .. v_{2n}) \in \mathcal{E}[\text{out} \mapsto \text{τ'}; \text{φ}_o \mapsto \zeta] \rho
\]

where

\[
v_i = \lambda \text{φ}_1 \text{φ}_o (x_{1}; \tau_1). \tau' F T (\text{protect } \text{φ}_1 \zeta; \text{import } r_1, \tau F^n x_1; \text{salloc } l; \text{sst } 0, r_1; \ldots; \text{import } r_1, \tau F^n x_0; \text{salloc } l; \text{sst } 0, r_1;
\]

\[
\text{mv ra}, \ell_{\text{end}}[\text{φ}_o \mapsto \zeta]; \text{jmp } w[\zeta][\text{end} \{\tau'/\zeta \}; \text{φ}_o \mapsto \zeta], \cdot
\]

We proceed by appealing to Lemma 3.4.3.

Consider arbitrary \( (M_1, M_2) : W' \). We first take one step

\[
(M_1 | v_{1i} v_{1o} .. v_{1n}) \rightarrow (M_1 | \tau' F T (\text{protect } \text{φ}_1 \zeta; \text{import } r_1, \tau F^n v_{1i}; \text{salloc } l; \text{sst } 0, r_1; \ldots; \text{import } r_1, \tau F^n v_{1n}; \text{salloc } l; \text{sst } 0, r_1;
\]

\[
\text{mv ra}, \ell_{\text{end}}[\text{φ}_o \mapsto \zeta]; \text{jmp } w[\zeta][\text{end} \{\tau'/\zeta \}; \text{φ}_o \mapsto \zeta], \cdot
\]

115
To show that this is in $E^{\text{out}} \vdash \tau'; \phi_o :: \zeta'$, it suffices to show that what is within the boundary is in

$$E^{\text{end}}\{\tau'; \phi_o :: \zeta\} \vdash \tau'/T; \phi_o :: \zeta$$

as part 1(a) will then yield the result.

We now appeal to Lemma 3.4.3 again, noting we can take $3n + 1$ steps:

$$(M_i \mid (\text{protect } \phi_i, \zeta; \text{import } r_1, \zeta T F w_{ij}; \text{Hallo } 1; \text{sst } 0, r_1; \ldots))$$

$$(\text{import } r_1, \zeta T F w_{in}; \text{Hallo } 1; \text{sst } 0, r_1; \text{mv } ra, \zeta \text{end } [\phi_o :: \zeta]; \text{jmp } w_i[\zeta]\{\text{end } \tau'/T; \phi_o :: \zeta\}, \cdot)$$

$$\rightarrow 3n (M_i^* \mid (\text{mv } ra, \zeta \text{end } [\phi_o :: \zeta]; \text{jmp } w_i[\zeta]\{\text{end } \tau'/T; \phi_o :: \zeta\}, \cdot)$$

Where $M_{i0a} = M_i$

$TF^T(v_{ij}, M_{ija}) = (w_{ij}, M_{ijb})$

$M_{ijc} = M_{ijab}[r_1 \mapsto w_{ij}]$

$M_{ijs} = M_{ije}([]) ::$

$M_{ijes} = M_{ije}[w_{ij} ::]$

$M_{ij(i+1)s} = M_{ije}$

$M_{i}^* = M_{in}$

We take one more step, resulting in:

$$\langle M_i^*|ra \mapsto \zeta \text{end } [\phi_o :: \zeta]\rangle | (\text{jmp } w_i[\zeta]\{\text{end } \tau'/T; \phi_o :: \zeta\}, \cdot)$$

We know

$$(W, w_1, w_2) \in W[\text{box } \forall[\zeta, \epsilon].\{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c; \sigma'\}^{rs}]\rho$$

where $\sigma' = \sigma' = \tau_n T :: \cdots :: \tau_1 T :: \phi_i :: \zeta$.

This means that

$$(W, w_1[\zeta]\{\text{end } \tau'/T; \phi_o :: \zeta\}, w_1[\zeta]\{\text{end } \tau'/T; \phi_o :: \zeta\})$$

$$\in W[\text{box } \forall[] \cdot \{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c\text{end } \tau'/T; \phi_o :: \zeta}; \sigma'\}^{rs}]\rho$$

And thus $w_i$ is a location that points to $\text{code}[\zeta, \epsilon] \{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c; \sigma'\}^{rs}.I_1$.

This means that the next reduction step results in:

$$\langle M_i^*|ra \mapsto \zeta \text{end } [\phi_o :: \zeta]\rangle | (I_1, \cdot)$$

From $W[\text{box } \forall[] \cdot \{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c\text{end } \tau'/T; \phi_o :: \zeta}; \sigma'\}^{rs}]\rho$ we know that

$$(W, \text{code}[\{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c\text{end } \tau'/T; \phi_o :: \zeta}; \sigma'\}^{rs}.I_1,$$

$\text{code}[\{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c\text{end } \tau'/T; \phi_o :: \zeta}; \sigma'\}^{rs}.I_2$$

$$\in W[\forall[] \cdot \{ra: \forall[] \cdot \{r_1: \tau'/T; \phi_o :: \zeta\}^c\text{end } \tau'/T; \phi_o :: \zeta}; \sigma'\}^{rs}]\rho$$

This means that for any $W'' \supseteq W$, fulfilling the stack and register requirements, in particular, $W'' = W'' \boxplus (M_i^* \setminus M_1, M_2^2 \setminus M_2)$, we have that

$$(W'', (I_1, \cdot), (I_2, \cdot)) \in E[ra \mapsto \tau'/T; \sigma']\rho$$

Recall that what we need is that the reduced expression is in
\[ \mathcal{E}\|\text{end}\{\tau'; \phi_\alpha :: \zeta\} \vdash \tau'; \phi_\alpha :: \zeta\|^\rho \]

In order to show that, consider arbitrary \(E_1, E_2\) such that

\[ (W', E_1, E_2) \in \mathcal{K}\|\text{end}\{\tau'; \phi_\alpha :: \zeta\} \vdash \tau'; \phi_\alpha :: \zeta\|^\rho \]

We need to show that \((W', E_1[e_1], E_2[e_2]) \in \mathcal{O}\) where \(e_i\) are the reduced expressions.
Note that this is equivalent to showing that \((W^*, E_1[e_1], E_2[e_2]) \in \mathcal{O}\), since \(W^* \equiv W'\).
Since we know that \((W^*, e_1, e_2) \in \mathcal{E}\|\text{ra} \vdash \tau'; \phi_\sigma \|^\rho\), we know that for any continuations \(E_1' \text{ and } E_2' \in \mathcal{K}\|\text{ra} \vdash \tau'; \phi_\sigma \|^\rho\), \((W^*, E_1', E_2') \in \mathcal{O}\).
We argue that \(E_1\) and \(E_2\) are suitable choices for \(E_1'\) and \(E_2'\). By the definition of the \(\mathcal{K}\|\cdot\) relation, we need that

\[ (W^*, E_1[(\text{ret ra } \{r_1\}, \cdot)], E_2[(\text{ret ra } \{r_1\}, \cdot)]) \in \mathcal{O} \]

Since

\[ \text{currentMR}(W^*(i_{\text{reg}})) \in W_\ast \{ \text{ra} : \forall \emptyset, \{x_1 : \tau'; \phi_\alpha :: \zeta\} \text{end}\{\tau'; \phi_\alpha :: \zeta\} \}
\]

this reduces, by the operational semantics, to \((W^*, E_1[I_1', \cdot]), E_2[I_2', \cdot])\), where

\[ (W^*, (I_1', \cdot), (I_2', \cdot)) \in \mathcal{E}\|\text{end}\{\tau'; \phi_\alpha :: \zeta\} \vdash \tau'; \phi_\alpha :: \zeta\|^\rho \]

But this exactly means that

\[ (W^*, E_1[(I_1', \cdot)], E_2[(I_2', \cdot)]) \in \mathcal{O} \]

as

\[ (W^*, E_1, E_2) \in \mathcal{K}\|\text{end}\{\tau'; \phi_\alpha :: \zeta\} \vdash \tau'; \phi_\alpha :: \zeta\|^\rho \]

so we are done.

(b) Proceed by induction first on the step index \(W.k\) and then on the structure of \(\tau\).

Case unit Follows trivially from the value translation and the definitions of \(V\|\text{unit}\|^\rho\) and \(V\|\text{unit}\|^\rho\).

Case int Follows trivially from the value translation and the definitions of \(V\|\text{int}\|^\rho\) and \(V\|\text{int}\|^\rho\).

Case \(\mu\alpha.\tau\)
This case proceeds analogously to the \(\mu\alpha.\tau\) case in the other direction.

Case \(\langle \tau_1, \ldots, \tau_n \rangle\) To show that that \((W, \ell_1, \ell_2)\) is in

\[ \mathcal{V}\|\langle \tau_1, \ldots, \tau_n \rangle \|^\rho = \mathcal{V}\|\text{box } \langle \tau_1 \tau, \ldots, \tau_n \tau \rangle \|^\rho \]

we need to show that

\[ (W, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{H}\|\langle \tau_1 \tau, \ldots, \tau_n \tau \rangle \|^\rho \]

for any \(M_i\) satisfying the box memory relation.
This, in turn, requires that for each \(j, (W, w_{ji}, w_{2j}) \in \mathcal{V}\|\tau_j \|^\rho\).
We’re given that

\[ (W, (v_{10}, \ldots, v_{1n}), (v_{10}, \ldots, v_{1n})) \in \mathcal{V}\|\langle \tau_1, \ldots, \tau_n \rangle \|^\rho \]

which means that \((W, v_{ji}, v_{2j}) \in \mathcal{V}\|\tau_j \|^\rho\).
From the value translation, we know that

- \(T\|^{\rho}(\tau_j)(v_{ji}, M_{ij}) = (w_{ij})\)
- \(M_{ij} \uplus M'_{ij}\)

117
where

- \( M_{i0} = M_i \)
- \( M_{ij} = M_{i(j-1)} \uplus M'_{i(j-1)} \)
- \( M'_{i} = \cup j M'_{ij} \)

This follows from the fact that

\[
\text{TF}^{\rho_i((\tau_1, \ldots, \tau_n))}(w_i, M_i) = (w_i, M_i \uplus M'_i)
\]

Since \((M_{1j}, M_{2j}) : W_i\), as they are extensions of \(M_1\) and \(M_2\), we can use the induction hypotheses and conclude that \((W \uplus (M'_1, M'_2), w_{1j}, w_{2j}) \in W[\tau]^T\rho\). This, combined with the value translation placing these values in a tuple on the heap, yields the required result.

**Case \((\tau) \rightarrow \tau'\)**

To show that that \((W, \ell_1, \ell_2)\) is in

\[
W[\tau] \rightarrow \tau'^T\| \rho = W[\text{box } \forall \zeta, \epsilon, \{\text{ra: box } \forall[], \{\text{r1: } \tau'^T; \zeta\}^c; \sigma'^*\}^\tau]\rho
\]

where \(\sigma' = \tau_n^T : \cdots : \tau_1^T \vdash \zeta\), we need to show that for any future world \(\tilde{W} \supseteq W\), and any memories \(M_1, M_2\) such that \((\tilde{W}, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))\), the following conditions hold:

- \(M_{i}(\ell_1) = \text{code}[\zeta, \epsilon, \{\text{ra: } \forall[], \{\text{r1: } \rho_1(\tau'^T); \zeta\}^c; \rho_1(\sigma') \}^\tau, I_i\]
- \((\tilde{W}, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{H}[\forall \zeta, \epsilon, \{\text{ra: } \forall[], \{\text{r1: } \tau'^T; \zeta\}^c; \sigma'^*\}^\tau]\rho\)

The latter requires that, in a further future world \(W' \supseteq \tilde{W}\), with \(\rho^* \in \mathcal{D}[\zeta, \epsilon]\), letting \(\rho' = \rho \cup \rho^*\), such that the world fulfills the following restrictions:

- \(\text{currentMR}(W'(i_{\text{reg}})) \in W' \mathcal{R}[\text{ra: } \forall[], \{\text{r1: } \rho_1(\tau'^T); \zeta\}^c]\rho'\)
- \(\text{currentMR}(W'(i_{\text{stk}})) \in W' \mathcal{S}[\sigma']\rho'\)

We must show that

\[
(W', (\rho_1^*(I_1), \cdot), (\rho_2^*(I_2), \cdot)) \in \mathcal{E}[\text{ra } \vdash \tau'^T; \zeta]\rho'
\]

Where \(I_1\) and \(I_2\) are defined by the value translation.

In order to do that, consider arbitrary \(E_1, E_2\) such that

\[
(W', E_1, E_2) \in \mathcal{K}[\text{ra } \vdash \tau'^T; \zeta]\rho'
\]

The definition of \(\mathcal{E}\) dictates we show that

\[
(W', E_1[\rho_1^*(I_1), \cdot], E_2[\rho_2^*(I_2), \cdot]) \in \mathcal{O}
\]

Expanding \(I_i\), we are considering:

\[
E_i [(\text{salloc 1; sst 0, ra; import r1, } \rho_i^*(\zeta)) \mathcal{F}^p_i(\tau') e_i; \text{sld ra, 0; sfree n+1; ret ra } \{\text{r1}\}, \cdot)]
\]

where

\[
e_i = v_i \rho_i^*(\tau) \mathcal{F}^p_i ((\text{sld r1, n-1; ret end } (\rho_i^*(\tau) \mathcal{F}^p_i (\{r1\}), \cdot))
\]

and

\[
\sigma = \rho_i^*(\forall [], \{r1: \tau'^T; \zeta\}^c : \tau'^T \vdash \zeta)
\]

Consider arbitrary \((M_1, M_2) : W'\). From the definition of \(\mathcal{O}\), we need to show that either both terms terminate or are both running after \(W, k\) steps. From Lemma 3.7, we can take two steps and consider \(M'_i\) which is like \(M_i\) but has stack with type \(\sigma\) instead of \(\sigma'\), and \(W^*\) which has a corresponding stack island, noting that the values in the register \(\text{ra}\) are related from the condition on \(W'\), which means that \(\text{currentMR}(W^*(i_{\text{stk}})) \in W^* \mathcal{S}[\sigma]\rho'\).

Summarizing, we must now show:
To show this, for any future world we will have that
\[ \forall \rho^2(\tau') \exists \rho^1(\tau) \text{ such continuations, which will complete the proof.} \]

Consider the sequence of evaluation contexts

\[ E_i = \left( \text{import } r_1, \rho^i(\tau) \right) \vdash \left( \text{import } r_1, \rho^2(\tau') \right) \]

\[ \vdash \left( \text{import } r_1, \rho^1(\tau) \right) \]

\[ (\text{import } r_1, \rho^i(\tau)) \; \text{evaluated contexts} \]

and target of reduction

\[ \text{sld } r_1, n-j; \text{ret end} \{ \rho^i(\tau); \rho^1(\sigma) \} \{ r_1 \} \]

where \( \overrightarrow{w} \) starts empty and accumulates argument values in subsequent contexts. In order to step the target, we note that after one step the \( M_i \)'s have, in register \( r_1 \), related values \( \overrightarrow{w}_{ji} \) which have type \( \rho^i(\tau) \). This means that

\[ \rho^i(\tau) \vdash \left( \text{import } r_1, \rho^i(\tau); \rho^1(\sigma) \right) \{ r_1 \} \]

reduces in another step to a \( \overrightarrow{v}' \), defined and in \( \mathcal{V}[\tau'] \) by appealing to 2a on the structurally smaller type \( \tau \).

After 2n steps, we are in a world similar to \( W'^* \), denote \( W'^* \) but with the register island having type \( \tau_{n-1} \) for register \( r_1 \). We now must show that:

\[ (W'^*, E'_1[(\text{import } r_1, \rho^1(\tau)); \text{sld } r_1, 0; \text{ret } r_1 \{ r_1 \}], E'_2[(\text{import } r_1, \rho^2(\tau')); \text{sld } r_1, 0; \text{ret } r_1 \{ r_1 \}]) \in \mathcal{O} \]

From \( \overrightarrow{v}_i \) in the \( \mathcal{V}[\cdot] \) relation, we know that

\[ \overrightarrow{v}_i \in \mathcal{E}[\text{out} \vdash \tau'; \xi[\rho] \Rightarrow (\sigma, \tau, \mathcal{S}[\sigma] \rho)] \]

This means that for any chosen continuations \( E'_1 \) and \( E'_2 \) drawn from

\[ \mathcal{K}[\text{out} \vdash \tau'; \xi[\rho] \Rightarrow (\sigma, \tau, \mathcal{S}[\sigma] \rho)] = \mathcal{K}[\text{out} \vdash \tau'; \sigma] \rho \]

we will have that

\[ (W'^*, E'_1[\overrightarrow{v}_1], E'_2[\overrightarrow{v}_2]) \in \mathcal{O} \]

In particular, we will argue that

\[ E'_i = E_i[(\text{import } r_1, \rho^i(\tau) \right) \; \text{evaluated contexts} \]

are such continuations, which will complete the proof.

To show this, for any future world \( W'^* \) such that

\[ \text{currentMR}(W'^{i_{\text{stk}}}) \in W'^* \mathcal{S}[\sigma] \rho' \]

and any values \( (W', \overrightarrow{v}_1, \overrightarrow{v}_2) \in \mathcal{V}[\tau'] \rho' \), we must show that
In order to do that, consider \((M_1, M_2) : W^f\). The first step results in related values \(w_1\) and \(w_2\) being placed in register \(r_1\), appealing to the inductive case of 2b for structurally smaller type \(\tau'\).

The next instruction results in register \(ra\) having, based on the stack island, related values with type \(\forall[] \cdot \{r_1 : \tau'^T; \zeta\}^\varepsilon\).

The next instruction results in the stack being composed of just \(\zeta\), which combined with the register typing means that we can appeal to the definition of \(E_i\) and get the desired result.

**Case \((\tau) \xrightarrow{\phi_i : \phi_o} \tau'\)**

To show that that \((W, \ell_1, \ell_2)\) is in

\[
W\![\tau; \phi_i : \phi_o; \tau'^T]\rho = W\![\text{box } \forall[\zeta, \epsilon].\{ra : \text{box } \forall[].\{r_1 : \tau'^T; \phi_o :: \zeta\}^\varepsilon; \sigma']\rho]
\]

we need to show that for any future world \(\hat{W} \sqsubseteq W\), and any memories \(M_1, M_2\) such that \((W, M_1, M_2) \in \text{currentMR}(W(i_{\text{box}}))\), the following conditions hold:

- \(M_2(\ell_2) = \text{code}[\zeta, \epsilon, \{ra : \forall[].\{r_1 : \rho_1(\tau'^T); \phi_o :: \zeta\}^\varepsilon; \rho_1(\sigma')\}^\varepsilon, \ell_i\)
- \((W, M_1(\ell_1), M_2(\ell_2)) \in \mathcal{H}[\forall[\zeta, \epsilon].\{ra : \text{box } \forall[].\{r_1 : \tau'^T; \phi_o :: \zeta\}^\varepsilon; \sigma']\rho\)

The latter requires that, in a further future world \(W' \sqsubseteq W\), with \(\rho' \in \mathcal{D}[\zeta, \epsilon]\), letting \(\rho' = \rho \cup \rho^*\), such that the world fulfills the following restrictions:

- \(\text{currentMR}(W'(i_{\text{reg}})) \in W \cdot \mathcal{R}[ra : \text{box } \forall[].\{r_1 : \rho_1(\tau'^T); \phi_o :: \zeta\}^\varepsilon]\rho'\)
- \(\text{currentMR}(W'(i_{\text{stk}})) \in W \cdot \mathcal{S}[\sigma']\rho'\)

We must show that

\[
(W', (\rho_1^*(I_1), \cdot), (\rho_2^*(I_2), \cdot)) \in \mathcal{E}[ra \vdash \tau'^T; \phi_o :: \zeta]#\rho'
\]

Where \(I_1\) and \(I_2\) are defined by the value translation.

In order to do that, consider arbitrary \(E_1, E_2\) such that

\[
(W', E_1, E_2) \in \mathcal{K}[ra \vdash \tau'^T; \phi_o :: \zeta]#\rho'
\]

The definition of \(\mathcal{E}[\cdot]\) dictates we show that

\[
(W', E_1[\rho_1^*(I_1), \cdot], E_2[\rho_2^*(I_2), \cdot]) \in \mathcal{O}
\]

Expanding \(I_i\), we are considering:

\[
E_i[\text{salloc } 1 + | \phi_i| ; \text{sst } | \phi_i|, ra; \\
\text{sldr1, } | \phi_i| + n; \text{sst1, r1}; \ldots; \text{sldr1, } | \phi_i| + n + | \phi_i| - 1; \text{sst } | \phi_i|, r1; \\
\text{import } r_1, \zeta T F' \epsilon; \\
\text{sldr } ra, | \phi_o| ; \\
\text{sldr2, } | \phi_o| - 1; \text{sst } | \phi_o| + n + | \phi_i| - 1, r2; \ldots; \text{sldr2, 0; sst } | \phi_o| + n + | \phi_i| - \text{pref}_o, r2; \\
\text{sfree } n + | \phi_o| + 1; \text{retra } \{r1\}]
\]

where

\[
e_i = \nu^{\rho_1^*(\zeta)} F' T (((\text{sldr1, } | \phi_i| + n - j; \text{ret } \{\rho_1^*(\zeta) T; \sigma\} \{r1\}), \cdot)
\]

and

\[
\sigma = \rho_1^*(\phi_i :: \forall[].\{r_1 : \tau'^T; \zeta\}^\varepsilon :: \tau'^T :: \zeta)
\]
Consider arbitrary \((M_1, M_2) : W'\). From the definition of \(O\), we need to show that either both terms terminate or are both running after \(W.k\) steps. From Lemma 3.7, we can take \(2 + 2 \cdot |\phi_i|\) steps and consider \(M'_1\) which is like \(M_1\) but has stack with type \(\sigma'\) instead of \(\sigma\), and \(W*\) which has a corresponding stack island, noting that the values in the register \(ra\) are related from the condition on \(W'\), and the rest of the instructions are moving related values from one point of the stack to another. This means that currentMR\((W^*(\text{intk})) \in W\cdot S[\sigma]')'. Summarizing, we must now show:

\[
(W^*, E_1[\text{import } r_1, \rho_1^{(\cdot)}; \tau; \rho_1^{(\cdot)}; \rho_1^{(\cdot)} (r_1)]; \text{ret } r_1(1), \cdot), E_2[\text{import } r_1, \rho_2^{(\cdot)}; \tau; \rho_2^{(\cdot)} (r_2)]; \text{ret } r_1(1), \cdot) \in O
\]

Consider the sequence of evaluation contexts

\[
E_i[\text{import } r_1, \rho_i^{(\cdot)}; \tau; \rho_i^{(\cdot)} (r_1)]_{\mathbb{V}}_{\mathbb{V}_i}^{\mathbb{V}_i} (\cdot); \text{ret } r_1(1), \cdot)
\]

and target of reduction

\[
\text{sld } r_1, \phi_i + n - j; \text{ret } r_1(1), \cdot)
\]

where \(\mathbb{V}_i\) starts empty and accumulates argument values in subsequent contexts. In order to step the target, we note that after one step the \(M_i\)'s have, in register \(r_1\), related values \(w_{ji}\) which have type \(\rho_i^{(\cdot)} (r_1)\). This means that

\[
\rho_i^{(\cdot)} (r_1) \text{-reduce to } (\text{ret } r_1(1), \cdot)_{\mathbb{V}}_{\mathbb{V}_i}^{\mathbb{V}_i} (\cdot; \text{ret } r_1(1), \cdot)
\]

reduces in another step to a \(\mathbb{V}_i\), defined and in \(\mathbb{V}_i \in \mathbb{V}^T\) by appealing to \(2a\) on the structurally smaller type \(\tau\).

After \(2n\) steps, we are in a world similar to \(W^*\), denote \(W^{**}\) but with the register island having type \(\tau_n\) for register \(r_1\). We now must show that:

\[
(W^{**}; E_1[\text{import } r_1, \rho_1^{(\cdot)}; \tau; \rho_1^{(\cdot)} (r_1)]; \text{ret } r_1(1), \cdot), E_2[\text{import } r_1, \rho_2^{(\cdot)}; \tau; \rho_2^{(\cdot)} (r_2)]; \text{ret } r_1(1), \cdot) \in O
\]

From \(\mathbb{V}_i\) in the \(\mathbb{V}_i^{(\cdot)}\) relation, we know that

\[
\mathbb{V}_i \in \mathbb{E}(\text{out} \vdash \tau' ; \phi_o :: \zeta ; \rho [\zeta \mapsto \text{SR}])
\]

for any \(\mathbb{E}\), and in particular, for \(\mathbb{S}\) as \(\mathbb{S} = (\sigma''', \sigma'', \mathbb{S}[\sigma'''] \rho)\) where \(\sigma'' = \forall[, \{r_1: \tau' T; \zeta \} \cdot, \tau''', \sigma'']\). This means that for any chosen continuations \(E_1\) and \(E_2\) drawn from

\[
K[\text{out} \vdash \tau' ; \phi_o :: \zeta ; \rho [\zeta \mapsto (\sigma''', \sigma'', \mathbb{S}[\sigma'''] \rho)] = K[\text{out} \vdash \tau'; \phi_o :: \sigma'' [\rho]
\]

we will have that

\[
(W^{**}; E'_1[\mathbb{V}_1\mathbb{V}_i], E'_2[\mathbb{V}_2]) \in O
\]

In particular, we will argue that

\[
E'_i = E_i[\text{import } r_1, \rho_i^{(\cdot)}; \tau; \rho_i^{(\cdot)} (r_1)];
\]

\[
\text{sld } r_1, \phi_o 
\]

\[
\text{sld } r_2, \phi_o - 1; \text{sst } |\phi_o| + n + |\phi_i| - 1, r_2; \text{sld } r_2, 0; \text{sst } |\phi_o| + n + |\phi_i| - |\text{pref}_o|, r_2;
\]

\[
\text{sfree } n + |\phi_o| + 1; \text{ret } r_1(1), \cdot)
\]

121
are such continuations, which will complete the proof.

To show this, for any future world \( W^f \sqsupseteq W^{**} \) such that currentMR\( (W^f(i_{stk})) \subseteq W^f \)
\( \mathcal{S}[\phi_o :: \sigma'' \rho'] \), and any values \( (W^f, v_1, v_2) \in \mathcal{V}[[\tau'] \rho'] \), we must show that

\[
(W^f, E_1[v_1], E_2[v_2]) \in \mathcal{O}
\]

In order to do that, consider \( (M_1, M_2) : W^f \). The first step results in related values \( w_1 \) and \( w_2 \) being placed in register \( r_1 \), appealing to the inductive case of 2b for structurally smaller type \( \tau' \).

The next instruction results in register \( r_a \) having, based on the stack island, related values with type \( \forall[] \{ r_1 : \tau'T; \zeta \}^c \).

The next \( 2 \star |\phi_o| + 1 \) instructions result in the stack being composed of \( \phi_o :: \zeta \), based on the related values on the stack specified by the constraint on the world, and the freeing of part of the stack.

Combined with the register typing, this means that we can appeal to the definition of \( E_i \) and get the desired result.

\[\square\]
3.6 Compatibility Lemmas

NOTE: The proofs for TAL terms in the multi-language F+T are elided below since they are essentially the same as the proofs for the corresponding compatibility lemmas for T.

Lemma 3.29 (Component)
If \( \psi \vdash H_1 \approx_{H_2} H_1 \), boxheap(\( \psi' \)), ret-type(q, \( \chi, \sigma \)) = \( \tau; \sigma' \), and \( (\psi; \psi') : \Gamma; \chi; \sigma ; q \vdash I_1 \approx_I I_2 \), then \( \psi; \Delta; \Gamma; \chi; \sigma ; q \vdash (I_1, H_1) \approx (I_2, H_2) : \tau; \sigma' \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.30 (Heap Fragment)
Let \( \cdot \vdash \psi_1, \ldots, \cdot \vdash \psi_n \) and \( \psi' = \ell_1 : \nu_1 \psi_1, \ldots, \ell_n : \nu_n \psi_n \) such that \( \text{dom}(\psi) \cap \text{dom}(\psi') = \emptyset \). If for each \( i \), we have \( \psi, \psi' \vdash h_{i1} \approx_{h_{i2}} \nu_i \psi_i \), then \( \psi \vdash \ell_1 \mapsto h_{11}, \ldots, \ell_n \mapsto h_{1n} \approx_{h_{21}, \ldots, h_{2n}} \psi' \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.31 (Code Block)
If \( \cdot \vdash \forall[\Delta]. \{ \chi; \sigma \}^q \) and \( \psi; \Delta; \cdot \vdash \chi; \sigma ; q \vdash I_1 \approx_I I_2 \), then \( \psi \vdash \text{code}[\Delta]\{ \chi; \sigma \}^q.I_1 \approx_{hv} \text{code}[\Delta]\{ \chi; \sigma \}^q.I_2 : \text{box}\forall[\Delta]. \{ \chi; \sigma \}^q \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.32 (Tuple)
If for each \( i \), we have \( \psi; \cdot \vdash w_{i1} \approx_{w_{i2}} \tau_i \), then \( \psi \vdash \langle w_{10}, \ldots, w_{in} \rangle \approx_{hv} \langle w_{20}, \ldots, w_{2n} \rangle : \nu(\tau_0, \ldots, \tau_n) \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.33 (Unit)
\( \psi; \Delta \vdash () \approx_w () : \text{unit} \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.34 (Integer)
\( \psi; \Delta \vdash n \approx_w n : \text{int} \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.35 (Mutable Location)
If \( \ell : \text{ref} \psi \in \psi \), then \( \psi; \Delta \vdash \ell \approx_w \ell : \text{ref} \psi \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.36 (Immutable Location)
If \( \ell : \text{box} \psi \in \psi \), then \( \psi; \Delta \vdash \ell \approx_w \ell : \text{box} \psi \).

Proof
Analogous to proof of corresponding compatibility lemma for T.
Lemma 3.37 (Pack)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \tau[\tau'/\alpha]$, then $\psi; \Delta \vdash \text{pack}(\tau',w_1) \text{ as } \exists \alpha. \tau \approx_w \text{pack}(\tau',w_2) \text{ as } \exists \alpha. \tau : \exists \alpha. \tau$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.38 (Fold)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \tau[\mu \alpha. \tau/\alpha]$, then $\psi; \Delta \vdash \text{fold}_{\mu \alpha. \tau} w_1 \approx_w \text{fold}_{\mu \alpha. \tau} w_2 : \mu \alpha. \tau$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.39 (Word Type Application)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \text{box}[\alpha, \Delta'].\{\chi; \sigma\}^n$ and $\Delta \vdash \tau$, then

$$\psi; \Delta \vdash w_1[\tau] \approx_w w_2[\tau] : \text{box}[\Delta'].\{\chi[\tau/\alpha]; \sigma[\tau/\alpha]\}^n[\tau/\alpha]$$

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.40 (Stack Type Application)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \text{box}[\zeta, \Delta' \tau', \zeta].\{\chi; \sigma\}^n$ and $\Delta \vdash \tau'$, then

$$\psi; \Delta \vdash w_1[\tau'] \approx_w w_2[\tau'] : \text{box}[\Delta'.\{\chi[\tau'/\zeta]; \sigma[\tau'/\zeta]\}^n[\tau'/\zeta]$$

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.41 (Return Marker Type Application)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \text{box}[\epsilon, \Delta'.\{\chi; \sigma\}^n, \text{ftv}(q') \subseteq \Delta$, and $\Delta \vdash \forall[\Delta'.\{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^n[q'/\epsilon]]$, then

$$\psi; \Delta \vdash w_1[q'] \approx_w w_2[q'] : \text{box}[\Delta'.\{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^n[q'/\epsilon]]$$

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.42 (Word Value)
If $\psi; \Delta \vdash w_1 \approx_w w_2 : \tau$, then $\psi; \Delta; \chi \vdash w_1 \approx_u w_2 : \tau$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.43 (Register)
If $r : \tau \in \chi$, then $\psi; \Delta; \chi \vdash r \approx_u r : \tau$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.44 (Pack)
If $\psi; \Delta; \chi \vdash u_1 \approx_u u_2 : \tau[\tau'/\alpha]$, then $\psi; \Delta; \chi \vdash \text{pack}(\tau',u_1) \text{ as } \exists \alpha. \tau \approx_u \text{pack}(\tau',u_2) \text{ as } \exists \alpha. \tau : \exists \alpha. \tau$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.45 (Fold)
If $\psi; \Delta; \chi \vdash u_1 \approx_u u_2 : \tau[\mu \alpha. \tau/\alpha]$, then $\psi; \Delta; \chi \vdash \text{fold}_{\mu \alpha. \tau} u_1 \approx_u \text{fold}_{\mu \alpha. \tau} u_2 : \mu \alpha. \tau$.
Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.46 (Word Type Application)
If $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2: \text{box} \forall[\alpha, \Delta'].\{\chi; \sigma\}^q$ and $\Delta \vdash \tau$, then

$$\Psi; \Delta; \chi \vdash u_1[\tau] \approx_u u_2[\tau]: \text{box} \forall[\Delta'].\{\chi[\tau/\alpha]; \sigma[\tau/\alpha]\}^q[\tau/\alpha].$$

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.47 (Stack Type Application)
If $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2: \text{box} \forall[\zeta, \Delta'].\{\chi; \sigma\}^q$ and $\Delta \vdash \sigma'$, then

$$\Psi; \Delta; \chi \vdash u_1[\sigma'] \approx_u u_2[\sigma']: \text{box} \forall[\Delta'].\{\chi[\sigma'/\zeta]; \sigma[\sigma'/\zeta]\}^q[\sigma'/\zeta].$$

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.48 (Return Marker Type Application)
If $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2: \text{box} \forall[\epsilon, \Delta'].\{\chi; \sigma\}^q$, ftv($q'$) $\subseteq \Delta$, and $\Delta \vdash \forall[\Delta'].\{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q[q'/\epsilon]$, then

$$\Psi; \Delta; \chi \vdash u_1[q'] \approx_u u_2[q']: \text{box} \forall[\Delta'].\{\chi[q'/\epsilon]; \sigma[q'/\epsilon]\}^q[q'/\epsilon].$$

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.49 (Arithmetic Operation)
If $\Psi; \Delta; \chi \vdash r_1 \approx_u r_2: \text{int}$, $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2: \text{int}$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d: \text{int}]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{aop} \ r_d, r_1, u_1; I_1 \approx \text{aop} \ r_d, r_2, u_2; I_2$.

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.50 (Branch)
If $\Psi; \Delta; \chi \vdash r_1 \approx_u r_2: \text{int}$, $\Psi; \Delta; \chi \vdash u_1 \approx_u u_2: \text{box} \forall[]\{\chi'; \sigma\}^q$, $\Delta \vdash \chi \leq \chi'$, and $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{bnz} \ r_1, u_1; I_1 \approx \text{bnz} \ r_2, u_2; I_2$.

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.51 (Load from Mutable Tuple)
If $\Psi; \Delta; \chi \vdash r_1 \approx_u r_2: \text{ref} \langle \tau_0, \ldots, \tau_n \rangle$, $0 \leq i \leq n$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d: \tau_i]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{ld} \ r_d, r_1[i]; I_1 \approx \text{ld} \ r_d, r_2[i]; I_2$.

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □

Lemma 3.52 (Load from Immutable Tuple)
If $\Psi; \Delta; \chi \vdash r_1 \approx_u r_2: \text{box} \langle \tau_0, \ldots, \tau_n \rangle$, $0 \leq i \leq n$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d: \tau_i]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{ld} \ r_d, r_1[i]; I_1 \approx \text{ld} \ r_d, r_2[i]; I_2$.

Proof

Analogous to proof of corresponding compatibility lemma for $T$. □
Lemma 3.53 (Store to Mutable Tuple)
If $\Psi; \Delta; \chi \vdash r_d \approx u \approx r_d\colon \text{ref}(\tau_0, \ldots, \tau_n)$, $0 \leq i \leq n$, $\Psi; \Delta; \chi \vdash r_s \approx u \approx r_s\colon \tau_i$, and $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{st}(r_d[i]); r_s; I_1 \approx \text{st}(r_d[i]); r_s; I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.54 (Allocate Immutable Tuple)
If $\text{len}(\tau) = n$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d:\text{ref}(\tau)]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \tau :: \sigma; q \vdash \text{alloc}(r_d, n); I_1 \approx \text{alloc}(r_d, n); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.55 (Allocate Immutable Tuple)
If $\text{len}(\tau) = n$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d:\text{box}(\tau)]; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \tau :: \sigma; q \vdash \text{alloc}(r_d, n); I_1 \approx \text{alloc}(r_d, n); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.56 (Move)
If $\Psi; \Delta; \chi \vdash u_1 \approx u_2\colon \tau$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d:\tau]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{mv}(r_d, u_1); I_1 \approx \text{mv}(r_d, u_2); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.57 (Move Return Address)
If $\chi(r_s) = \tau$ and $\Psi; \Delta; \Gamma; \chi[r_d:\tau]; \sigma; r_d \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; r_s \vdash \text{mv}(r_d, r_s); I_1 \approx \text{mv}(r_d, r_s); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.58 (Unpack)
If $\Psi; \Delta; \chi \vdash u_1 \approx u_2\colon \exists \alpha.\tau$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d:\tau]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{unpack}(\alpha, r_d); u_1; I_1 \approx \text{unpack}(\alpha, r_d); u_2; I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.59 (Unfold)
If $\Psi; \Delta; \chi \vdash u_1 \approx u_2\colon \mu\alpha.\tau$, $q \neq r_d$, and $\Psi; \Delta; \Gamma; \chi[r_d:\tau[\mu\alpha.\tau/\alpha]]; \sigma; q \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{unfold}(r_d, u_1); I_1 \approx \text{unfold}(r_d, u_2); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.60 (Allocate Stack Space)
If $\Psi; \Delta; \Gamma; \chi; \text{unit} :: \cdots :: \text{unit} :: \sigma; \text{inc}(q, n) \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{alloc}(n); I_1 \approx \text{alloc}(n); I_2$.

Proof
Analogous to proof of corresponding compatibility lemma for $T$.

Lemma 3.61 (Free Stack Space)
If $\Psi; \Delta; \Gamma; \chi; \sigma; \text{dec}(q, n) \vdash I_1 \approx I_2$, then $\Psi; \Delta; \Gamma; \chi; \tau_0 :: \cdots :: \tau_{n-1} :: \sigma; q \vdash \text{sfree}(n); I_1 \approx \text{sfree}(n); I_2$.
Proof

Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.62 (Load from Stack)
If \( \sigma(i) = \tau, q \neq r_d \), and \( \Psi; \Delta; \Gamma; \chi[r_d: \tau]; \sigma; q \vdash I_1 \approx I_2 \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{sld} r_d, i; I_1 \approx I_2; sld r_d, i; I_2 \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.63 (Load Return Address from Stack)
If \( \sigma(i) = \tau \) and \( \Psi; \Delta; \Gamma; \chi[r_d: \tau]; \sigma; r_d \vdash I_1 \approx I_2 \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; i \vdash \text{sld} r_d, i; I_1 \approx I_2; \text{sld} r_d, i; I_2 \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.64 (Store to Stack)
If \( \Psi; \Delta; \chi \vdash r_s1 \approx r_s2; \tau', q \neq i \), and \( \Psi; \Delta; \Gamma; \chi; \tau_0; \cdots; \tau_{i-1}; \cdots; \tau'_{i-1}; \cdots; \tau'_{i} :: \sigma; q \vdash I_1 \approx I_2 \), then \( \Psi; \Delta; \Gamma; \chi; \tau_0; \cdots; \tau_{i}; : \sigma; q \vdash \text{sst} i, r_s1 \approx \text{sst} i, r_s2; I_1 \approx I_2 \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.65 (Store Return Address to Stack)
If \( \chi(r_s) = \tau' \) and \( \Psi; \Delta; \Gamma; \chi; \tau_0; \cdots; \tau_{i-1}; \cdots; \tau'_{i-1}; \cdots; \tau'_{i} :: \sigma; i \vdash I_1 \approx I_2 \), then \( \Psi; \Delta; \Gamma; \chi; \tau_0; \cdots; \tau_{i}; :: \sigma; r_s \vdash \text{sst} i, r_s1 \approx \text{sst} i, r_s2; I_1 \approx I_2 \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.66 (Return from Call)
If \( \chi(r) = \text{box} \forall []). \{ \tau': \tau; \sigma \}^q \) and \( \chi(r') = \tau \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; r \vdash \text{return} \{ r' \} \approx \text{return} \{ r \} \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.67 (Return at End)
If \( \chi(r) = \tau \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; \text{end} \{ \tau; \sigma \} \{ r \} \approx \text{return} \{ \tau; \sigma \} \{ r \} \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.68 (Jump)
If \( \Psi; \Delta; \chi \vdash u_1 \approx u_2 : \text{box} \forall [] \{ \chi'; \sigma \}^q \), \( \Delta \vdash \chi \leq \chi' \), and \( \cdot [\Delta]; \chi; \sigma \vdash q \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{jmp} u_1 \approx \text{jmp} u_2 \).

Proof
Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.69 (Call)
Given the following:

- \( \Psi; \Delta; \chi \vdash u_1 \approx u_2 : \text{box} \forall [\zeta, \epsilon]. \{ \chi; \delta \}^q \),
- \( \text{ret-addr-type}(\hat{q}, \hat{\chi}, \hat{\sigma}) = \forall [] \{ \tau; \sigma' \}^\epsilon \),
- \( \Delta \vdash \sigma_0 \),
\[ \Delta \vdash \forall \cdot \{ \hat{\chi} | \sigma_0/\zeta | (i + k - j)/\epsilon \}; \hat{\sigma} | \sigma_0/\zeta | (i + k - j)/\epsilon \} \hat{q}, \]
\[ \Delta \vdash \chi \leq \hat{\chi} | \sigma_0/\zeta | (i + k - j)/\epsilon, \]
\[ \sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0, \]
\[ \hat{\sigma} = \tau_0 :: \cdots :: \tau_j :: \zeta, \]
\[ j < i, \] and
\[ \hat{\sigma}' = \tau'_0 :: \cdots :: \tau'_k :: \zeta, \]
we have that \( \Psi; \Delta; \Gamma; \chi; \sigma; i \vdash \text{call } u_1 \{ \sigma_0, i + k - j \} \approx_1 \text{call } u_2 \{ \sigma_0, i + k - j \}. \)

**Proof**

Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.70 (Call from Top Level)

Given the following:

\[ \Psi; \Delta; \chi \vdash u_1 \approx u_2 : \text{box } \forall \cdot \{ \hat{\chi}; \hat{\sigma} \} \hat{q}, \]
\[ \text{ret-addr-type}(\hat{q}, \hat{\chi}, \hat{\sigma}) = \forall \cdot \{ r; \tau; \hat{\sigma}' \} \epsilon, \]
\[ \Delta \vdash \sigma_0, \]
\[ \Delta \vdash \forall \cdot \{ \hat{\chi}|\sigma_0/\zeta|\text{end}\{\tau^*;\sigma^*\}/\epsilon; \hat{\sigma}|\sigma_0/\zeta|\text{end}\{\tau^*;\sigma^*\}/\epsilon \} \hat{q}, \]
\[ \Delta \vdash \chi \leq \hat{\chi}|\sigma_0/\zeta|\text{end}\{\tau^*;\sigma^*\}/\epsilon, \]
\[ \sigma = \tau :: \sigma_0, \]
\[ \hat{\sigma} = \tau :: \zeta, \] and
\[ \hat{\sigma}' = \tau'_0 :: \cdots :: \tau'_k :: \zeta, \]
we have that \( \Psi; \Delta; \Gamma; \chi; \sigma; \text{end}\{\hat{\tau}; \hat{\sigma}\} \vdash \text{call } u_1 \{ \sigma_0, \text{end}\{\tau^*;\sigma^*\} \} \approx_1 \text{call } u_2 \{ \sigma_0, \text{end}\{\tau^*;\sigma^*\} \}. \)

**Proof**

Analogous to proof of corresponding compatibility lemma for T.

Lemma 3.71 (Import)

Given the following:

\[ \Psi; \Delta; \Gamma; \chi; \tau_0 :: \cdots :: \tau_j :: \zeta; \text{out } \vdash e_1 \approx e_2 : \tau; \tau'_0 :: \cdots :: \tau'_k :: \zeta, \]
\[ \sigma = \tau_0 :: \cdots :: \tau_j :: \sigma_0, \]
\[ \sigma' = \tau'_0 :: \cdots :: \tau'_k :: \sigma_0, \]
\[ q = i > j \] or \( q = \text{end}\{\hat{\tau}; \hat{\sigma}\}, \]
\[ \Psi; \Delta; \chi[r_d: \tau^*]; \sigma'; \text{inc}(q, k - j) \vdash I_1 \approx_1 I_2, \]
we have that \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{import } r_d, \sigma_0 \mathcal{F}^r e_1; I_1 \approx_1 \text{import } r_d, \sigma_0 \mathcal{F}^r e_2; I_2. \)

**Proof**
Consider arbitrary \( W, \rho \), and \( \gamma \) such that \( W \in H[\Psi], \rho \in D[\Delta], \gamma \in G[\Gamma], \) currentMR(\( W(i_{\text{reg}}) \)) \( \in W \mathcal{R}[\chi]\rho \), and currentMR(\( W(i_{\text{stk}}) \)) \( \in W S[\sigma]\rho \).

We need to show

\[
(W, \rho_1(\gamma_1((\text{import } r_d, \sigma^{a} T F^{r} e_1; I_1, \cdot))), \rho_2(\gamma_2((\text{import } r_d, \sigma^{a} T F^{r} e_2; I_2, \cdot))))
\]

\[
= (W, (\text{protect } \rho_1(\phi), \rho_1(\zeta); \rho_1(\gamma_1(I_1)), \cdot), (\text{protect } \rho_2(\phi), \rho_2(\zeta); \rho_2(\gamma_2(I_2)), \cdot))
\]

\[\in E[q \vdash \text{ret-type}(q, \chi, \sigma)]\rho.\]

The result follows from Lemma 3.28, 1(b), using what is given in the hypotheses.

\[\square\]

**Lemma 3.72 (Protect)**

Given the following:

- \( \sigma = \phi :: \sigma' \)
- \( \Psi; \Delta, \zeta; \chi; \phi :: \zeta; q \vdash I_1 \approx I_2 \),

we have that \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \text{protect } \phi, \zeta; I_1 \approx \text{protect } \phi, \zeta; I_2 \).

**Proof**

Consider arbitrary \( W, \rho \), and \( \gamma \) such that \( W \in H[\Psi], \rho \in D[\Delta], \gamma \in G[\Gamma], \) currentMR(\( W(i_{\text{reg}}) \)) \( \in W \mathcal{R}[\chi]\rho \), and currentMR(\( W(i_{\text{stk}}) \)) \( \in W S[\sigma]\rho \).

We need to show

\[
(W, \rho_1(\gamma_1((\text{protect } \phi, \zeta; I_1, \cdot))), \rho_2(\gamma_2((\text{protect } \phi, \zeta; I_2, \cdot))))
\]

\[
= (W, (\text{protect } \rho_1(\phi), \rho_1(\zeta); \rho_1(\gamma_1(I_1)), \cdot), (\text{protect } \rho_2(\phi), \rho_2(\zeta); \rho_2(\gamma_2(I_2)), \cdot))
\]

\[\in E[q \vdash \text{ret-type}(q, \chi, \sigma)]\rho.\]

Let \( \rho' = \rho[\zeta \mapsto S[\sigma']] \), noting the latter exists due to last given. We can then instantiate the second hypothesis, since \( \rho' \in D[\Delta, \zeta] \), and \( S[\sigma]\rho = S[\phi :: \sigma']\rho = S[\phi :: \zeta]\rho' \) due to substitution. This gives us that:

\[
(W, \rho_1(\gamma_1((I_1, \cdot)), \rho_1(\gamma_1((I_1, \cdot)))) \in E[q \vdash \text{ret-addr-type}(q, \chi, \phi :: \zeta)]\rho'
\]

With which the result can follow from Lemma 3.4.3, since the \text{protect } \phi, \zeta instruction reduces without any operational consequences.

\[\square\]

**Lemma 3.73 (F Variable)**

If \( x : \tau \in \Gamma \), then \( \Psi; \Delta; \Gamma; \chi; \sigma; \text{out} \vdash x \approx x : \tau; \sigma \).

**Proof**

Consider arbitrary \( W, \rho \), and \( \gamma \) such that \( W \in H[\Psi], \rho \in D[\Delta], \gamma \in G[\Gamma], \) currentMR(\( W(i_{\text{reg}}) \)) \( \in W \mathcal{R}[\chi]\rho \), and currentMR(\( W(i_{\text{stk}}) \)) \( \in W S[\sigma]\rho \).

We must show that \( (W, \rho_1(\gamma_1(x)), \rho_2(\gamma_2(x))) \in E[\text{out} \vdash \text{int}; \sigma_2] \)

This follows immediately from the definition of \( G[\Gamma] \) and Lemma 3.25.

\[\square\]

**Lemma 3.74 (F Unit)**

\( \Psi; \Delta; \Gamma; \chi; \sigma; \text{out} \vdash () \approx () : \text{unit}; \sigma \).

**Proof**

Immediate from Lemma 3.25.

\[\square\]

129
Lemma 3.75 (F Integer)
Ψ; Δ; Γ; χ; σ; out ⊢ n ≈ n: int; σ.

Proof
Immediate from Lemma 3.25.

Lemma 3.76 (F Arith Op)
If Ψ; Δ; Γ; χ; σ₀; out ⊢ t₁ ≈ t₂: int; σ₁, and Ψ; Δ; Γ; χ; σ₁; out ⊢ t’₁ ≈ t’₂: int; σ₂, then Ψ; Δ; Γ; χ; σ₀; out ⊢ t₁ p t’₁ ≈ t₂ p t’₂: int; σ₂.

Proof
Consider arbitrary W, ρ, and γ such that W ∈ H[Ψ], ρ ∈ D[Δ], γ ∈ G[Γ], currentMR(W(i_reg)) ≈ W R[γ]ρ, and currentMR(W(i_stk)) ≈ W S[σ₀]ρ.

We must show that
(W, ρ₁(γ₁(t₁ p t’₁)), ρ₂(γ₂(t₂ p t’₂))) = (W, ρ₁(γ₁(t₂)), ρ₂(γ₂(t₂))) ∈ E[out ⊢ int; σ₂]

We proceed by applying Lemma 3.26 twice, first with evaluation contexts Eᵣ = [ ] p ρ₁(γ₁(t’)) and then with Eᵣ = v₁ p [ ].

We then must show that (W, v₁ p v’₁, v₂ p v’₂) ∈ E[out ⊢ int; σ₂].

The result then follows from Lemma 3.13 and Lemma 3.25.

Lemma 3.77 (F If0)
If Ψ; Δ; Γ; χ; σ₀; out ⊢ t₁ ≈ t’₁: int; σ₁, Ψ; Δ; Γ; χ; σ₁; out ⊢ t₂ ≈ t’₂: τ; σ₂, and Ψ; Δ; Γ; χ; σ₁; out ⊢ if0 t₁ t₂ t₃ ≈ if0 t’₁ t’₂ t’₃: τ; σ₂.

Proof
Consider arbitrary W, ρ, and γ such that W ∈ H[Ψ], ρ ∈ D[Δ], γ ∈ G[Γ], currentMR(W(i_reg)) ≈ W R[γ]ρ, and currentMR(W(i_stk)) ≈ W S[σ₀]ρ.

We must show that
(W, ρ₁(γ₁(if0 t₁ t₂ t₃)), ρ₂(γ₂(if0 t’₁ t’₂ t’₃))) ∈ E[out ⊢ τ; σ₂]

After pushing in the substitutions, we apply Lemma 3.26 three times, which requires us to then show that:

(W, if0 v₁ v₂ v₃, if0 v’₁ v’₂ v’₃) ∈ E[out ⊢ τ; σ₂]


Inspecting the V[int]ρ, we can see that v₁ and v’₁ are integer values which are either 0 or not.

In either case, we proceed by combination of Lemma 3.13 and Lemma 3.25.

Lemma 3.78 (F Pure Function)
If Ψ; Δ, ζ; Γ, x: τ; χ; ζ; out ⊢ t₁ ≈ t₂: τ’; ζ,
then Ψ; Δ; Γ; χ; σ; out ⊢ λ(x:τ).t₁ ≈ λ(x:τ).t₁: (τ → τ’; σ)

Proof
Consider arbitrary W, ρ, and γ such that W ∈ H[Ψ], ρ ∈ D[Δ], γ ∈ G[Γ], currentMR(W(i_reg)) ≈ W R[γ]ρ, and currentMR(W(i_stk)) ≈ W S[σ]ρ.

We need to show that
(W, ρ₁(γ₁(λ(x: τ₁), τ₁)), ρ₂(γ₂(λ(x: τ₂), τ₂)))
= (W, λ(x: ρ₁(γ₁(τ₁))), ρ₁(γ₁(τ₁)), λ(x: ρ₂(γ₂(τ₂))), ρ₂(γ₂(τ₂))) ∈ E[out ⊢ (τ) → τ′]ρ

Using Lemma 3.25, it suffices to show that (W, ρ₁(γ₁(λ(x: ρ₁(γ₁(τ₁)))), ρ₁(γ₁(τ₁))), ρ₂(γ₂(τ₂))) ∈ Y[τ → τ′]ρ.

This means we must consider arbitrary W' ⊆ pub, SR ∈ TStackRel, v̄₁, v̄₂ such that currentMR(W'(i_stk)) ∈ W'.
S[γ]′ where ′ = ′[γ ↦ SR] and (W', v̄₁, v̄₂) ∈ Y[τ]ρ.

Our obligation is to show that (W', ρ₁(γ₁(λ(x: ρ₁(γ₁(τ₁)))), ρ₁(γ₁(τ₁))), ρ₂(γ₂(τ₂))) ∈ E[out ⊢ τ'; ζ]ρ.

We appeal to Lemma 3.13, which means we must show that
(W', ρ₁(γ₁(τ₁)), ρ₂(γ₂(τ₂))) ∈ E[τ; ζ]ρ.

Let γ' = γ[x → (v̄₁, v̄₂)]. Note that γ' ∈ G[Γ, x: τ]ρ, and that, appealing to substitution, the above is equivalent to (W', ρ₁(γ₁(τ₁))), ρ₂(γ₂(τ₂))) ∈ E[τ; ζ]ρ.

Which follows from our first hypothesis.

□

Lemma 3.79 (F Stack Modifying Function)
If Ψ; Δ, ζ; Γ, x: τ; χ; φ₁ :: ζ; out ⊢ t₁ ≈ t₂: τ'; φ₀ :: ζ,
then Ψ; Δ; Γ; χ; σ; out ⊢ λφ₁(x: τ).t₁ ≈ λφ₁(x: τ).t₁: (τ) φ₁:φ₀ τ'; σ

Proof
Consider arbitrary W, ρ, and γ such that W ∈ H[Ψ], ρ ∈ D[Δ], γ ∈ G[Γ], currentMR(W(i_reg)) ∈ W.

We need to show that

(W, ρ₁(γ₁(λ(φ₁)(x: τ₁), τ₁)), ρ₂(γ₂(λ(φ₁)(x: τ₂), τ₂)))
= (W, λ(ρ₁(φ₁)(x: ρ₁γ₁(τ₁))), ρ₁(γ₁(τ₁)), λ(ρ₂(φ₁)(x: ρ₂γ₂(τ₂))), ρ₂(γ₂(τ₂))) ∈ E[out ⊢ (τ) → τ′]φ₁:φ₀ τ'ρ.

Using Lemma 3.25, it suffices to show that (W, λ(ρ₁(φ₁)(x: ρ₁γ₁(τ₁))), ρ₁(γ₁(τ₁)), λ(ρ₂(φ₁)(x: ρ₂γ₂(τ₂))), ρ₂(γ₂(τ₂))) ∈ Y[τ → τ′]φ₁:φ₀ τ'ρ.

This means we must consider arbitrary W' ⊆ pub, SR ∈ TStackRel, v̄₁, v̄₂ such that currentMR(W'(i_stk)) ∈ W'.
S[φ₁ :: ζ]′ where ′ = ′[ζ ↦ SR] and (W', v̄₁, v̄₂) ∈ Y[τ]φ₁:φ₀ τ'ρ.

Our obligation is to show that (W', λ(ρ₁(φ₁)(x: ρ₁γ₁(τ₁))), ρ₁(γ₁(τ₁)), λ(ρ₂(φ₁)(x: ρ₂γ₂(τ₂))), ρ₂(γ₂(τ₂))) ∈ E[out ⊢ τ'; ζ]φ₁:φ₀ τ'ρ.

We appeal to Lemma 3.13, which means we must show that
(W', ρ₁(γ₁(τ₁)), ρ₂(γ₂(τ₂))) ∈ E[τ'; ζ]φ₁:φ₀ τ'ρ.

Let γ' = γ[x → (v̄₁, v̄₂)]. Note that γ' ∈ G[Γ, x: τ]φ₁:φ₀ τ'ρ, and that, appealing to substitution, the above is equivalent to (W', ρ₁(γ₁(τ₁)), ρ₂(γ₂(τ₂))) ∈ E[τ'; ζ]φ₁:φ₀ τ'ρ.

Which follows from our first hypothesis.

□
**Lemma 3.80 (F Pure Application)**

If $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t \approx t': (\tau_1 \cdots \tau_n) \rightarrow \tau'; \sigma_0$, and $\Psi; \Delta; \Gamma; \chi; \sigma_{i-1}; \text{out} \vdash t_i \approx t'_i: \tau_i; \sigma_i$ for $i \in \{1, \ldots, n\}$, then $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t_1 \cdots t_n \approx t'_1 \cdots t'_n: \tau'; \sigma_n$.

**Proof**

Consider arbitrary $W$, $\rho$, and $\gamma$ such that $W \in \mathcal{H}[\Psi]$, $\rho \in D[\Delta]$, $\gamma \in \mathcal{G}[\Gamma]$, currentMR($W(i_{reg})$) $\in W \mathcal{R}[\chi] \rho$, and currentMR($W(i_{stk})$) $\in W \mathcal{S}[\sigma_0] \rho$.

We need to show that

$$(W, \rho_1(\gamma_1(t_1 \cdots t_n)), \rho_2(\gamma_2(t'_1 \cdots t'_n))) = (W, \rho_1(\gamma_1(t_1)) \cdots \rho_1(\gamma_1(t_n)), \rho_2(\gamma_2(t'_1)) \cdots \rho_2(\gamma_2(t'_n)))$$

$\in \mathcal{E}[\text{out} \vdash \tau'; \sigma_n] \rho$

We proceed by applying Lemma 3.26 $n + 1$ times, with the first evaluation contexts being $E_0 = [\cdot] \rho_1(\gamma_1(t_1)) \cdots \rho_1(\gamma_1(t_n))$ and $E'_0 = [\cdot] \rho_2(\gamma_2(t'_1)) \cdots \rho_2(\gamma_2(t'_n))$.

And the $i$th being $E_i = v_0 v_1 \cdots v_{i-1}[\cdot] \cdots \rho_1(\gamma_1(t_n))$ and $E'_i = v'_0 v'_1 \cdots v'_{i-1}[\cdot] \cdots \rho_2(\gamma_2(t'_n))$.

Where $(W, v_0, v'_0) \in \mathcal{V}[(\tau_1 \cdots \tau_n) \rightarrow \tau'] \rho$ and $(W, v_i, v'_i) \in \mathcal{V}[\tau_i] \rho$.

Once we show that

$$(W, E_n[v_n], E'_n[v'_n]) = (W, v_0 v_1 \cdots v_n, v'_0 v'_1 \cdots v'_n) \in \mathcal{E}[\text{out} \vdash \tau'; \sigma_n] \rho$$

all prior applications of Lemma 3.26 follow.

We instantiate $(W, v_0, v'_0) \in \mathcal{V}[(\tau_1 \cdots \tau_n) \rightarrow \tau'] \rho$ with $W'$, SR = $(\rho_1(\sigma_n), \rho_2(\sigma_n), \mathcal{S}[\sigma_n] \rho)$, $\overline{v}$, $\overline{v'}$ to yield the desired result.

**Lemma 3.81 (F Stack Modifying Application)**

If $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t \approx t': (\tau_1 \cdots \tau_n) \xrightarrow{\phi_i(\phi'_0)} \tau'; \sigma_0$, and $\Psi; \Delta; \Gamma; \chi; \sigma_{i-1}; \text{out} \vdash t_i \approx t'_i: \tau_i; \sigma_i$ for $i \in \{1, \ldots, n\}$, where $\sigma_n = \phi_i :: \sigma$ and $\sigma' = \phi'_0 :: \sigma'$, then $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t_1 \cdots t_n \approx t'_1 \cdots t'_n: \tau'; \sigma'$.

**Proof**

Consider arbitrary $W$, $\rho$, and $\gamma$ such that $W \in \mathcal{H}[\Psi]$, $\rho \in D[\Delta]$, $\gamma \in \mathcal{G}[\Gamma]$, currentMR($W(i_{reg})$) $\in W \mathcal{R}[\chi] \rho$, and currentMR($W(i_{stk})$) $\in W \mathcal{S}[\sigma_0] \rho$.

We need to show that

$$(W, \rho_1(\gamma_1(t_1 \cdots t_n)), \rho_2(\gamma_2(t'_1 \cdots t'_n))) = (W, \rho_1(\gamma_1(t)) \rho_1(\gamma_1(t_1)) \cdots \rho_1(\gamma_1(t_n)), \rho_2(\gamma_2(t')) \rho_2(\gamma_2(t'_1)) \cdots \rho_2(\gamma_2(t'_n)))$$

$\in \mathcal{E}[\text{out} \vdash \tau'; \sigma'] \rho$

We proceed by applying Lemma 3.26 $n + 1$ times, with the first evaluation contexts being
$E_0 = [\cdot] \rho_1(\gamma_1(t_1)) \cdots \rho_1(\gamma_1(t_n))$ and
$E'_0 = [\cdot] \rho_2(\gamma_2(t'_1)) \cdots \rho_2(\gamma_2(t'_n))$
And the $i$th being
$E_i = v_0 v_1 \cdots v_{i-1}[\cdot] \cdots \rho_1(\gamma_1(t_n))$ and
$E'_i = v'_0 \rho_2(\gamma_2(v'_i)) \cdots v'_{i-1}[\cdot] \cdots \rho_2(\gamma_2(t'_n))$
Where $(W, v_0, v'_0) \in \mathcal{V}[(\tau_1 \cdots \tau_n) \rightarrow \tau'] \rho$ and $(W, v_i, v'_i) \in \mathcal{V}[\pi_i] \rho$.
Once we show that
$$(W, E_n[v_n], E'_n[v'_n]) = (W, v_0 v_1 \cdots v_n, v'_0 v'_1 \cdots v'_n) \in \mathcal{E}[\tau \vdash \tau'; \sigma] \rho$$
all prior applications of Lemma 3.26 follow.

We instantiate $(W, v_0, v'_0) \in \mathcal{V}[(\pi_1 \cdots \pi_n) \rightarrow \pi'] \rho$ with $W'$, $SR = (\rho_1(\sigma_n), \rho_2(\sigma_n), S[\sigma_n] \rho)$ (noting that $\sigma_n = \phi_1 \vdash \zeta$, $\nu$, and $\bar{\nu}$) to yield the desired result.

**Lemma 3.82 (F Fold)**

If $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t_1 \approx t_2 : \tau[\mu \alpha. \tau/\alpha] ; \sigma_1$,
then $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash \text{fold}_{\mu \alpha. \tau} t_1 \approx \text{fold}_{\mu \alpha. \tau} t_2 : \mu \alpha. \tau ; \sigma_1$

**Proof**

Consider arbitrary $W, \rho,$ and $\gamma$ such that $W \in \mathcal{H}[\Psi], \rho \in D[\Delta], \gamma \in G[\Gamma], \text{currentMR}(W(i_{\text{reg}})) \in W \mathcal{R}[\chi] \rho$, and currentMR$(W(i_{\text{stk}})) \in W \mathcal{S}[\sigma_0] \rho$.

We proceed by applying Lemma 3.26, with evaluation contexts $E_i = \text{fold}_{\mu \alpha. \tau} [\cdot]$.

Given $(W, v_1, v_2) \in \mathcal{V}[\tau[\mu \alpha. \tau/\alpha]] \rho$, we must show $(W, \text{fold}_{\mu \alpha. \tau} v_1, \text{fold}_{\mu \alpha. \tau} v_2) \in \mathcal{E}[\text{out} \vdash \mu \alpha. \tau ; \sigma_1]$.

From Lemma 3.8, noting that $\triangledown W \vdash W$, we have that $(W, v_1, v_2) \in \triangledown \mathcal{V}[\tau[\mu \alpha. \tau/\alpha]] \rho$, which means $(W, \text{fold}_{\mu \alpha. \tau} v_1, \text{fold}_{\mu \alpha. \tau} v_2) \in \mathcal{V}[\mu \alpha. \tau] \rho$ and the result follows from Lemma 3.25.

**Lemma 3.83 (F Unfold)**

If $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t_1 \approx t_2 : \mu \alpha. \tau ; \sigma_1$,
then $\Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash \text{unfold} t_1 \approx \text{unfold} t_2 : \tau[\mu \alpha. \tau/\alpha] ; \sigma_1$

**Proof**

Consider arbitrary $W, \rho,$ and $\gamma$ such that $W \in \mathcal{H}[\Psi], \rho \in D[\Delta], \gamma \in G[\Gamma], \text{currentMR}(W(i_{\text{reg}})) \in W \mathcal{R}[\chi] \rho$, and currentMR$(W(i_{\text{stk}})) \in W \mathcal{S}[\sigma_0] \rho$.

We proceed by applying Lemma 3.26, with evaluation contexts $E_i = \text{unfold} [\cdot]$.

Given $(W, v_1, v_2) \in \mathcal{V}[\mu \alpha. \tau] \rho$, we must show $(W, \text{unfold} v_1, \text{unfold} v_2) \in \mathcal{E}[\text{out} \vdash \tau[\mu \alpha. \tau/\alpha] ; \sigma_1]$.

From the definition of $\mathcal{V}[\mu \alpha. \tau] \rho$, we know that the above is equivalent to $(W, \text{unfold fold}_{\mu \alpha. \tau} v'_1, \text{unfold fold}_{\mu \alpha. \tau} v'_2) \in \mathcal{E}[\text{out} \vdash \tau[\mu \alpha. \tau/\alpha] ; \sigma_1]$, where $(W, v'_1, v'_2) \in \triangledown \mathcal{V}[\tau[\mu \alpha. \tau/\alpha]] \rho$.

And we can then appeal to Lemma 3.4.3, noting that Lemma 3.25 provides us with the needed condition once we take one reduction step.

**Lemma 3.84 (F Tuple)**

If $\Psi; \Delta; \Gamma; \chi; \sigma_{i-1}; \text{out} \vdash t_i \approx t'_i ; \tau_i ; \sigma_i$ for $i \in \{1, \ldots, n\}$,
then $\Psi; \Delta; \Gamma; \chi; \sigma_{n-1}; \text{out} \vdash (t_1, \ldots, t_n) \approx (t'_1, \ldots, t'_n) : \langle \tau_1, \ldots, \tau_n \rangle ; \sigma_n$.

**Proof**

133
Consider arbitrary \( W, \rho, \) and \( \gamma \) such that \( W \in \mathcal{H}[[\Psi]], \rho \in D[[\Delta]], \gamma \in G[[\Gamma]], \) currentMR\((W(i_{\text{reg}})) \in W \mathcal{R}[[\chi]]\rho, \) and currentMR\((W(i_{\text{stk}})) \in W \mathcal{S}[[\sigma_{n-1}]]\rho. \)

We must show that
\[
(W, \rho_1(\gamma_1((t_1, \ldots, t_n))), \rho_2(\gamma_2((t'_1, \ldots, t'_n)))) \in \mathcal{E}[[\text{out} \vdash \text{int}; \sigma_2]]
\]

We proceed by pushing the substitutions in and applying Lemma 3.26 \( n \) times, with \( i \)th evaluation contexts \( E_{i1} = \langle v_1, \ldots, [\cdot], \ldots, \rho_1(\gamma_1(t_n)) \rangle \) and \( E_{2i} = \langle v'_1, \ldots, [\cdot], \ldots, \rho_2(\gamma_2(t'_n)) \rangle \)

Where \( W, \rho_1, \rho_2 \) and \( \gamma_2(\gamma_2(t'_n)) \)

We then must show that \( (W, \langle v_1, \ldots, v_n \rangle, \langle v'_1, \ldots, v'_n \rangle) \in \mathcal{E}[[\text{out} \vdash \langle \tau_1, \ldots, \tau_n \rangle; \sigma_n]]\rho. \)

But this follows from Lemma 3.25.

\[ \square \]

**Lemma 3.85 (F Projection)**

If \( \Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash t_1 \approx t_2 : \langle \tau_1, \ldots, \tau_n \rangle; \sigma_1, \)
then \( \Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash \pi_i(t_1) \approx \pi_i(t_2) : \tau_i; \sigma_1. \)

**Proof**

Consider arbitrary \( W, \rho, \) and \( \gamma \) such that \( W \in \mathcal{H}[[\Psi]], \rho \in D[[\Delta]], \gamma \in G[[\Gamma]], \) currentMR\((W(i_{\text{reg}})) \in W \mathcal{R}[[\chi]]\rho, \) and currentMR\((W(i_{\text{stk}})) \in W \mathcal{S}[[\sigma_0]]\rho. \)

We must show that
\[
(W, \rho_1(\gamma_1(\pi_i(t_1))), \rho_2(\gamma_2(\pi_i(t_2)))) = (W, \pi_i(\rho_1(\gamma_1(t_1))), \pi_i(\rho_2(\gamma_2(t_2)))) \in \mathcal{E}[[\text{out} \vdash \tau_i; \sigma_1]]\rho
\]

We appeal to Lemma 3.26, with \( E_i = \pi_i([\cdot]). \) We must show that
\[
(W, \pi_i(v_1), \pi_i(v_2)) \in \mathcal{E}[[\tau_i]]\rho \text{ given that } (W, v_1, v_2) \in \mathcal{V}[[\langle \tau_1, \ldots, \tau_n \rangle]]\rho
\]

We appeal to Lemma 3.13, noting that the definition of \( \mathcal{V}[[\langle \tau_1, \ldots, \tau_n \rangle]]\rho \) and Lemma 3.25 yields the desired result.

\[ \square \]

**Lemma 3.86 (FT Boundary)**

If \( \Psi; \Delta; \Gamma; \chi; \sigma_0; \text{end}\{\tau^T; \sigma_1\} \vdash e_1 \approx e_2 : \tau^T; \sigma_1, \)
then \( \Psi; \Delta; \Gamma; \chi; \sigma_0; \text{out} \vdash \tau^T e_1 \approx \tau^T e_2 : \tau^T; \sigma_1. \)

**Proof**

Consider arbitrary \( W, \rho, \) and \( \gamma \) such that \( W \in \mathcal{H}[[\Psi]], \rho \in D[[\Delta]], \gamma \in G[[\Gamma]], \) currentMR\((W(i_{\text{reg}})) \in W \mathcal{R}[[\chi]]\rho, \) and currentMR\((W(i_{\text{stk}})) \in W \mathcal{S}[[\sigma]]\rho. \)

We need to show that
\[
(W, \rho_1(\gamma_1(\tau^T e_1)), \rho_2(\gamma_2(\tau^T e_2))) = (W, \rho_1(\tau^T e_1), \rho_2(\tau^T e_2)) \in \mathcal{E}[[\text{out} \vdash \tau; \sigma_1]]
\]

From the hypothesis, we know that \( (W, \rho_1(\gamma_1(e_1)), \rho_2(\gamma_2(e_2))) \in \mathcal{E}[[\text{end}\{\tau^T; \sigma_1\} \vdash \tau^T; \sigma_1]]. \)

The result is then immediate from Lemma 3.28, 1(a).

\[ \square \]
3.7 Fundamental Property and Soundness

Lemma 3.87 (Fundamental Property) • If \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash e : \tau ; \sigma' \) then \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash e \approx e : \tau ; \sigma' \).
  • If \( \Psi \vdash H : \Psi' \) then \( \Psi \vdash H \approx_H H : \Psi' \).
  • If \( \Psi \vdash h : \nu \psi \) then \( \Psi \vdash h \approx_{\nu \psi} h : \nu \psi \) def.
  • If \( \Psi ; \Delta \vdash w : \tau \) \& \( \Psi \vdash w \approx_w w : \tau \).
  • If \( \Psi ; \Delta ; \chi \vdash u : \tau \) then \( \Psi ; \Delta ; \chi \vdash u \approx_u u : \tau \).
  • If \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash I \) then \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash I \approx_I I \).

Proof

We prove all the claims simultaneously, by induction on the typing derivations, using the compatibility lemmas.

Lemma 3.88 (Weakening)
If \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash e_1 \approx e_2 : \tau ; \sigma' \) and \( \Psi \subseteq \Psi' \), \( \Delta \subseteq \Delta' \), \( \Gamma \subseteq \Gamma' \), \( \chi \subseteq \chi' \), then \( \Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma ; q \vdash e_1 \approx e_2 : \tau ; \sigma' \).

Proof

Let \( W \in \mathcal{H}[\Psi] \), \( \rho' \in \mathcal{D}[\Delta] \), \( (W, \gamma') \in \mathcal{G}[\Gamma] \rho' \) such that currentMR\((W(i_{reg}))\) \( \subseteq_W R[\chi']\rho' \) and currentMR\((W(i_{stk}))\) \( \subseteq_W S[\sigma]\rho' \).

Let \( \rho = \rho'|\Delta \) and \( \gamma = \gamma'|\Gamma \). Note that \( W \in \mathcal{H}[\Psi] \) and \( \rho \in \mathcal{D}[\Delta] \) immediately.

We must further show:

1. currentMR\((W(i_{reg}))\) \( \subseteq_W R[\chi]\rho \)
2. currentMR\((W(i_{stk}))\) \( \subseteq_W S[\sigma]\rho \)
3. \((W, \gamma) \in \mathcal{G}[\Gamma]\rho \)

Since the free type variables in \( \chi \) are in \( \Delta \), \( R[\chi]\rho' = R[\chi]\rho \), after which the first follows from the definition.

Similarly, the free type variables in \( \sigma \) are in \( \Delta \), after which the second follows.

Finally, we must show that \((W, \gamma) \in \mathcal{G}[\Gamma]\rho \). But clearly \((W, \gamma) \in \mathcal{G}[\Gamma]\rho' \) and since the free type variables in \( \Gamma \) are in \( \Delta \), \( \mathcal{G}[\Gamma]\rho' = \mathcal{G}[\Gamma]\rho \), so we are done.

Lemma 3.89 (Congruence)
If \( \Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash e_1 \approx e_2 : \tau ; \sigma' \) and \( \vdash C : (\Psi ; \Delta ; \Gamma ; \chi ; \sigma ; q \vdash \tau ; \sigma') \rightarrow (\Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q' \vdash \tau ; \sigma_1) \), then \( \Psi' ; \Delta' ; \Gamma' ; \chi' ; \sigma_0 ; q \vdash C[e_1] \approx C[e_2] : \tau ; \sigma_1 \).

Proof

This follows by induction on the type derivation for \( C \), using Lemma 3.88 for the cases when \( C \) is empty, and the compatibility lemmas in the rest of the cases.

Lemma 3.90 (Canonical World)
If \( \vdash M = (H, R, S) : (\Psi, \chi, \sigma) \), then for any \( k \), \( \exists W. W.k = k \wedge W \in \mathcal{H}[\Psi] \wedge \text{currentMR}(W(i_{reg})) \subseteq_W R[\chi] \wedge \text{currentMR}(W(i_{stk})) \subseteq_W S[\sigma] \wedge (M, M) : W \).

Proof
Say that $\Psi = \Psi', \ell_1 : \text{ref } \psi_1, \ldots, \ell_n : \text{ref } \psi_n$ where boxheap($\Psi'$). Let

$$\theta_i = (\bullet, \bullet, \emptyset, \emptyset, \lambda s.((W', M_1, M_2) \in \text{MemAtom}_k | (W', M_1(\ell_i), M_2(\ell_i)) \in \mathcal{H}[\psi_i]|\emptyset, \lambda s.\{(\ell_i, \ell_i)\})$$

for $1 \leq i \leq n$. We construct

$$W = (k, \Psi, \Psi, (\text{island}_{reg}((R, \chi, R, \chi), k), \text{island}_{stk}((S, \sigma, S, \sigma), k), \text{island}_{box}((H|\Psi'|, H|\Psi'), k), \theta_1, \ldots, \theta_n)$$

We need to show the following:

- For each $\ell : \text{box } \psi \in \Psi'$, $(W, \ell, \ell) \in W[\text{box } \psi]|\emptyset$,
- For each $i$, $(W, \ell_i, \ell_i) \in W[\text{ref } \psi]|\emptyset$,
- currentMR($W(i_{\text{reg}})$) $\subseteq_W \mathcal{R}[\chi]$,
- currentMR($W(i_{\text{stk}})$) $\subseteq_W \mathcal{S}[\sigma]$,
- $(M, M) : W$

All of these follow from the respective definitions and the Fundamental Property for word and heap values.

\[\square\]

**Lemma 3.91 (Adequacy)**

If $\Psi; \vdash \vdash \chi; \sigma; q \vdash e_1 \approx e_2 : \tau; \sigma'$, then $(M | e_1) \downarrow$ if and only if $(M | e_2) \downarrow$.

**Proof**

We show that $(M | e_1) \downarrow$ implies $(M | e_2) \downarrow$, and the converse holds by an identical argument.

Suppose $(M | e_1) \downarrow^k$. By Lemma 3.90, there is some $W \in \mathcal{H}[\psi]$ with currentMR($W(i_{\text{reg}})$) $\subseteq_W \mathcal{R}[\chi] \land$ currentMR($W(i_{\text{stk}})$) $\subseteq_W \mathcal{S}[\sigma]$ such that $(M, M) : W$ and $W.K \geq k$.

So by our assumption, $(W, e_1, e_2) \in E[q \vdash \tau; \sigma']|\emptyset$. We claim that $(W, E, E) \in \mathcal{K}[q \vdash \tau; \sigma']|\emptyset$, where

$$E = \begin{cases} [\cdot] & \tau = \tau \\ ([\cdot], \cdot) & \tau = \tau \end{cases}$$

If the claim holds, then $(W, E[e_1], E[e_2]) = (W, e_1, e_2) \in \mathcal{O}$. Since running($W.K, (M | e_1)$) contradicts our assumption, we must have $(M | e_2) \downarrow^k$, as desired.

To prove the claim, we must consider the two types of continuations, depending on whether $\tau = \tau$ or $\tau = \tau$.

In the first case, let $W' \equiv_{\text{pub}} W$ and consider $(W', \psi_1, \psi_2) \in \mathcal{V}[\tau]|\emptyset$, such that currentMR($W'(i_{\text{stk}})$) $\subseteq_W \mathcal{S}[\sigma]|\emptyset$. In that case

$$(W', E[\psi_1], E[\psi_2]) \in \mathcal{O}$$

trivially.

In the second case, the result is essentially as trivial, since while value forms are more complex in $T$, they are similarly observationally equivalent.

\[\square\]

**Lemma 3.92 (Logical Equivalence Implies Contextual Equivalence)**

If $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx e_2 : \tau; \sigma'$ then $\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx e_2 : \tau; \sigma'$.
Proof

Let \( C : (\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \tau; \delta) \rightsquigarrow (\Psi'; \ldots; \chi'; \sigma'; q' \vdash \tau'; \delta') \) and \( \vdash M : (\Psi', \chi', \sigma') \).

By congruence (Lemma 3.89), \( \Psi' ; \ldots ; \chi ; \sigma ; q \vdash C[e_1] \approx C[e_2] : \tau ; \sigma' \).

By adequacy (Lemma 3.91), \( \langle M \mid C[e_1] \rangle \downarrow \) if and only if \( \langle M \mid C[e_2] \rangle \downarrow \), as desired.

\( \square \)

3.8 Completeness

Lemma 3.93 (Contextual Equivalence Implies CIU Equivalence)

If \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{\text{ctx}} e_2 : \tau ; \sigma' \) then

\( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{\text{ciu}} e_2 : \tau ; \sigma' \).

Proof

We have that \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 : \tau ; \sigma' \), \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_2 : \tau ; \sigma' \), and

\[
\forall C, M, \Psi', \chi', \sigma', q', \tau', \delta'. \quad \vdash C : (\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \tau; \delta) \rightsquigarrow (\Psi'; \ldots; \chi'; \sigma'; q' \vdash \tau'; \delta') \quad \wedge \quad \vdash M : (\Psi', \chi', \sigma')
\]

\[
\implies (\langle M \mid C[e_1] \rangle \downarrow \iff \langle M \mid C[e_2] \rangle \downarrow)
\]

We need to show that

\[
\forall \delta, \gamma, E, M, \Psi_E, q_E, \tau_E, \sigma_E, \cdot \vdash \delta : \Delta \wedge \Psi_E ; \ldots ; \cdot ; \cdot ; \cdot ; \cdot ; \text{out} \vdash \gamma : \Gamma \wedge
\]

\[
\vdash E : (\Psi_E ; \ldots ; \chi_E ; \sigma_E ; q_E \vdash \tau_E ; \delta_E) \rightsquigarrow (\Psi'_E ; \ldots ; \chi'_E ; \sigma_E ; q'_E \vdash \tau'_E ; \delta'_E) \wedge \vdash M : (\Psi'_E, \chi_E, \sigma_E)
\]

\[
\implies (\langle M \mid E[\delta(\gamma(e_1))] \rangle \downarrow \iff \langle M \mid E[\delta(\gamma(e_2))] \rangle \downarrow)
\]

Assume all the premises in the implication. It suffices to find a \( C \) such that co-termination of \( \langle M \mid C[e_1] \rangle \) and \( \langle M \mid C[e_2] \rangle \) is equivalent to co-termination of \( \langle M \mid E[\delta(\gamma(e_1))] \rangle \) and \( \langle M \mid E[\delta(\gamma(e_2))] \rangle \).

We need a \( C \) such that:

\[
\vdash C : (\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash \tau; \delta) \rightsquigarrow (\Psi'_E; \ldots; \chi'_E; \sigma_E; q'_E \vdash \tau'_E; \delta'_E)
\]

This amounts to constructing a syntactically valid \( C \) that accomplishes what \( E[\delta(\gamma([\cdot])]) \) would do, were it a valid context, which is essentially an eta-expansion with boundaries inserted appropriately.

Let

\[
C_{\text{inner}} = \begin{cases} [] & \text{if } \tau = \tau' \\ \tau \mathcal{FT} [\cdot] & \text{if } \tau = \tau' \end{cases}
\]

\[
C_{\text{gamma}} = (\lambda(\text{dom}(\Gamma)).C_{\text{inner}}) \gamma(\text{dom}(\Gamma))
\]

\[
C_{\text{delta}} = (\text{jmp } \ell[\delta(\text{dom}(\delta))], \\
\ell \mapsto \text{code}(\Delta)\{\cdot ; \cdot \}^q \text{import } r_d, \sigma \mathcal{FT} \mathcal{R} \quad C_{\text{gamma}}) \\
\quad \text{ret } q \{ r_d \}
\]

\[
C_{\text{outer}} = \begin{cases} C_{\text{delta}} & \text{if } \tau = \tau' \\ \tau \mathcal{FT} C_{\text{delta}} & \text{if } \tau = \tau' \end{cases}
\]

\[
C = E[C_{\text{outer}}]
\]

By inspection of the operational semantics,
Proof

We have that \( \Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{\text{eq}} e_2 : \tau; \sigma' \), and

\[
\forall \delta, \gamma, E, M, \Psi_E', \sigma_E', \delta_E', \vdash \delta : \Delta \land \Psi_E' ; \cdots ; \bullet ; \text{out} \vdash \gamma : \Gamma ; \bullet \land
\]

\[
\vdash E : (\Psi_E' ; \cdots ; \chi; \sigma; q \vdash \tau; \sigma) \Rightarrow (\Psi_E' ; \cdots ; \chi; \sigma; q_E' \vdash \tau_E' ; \delta_E') \land \vdash M : (\Psi_E', \chi; \sigma)
\]

\[
\Rightarrow ((M \mid E[\delta(\gamma(e_1))]) \downarrow \iff (M \mid E[\delta(\gamma(e_2))]) \downarrow
\]

We need to show that

\[
\forall W, \rho. W \in H[\Psi] \land \rho \in D[\Delta] \land
\]

\[
currentMR(W(i_reg)) \in_W R[\chi]\rho \land
\]

\[
currentMR(W(i_stk)) \in_W S[\sigma]\rho
\]

\[
\Rightarrow (W, \rho_1(e_1), \rho_2(e_2)) \in E[q \vdash \tau; \sigma']\rho
\]

Assume all the premises of this implication.

Let \((W, E_1, E_2) \in K[q \vdash \tau; \sigma']\rho\). We need to show that \((W, E_1[\rho_1(\gamma_1(e_1))], E_2[\rho_2(\gamma_2(e_2))]) \in \mathcal{O}\).

Let \((M_1, M_2) : W\). It suffices to show that

\[
(M_1 \mid E_1[\rho_1(\gamma_1(e_1))]) \downarrow \iff (M_2 \mid E_2[\rho_2(\gamma_2(e_2))]) \downarrow.
\]

By the Fundamental Property, \(\Psi; \Delta; \chi; \sigma; q \vdash e_1 \approx_{\text{eq}} e_1 : \tau; \sigma'\). Therefore,

\[
(W, \rho_1(\gamma_1(e_1)), \rho_2(\gamma_2(e_1))) \in E[q \vdash \tau; \sigma']\rho
\]

and thus

\[
(M_1 \mid E_1[\rho_1(\gamma_1(e_1))]) \downarrow \iff (M_2 \mid E_2[\rho_2(\gamma_2(e_1))]) \downarrow.
\]

It remains to show that

\[
(M_1 \mid E_1[\rho_1(\gamma_1(e_1))]) \downarrow \iff (M_2 \mid E_2[\rho_2(\gamma_2(e_2))]) \downarrow.
\]

But this follows from our hypothesis that \(\Psi; \Delta; \Gamma; \chi; \sigma; q \vdash e_1 \approx_{\text{eq}} e_2 : \tau; \sigma'\).

\[ \Box \]
3.9 Examples

In some of the following examples, we will use the following syntactic shorthands to improve readability:

\[
\text{let } x = e_1 \text{ in } e_2 \text{ means } (\lambda(x :: t). e_2) e_1 \text{ where } e_1 :: t
\]

\[
e_1 ; e_2 \text{ means } (\lambda(\text{ignored} :: t_1, \text{result} :: t_2). \text{result}) e_1 e_2 \text{ where } e_1 :: t_1 \text{ and } e_2 :: t_2
\]

Noting that for any well-typed term we can mechanically synthesize its type, so elliding them in this context introduces no ambiguity.

3.9.1 Calculations with Different Number of Basic T Blocks

\[
f_1 = \lambda(x :: \text{int}).(\text{int}) \to \text{intFT} (\text{mv r1}; \ell; \text{ret end}\{\text{intT}; \cdot\} \{r1\}, H_1) x
\]

where

\[
H_1(\ell) = \text{code}[\zeta, \epsilon]\{\text{ra}; \forall[]\{r1 :: \text{intT}; \zeta\}^e; \text{intT} :: \zeta\}^{ra}.
\]

\[
\text{sld r1}, 0; + r1, r1, 1; + r1, r1, 1; \text{sfre 1}; \text{reta} \{r1\}
\]

\[
f_2 = \lambda(x :: \text{int}).(\text{int}) \to \text{intFT} (\text{mv r1}; \ell; \text{ret end}\{\text{intT}; \cdot\} \{r1\}, H_2) x
\]

where

\[
H_2(\ell) = \text{code}[\zeta, \epsilon]\{\text{ra}; \forall[]\{r1 :: \text{intT}; \zeta\}^e; \text{intT} :: \zeta\}^{ra}.
\]

\[
\text{sld r1}, 0; + r1, r1, 1; \text{sst 0}, r1; \text{jmp } \ell'[\zeta][\epsilon]
\]

\[
H_2(\ell') = \text{code}[\zeta, \epsilon]\{\text{ra}; \forall[]\{r1 :: \text{intT}; \zeta\}^e; \text{intT} :: \zeta\}^{ra}.
\]

\[
\text{sld r1}, 0; + r1, r1, 1; \text{sfre 1}; \text{reta} \{r1\}
\]

**Claim 3.95**

The two functions are equivalent, i.e., \( \cdots; \cdots; \cdot \text{out} \vdash f_1 \approx f_2 :: (\text{int}) \to \text{int} :: \cdot \), which shows that we can reason about assembly components that have similar externally visible behavior but different structure and control flow.

**Proof**

Expanding the definition of \( \approx \), we see we need to show first that the terms are well-typed. This follows from the typing rules, noting in particular that:

\[
(\text{int}) \to \text{intFT} (\text{mv r1}; \ell; \text{ret end}\{\text{intT}; \cdot\} \{r1\}, H)
\]

Requires that the inner component have translation type

\[
(\text{int}) \to \text{intT} = \text{box } \forall[\zeta, \epsilon].\{\text{ra}; \forall[]\{r1 :: \text{intT}; \zeta\}^e; \text{intT} :: \zeta\}^{ra}
\]

And the heap must be well-typed, at the type that we’ll denote \( \Psi \):

\[
\cdot \vdash H :: \Psi
\]

For program \( f_2 \), checking that the heap is well-typed involves checking the \( \text{jmp } \ell'[\zeta][\epsilon] \) instruction. At that point, the register file typing is the following:

\[
\chi = \{r1 :: \text{int}, \text{ra}; \forall[]\{r1 :: \text{int}; \zeta\}^e\}
\]

The full typing rule that we must satisfy at this point is:

\[
\Psi; \zeta, \epsilon; \chi \vdash \ell'[\zeta][\epsilon] :: \text{box } \forall[]\{\text{ra}; \forall[]\{r1 :: \text{int}; \zeta\}^e; \text{int} :: \zeta\}^{ra}
\]

\[
\zeta, \epsilon \vdash \chi \leq \{\text{ra}; \forall[]\{r1 :: \text{int}; \zeta\}^e\} \cdot [\zeta, \epsilon]; \chi :: \text{int} :: \zeta \vdash \text{ra}
\]

\[
\Psi; \zeta, \epsilon; \chi :: \zeta; \text{ra} \vdash \text{jmp } \ell'[\zeta][\epsilon]
\]
Which can easily be seen to be true.

We further must show that given a suitable world $W$,

$$(W, f_1, f_2) \in E[\text{out} \vdash (\text{int} \to \text{int})]$$

For this, we must show:

$$(E_1, E_2) \in K[\text{out} \vdash (\text{int} \to \text{int})] \implies (W, E_1[f_1], E_2[f_2]) \in O$$

From the definition of related contexts, it suffices to show $(W, f_1, f_2) \in V[(\text{int}) \to \text{int}]$.

In order to do this, given

- $\text{SR} \in TStackRel$
- $\rho = [\zeta \mapsto \text{SR}]$
- integer $v$ (for which $(W, v, v) \in V[\text{int}]$ trivially)
- World $W'$ such that $W' \equiv W$ and currentMR($W'(i_{stk})$) $\in W$. $S[\zeta] \rho$

We must show:

$$(W', f_1v, f_2v) \in E[\text{out} \vdash \zeta; \rho]$$

As before, to do this we must show:

$$(W', E'_1, E'_2) \in K[\text{out} \vdash \zeta; \rho] \implies (W', E'_1[f_1v], E'_2[f_2v]) \in O$$

In order to do this, consider arbitrary memories $(M_1, M_2) : W'$.

We will then show that

$$\langle M_1 | f_1v \rangle \rightarrow j_1 \langle M'_1 | v_1 \rangle$$
$$\langle M_2 | f_2v \rangle \rightarrow j_2 \langle M'_2 | v_2 \rangle$$

Where $(M'_1, M'_2) : W^*$ for some $W^* \equiv W'$, which we can lift to contexts to show that:

$$\langle M_1 | E'_1[f_1v] \rangle \rightarrow \langle M'_1 | E'_1[v_1] \rangle$$
$$\langle M_2 | E'_2[f_2v] \rangle \rightarrow \langle M'_2 | E'_2[v_2] \rangle$$

We can then instantiate the contexts to show that $(W^*, E'_1[v_1], E'_2[v_2]) \in O$, which we can then compose together to show that $(W', E'_1[f_1v], E'_2[f_2v]) \in O$.

We argue the central reduction by appealing to the operational semantics, which will show that:

$$\langle M_1 | f_1v \rangle \rightarrow j_1 \langle M'_1 | v_1 \rangle$$
$$\langle M_2 | f_2v \rangle \rightarrow j_2 \langle M'_2 | v_2 \rangle$$

Consider the first program, letting $M_1 = (H, R, S)$. That is, we want to show that:

$$\langle M_1 | f_1v \rangle \rightarrow j_1 \langle M'_1 | v_1 \rangle$$

For some $v_1$.  

140
First we combine the heap fragment, reduce and then carry out the value translation for the $T$ codeblock to an $F$ function.

$$\langle H, R, S \mid f_1 \psi \rangle \leadsto^* \langle f_1 \psi \rangle \mapsto_{\text{int}^F} (\text{protect } \cdot, \zeta;$$

$$\lambda(x : \text{int}). \text{int}F^T (\text{import } r_1, \cdot \text{ int } x;$$

$$\text{salloc } l_1; \text{sst } 0, r_1;$$

$$\text{mv } ra, \ell_{\text{end}}[\zeta]; \text{jmp } \ell[\zeta][\text{end}\{\text{int}^T; \zeta\}];.,)$$

This further reduces:

$$\langle (H \sqcup H_1, R, S) \mid \lambda(x : \text{int}). \text{int}F^T (\text{import } r_1, \cdot \text{ int } x;$$

$$\text{salloc } l_1; \text{sst } 0, r_1;$$

$$\text{mv } ra, \ell_{\text{end}}[\zeta]; \text{jmp } \ell[\zeta][\text{end}\{\text{int}^T; \zeta\}];.,)$$

$$\leadsto^*$$

$$\langle (H \sqcup H_1, R[r_1 \mapsto v, ra \mapsto \ell_{\text{end}}[\zeta], v :: S) \mid \text{int}F^T (\text{import } r_1, \cdot \text{ int } x;$$

$$\text{salloc } l_1; \text{sst } 0, r_1;$$

$$\text{mv } ra, \ell_{\text{end}}[\zeta]; \text{jmp } \ell[\zeta][\text{end}\{\text{int}^T; \zeta\}];.,)$$

At which point the top of the stack contains the argument $v$, the register $ra$ contains the address of the return continuation created by the value translation, and we can jump to the code at $\ell$.

This reduces as:

$$\langle (H \sqcup H_1, R[r_1 \mapsto v, ra \mapsto \ell_{\text{end}}[\zeta], v :: S) \mid \text{int}F^T (\text{import } r_1, \cdot \text{ int } x;$$

$$\text{salloc } l_1; \text{sst } 0, r_1;$$

$$\text{mv } ra, \ell_{\text{end}}[\zeta]; \text{jmp } \ell[\zeta][\text{end}\{\text{int}^T; \zeta\}];.,)$$

Which we can then see will load the value $v$ off the top of the stack, add two to it, and return:

$$\langle (H \sqcup H_1, R[r_1 \mapsto v, ra \mapsto \ell_{\text{end}}[\zeta], v :: S) \mid \text{int}F^T (\text{import } r_1, \cdot \text{ int } x;$$

$$\text{salloc } l_1; \text{sst } 0, r_1;$$

$$\text{mv } ra, \ell_{\text{end}}[\zeta]; \text{jmp } \ell[\zeta][\text{end}\{\text{int}^T; \zeta\}];.,)$$

At which point, we jump to the return continuation, which then steps to $\text{ret end}\{\text{int}; \zeta\} \{r_1\}$, which allows the boundary to cancel:
\[ \langle H \uplus H_1, R[r1 \mapsto v+2, ra \mapsto \ell_{\text{end}}(\zeta)], S \rangle \mid ^{\text{int}F_T} (\text{ret} \{r1\}, \cdot) \]

\[ \mapsto \]

\[ \langle H \uplus H_1, R[r1 \mapsto v+2, ra \mapsto \ell_{\text{end}}(\zeta)], S \rangle \mid ^{\text{int}F_T} (\text{ret} \{\text{int}^T; \zeta\} \{r1\}, \cdot) \]

\[ \mapsto \]

\[ \langle (H \uplus H_1, R[r1 \mapsto v+2, ra \mapsto \ell_{\text{end}}(\zeta)], S) \mid v+2 \rangle \]

Which completes the reduction of the first program.

The proof for the reduction of \( f_2v \) proceeds similarly at first, but the first block jumps to the second with \( v+1 \) stored at the top of the stack (replacing \( v \)). The second block is analogous to the first function, except it only adds 1 once, so the reduction proceeds as (here letting \( M_2 = (H, R, S) \)):

\[ \langle (H, R, S) \mid f_2v \rangle \mapsto \langle (H \uplus H_2, R[r1 \mapsto v+2, ra \mapsto \ell_{\text{end}}(\zeta)], S) \mid v+2 \rangle. \]

\( v+2 \) and \( v+2 \) are trivially related, which means in the world \( W^* \) that the final memories fulfill (which only varies in private merged heaps and registers from the original \( W' \)),

\[ (W^*, E'_1[v+2], E'_2[v+2]) \in O \]

This can then be composed with the above reduction lifted to contexts (ie, \( E'_i[f'v] \mapsto E'_i[v+2] \)) to yield the result.

\[ \Box \]

### 3.9.2 Factorial Two Ways

\[ \text{fact}_F = \lambda(x : \text{int}).(F \text{fold}_{\mu\alpha.(\alpha \rightarrow \text{int})} F) x \]

where

\[ F = \lambda(f : \mu\alpha.(\alpha \rightarrow \text{int}).\lambda(x : \text{int}).\text{if}0 x 1 ((\text{unfold} f f) (x - 1)) \ast x \]

\[ \text{fact}_T = \lambda(x : \text{int}).(^{\text{int}}{\text{F}^T} (\ell_{\text{fact}}, H) x \]

where

\[ H(\ell_{\text{fact}}) = \text{code}[\zeta, e]\{ra : \forall[.]\{r1 : \text{int}^T; \zeta\}^e; \text{int}^T :: \zeta\}^r_a. \]

\[ \text{sld} \text{rn}, 0; \text{mv} \text{rr}, 1; \text{bnz} \text{rn}, \ell_{\text{aux}}[\zeta][e]; \text{sfree} 1; \text{ret} \{\text{rr} \}
\]

\[ H(\ell_{\text{aux}}) = \text{code}[\zeta, e]\{rr : \text{int}, r1 : \text{int}, \text{rn} : \text{int}, \text{ra} : \forall[.]\{r1 : \text{int}^T; \zeta\}^e; \text{int}^T :: \zeta\}^r_a. \]

\[ \ast \text{rr}, \text{rr}, \text{rn}; - \text{rn}, \text{rn}, 1; \text{bnz} \text{rn}, \ell_{\text{aux}}[\zeta][e]; \text{sfree} 1; \text{ret} \{\text{rr} \}
\]

Claim 3.96

Here we again claim equivalence, this time showing how a high-level functional implementation can be shown equivalent to a low-level implementation implemented with direct jumps and register mutation. This is interesting because the the first program uses recursive types but the control flow mechanisms in \( T \) make this not necessary in the second program.

\[ \vdash \vdash; \vdash; \vdash; \text{out} \vdash \text{fact}_F \approx \text{fact}_T : (\text{int}) \rightarrow \text{int}; \cdot \]

**Proof**

To prove this equivalence, we use the same general approach as the (much simpler) programs with different numbers of basic blocks shown equivalent in the previous section.

First we note that both terms are well-typed, which follows from the typing rules.

We must next show that given a suitable world \( W \),

\[ (W, \text{fact}_F, \text{fact}_T) \in \mathcal{E}[\text{out} \vdash (\text{int}) \rightarrow \text{int}; \cdot] \]

142
For this, we must show:

$$(E_1, E_2) \in K[\text{out} \vdash \text{int} ; \cdot] \implies (W, E_1[\text{fact}_F], E_2[\text{fact}_T]) \in \mathcal{O}$$

From the definition of related contexts, it suffices to show $$(W, \text{fact}_F, \text{fact}_T) \in V[\text{int} \vdash \text{int}]$$. In order to do this, given

- $$SR \in T\text{StackRel}$$
- $$\rho = [\zeta \mapsto SR]$$
- integer $$v$$ (for which $$(W, v, v) \in V[\text{int}]$$ trivially)
- World $$W'$$ such that $$W' \sqsubseteq W$$ and currentMR$$(W'(i_{stk})) \in W' S[\zeta]$$

We must show:

$$(W', \text{fact}_F v, \text{fact}_T v) \in E[\text{out} \vdash \text{int}; \zeta]\rho$$

As before, to do this we must show:

$$(W', E'_1, E'_2) \in K[\text{out} \vdash \text{int}; \cdot] \rho \implies (W', E'_1[\text{fact}_F v], E'_2[\text{fact}_T v]) \in \mathcal{O}$$

At this point, deviating from the previous example, we consider two cases. In the case that $$v \geq 0$$, we will prove this in a similar manner to the previous example, showing that:

$$\langle M_1 | \text{fact}_F v \rangle \mapsto \ast \langle M'_1 \mid v_1 \rangle\n\langle M_2 \mid \text{fact}_T v \rangle \mapsto \ast \langle M'_2 \mid v_2 \rangle$$

Where $$v_1$$ and $$v_2$$ are related (in particular, are the same integer), at which point we will show that $$(M'_1, M'_2) : W^*$$ for some $$W^* \sqsubseteq W'$$, from which the rest follows as in the previous example.

If, on the other hand, $$v < 0$$, both programs will diverge, so for arbitrary $$W.k$$ we will show that

$$\text{running}(k, \langle M_1 \mid \text{fact}_F v \rangle) \land \text{running}(k, \langle M_2 \mid \text{fact}_T v \rangle)$$

Which is equivalent to

$$\text{running}(k, \langle M_1 \mid E'_1[\text{fact}_F v] \rangle) \land \text{running}(k, \langle M_2 \mid E'_2[\text{fact}_T v] \rangle)$$

Which is sufficient to show membership in $$\mathcal{O}$$.

We will prove the first case in a similar way to the previous example, by appealing to the operational semantics.

$$\langle M_1 \mid \text{fact}_F v \rangle$$

$$\mapsto \ast$$

$$\langle M_1 \mid \lambda(f : \mu \alpha.(\alpha) \rightarrow \text{int} \cdot \lambda(x : \text{int}). \text{if} 0 \cdot x 1 \cdot (\text{unfold} f f) (x - 1) \cdot x) (\text{fold}_{\mu \alpha.(\alpha) \rightarrow \text{int}} F) v \rangle$$

$$\mapsto \ast$$

$$\langle M_1 \mid \text{if} 0 \cdot v 1 \cdot (\text{unfold} (\text{fold}_{\mu \alpha.(\alpha) \rightarrow \text{int}} F) (\text{fold}_{\mu \alpha.(\alpha) \rightarrow \text{int}} F) (v - 1)) \cdot * \rangle$$

If $$v$$ is 0, we can see this will reduce to 1, and otherwise it will step further:
\( \langle M_1 \mid \text{if} 0 \ v \ 1 \ (\text{unfold} \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ F) \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ (v - 1) \ast v \rangle \)
\[\mapsto \ast \]
\( \langle M_1 \mid \text{if} 0 \ (v - 1) \ 1 \ (\text{unfold} \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ F) \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ (v - 2) \ast (v - 1) \ast v \rangle \)

Since we know \( v \geq 0 \), this will eventually reduce to:
\( \langle M_1 \mid \text{if} 0 \ v \ 1 \ (\text{unfold} \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ F) \ (f_{\mu \alpha.(\alpha)} \to \text{int}) \ (v - 1) \ast v \rangle \)
\[\mapsto \ast \]
\( \langle M_1 \mid 1 \ast 2 \ast \ldots \ast (v - 2) \ast (v - 1) \ast v \rangle \)

We now consider the second program, again for the case when \( v \geq 0 \).

Let \( (H_2, R_2, S_2) = M_2 \).

As in the previous example, first we combine the heap fragment, reduce and then carry out the value translation for the \( T \) codeblock to an \( F \) function, stepping to the point of jumping into the first code block.

\[ \langle (H_2, R_2, S_2) \mid \text{fact}_T v \rangle \]
\[\mapsto \ast \]
\[ \langle (H_2 \uplus H, R_2, S_2) \mid \lambda (x : \text{int})^{\text{int}F_T} (\text{protect} \cdot, \xi; \ v) \]
\[\text{import } r_1, \xi^{\text{int}F_T} x; \]
\[\text{salloc } l; \text{sst } 0, r_1; \]
\[\text{mv } r_1, \xi_{\text{end}}[\xi]; \text{jmp } \xi_{\text{fact}}[\xi][\text{end} \{ \text{int}^T; \xi \}]; \cdot \]
\[\mapsto \ast \]
\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{jmp } \xi_{\text{fact}}[\xi][\text{end} \{ \text{int}^T; \xi \}]; \cdot) \rangle \]

Once we jump, we load the argument off of the stack and then branch on it being zero.

\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{jmp } \xi_{\text{fact}}[\xi][\text{end} \{ \text{int}^T; \xi \}]; \cdot) \rangle \]
\[\mapsto \ast \]
\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{sld } r_1; 0; \text{mr } r_1, 1; \}
\[\text{bnz } r_1, \xi_{\text{aux}}[\xi][\cdot]; \text{sfree } 1; \text{ret } r_1 \{ \cdot \}) \]
\[\mapsto \ast \]
\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{bnz } r_1, \xi_{\text{aux}}[\xi][\cdot]; \text{sfree } 1; \text{ret } r_1 \{ \cdot \}) \rangle \]

As in the high-level version, if \( v = 0 \) we can see that this reduces, via the same jump through the return continuation shown in the previous example, to 1.

Otherwise, we jump to the \( \xi_{\text{aux}} \) codeblock:

\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{bnz } r_1, \xi_{\text{aux}}[\xi][\cdot]; \text{sfree } 1; \text{ret } r_1 \{ \cdot \}) \rangle \]
\[\mapsto \ast \]
\[ \langle (H_2 \uplus H, R_2[r_1 \mapsto v, r_1 \mapsto \xi_{\text{end}}[\xi]], v :: S_2) \mid \text{int}^{\text{int}F_T} (\text{bnz } r_1, \xi_{\text{aux}}[\xi][\cdot]; \text{sfree } 1; \text{ret } r_1 \{ \cdot \}) \rangle \]

This then steps by updating the return value register \( (\cdot r) \) by multiplying it by \( v \) and then decreasing \( v \) by 1, before branching again on the updated value being 0.
3.9.3 Implementing a Mutable Reference

We can use inline \( T \) code to implement a basic mutable reference for use it \( F \) code. While the basic multi-language allows global mutable tuples, leading to trivial mutation of a statically defined value, in this case we show how we can dynamically allocate a new reference and access it from the functional language. While
multiple can be created, the stack-based nature means that only the most recently created can be accessed until the continuation for it returns.

\[
\text{withref} = \lambda (\text{init} : \text{int}, \quad \quad)
\]

\[
k = ((\langle \text{int} \rangle, \langle \text{int} \rangle) \rightarrow \langle \text{int} \rangle) \rightarrow \langle \text{int} \rangle
\]

\[
\text{FT}(\text{protect} \cdot, \zeta; \text{salloc} \ 1; \quad \quad)
\]

\[
\text{import} \ r1, \zeta \text{FT}\langle \text{int}\rangle \quad \quad
\]

\[
\text{sst} \ 0, \ r1; \text{ralloc} \ rc, \ 1;
\]

\[
\text{salloc} \ 1; \text{sst} \ 0, rc; \text{mv} \ r1, ();
\]

\[
\text{ret} \ \end{unit}; \langle \text{int} \rangle :: \zeta \ \{r1\}, \cdot
\]

\[
\text{let} \ r = k (\lambda (\langle \text{int} \rangle \langle \text{int} \rangle) (x : \text{int}). \quad \quad)
\]

\[
\text{FT}(\text{protect} \langle \text{int} \rangle, \zeta; \quad \quad)
\]

\[
\text{sld} \ r1, \ 0;
\]

\[
\text{import} \ r2, \langle \text{int} \rangle :: \zeta \text{FT}\langle \text{int}\rangle \quad \quad
\]

\[
\text{str} \ r1[0], r2; \text{mv} \ r1, ();
\]

\[
\text{ret} \ \end{unit}; \langle \text{int} \rangle :: \zeta \ \{r1\}, \cdot
\]

\[
\lambda (\langle \text{int} \rangle) ().
\]

\[
\text{in}
\]

\[
\text{FT}(\text{protect} \langle \text{int} \rangle, \zeta; \text{sfree} \ 1; \quad \quad)
\]

\[
\text{mv} \ r1, (); \text{ret} \ \end{unit}; \zeta \ \{r1\}, \cdot
\]

\[
3.9.4 \text{ Higher Order}
\]

In this example we demonstrate how the multi-language supports higher order calls between languages.
\[ \tau = ((\text{int}) \rightarrow \text{int}) \rightarrow \text{int} \]
\[ g = \lambda (h : (\text{int}) \rightarrow \text{int}).h \]
\[ e = \left( \text{int} \rightarrow (\text{int} \rightarrow \text{int}) \right) g \]
\[ \text{ret} \end{\left( \tau \rightarrow \text{int} ; \bullet \right) \{ r1 \}, \cdot} \]

\[ H(\ell) = \]
\[ \text{code[}\zeta,\epsilon\{ra:}\forall[]],\{r1:}\text{int}^\tau;\zeta^e;\tau^T :: \zeta^r]a. \]
\[ \text{sld} r1, 0; \text{salloc} 1; \text{sst} 0, \ell_h; \text{sst} 1, \text{ra}; \]
\[ \text{mv} \text{ra}, \ell_{\text{gret}}[\zeta]; \text{call} r1 \{ \forall[]],\{r1:}\text{int}^T;\zeta^e :: \zeta, 0\} \]

3.9.5 Calls

This example shown the call/return structure of the call instruction.

\[ f = (\text{mv} \text{ra}, \ell_{1\text{ret}}; \text{call} \ell_1 \{ \bullet, \text{end}\{\text{int};\bullet\}\}, \text{H}) \]
\[ H(\ell_1) = \]
\[ \text{code[}\zeta,\epsilon\{ra:}\forall[]],\{r1:}\text{int}^T;\zeta^e;\zeta^r]a. \]
\[ \text{salloc} 1; \text{sst} 0, \text{ra}; \text{mv} \text{ra}, \ell_{2\text{ret}}; \]
\[ \text{call} \ell_2 \{ \forall[]],\{r1:}\text{int};\zeta^e :: \zeta, 0\} \]
\[ H(\ell_{1\text{ret}}) = \text{code[}\{r1:}\text{int};\bullet\} \text{end}\{\text{int};\bullet\}. \]
\[ \text{ret} \end{\left( \text{int};\bullet \right) \{ r1 \} \]

\[ H(\ell_2) = \text{code[}\zeta,\epsilon\{ra:}\forall[]],\{r1:}\text{int}^T;\zeta^e;\zeta^r]a. \]
\[ \text{mv} r1, 1; \text{jmp} \ell_{2\text{aux}} \]
\[ H(\ell_{2\text{aux}}) = \text{code[}\zeta,\epsilon\{ra:}\forall[]],\{r1:}\text{int};\zeta^e;\zeta^r]a. \]
\[ \text{mul} r1, r1, 2; \text{ret} \text{ra} \{ r1 \} \]
\[ H(\ell_{2\text{ret}}) = \text{code[}\{r1:}\text{int};\forall[]],\{r1:}\text{int};\bullet\} \text{end}\{\text{int};\bullet \} :: \bullet^0. \]
\[ \text{sld} \text{ra}, 0; \text{sfree} 1; \text{ret} \text{ra} \{ r1 \} \]
References
