

# Signature Restriction for Polymorphic Algebraic Effects (Supplementary Material)

Taro Sekiyama<sup>1</sup>, Takeshi Tsukada<sup>2</sup>, and Atsushi Igarashi<sup>3</sup>

<sup>1</sup>National Institute of Informatics & SOKENDAI

tsekiyama@acm.org

<sup>2</sup>Chiba University

tsukada@math.s.chiba-u.ac.jp

<sup>3</sup>Kyoto University

igarashi@kuis.kyoto-u.ac.jp

This is the supplementary material for “Signature Restriction for Polymorphic Algebraic Effects” submitted to Journal of Functional Programming, providing the full definitions of the language, the polymorphic type system, and the type-and-effect system and the full proofs of the properties presented in the article.

## 1 Definition

### 1.1 Syntax

<b>Variables</b> $x, y, z, f, k$	<b>Type variables</b> $\alpha, \beta, \gamma$	<b>Effect operations</b> $\text{op}$
<b>Base types</b> $\iota$	$::= \text{bool} \mid \text{int} \mid \dots$	
<b>Types</b> $A, B, C, D$	$::= \alpha \mid \iota \mid A \rightarrow B \mid \forall \alpha. A \mid A \times B \mid A + B \mid A \text{ list}$	
<b>Constants</b> $c$	$::= \text{true} \mid \text{false} \mid 0 \mid + \mid \dots$	
<b>Terms</b> $M$	$::= x \mid c \mid \lambda x. M \mid M_1 M_2 \mid \# \text{op}(M) \mid \text{handle } M \text{ with } H \mid$ $(M_1, M_2) \mid \pi_1 M \mid \pi_2 M \mid$ $\text{inl } M \mid \text{inr } M \mid \text{case } M \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2 \mid$ $\text{nil} \mid \text{cons } M \mid \text{case } M \text{ of } \text{nil} \rightarrow M_1; \text{cons } x \rightarrow M_2 \mid \text{fix } f. \lambda x. M$	
<b>Handlers</b> $H$	$::= \text{return } x \rightarrow M \mid H; \text{op}(x, k) \rightarrow M$	
<b>Values</b> $v$	$::= c \mid \lambda x. M \mid (v_1, v_2) \mid \text{inl } v \mid \text{inr } v \mid \text{nil} \mid \text{cons } v$	
<b>Typing contexts</b> $\Gamma$	$::= \emptyset \mid \Gamma, x : A \mid \Gamma, \alpha$	
<b>Evaluation contexts</b> $E$	$::= [] \mid E M_2 \mid v_1 E \mid \# \text{op}(E) \mid \text{handle } E \text{ with } H \mid$ $(E, M_2) \mid (v_1, E) \mid \pi_1 E \mid \pi_2 E \mid$ $\text{inl } E \mid \text{inr } E \mid \text{case } E \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2 \mid$ $\text{cons } E \mid \text{case } E \text{ of } \text{nil} \rightarrow M_1; \text{cons } x \rightarrow M_2$	

**Convention 1.** *This work follows the conventions as below.*

- We write  $\alpha^I$  for  $\alpha = \alpha_1, \dots, \alpha_n$  with  $I = \{1, \dots, n\}$ . We often omit index sets ( $I$  and  $J$ ) if they are not important: for example, we often abbreviate  $\alpha^I$  to  $\alpha$ . We apply this bold-font notation to other syntax categories as well; for example,  $\mathbf{A}^I$  denotes a sequence of types.
- We write  $\{s\}$  to view the sequence  $s$  as a set by ignoring the order.
- We write  $\forall \alpha^I. A$  for  $\forall \alpha_1. \dots \forall \alpha_n. A$  with  $I = \{1, \dots, n\}$ . We may omit index sets ( $\forall \alpha. A$ ). We write  $\forall \alpha^I. \mathbf{A}^J$  for a sequence of types  $\forall \alpha^I. A_1, \dots, \forall \alpha^I. A_n$  with  $J = \{1, \dots, n\}$ .
- We write  $\Gamma_1, \Gamma_2$  for the concatenation of  $\Gamma_1$  and  $\Gamma_2$ , and  $x : A$  and  $\alpha$  for  $(\emptyset, x : A)$ ,  $(\emptyset, \alpha)$ , respectively.

- We write  $H^{\text{return}}$  for the return clause in  $H$  and  $H(\text{op})$  for the operation clause of  $\text{op}$  in  $H$ .

**Definition 1** (Domain of typing contexts). We define  $\text{dom}(\Gamma)$  as follows.

$$\begin{aligned} \text{dom}(\emptyset) &\stackrel{\text{def}}{=} \emptyset \\ \text{dom}(\Gamma, x : A) &\stackrel{\text{def}}{=} \text{dom}(\Gamma) \cup \{x\} \\ \text{dom}(\Gamma, \alpha) &\stackrel{\text{def}}{=} \text{dom}(\Gamma) \cup \{\alpha\} \end{aligned}$$

**Definition 2** (Free type variables and type substitution in types). Free type variables  $\text{ftv}(A)$  in a type  $A$  and type substitution  $B[\mathbf{A}/\boldsymbol{\alpha}]$  of types  $\mathbf{A}$  for type variables  $\boldsymbol{\alpha}$  in  $B$  are defined as usual. Type  $A$  is closed if and only if  $\text{ftv}(A)$  is empty.

**Assumption 1.** We suppose that the metafunction  $\text{ty}$  assigns to each constant  $c$  a first-order closed type  $\text{ty}(c)$  of the form  $\iota_1 \rightarrow \dots \rightarrow \iota_n$ . We also suppose that, for any  $\iota$ , there exists the set  $\mathbb{K}_\iota$  of constants of  $\iota$ . For any constant  $c$ ,  $\text{ty}(c) = \iota$  if and only if  $c \in \mathbb{K}_\iota$ . The partial function  $\zeta$  gives a denotation to pairs of constants. In particular, for any constants  $c_1$  and  $c_2$ : (1)  $\zeta(c_1, c_2)$  is defined if and only if  $\text{ty}(c_1) = \iota_0 \rightarrow A$  and  $\text{ty}(c_2) = \iota_0$  for some  $\iota_0$  and  $A$ ; and (2) if  $\zeta(c_1, c_2)$  is defined,  $\zeta(c_1, c_2)$  is a constant and  $\text{ty}(\zeta(c_1, c_2)) = A$  where  $\text{ty}(c_1) = \iota_0 \rightarrow A$  for some  $\iota_0$  and  $A$ .

**Definition 3** (Polarity of type variable occurrence). The sets  $\text{ftv}(A)^+$  and  $\text{ftv}(A)^-$  of type variables that occur positively and negatively, respectively, in type  $A$  are defined by induction on  $A$ , as follows. We write  $\text{ftv}(A)^\pm$  for either  $\text{ftv}(A)^+$  or  $\text{ftv}(A)^-$  and  $\text{ftv}(A)^\mp$  for the other.

$$\begin{aligned} \text{ftv}(\alpha)^+ &\stackrel{\text{def}}{=} \{\alpha\} \\ \text{ftv}(\alpha)^- &\stackrel{\text{def}}{=} \emptyset \\ \text{ftv}(A \rightarrow B)^\pm &\stackrel{\text{def}}{=} \text{ftv}(A)^\mp \cup \text{ftv}(B)^\pm \\ \text{ftv}(\forall \alpha. A)^\pm &\stackrel{\text{def}}{=} \text{ftv}(A)^\pm \setminus \{\alpha\} \\ \text{ftv}(A \times B)^\pm &\stackrel{\text{def}}{=} \text{ftv}(A)^\pm \cup \text{ftv}(B)^\pm \\ \text{ftv}(A + B)^\pm &\stackrel{\text{def}}{=} \text{ftv}(A)^\pm \cup \text{ftv}(B)^\pm \\ \text{ftv}(A \text{ list})^\pm &\stackrel{\text{def}}{=} \text{ftv}(A)^\pm \end{aligned}$$

The set  $\text{ftv}(A)_{\text{ns}}^+$  of type variables that occur non-strictly positively in type  $A$  is defined as follows.

$$\begin{aligned} \text{ftv}(\alpha)_{\text{ns}}^+ &\stackrel{\text{def}}{=} \emptyset \\ \text{ftv}(A \rightarrow B)_{\text{ns}}^+ &\stackrel{\text{def}}{=} \text{ftv}(A)^- \cup \text{ftv}(B)_{\text{ns}}^+ \\ \text{ftv}(\forall \alpha. A)_{\text{ns}}^+ &\stackrel{\text{def}}{=} \text{ftv}(A)_{\text{ns}}^+ \setminus \{\alpha\} \\ \text{ftv}(A \times B)_{\text{ns}}^+ &\stackrel{\text{def}}{=} \text{ftv}(A)_{\text{ns}}^+ \cup \text{ftv}(B)_{\text{ns}}^+ \\ \text{ftv}(A + B)_{\text{ns}}^+ &\stackrel{\text{def}}{=} \text{ftv}(A)_{\text{ns}}^+ \cup \text{ftv}(B)_{\text{ns}}^+ \\ \text{ftv}(A \text{ list})_{\text{ns}}^+ &\stackrel{\text{def}}{=} \text{ftv}(A)_{\text{ns}}^+ \end{aligned}$$

**Definition 4** (Type signature). The metafunction  $\text{ty}$  assigns to each effect operation  $\text{op}$  a type signature  $\text{ty}(\text{op})$  of the form  $\forall \alpha_1. \dots \forall \alpha_n. A \hookrightarrow B$  for some  $n$ , where  $\alpha_1, \dots, \alpha_n$  are bound in the parameter type  $A$  and arity type  $B$ . It may be abbreviated to  $\forall \boldsymbol{\alpha}^I. A \hookrightarrow B$  or, more simply, to  $\forall \boldsymbol{\alpha}. A \hookrightarrow B$ . We suppose that  $\forall \alpha_1. \dots \forall \alpha_n. A \hookrightarrow B$  is closed, i.e.,  $\text{ftv}(A), \text{ftv}(B) \subseteq \{\alpha_1, \dots, \alpha_n\}$ .

**Definition 5** (Signature restriction). An operation  $\text{op}$  with type signature  $\text{ty}(\text{op}) = \forall \boldsymbol{\alpha}. A \hookrightarrow B$  satisfies the signature restriction if and only if: (1) the occurrences of each type variable of  $\boldsymbol{\alpha}$  in the parameter type  $A$  are only negative or strictly positive (i.e.,  $\{\boldsymbol{\alpha}\} \cap \text{ftv}(A)_{\text{ns}}^+ = \emptyset$ ); and (2) the occurrences of each type variable of  $\boldsymbol{\alpha}$  in the arity type  $B$  are only positive (i.e.,  $\{\boldsymbol{\alpha}\} \cap \text{ftv}(B)^- = \emptyset$ ).

## 1.2 Semantics

**Definition 6** ( $\text{op}$ -transparent evaluation contexts). Evaluation context  $E$  is  $\text{op}$ -transparent, written  $\text{op} \notin E$ , if and only if, there exist no  $E_1$ ,  $E_2$ , and  $H$  such that  $E = E_1[\text{handle } E_2 \text{ with } H]$  and  $H$  has an operation clause for  $\text{op}$ .

Reduction rules

$$\boxed{M_1 \rightsquigarrow M_2}$$

$c\ v$	$\rightsquigarrow$	$\zeta(c, v)$	(R_CONST)
$(\lambda x.M)\ v$	$\rightsquigarrow$	$M[v/x]$	(R_BETA)
handle $v$ with $H$	$\rightsquigarrow$	$M[v/x]$	(R_RETURN)
handle $E[\#op(v)]$ with $H$	$\rightsquigarrow$	$M[v/x][\lambda y.\text{handle } E[y] \text{ with } H/k]$	(R_HANDLE)
		(where $H^{\text{return}} = \text{return } x \rightarrow M$ )	
		(where $op \notin E$ and $H(op) = op(x, k) \rightarrow M$ )	
$\pi_1(v_1, v_2)$	$\rightsquigarrow$	$v_1$	(R_PROJ1)
$\pi_2(v_1, v_2)$	$\rightsquigarrow$	$v_2$	(R_PROJ2)
case inl $v$ of inl $x \rightarrow M_1$ ; inr $y \rightarrow M_2$	$\rightsquigarrow$	$M_1[v/x]$	(R_CASEL)
case inr $v$ of inl $x \rightarrow M_1$ ; inr $y \rightarrow M_2$	$\rightsquigarrow$	$M_2[v/y]$	(R_CASER)
case nil of nil $\rightarrow M_1$ ; cons $x \rightarrow M_2$	$\rightsquigarrow$	$M_1$	(R_NIL)
case cons $v$ of nil $\rightarrow M_1$ ; cons $x \rightarrow M_2$	$\rightsquigarrow$	$M_2[v/x]$	(R_CONS)
fix $f.\lambda x.M$	$\rightsquigarrow$	$(\lambda x.M)[\text{fix } f.\lambda x.M / f]$	(R_FIX)

Evaluation rules

$$\boxed{M_1 \longrightarrow M_2}$$

$$\frac{M_1 \rightsquigarrow M_2}{E[M_1] \longrightarrow E[M_2]} \quad \text{E\_EVAL}$$

Figure 1: Semantics.

**Definition 7.** Relations  $\longrightarrow$  and  $\rightsquigarrow$  are the smallest relations satisfying the rules in Figure 1.

**Definition 8** (Multi-step evaluation). Binary relation  $\longrightarrow^*$  over terms is the reflexive and transitive closure of  $\longrightarrow$ .

**Definition 9** (Nonreducible terms). We write  $M \not\rightarrow$  if there exists no term  $M'$  such that  $M \longrightarrow M'$ .

### 1.3 Typing

**Definition 10.** Well-formedness judgment  $\vdash \Gamma$  is the smallest relations satisfying the rules in Figure 3. We write  $\Gamma \vdash A$  if and only if  $\text{ftv}(A) \subseteq \text{dom}(\Gamma)$  and  $\vdash \Gamma$  is derived. Type containment judgment  $\Gamma \vdash A \sqsubseteq B$  is the least relation satisfying the rules in Figure 2. Typing judgments  $\Gamma \vdash M : A$  and  $\Gamma \vdash H : A \Rightarrow B$  are the smallest relations satisfying the rules in Figure 4.

**Type containment**

$$\boxed{\Gamma \vdash A \sqsubseteq B}$$

$$\begin{array}{c}
\frac{\vdash \Gamma}{\Gamma \vdash A \sqsubseteq A} \quad \text{C\_REFL} \quad \frac{\Gamma \vdash A \sqsubseteq C \quad \Gamma \vdash C \sqsubseteq B}{\Gamma \vdash A \sqsubseteq B} \quad \text{C\_TRANS} \quad \frac{\Gamma \vdash B_1 \sqsubseteq A_1 \quad \Gamma \vdash A_2 \sqsubseteq B_2}{\Gamma \vdash A_1 \rightarrow A_2 \sqsubseteq B_1 \rightarrow B_2} \quad \text{C\_FUN} \\
\\
\frac{\Gamma \vdash B}{\Gamma \vdash \forall \alpha. A \sqsubseteq A[B/\alpha]} \quad \text{C\_INST} \quad \frac{\vdash \Gamma \quad \alpha \notin \text{ftv}(A)}{\Gamma \vdash A \sqsubseteq \forall \alpha. A} \quad \text{C\_GEN} \quad \frac{\Gamma, \alpha \vdash A \sqsubseteq B}{\Gamma \vdash \forall \alpha. A \sqsubseteq \forall \alpha. B} \quad \text{C\_POLY} \\
\\
\frac{\Gamma \vdash A_1 \sqsubseteq B_1 \quad \Gamma \vdash A_2 \sqsubseteq B_2}{\Gamma \vdash A_1 \times A_2 \sqsubseteq B_1 \times B_2} \quad \text{C\_PROD} \quad \frac{\Gamma \vdash A_1 \sqsubseteq B_1 \quad \Gamma \vdash A_2 \sqsubseteq B_2}{\Gamma \vdash A_1 + A_2 \sqsubseteq B_1 + B_2} \quad \text{C\_SUM} \quad \frac{\Gamma \vdash A \sqsubseteq B}{\Gamma \vdash A \text{ list} \sqsubseteq B \text{ list}} \quad \text{C\_LIST} \\
\\
\frac{\vdash \Gamma \quad \alpha \notin \text{ftv}(A)}{\Gamma \vdash \forall \alpha. A \rightarrow B \sqsubseteq A \rightarrow \forall \alpha. B} \quad \text{C\_DFUN} \quad \frac{\vdash \Gamma}{\Gamma \vdash \forall \alpha. A \times B \sqsubseteq (\forall \alpha. A) \times (\forall \alpha. B)} \quad \text{C\_DPROD} \\
\\
\frac{\vdash \Gamma}{\Gamma \vdash \forall \alpha. A + B \sqsubseteq (\forall \alpha. A) + (\forall \alpha. B)} \quad \text{C\_DSUM} \quad \frac{\vdash \Gamma}{\Gamma \vdash \forall \alpha. A \text{ list} \sqsubseteq (\forall \alpha. A) \text{ list}} \quad \text{C\_DLIST}
\end{array}$$

Figure 2: Type containment.

**Well-formedness**

$$\boxed{\vdash \Gamma}$$

$$\begin{array}{c}
\frac{}{\vdash \emptyset} \quad \text{WF\_EMPTY} \quad \frac{x \notin \text{dom}(\Gamma) \quad \Gamma \vdash A}{\vdash \Gamma, x : A} \quad \text{WF\_EXTVAR} \quad \frac{\alpha \notin \text{dom}(\Gamma) \quad \vdash \Gamma}{\vdash \Gamma, \alpha} \quad \text{WF\_EXTTYVAR}
\end{array}$$

Figure 3: Well-formedness.

**Term typing**  $\boxed{\Gamma \vdash M : A}$

$$\begin{array}{c}
\frac{\vdash \Gamma \quad x : A \in \Gamma}{\Gamma \vdash x : A} \quad \text{T\_VAR} \qquad \frac{\vdash \Gamma}{\Gamma \vdash c : ty(c)} \quad \text{T\_CONST} \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x. M : A \rightarrow B} \quad \text{T\_ABS} \\
\\
\frac{\Gamma \vdash M_1 : A \rightarrow B \quad \Gamma \vdash M_2 : A}{\Gamma \vdash M_1 M_2 : B} \quad \text{T\_APP} \quad \frac{\Gamma, \alpha \vdash M : A}{\Gamma \vdash M : \forall \alpha. A} \quad \text{T\_GEN} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash A \sqsubseteq B \quad \Gamma \vdash B}{\Gamma \vdash M : B} \quad \text{T\_INST} \\
\\
\frac{ty(\text{op}) = \forall \alpha. A \hookrightarrow B \quad \Gamma \vdash M : A[C/\alpha] \quad \Gamma \vdash C}{\Gamma \vdash \#op(M) : B[C/\alpha]} \quad \text{T\_OP} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash H : A \Rightarrow B}{\Gamma \vdash \text{handle } M \text{ with } H : B} \quad \text{T\_HANDLE} \\
\\
\frac{\Gamma \vdash M_1 : A \quad \Gamma \vdash M_2 : B}{\Gamma \vdash (M_1, M_2) : A \times B} \quad \text{T\_PAIR} \quad \frac{\Gamma \vdash M : A \times B}{\Gamma \vdash \pi_1 M : A} \quad \text{T\_PROJ1} \quad \frac{\Gamma \vdash M : A \times B}{\Gamma \vdash \pi_2 M : B} \quad \text{T\_PROJ2} \\
\\
\frac{\Gamma \vdash M : A \quad \Gamma \vdash B}{\Gamma \vdash \text{inl } M : A + B} \quad \text{T\_INL} \qquad \frac{\Gamma \vdash M : B \quad \Gamma \vdash A}{\Gamma \vdash \text{inr } M : A + B} \quad \text{T\_INR} \\
\\
\frac{\Gamma \vdash M : A + B \quad \Gamma, x : A \vdash M_1 : C \quad \Gamma, y : B \vdash M_2 : C}{\Gamma \vdash \text{case } M \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2 : C} \quad \text{T\_CASE} \\
\\
\frac{\Gamma \vdash A}{\Gamma \vdash \text{nil} : A \text{ list}} \quad \text{T\_NIL} \qquad \frac{\Gamma \vdash M : A \times A \text{ list}}{\Gamma \vdash \text{cons } M : A \text{ list}} \quad \text{T\_CONS} \\
\\
\frac{\Gamma \vdash M : A \text{ list} \quad \Gamma \vdash M_1 : B \quad \Gamma, x : A \times A \text{ list} \vdash M_2 : B}{\Gamma \vdash \text{case } M \text{ of } \text{nil} \rightarrow M_1; \text{cons } x \rightarrow M_2 : B} \quad \text{T\_CASELIST} \quad \frac{\Gamma, f : A \rightarrow B, x : A \vdash M : B}{\Gamma \vdash \text{fix } f. \lambda x. M : A \rightarrow B} \quad \text{T\_FIX}
\end{array}$$

**Handler typing**  $\boxed{\Gamma \vdash H : A \Rightarrow B}$

$$\begin{array}{c}
\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \text{return } x \rightarrow M : A \Rightarrow B} \quad \text{TH\_RETURN} \\
\\
\frac{\Gamma \vdash H : A \Rightarrow B \quad ty(\text{op}) = \forall \alpha. C \hookrightarrow D \quad \Gamma, \alpha, x : C, k : D \rightarrow B \vdash M : B}{\Gamma \vdash H; \text{op}(x, k) \rightarrow M : A \Rightarrow B} \quad \text{TH\_OP}
\end{array}$$

Figure 4: Typing.

**Effects**  $\epsilon ::= \{\text{op}_1, \dots, \text{op}_n\}$   
**Types**  $A, B, C, D ::= \alpha \mid \iota \mid A \rightarrow^\epsilon B \mid \forall \alpha. A \mid A \times B \mid A + B \mid A \text{ list}$

Figure 5: Type language for the effect-and-type system.

**Type containment**  $\boxed{\Gamma \vdash A \sqsubseteq B}$

$$\frac{\Gamma \vdash B_1 \sqsubseteq A_1 \quad \Gamma \vdash A_2 \sqsubseteq B_2}{\Gamma \vdash A_1 \rightarrow^\epsilon A_2 \sqsubseteq B_1 \rightarrow^\epsilon B_2} \text{C\_FUNEFF} \qquad \frac{\vdash \Gamma \quad \alpha \notin \text{ftv}(A) \quad \text{SR}(\epsilon)}{\Gamma \vdash \forall \alpha. A \rightarrow^\epsilon B \sqsubseteq A \rightarrow^\epsilon \forall \alpha. B} \text{C\_DFUNEFF}$$

Figure 6: Change from Figure 2 for type containment of the effect-and-type system. It gets rid of (C\_FUN) and (C\_DFUN) instead of adding (C\_FUNEFF) and (C\_DFUNEFF).

## 1.4 Type-and-effect system

The type language for the type-and-effect system is shown Figure 5. Figure 6 describes only the change of the type containment rules from those of the polymorphic type system.

**Definition 11** (Signature restriction on effects). *The predicate  $\text{SR}(\epsilon)$  holds if and only if, for any  $\text{op} \in \epsilon$  such that  $\text{ty}(\text{op}) = \forall \alpha. A \hookrightarrow B$ :*

- $\{\alpha\} \cap \text{ftv}(A)_{\text{ns}}^+ = \emptyset$ ;
- $\{\alpha\} \cap \text{ftv}(B)^- = \emptyset$ ; and
- for any function type  $C \rightarrow^{\epsilon'} D$  occurring at a strictly positive position in the type  $A$ , if  $\{\alpha\} \cap \text{ftv}(D) \neq \emptyset$ , then  $\text{SR}(\epsilon')$ .

**Definition 12.** *Typing judgments  $\Gamma \vdash M : A \mid \epsilon$  and  $\Gamma \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon'$  are the smallest relations satisfying the rules in Figure 7.*

**Term typing**

$$\boxed{\Gamma \vdash M : A \mid \epsilon}$$

$$\begin{array}{c}
\frac{\vdash \Gamma \quad x : A \in \Gamma}{\Gamma \vdash x : A \mid \epsilon} \quad \text{TE\_VAR} \qquad \frac{\vdash \Gamma}{\Gamma \vdash c : ty(c) \mid \epsilon} \quad \text{TE\_CONST} \\
\\
\frac{\Gamma, x : A \vdash M : B \mid \epsilon'}{\Gamma \vdash \lambda x. M : A \rightarrow^{\epsilon'} B \mid \epsilon} \quad \text{TE\_ABS} \qquad \frac{\Gamma \vdash M_1 : A \rightarrow^{\epsilon'} B \mid \epsilon \quad \Gamma \vdash M_2 : A \mid \epsilon \quad \epsilon' \subseteq \epsilon}{\Gamma \vdash M_1 M_2 : B \mid \epsilon} \quad \text{TE\_APP} \\
\\
\frac{\Gamma, \alpha \vdash M : A \mid \epsilon \quad SR(\epsilon)}{\Gamma \vdash M : \forall \alpha. A \mid \epsilon} \quad \text{TE\_GEN} \qquad \frac{\Gamma \vdash M : A \mid \epsilon \quad \Gamma \vdash A \subseteq B \quad \Gamma \vdash B}{\Gamma \vdash M : B \mid \epsilon} \quad \text{TE\_INST} \\
\\
\frac{ty(\text{op}) = \forall \alpha. A \hookrightarrow B \quad \text{op} \in \epsilon \quad \Gamma \vdash M : A[C/\alpha] \mid \epsilon \quad \Gamma \vdash C}{\Gamma \vdash \# \text{op}(M) : B[C/\alpha] \mid \epsilon} \quad \text{TE\_OP} \\
\\
\frac{\Gamma \vdash M : A \mid \epsilon \quad \Gamma \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon'}{\Gamma \vdash \text{handle } M \text{ with } H : B \mid \epsilon'} \quad \text{TE\_HANDLE} \\
\\
\frac{\Gamma \vdash M_1 : A \mid \epsilon \quad \Gamma \vdash M_2 : B \mid \epsilon}{\Gamma \vdash (M_1, M_2) : A \times B \mid \epsilon} \quad \text{TE\_PAIR} \quad \frac{\Gamma \vdash M : A \times B \mid \epsilon}{\Gamma \vdash \pi_1 M : A \mid \epsilon} \quad \text{TE\_PROJ1} \quad \frac{\Gamma \vdash M : A \times B \mid \epsilon}{\Gamma \vdash \pi_2 M : B \mid \epsilon} \quad \text{TE\_PROJ2} \\
\\
\frac{\Gamma \vdash M : A \mid \epsilon \quad \Gamma \vdash B}{\Gamma \vdash \text{inl } M : A + B \mid \epsilon} \quad \text{TE\_INL} \qquad \frac{\Gamma \vdash M : B \mid \epsilon \quad \Gamma \vdash A}{\Gamma \vdash \text{inr } M : A + B \mid \epsilon} \quad \text{TE\_INR} \\
\\
\frac{\Gamma \vdash M : A + B \mid \epsilon \quad \Gamma, x : A \vdash M_1 : C \mid \epsilon \quad \Gamma, y : B \vdash M_2 : C \mid \epsilon}{\Gamma \vdash \text{case } M \text{ of inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2 : C \mid \epsilon} \quad \text{TE\_CASE} \\
\\
\frac{\Gamma \vdash A}{\Gamma \vdash \text{nil} : A \text{ list} \mid \epsilon} \quad \text{TE\_NIL} \qquad \frac{\Gamma \vdash M : A \times A \text{ list} \mid \epsilon}{\Gamma \vdash \text{cons } M : A \text{ list} \mid \epsilon} \quad \text{TE\_CONS} \\
\\
\frac{\Gamma \vdash M : A \text{ list} \mid \epsilon \quad \Gamma \vdash M_1 : B \mid \epsilon \quad \Gamma, x : A \times A \text{ list} \vdash M_2 : B \mid \epsilon}{\Gamma \vdash \text{case } M \text{ of nil} \rightarrow M_1; \text{cons } x \rightarrow M_2 : B \mid \epsilon} \quad \text{TE\_CASELIST} \\
\\
\frac{\Gamma, f : A \rightarrow^{\epsilon} B, x : A \vdash M : B \mid \epsilon}{\Gamma \vdash \text{fix } f. \lambda x. M : A \rightarrow^{\epsilon} B \mid \epsilon'} \quad \text{TE\_FIX} \qquad \frac{\Gamma \vdash M : A \mid \epsilon' \quad \epsilon' \subseteq \epsilon}{\Gamma \vdash M : A \mid \epsilon} \quad \text{TE\_WEAK}
\end{array}$$

**Handler typing**

$$\boxed{\Gamma \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon'}$$

$$\begin{array}{c}
\frac{\Gamma, x : A \vdash M : B \mid \epsilon' \quad \epsilon \subseteq \epsilon'}{\Gamma \vdash \text{return } x \rightarrow M : A \mid \epsilon \Rightarrow B \mid \epsilon'} \quad \text{THE\_RETURN} \\
\\
\frac{\Gamma \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon' \quad ty(\text{op}) = \forall \alpha. C \hookrightarrow D \quad \Gamma, \alpha, x : C, k : D \rightarrow^{\epsilon'} B \vdash M : B \mid \epsilon'}{\Gamma \vdash H; \text{op}(x, k) \rightarrow M : A \mid \epsilon \uplus \{\text{op}\} \Rightarrow B \mid \epsilon'} \quad \text{THE\_OP}
\end{array}$$

Figure 7: Typing of the effect-and-type system.

## 2 Proofs

### 2.1 Soundness of the Type System

**Lemma 1** (Weakening). *Suppose that  $\vdash \Gamma_1, \Gamma_2$ . Let  $\Gamma_3$  be a typing context such that  $\text{dom}(\Gamma_2) \cap \text{dom}(\Gamma_3) = \emptyset$ .*

1. *If  $\vdash \Gamma_1, \Gamma_3$ , then  $\vdash \Gamma_1, \Gamma_2, \Gamma_3$ .*
2. *If  $\Gamma_1, \Gamma_3 \vdash A$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash A$ .*
3. *If  $\Gamma_1, \Gamma_3 \vdash A \sqsubseteq B$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash A \sqsubseteq B$ .*
4. *If  $\Gamma_1, \Gamma_3 \vdash M : A$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash M : A$ .*
5. *If  $\Gamma_1, \Gamma_3 \vdash H : A \Rightarrow B$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash H : A \Rightarrow B$ .*

*Proof.* By mutual induction on the derivations of the judgments. □

**Lemma 2** (Type substitution). *Suppose that  $\Gamma_1 \vdash A$ .*

1. *If  $\vdash \Gamma_1, \alpha, \Gamma_2$ , then  $\vdash \Gamma_1, \Gamma_2 [A/\alpha]$ .*
2. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash B$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash B[A/\alpha]$ .*
3. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash B \sqsubseteq C$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash B[A/\alpha] \sqsubseteq C[A/\alpha]$ .*
4. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash M : B$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash M : B[A/\alpha]$ .*
5. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash H : B \Rightarrow C$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash H : B[A/\alpha] \Rightarrow C[A/\alpha]$ .*

*Proof.* Straightforward by mutual induction on the derivations of the judgments. Note that the cases for (T\_OP) and (TH\_OP) depend on Definition 4, which states that, for any  $\text{op}$ , if  $\text{ty}(\text{op}) = \forall \beta. C \hookrightarrow D$ ,  $\text{ftv}(C) \cup \text{ftv}(D) \subseteq \{\beta\}$ . □

**Lemma 3.**

1. *If  $\vdash \Gamma_1, x : A, \Gamma_2$ , then  $\vdash \Gamma_1, \Gamma_2$ .*
2. *If  $\Gamma_1, x : A, \Gamma_2 \vdash B$ , then  $\Gamma_1, \Gamma_2 \vdash B$ .*
3. *If  $\Gamma_1, x : A, \Gamma_2 \vdash B \sqsubseteq C$ , then  $\Gamma_1, \Gamma_2 \vdash B \sqsubseteq C$ .*

*Proof.* By induction on the derivations of the judgments. □

**Lemma 4** (Term substitution). *Suppose that  $\Gamma_1 \vdash M : A$ .*

1. *If  $\Gamma_1, x : A, \Gamma_2 \vdash M' : B$ , then  $\Gamma_1, \Gamma_2 \vdash M'[M/x] : B$ .*
2. *If  $\Gamma_1, x : A, \Gamma_2 \vdash H : B \Rightarrow C$ , then  $\Gamma_1, \Gamma_2 \vdash H[M/x] : B \Rightarrow C$ .*

*Proof.* By mutual induction on the typing derivations with Lemma 3. The case for (T\_VAR) uses Lemma 1 (4). □

**Definition 13.** *The function  $\text{unqualify}$  returns the type obtained by removing all the  $\forall$ s at the top-level from a given type, defined as follows.*

$$\begin{aligned} \text{unqualify}(\forall \alpha. A) &\stackrel{\text{def}}{=} \text{unqualify}(A) \\ \text{unqualify}(A) &\stackrel{\text{def}}{=} A \quad (\text{if } A \neq \forall \alpha. B \text{ for any } \alpha \text{ and } B) \end{aligned}$$

**Lemma 5.** *Suppose  $\Gamma \vdash A \sqsubseteq B$ . If  $\text{unqualify}(A)$  is not a type variable, then  $\text{unqualify}(B)$  is not either.*

*Proof.* By induction on the type containment derivation. The only interesting case is for (C\_INST). In that case, we are given  $\Gamma \vdash \forall \alpha. C \sqsubseteq C[D/\alpha]$  ( $A = \forall \alpha. C$  and  $B = C[D/\alpha]$ ) for some  $\alpha$ ,  $C$ , and  $D$ , and, by inversion,  $\Gamma \vdash D$ . It is easy to see, if  $\text{unqualify}(\forall \beta. C) = \text{unqualify}(C)$  is not a type variable, then  $\text{unqualify}(C[D/\beta])$  is not either. □



**Lemma 6.** Suppose that  $\Gamma \vdash A \sqsubseteq B$  and  $\text{unqualify}(A)$  is not a type variable.

1. If  $\text{unqualify}(B) = \iota$ , then  $\text{unqualify}(A) = \iota$ .
2. If  $\text{unqualify}(B) = B_1 \rightarrow B_2$ , then  $\text{unqualify}(A) = A_1 \rightarrow A_2$  for some  $A_1$  and  $A_2$ .
3. If  $\text{unqualify}(B) = B_1 \times B_2$ , then  $\text{unqualify}(A) = A_1 \times A_2$  for some  $A_1$  and  $A_2$ .
4. If  $\text{unqualify}(B) = B_1 + B_2$ , then  $\text{unqualify}(A) = A_1 + A_2$  for some  $A_1$  and  $A_2$ .
5. If  $\text{unqualify}(B) = B' \text{ list}$ , then  $\text{unqualify}(A) = A' \text{ list}$  for some  $A'$ .

*Proof.* By induction on the type containment derivation. The case for (C\_TRANS) is shown by the IHs and Lemma 5. In the case for (C\_INST), we are given  $\Gamma \vdash \forall \alpha. C \sqsubseteq C[D/\alpha]$  for some  $\alpha$ ,  $C$ , and  $D$  ( $A = \forall \alpha. C$  and  $B = C[D/\alpha]$ ). Since  $\text{unqualify}(\forall \alpha. C) = \text{unqualify}(C)$  is not a type variable, it is easy to see that the top type constructor of  $\text{unqualify}(C)$  is the same as that of  $\text{unqualify}(C[D/\alpha])$ . Proving the other cases is straightforward.  $\square$

**Lemma 7.** If  $\Gamma \vdash v : A$ , then  $\text{unqualify}(A)$  is not a type variable.

*Proof.* By induction on the typing derivation for  $v$ . We can show the case for (T\_INST) by the IH and Lemma 5.  $\square$

**Lemma 8** (Canonical forms). Suppose that  $\Gamma \vdash v : A$ .

1. If  $\text{unqualify}(A) = \iota$ , then  $v = c$  for some  $c$ .
2. If  $\text{unqualify}(A) = B \rightarrow C$ , then  $v = c$  for some  $c$ , or  $v = \lambda x. M$  for some  $x$  and  $M$ .
3. If  $\text{unqualify}(A) = B \times C$ , then  $v = (v_1, v_2)$  for some  $v_1$  and  $v_2$ .
4. If  $\text{unqualify}(A) = B + C$ , then  $v = \text{inl } v'$  or  $v = \text{inr } v'$  for some  $v'$ .
5. If  $\text{unqualify}(A) = B \text{ list}$ , then  $v = \text{nil}$  or  $v = \text{cons } v'$  for some  $v'$ .

*Proof.* Straightforward by induction on the typing derivation for  $v$ . The only interesting case is for (T\_INST). In the case, we are given, by inversion,  $\Gamma \vdash v : B$  and  $\Gamma \vdash B \sqsubseteq A$  and  $\Gamma \vdash A$  for some  $B$ . By Lemma 7,  $\text{unqualify}(B)$  is not a type variable. Thus, by Lemma 6 and the IH, we finish.  $\square$

**Definition 14.** We use metavariable  $\Delta$  for ranging over typing contexts that consist of only type variables. Formally, they are defined by the following syntax.

$$\Delta ::= \emptyset \mid \Delta, \alpha$$

**Lemma 9** (Commutation of universal quantification in type containment). If  $\vdash \Gamma$ , then  $\Gamma \vdash \forall \alpha. \forall \beta. A \sqsubseteq \forall \beta. \forall \alpha. A$ .

*Proof.* Let  $\alpha'$  and  $\beta'$  be fresh, distinct type variables. Because  $\forall \beta. \forall \alpha. A$  is alpha-equivalent to  $\forall \beta'. \forall \alpha'. A[\alpha'/\alpha][\beta'/\beta]$ , it suffices to show that

$$\Gamma \vdash \forall \alpha. \forall \beta. A \sqsubseteq \forall \beta'. \forall \alpha'. A[\alpha'/\alpha][\beta'/\beta],$$

which is derived by (C\_TRANS) with the following type containment derivations:

$$\begin{aligned} & \frac{\vdash \Gamma \quad \alpha' \notin \text{ftv}(\forall \alpha. \forall \beta. A)}{\Gamma \vdash \forall \alpha. \forall \beta. A \sqsubseteq \forall \alpha'. \forall \alpha. \forall \beta. A} \text{ (C\_GEN)} \\ & \frac{\vdash \Gamma \quad \beta' \notin \text{ftv}(\forall \alpha'. \forall \alpha. \forall \beta. A)}{\Gamma \vdash \forall \alpha'. \forall \alpha. \forall \beta. A \sqsubseteq \forall \beta'. \forall \alpha'. \forall \alpha. \forall \beta. A} \text{ (C\_GEN)} \\ & \frac{\Gamma, \beta', \alpha' \vdash \alpha'}{\Gamma, \beta', \alpha' \vdash \forall \alpha. \forall \beta. A \sqsubseteq \forall \beta. A[\alpha'/\alpha]} \text{ (C\_INST)} \\ & \frac{\Gamma \vdash \forall \beta'. \forall \alpha'. \forall \alpha. \forall \beta. A \sqsubseteq \forall \beta'. \forall \alpha'. \forall \beta. A[\alpha'/\alpha]}{\Gamma \vdash \forall \beta'. \forall \alpha'. \forall \beta. A \sqsubseteq \forall \beta'. \forall \alpha'. A[\alpha'/\alpha][\beta'/\beta]} \text{ (C\_POLY)} \\ & \frac{\Gamma, \beta', \alpha' \vdash \beta'}{\Gamma, \beta', \alpha' \vdash \forall \beta. A[\alpha'/\alpha] \sqsubseteq A[\alpha'/\alpha][\beta'/\beta]} \text{ (C\_INST)} \\ & \frac{\Gamma \vdash \forall \beta'. \forall \alpha'. \forall \beta. A[\alpha'/\alpha] \sqsubseteq \forall \beta'. \forall \alpha'. A[\alpha'/\alpha][\beta'/\beta]}{\Gamma \vdash \forall \beta'. \forall \alpha'. \forall \beta. A[\alpha'/\alpha] \sqsubseteq \forall \beta'. \forall \alpha'. A[\alpha'/\alpha][\beta'/\beta]} \text{ (C\_POLY)} \end{aligned}$$

$\square$

**Lemma 10** (Type containment inversion: polymorphic function types). *If  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 \rightarrow A_2 \sqsubseteq \forall \alpha_2^{I_2}. B_1 \rightarrow B_2$ , then there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, \beta^J$ , and  $C^{I_{11}}$  such that*

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta^J \vdash C^{I_{11}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \beta^J. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}]$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_2[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$ , and
- type variables in  $\{\beta^J\}$  do not appear free in  $A_1$  and  $A_2$ .

*Proof.* By induction on the type containment derivation. Throughout the proof, we use the fact of  $\vdash \Gamma$  for applying (C\_REFL); it is shown easily by induction on the type containment derivation.

Case (C\_REFL): We have  $\alpha_1^{I_1} = \alpha_2^{I_2}$  and  $A_1 = B_1$  and  $A_2 = B_2$ . Let  $\alpha_{12}^{I_{12}}$  and  $\beta^J$  be the empty sequence,  $\alpha_{11}^{I_{11}} = \alpha_1^{I_1}$ , and  $C^{I_{11}} = \alpha_1^{I_1}$ . We have to show that

- $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq A_1$  and
- $\Gamma, \alpha_2^{I_2} \vdash A_2 \sqsubseteq B_2$ .

They are derived by (C\_REFL).

Case (C\_TRANS): By inversion, we have  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 \rightarrow A_2 \sqsubseteq D$  and  $\Gamma \vdash D \sqsubseteq \forall \alpha_2^{I_2}. B_1 \rightarrow B_2$  for some  $D$ . By Lemma 6,  $D = \forall \alpha_3^{I_3}. D_1 \rightarrow D_2$  for some  $\alpha_3^{I_3}$ ,  $D_1$ , and  $D_2$ . By the IH on  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 \rightarrow A_2 \sqsubseteq \forall \alpha_3^{I_3}. D_1 \rightarrow D_2$ , there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, C_1^{I_{11}}$ , and  $\beta_1^{J_1}$  such that

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_3^{I_3}, \beta_1^{J_1} \vdash C_1^{I_{11}}$ ,
- $\Gamma, \alpha_3^{I_3} \vdash D_1 \sqsubseteq \forall \beta_1^{J_1}. A_1[C_1^{I_{11}}/\alpha_{11}^{I_{11}}]$ ,
- $\Gamma, \alpha_3^{I_3} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_1^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq D_2$ , and
- type variables in  $\beta_1^{J_1}$  do not appear free in  $A_1$  and  $A_2$ .

By the IH on  $\Gamma \vdash \forall \alpha_3^{I_3}. D_1 \rightarrow D_2 \sqsubseteq \forall \alpha_2^{I_2}. B_1 \rightarrow B_2$ , there exist  $\alpha_{31}^{I_{31}}, \alpha_{32}^{I_{32}}, C_3^{I_{31}}$ , and  $\beta_3^{J_3}$  such that

- $\{\alpha_3^{I_3}\} = \{\alpha_{31}^{I_{31}}\} \uplus \{\alpha_{32}^{I_{32}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta_3^{J_3} \vdash C_3^{I_{31}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \beta_3^{J_3}. D_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}]$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. D_2[C_3^{I_{31}}/\alpha_{31}^{I_{31}}] \sqsubseteq B_2$ , and
- type variables in  $\beta_3^{J_3}$  do not appear free in  $D_1$  and  $D_2$ .

We show the conclusion by letting  $C^{I_{11}} = C_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}]^{I_{11}}$  and  $\beta^J = \alpha_{32}^{I_{32}}, \beta_3^{J_3}, \beta_1^{J_1}$ . We have to show that

- $\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}}, \beta_3^{J_3}, \beta_1^{J_1} \vdash C_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}]^{I_{11}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}]$ , and
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_2[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$ .

The first requirement is shown by  $\Gamma, \alpha_3^{I_3}, \beta_1^{J_1} \vdash C_1^{I_{11}}$  and  $\Gamma, \alpha_2^{I_2}, \beta_3^{J_3} \vdash C_3^{I_{31}}$  and Lemma 1 (2) and Lemma 2 (2).

Next, we show the second requirement. Since  $\Gamma, \alpha_3^{I_3} \vdash D_1 \sqsubseteq \forall \beta_1^{J_1}. A_1[C_1^{I_{11}}/\alpha_{11}^{I_{11}}]$  and  $\Gamma, \alpha_2^{I_2}, \beta_3^{J_3} \vdash C_3^{I_{31}}$ , we have  $\Gamma, \alpha_2^{I_2}, \alpha_3^{I_3}, \beta_3^{J_3} \vdash D_1 \sqsubseteq \forall \beta_1^{J_1}. A_1[C_1^{I_{11}}/\alpha_{11}^{I_{11}}]$  and  $\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}}, \beta_3^{J_3} \vdash C_3^{I_{31}}$  by Lemma 1 (3) and (2), respectively. Thus, by Lemma 2 (3),

$$\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}}, \beta_3^{J_3} \vdash D_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}] \sqsubseteq \forall \beta_1^{J_1}. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}]$$

(note that we can suppose that  $\alpha_{31}^{I_{31}}$  do not appear free in  $A_1$ ). By (C\_POLY),

$$\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}} \vdash \forall \beta_3^{J_3}. D_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}] \sqsubseteq \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}].$$

Since  $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \beta_3^{J_3}. D_1[C_3^{I_{31}}/\alpha_{31}^{I_{31}}]$ , we have

$$\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}} \vdash B_1 \sqsubseteq \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_1[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}]$$

by Lemma 1 (3) and (C\_TRANS). Since we can suppose that  $\alpha_{32}^{I_{32}}$  do not appear free in  $B_1$ , we have

$$\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_1[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}]$$

by (C\_GEN), (C\_POLY), and (C\_TRANS).

Finally, we show the third requirement. Since  $\Gamma, \alpha_3^{I_3} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_1^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq D_2$  and  $\Gamma, \alpha_2^{I_2}, \beta_3^{J_3} \vdash C_3^{I_{31}}$ , we have  $\Gamma, \alpha_2^{I_2}, \alpha_3^{I_3}, \beta_3^{J_3} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_1^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq D_2$  and  $\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}}, \beta_3^{J_3} \vdash C_3^{I_{31}}$  by Lemma 1 (3) and (2), respectively. Thus, by Lemma 2 (3),

$$\Gamma, \alpha_2^{I_2}, \alpha_{32}^{I_{32}}, \beta_3^{J_3} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq D_2[C_3^{I_{31}}/\alpha_{31}^{I_{31}}]$$

(note that we can suppose that  $\alpha_{31}^{I_{31}}$  do not appear free in  $A_2$ ). By (C\_POLY),

$$\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. D_2[C_3^{I_{31}}/\alpha_{31}^{I_{31}}].$$

Since  $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. D_2[C_3^{I_{31}}/\alpha_{31}^{I_{31}}] \sqsubseteq B_2$ , we have

$$\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$$

by (C\_TRANS). Because

$$\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \alpha_{12}^{I_{12}}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}]$$

by Lemma 9 (note that  $\vdash \Gamma, \alpha_2^{I_2}$  because we can assume that the type variables of  $\alpha_2^{I_2}$  do not occur in  $\Gamma$  without loss of generality), we have

$$\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \alpha_{32}^{I_{32}}. \forall \beta_3^{J_3}. \forall \beta_1^{J_1}. A_2[C_{01}^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$$

by (C\_TRANS).

Case (C\_FUN): Obvious by inversion.

Case (C\_INST): We have  $\alpha_1^{I_1} = \alpha, \alpha_2^{I_2}$  and  $B_1 = A_1[C/\alpha]$  and  $B_2 = A_2[C/\alpha]$  for some  $C$  such that  $\Gamma \vdash C$ . We show the conclusion by letting  $\alpha_{11}^{I_{11}} = \alpha, \alpha_2^{I_2}$ ,  $C^{I_{11}} = C, \alpha_2^{I_2}$ , and  $\alpha_{12}^{I_{12}}$  and  $\beta^J$  be the empty sequence. We have to show that

- $\Gamma, \alpha_2^{I_2} \vdash C$ ,
- $\Gamma, \alpha_2^{I_2} \vdash A_1[C/\alpha] \sqsubseteq A_1[C/\alpha]$ , and
- $\Gamma, \alpha_2^{I_2} \vdash A_2[C/\alpha] \sqsubseteq A_2[C/\alpha]$ .

The first is shown by Lemma 1 (1). The second is by (C\_REFL). The third is by (C\_REFL).

Case (C\_GEN): We have  $\alpha_2^{I_2} = \alpha, \alpha_1^{I_1}$  and  $A_1 = B_1$  and  $A_2 = B_2$  and  $\alpha \notin \text{ftv}(\forall \alpha_1^{I_1}. A_1 \rightarrow A_2)$ . We show the conclusion by letting  $\alpha_{11}^{I_{11}} = \alpha_1^{I_1}$ ,  $C^{I_{11}} = \alpha_1^{I_1}$ , and  $\alpha_{12}^{I_{12}}$  and  $\beta^J$  be the empty sequence. We have to show that

- $\Gamma, \alpha, \alpha_1^{I_1} \vdash A_1 \sqsubseteq A_1$  and
- $\Gamma, \alpha, \alpha_1^{I_1} \vdash A_2 \sqsubseteq A_2$ .

They are derived by (C\_REFL).

Case (C\_POLY): We have  $\alpha_1^{I_1} = \alpha, \alpha_{01}^{I_{01}}$  and  $\alpha_2^{I_2} = \alpha, \alpha_{02}^{I_{02}}$  and, by inversion,  $\Gamma, \alpha \vdash \forall \alpha_{01}^{I_{01}}. A_1 \rightarrow A_2 \sqsubseteq \forall \alpha_{02}^{I_{02}}. B_1 \rightarrow B_2$ . By the IH, there exist some  $\alpha_{011}^{I_{011}}$ ,  $\alpha_{12}^{I_{12}}$ ,  $\beta^J$ , and  $C_0^{I_{011}}$  such that

- $\{\alpha_{01}^{I_{01}}\} = \{\alpha_{011}^{I_{011}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha, \alpha_{02}^{I_{02}}, \beta^J \vdash C_0^{I_{011}}$ ,

- $\Gamma, \alpha, \alpha_{02}^{I_{02}} \vdash B_1 \sqsubseteq \forall \beta^J. A_1[C_0^{I_{011}}/\alpha_{011}^{I_{011}}]$ ,
- $\Gamma, \alpha, \alpha_{02}^{I_{02}} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_2[C_0^{I_{011}}/\alpha_{011}^{I_{011}}] \sqsubseteq B_2$ , and
- type variables in  $\beta^J$  do not appear free in  $A_1$  and  $B_1$ .

We can prove the conclusion by letting  $\alpha_{11}^{I_{11}} = \alpha, \alpha_{011}^{I_{011}}$  and  $C^{I_{11}} = \alpha, C_0^{I_{011}}$ .

Case (C\_DFUN): It is found that, for some  $\alpha$ ,  $\alpha_1^{I_1} = \alpha$  and  $\alpha_2^{I_2}$  is the empty sequence and  $B_1 = A_1$  and  $B_2 = \forall \alpha. A_2$ . We show the conclusion by letting  $\alpha_{12}^{I_{12}} = \alpha$  and  $\alpha_{11}^{I_{11}}, C^{I_{11}}$ , and  $\beta^J$  be the empty sequence. It suffices to show that  $\Gamma \vdash A_1 \sqsubseteq A_1$  and  $\Gamma \vdash \forall \alpha. A_2 \sqsubseteq \forall \alpha. A_2$ , which are derived by (C\_REFL).

Case (C\_PROD), (C\_SUM), (C\_LIST), (C\_DPROD), (C\_DSUM), and (C\_DLIST): Contradictory.

□

**Lemma 11** (Type containment inversion: monomorphic function types). *If  $\Gamma \vdash A_1 \rightarrow A_2 \sqsubseteq B_1 \rightarrow B_2$ , then  $\Gamma \vdash B_1 \sqsubseteq A_1$  and  $\Gamma \vdash A_2 \sqsubseteq B_2$ .*

*Proof.* By Lemma 10,  $\Gamma \vdash B_1 \sqsubseteq \forall \alpha. A_1$  and  $\Gamma \vdash \forall \alpha. A_2 \sqsubseteq B_2$  for some  $\alpha$  such that type variables in  $\alpha$  do not appear free in  $A_1$  and  $A_2$ . Since  $\Gamma \vdash \forall \alpha. A_1 \sqsubseteq A_1$  by (C\_INST) (we can substitute any type, e.g.,  $\forall \beta. \beta$ , for  $\alpha$ ), we have  $\Gamma \vdash B_1 \sqsubseteq A_1$  by (C\_TRANS). Since  $\Gamma \vdash A_2 \sqsubseteq \forall \alpha. A_2$  by (C\_GEN), we have  $\Gamma \vdash A_2 \sqsubseteq B_2$ . □

**Lemma 12** (Value inversion: constants). *If  $\Gamma \vdash c : A$ , then  $\Gamma \vdash ty(c) \sqsubseteq A$ .*

*Proof.* By induction on the typing derivation for  $c$ . There are only three typing rules that can be applied to  $c$ .

Case (T\_CONST): By (C\_REFL).

Case (T\_GEN): We are given  $\Gamma \vdash c : \forall \alpha. B$  for some  $\alpha$  and  $B$  (i.e.,  $A = \forall \alpha. B$ ), and, by inversion,  $\Gamma, \alpha \vdash c : B$ . By the IH,  $\Gamma, \alpha \vdash ty(c) \sqsubseteq B$ . By (C\_POLY),  $\Gamma \vdash \forall \alpha. ty(c) \sqsubseteq \forall \alpha. B$ . Since  $ty(c)$  is closed, we have  $\Gamma \vdash ty(c) \sqsubseteq \forall \alpha. ty(c)$  by (C\_GEN). Thus, by (C\_TRANS), we have the conclusion.

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 13** (Progress). *If  $\Delta \vdash M : A$ , then:*

- $M \longrightarrow M'$  for some  $M'$ ;
- $M$  is a value; or
- $M = E[\#op(v)]$  for some  $E, op$ , and  $v$  such that  $op \notin E$ .

*Proof.* By induction on the typing derivation for  $M$ . We proceed by case analysis on the typing rule applied last to derive  $\Delta \vdash M : A$ .

Case (T\_VAR): Contradictory.

Case (T\_CONST), (T\_ABS), and (T\_NIL): Obvious.

Case (T\_APP): We are given

- $M = M_1 M_2$ ,
- $\Delta \vdash M_1 M_2 : A$ ,
- $\Delta \vdash M_1 : B \rightarrow A$ , and
- $\Delta \vdash M_2 : B$

for some  $M_1, M_2$ , and  $B$ . By case analysis on the behavior of  $M_1$ . We have three cases to consider by the IH.

Case  $M_1 \longrightarrow M'_1$  for some  $M'_1$ : We have  $M \longrightarrow M'_1 M_2$ .

Case  $M_1 = E_1[\#op(v)]$  for some  $E_1$ ,  $op$ , and  $v$  such that  $op \notin E_1$ : We have the third case in the conclusion by letting  $E = E_1 M_2$ .

Case  $M_1 = v_1$  for some  $v_1$ : By case analysis on the behavior of  $M_2$  with the IH.

Case  $M_2 \longrightarrow M'_2$  for some  $M'_2$ : We have  $M \longrightarrow v_1 M'_2$ .

Case  $M_2 = E_2[\#op(v)]$  for some  $E_2$ ,  $op$ , and  $v$  such that  $op \notin E_2$ : We have the third case in the conclusion by letting  $E = v_1 E_2$ .

Case  $M_2 = v_2$  for some  $v_2$ : By Lemma 8 on  $v_1$ , we have two cases to consider.

Case  $v_1 = c_1$ : Since  $\Delta \vdash c_1 : B \rightarrow A$ , we have  $\Delta \vdash ty(c_1) \sqsubseteq B \rightarrow A$  by Lemma 12. By Lemma 6 (2), it is found that  $ty(c_1) = \iota \rightarrow C$  for some  $\iota$  and  $C$ . Since  $\Delta \vdash \iota \rightarrow C \sqsubseteq B \rightarrow A$ , we have  $\Delta \vdash B \sqsubseteq \iota$  by Lemma 11. Since  $\Delta \vdash v_2 : B$ ,  $unqualify(B)$  is not a type variable by Lemma 7. Thus, since  $\Delta \vdash B \sqsubseteq \iota$ , it is found that  $unqualify(B) = \iota$  by Lemma 6. Since  $\Delta \vdash v_2 : B$ , we have  $v_2 = c_2$  for some  $c_2$  by Lemma 8. Since  $\Delta \vdash c_2 : B$ , we have  $\Delta \vdash ty(c_2) \sqsubseteq B$  by Lemma 12. Since  $unqualify(B) = \iota$ , we have  $ty(c_2) = \iota$  by Lemma 6. Thus,  $\zeta(c_1, c_2)$  is defined, and  $M = c_1 c_2 \longrightarrow \zeta(c_1, c_2)$  by (R\_CONST)/(E\_EVAL).

Case  $v_1 = \lambda x.M'$ : By (R\_BETA)/(E\_EVAL),  $M = (\lambda x.M') v_2 \longrightarrow M'[v_2/x]$ .

Case (T\_GEN): By the IH.

Case (T\_INST): By the IH.

Case (T\_OP): We are given

- $M = \#op(M')$ ,
- $ty(op) = \forall \alpha. A' \hookrightarrow B'$ ,
- $\Delta \vdash \#op(M') : B'[C/\alpha]$ , and
- $\Delta \vdash M' : A'[C/\alpha]$

for some  $op$ ,  $M'$ ,  $\alpha$ ,  $A'$ ,  $B'$ , and  $C$ . By case analysis on the behavior of  $M'$  with the IH.

Case  $M' \longrightarrow M''$  for some  $M''$ : We have  $M \longrightarrow \#op(M'')$ .

Case  $M' = E'[\#op'(v)]$  for some  $E'$ ,  $op'$ , and  $v$  such that  $op' \notin E'$ : We have the third case in the conclusion by letting  $E = \#op(E')$ .

Case  $M' = v$  for some  $v$ : We have the third case in the conclusion by letting  $E = []$ .

Case (T\_HANDLE): We are given

- $M = \text{handle } M' \text{ with } H$ ,
- $\Delta \vdash M' : B$ , and
- $\Delta \vdash H : B \Rightarrow A$

for some  $M'$ ,  $H$ , and  $B$ . By case analysis on the behavior of  $M'$  with the IH.

Case  $M' \longrightarrow M''$  for some  $M''$ : We have  $M \longrightarrow \text{handle } M'' \text{ with } H$ .

Case  $M' = E'[\#op(v)]$  for some  $E'$ ,  $op$ , and  $v$  such that  $op \notin E'$ : If handler  $H$  contains an operation clause  $op(x, k) \rightarrow M''$ , then we have  $M \longrightarrow M''[v/x][\lambda y. \text{handle } E'[y] \text{ with } H/k]$  by (R\_HANDLE)/(E\_EVAL).

Otherwise, if  $H$  contains no operation clause for  $op$ , we have the third case in the conclusion by letting  $E = \text{handle } E' \text{ with } H$ .

Case  $M' = v$  for some  $v$ : By (R\_RETURN)/(E\_EVAL).

Case (T\_PAIR): We are given

- $M = (M_1, M_2)$ ,
- $\Delta \vdash M_1 : B_1$ , and
- $\Delta \vdash M_2 : B_2$

for some  $M_1, M_2, B_1$ , and  $B_2$ . By case analysis on the behavior of  $M_1$  with the IH.

Case  $M_1 \longrightarrow M'_1$  for some  $M'_1$ : We have  $M = (M'_1, M_2)$ .

Case  $M_1 = E_1[\#op(v)]$  for some  $E_1, op$ , and  $v$  such that  $op \notin E_1$ : We have the third case in the conclusion by letting  $E = (E_1, M_2)$ .

Case  $M_1 = v_1$  for some  $v_1$ : By case analysis on the behavior of  $M_2$  with the IH.

Case  $M_2 \longrightarrow M'_2$ : We have  $M_2 \longrightarrow (v_1, M'_2)$ .

Case  $M_2 = E_2[\#op(v)]$  for some  $E_2, op$ , and  $v$  such that  $op \notin E_2$ : We have the third case in the conclusion by letting  $E = (v_1, E_2)$ .

Case  $M_2 = v_2$ : We have the second case in the conclusion since  $M = (v_1, v_2)$ .

Case (T\_PROJ1): We are given

- $M = \pi_1 M'$  and
- $\Delta \vdash M' : A \times B$

for some  $M'$  and  $B$ . By case analysis on the behavior of  $M'$  with the IH.

Case  $M' \longrightarrow M''$  for some  $M''$ : We have  $M \longrightarrow \pi_1 M''$ .

Case  $M' = E'[\#op(v)]$  for some  $E', op$ , and  $v$  such that  $op \notin E'$ : We have the third case in the conclusion by letting  $E = \pi_1 E'$ .

Case  $M' = v'$  for some  $v'$ : Since  $\Delta \vdash M' : A \times B$  (i.e.,  $\Delta \vdash v' : A \times B$ ), we have  $v' = (v_1, v_2)$  for some  $v_1$  and  $v_2$  by Lemma 8. By (R\_PROJ1)/(E\_EVAL), we finish.

Case (T\_PROJ2): Similarly to the case for (T\_PROJ1).

Case (T\_INL), (T\_INR), and (T\_CONS): Similarly to the case for (T\_PAIR).

Case (T\_CASE): We are given

- $M = \text{case } M' \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2$  and
- $\Delta \vdash M' : B + C$

for some  $M', M_1, M_2, x, y, B$ , and  $C$ . By case analysis on the behavior of  $M'$  with the IH.

Case  $M' \longrightarrow M''$  for some  $M''$ : We have  $M \longrightarrow \text{case } M'' \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2$ .

Case  $M' = E'[\#op(v)]$  for some  $E', op$ , and  $v$  such that  $op \notin E'$ : We have the third case in the conclusion by letting  $E = \text{case } E' \text{ of } \text{inl } x \rightarrow M_1; \text{inr } y \rightarrow M_2$ .

Case  $M' = v$  for some  $v$ : By Lemma 8,  $v = \text{inl } v'$  or  $v = \text{inr } v'$  for some  $v'$ . We finish by (R\_CASEL)/(E\_EVAL) or (R\_CASER)/(E\_EVAL).

Case (T\_CASELIST): Similar to the case for (T\_CASE).

Case (T\_FIX): By (R\_FIX)/(E\_EVAL).

□

#### Lemma 14.

1. If  $\Gamma \vdash M : A$ , then  $\Gamma \vdash A$ .
2. If  $\Gamma \vdash H : A \Rightarrow B$ , then  $\Gamma \vdash B$ .

*Proof.* Straightforward by mutual induction on the typing derivations. The case for (T\_OP) depends on Lemma 2 and Definition 4, which states that, for  $op$  such that  $ty(op) = \forall \alpha. A \hookrightarrow B$ ,  $ftv(B) \subseteq \{\alpha\}$ . □

**Lemma 15** (Value inversion: lambda abstractions). *If  $\Gamma \vdash \lambda x. M : A$ , then  $\Gamma, \alpha, x : B \vdash M : C$  and  $\Gamma \vdash \forall \alpha. B \rightarrow C \sqsubseteq A$  for some  $\alpha, B$ , and  $C$ .*

*Proof.* By induction on the typing derivation for  $\lambda x.M$ . There are only three typing rules that can be applied to  $\lambda x.M$ .

Case (T\_ABS): We have  $A = B \rightarrow C$  for some  $B$  and  $C$ . Let  $\alpha$  be the empty sequence. We have the conclusion by inversion and (C\_REFL).

Case (T\_GEN): We are given  $\Gamma \vdash \lambda x.M : \forall \beta. D$  for some  $\beta$  and  $D$  (i.e.,  $A = \forall \beta. D$ ), and, by inversion,  $\Gamma, \beta \vdash \lambda x.M : D$ . By the IH,  $\Gamma, \beta, \gamma^I, x : B \vdash M : C$  and  $\Gamma, \beta \vdash \forall \gamma^I. B \rightarrow C \sqsubseteq D$  for some  $\gamma^I, B$ , and  $C$ . We show the conclusion by letting  $\alpha = \beta, \gamma^I$ . It suffices to show that  $\Gamma \vdash \forall \beta. \forall \gamma^I. B \rightarrow C \sqsubseteq \forall \beta. D$ , which is derived from  $\Gamma, \beta \vdash \forall \gamma^I. B \rightarrow C \sqsubseteq D$  with (C\_POLY).

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 16** (Value inversion: pairs). *If  $\Gamma \vdash (M_1, M_2) : A$ , then  $\Gamma, \alpha \vdash M_1 : B_1$  and  $\Gamma, \alpha \vdash M_2 : B_2$  and  $\Gamma \vdash \forall \alpha. B_1 \times B_2 \sqsubseteq A$  for some  $\alpha, B_1$ , and  $B_2$ .*

*Proof.* By induction on the typing derivation for  $(M_1, M_2)$ . There are only three typing rules that can be applied to  $(M_1, M_2)$ .

Case (T\_PAIR): Obvious by (C\_REFL).

Case (T\_GEN): We are given  $\Gamma \vdash (M_1, M_2) : \forall \beta. C$  (i.e.,  $A = \forall \beta. C$ ) and, by inversion,  $\Gamma, \beta \vdash (M_1, M_2) : C$ . By the IH,  $\Gamma, \beta, \gamma^I \vdash M_1 : B_1$  and  $\Gamma, \beta, \gamma^I \vdash M_2 : B_2$   $\Gamma, \beta \vdash \forall \gamma^I. B_1 \times B_2 \sqsubseteq C$  for some  $\gamma^I, B_1$ , and  $B_2$ . We show the conclusion by letting  $\alpha = \beta, \gamma^I$ . It suffices to show that  $\Gamma \vdash \forall \beta. \forall \gamma^I. B_1 \times B_2 \sqsubseteq \forall \beta. C$ , which is derived from  $\Gamma, \beta \vdash \forall \gamma^I. B_1 \times B_2 \sqsubseteq C$  with (C\_POLY).

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 17** (Value inversion: left injections). *If  $\Gamma \vdash \text{inl } M : A$ , then  $\Gamma, \alpha \vdash M : B$  and  $\Gamma \vdash \forall \alpha. B + C \sqsubseteq A$  for some  $\alpha, B$ , and  $C$ .*

*Proof.* By induction on the typing derivation for  $\text{inl } M$ . There are only three typing rules that can be applied to  $\text{inl } M$ .

Case (T\_INL): Obvious by (C\_REFL).

Case (T\_GEN): We are given  $\Gamma \vdash \text{inl } M : \forall \beta. D$  (i.e.,  $A = \forall \beta. D$ ) and, by inversion,  $\Gamma, \beta \vdash \text{inl } M : D$ . By the IH,  $\Gamma, \beta, \gamma^I \vdash M : B$  and  $\Gamma, \beta \vdash \forall \gamma^I. B + C \sqsubseteq D$  for some  $\gamma^I, B$ , and  $C$ . We show the conclusion by letting  $\alpha = \beta, \gamma^I$ . It suffices to show that  $\Gamma \vdash \forall \beta. \forall \gamma^I. B + C \sqsubseteq \forall \beta. D$ , which is derived from  $\Gamma, \beta \vdash \forall \gamma^I. B + C \sqsubseteq D$  with (C\_POLY).

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 18** (Value inversion: right injections). *If  $\Gamma \vdash \text{inr } M : A$ , then  $\Gamma, \alpha \vdash M : C$  and  $\Gamma \vdash \forall \alpha. B + C \sqsubseteq A$  for some  $\alpha, B$ , and  $C$ .*

*Proof.* Similarly to the proof of Lemma 17.

□

**Lemma 19** (Value inversion: cons). *If  $\Gamma \vdash \text{cons } M : A$ , then  $\Gamma, \alpha \vdash M : B \times B \text{list}$  and  $\Gamma \vdash \forall \alpha. B \text{list} \sqsubseteq A$  for some  $\alpha$  and  $B$ .*

*Proof.* By induction on the typing derivations for  $\text{cons } M$ . There are only three typing rules that can be applied to  $\text{cons } M$ .

Case (T\_CONS): Obvious by (C\_REFL).

Case (T\_GEN): We are given  $\Gamma \vdash \text{cons } M : \forall \beta. C$  (i.e.,  $A = \forall \beta. C$ ) and, by inversion,  $\Gamma, \beta \vdash \text{cons } M : C$ . By the IH,  $\Gamma, \beta, \gamma^I \vdash M : B \times B \text{ list}$  and  $\Gamma, \beta \vdash \forall \gamma^I. B \text{ list} \sqsubseteq C$  for some  $\gamma^I$  and  $B$ . We show the conclusion by letting  $\alpha = \beta, \gamma^I$ . It suffices to show that  $\Gamma \vdash \forall \beta. \forall \gamma^I. B \text{ list} \sqsubseteq \forall \beta. C$ , which is derived from  $\Gamma, \beta \vdash \forall \gamma^I. B \text{ list} \sqsubseteq C$  with (C\_POLY).

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 20.** *If  $\text{ty}(\text{op}) = \forall \alpha^I. A \hookrightarrow B$  and  $\Gamma \vdash \# \text{op}(v) : C$ , then*

- $\Gamma, \beta^J \vdash D^I$ ,
- $\Gamma, \beta^J \vdash v : A[D^I/\alpha^I]$ , and
- $\Gamma \vdash \forall \beta^J. B[D^I/\alpha^I] \sqsubseteq C$

for some  $\beta^J$  and  $D^I$ .

*Proof.* By induction on the typing derivation for  $\# \text{op}(v)$ . There are only three typing rules that can be applied to  $\# \text{op}(v)$ .

Case (T\_OP): We have  $C = B[D^I/\alpha^I]$  and  $\Gamma \vdash D^I$  and  $\Gamma \vdash v : A[D^I/\alpha^I]$  for some  $D^I$ . We have the conclusion by letting  $\beta^J$  be the empty sequence; note that  $\Gamma \vdash B[D^I/\alpha^I] \sqsubseteq B[D^I/\alpha^I]$  by (C\_REFL).

Case (T\_GEN): We are given  $C = \forall \beta. C_0$  and, by inversion,  $\Gamma, \beta \vdash \# \text{op}(v) : C_0$  for some  $\beta$  and  $C_0$ . By the IH, there exist some  $\beta_0^{J_0}$  and  $D^I$  such that

- $\Gamma, \beta, \beta_0^{J_0} \vdash D^I$ ,
- $\Gamma, \beta, \beta_0^{J_0} \vdash v : A[D^I/\alpha^I]$  and
- $\Gamma, \beta \vdash \forall \beta_0^{J_0}. B[D^I/\alpha^I] \sqsubseteq C_0$ .

We show the conclusion by letting  $\beta^J = \beta, \beta_0^{J_0}$ . It suffices to show  $\Gamma \vdash \forall \beta. \forall \beta_0^{J_0}. B[D^I/\alpha^I] \sqsubseteq \forall \beta. C_0$ , which is proven from  $\Gamma, \beta \vdash \forall \beta_0^{J_0}. B[D^I/\alpha^I] \sqsubseteq C_0$  with (C\_POLY).

Case (T\_INST): By the IH and (C\_TRANS).

□

**Lemma 21.** *If  $\Gamma, \alpha^I \vdash E[M] : A$ , then*

- $\Gamma, \alpha^I, \beta^J \vdash M : B$  and
- $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E[y] : A$  for any  $y \notin \text{dom}(\Gamma)$

for some  $\beta^J$  and  $B$ .

*Proof.* By induction on the typing derivation of  $\Gamma, \alpha^I \vdash E[M] : A$ .

Suppose that  $E = []$ . Since  $\Gamma, \alpha^I \vdash E[M] : A$ , we have  $\Gamma, \alpha^I \vdash M : A$ . We let  $\beta^J$  be the empty sequence and  $B = A$ . It is then trivial that  $\Gamma, y : \forall \alpha^I. B, \alpha^I \vdash E[y] : A$  by (T\_INST). Note that  $\vdash \Gamma$  and  $\Gamma \vdash \forall \alpha. B$  by Lemma 14.

In what follows, we suppose that  $E \neq []$ . We proceed by case analysis on the typing rule applied last to derive  $\Gamma, \alpha^I \vdash E[M] : A$ .

Case (T\_VAR), (T\_CONST), (T\_ABS), (T\_NIL), and (T\_FIX): Contradictory with the assumption that  $E \neq []$ .

Case (T\_APP): By case analysis on  $E$ .

Case  $E = E' M_2$ : By inversion of the typing derivation, we have  $\Gamma, \alpha^I \vdash E'[M] : C \rightarrow A$  and  $\Gamma, \alpha^I \vdash M_2 : C$  for some  $C$ . By the IH, (1)  $\Gamma, \alpha^I, \beta^J \vdash M : B$  for some  $\beta^J$  and  $B$  and (2) for any  $y \notin \text{dom}(\Gamma)$ ,  $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E'[y] : C \rightarrow A$ . By Lemma 1 (4) and (T\_APP),  $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E'[y] M_2 : A$ , i.e.,  $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E[y] : A$ .



Case  $E = v_1 E'$ : Similarly to the above case.

Case (T\_GEN): We have  $\Gamma, \alpha^I \vdash E[M] : \forall \gamma. A'$  and, by inversion,  $\Gamma, \alpha^I, \gamma \vdash E[M] : A'$  for some  $\gamma$  and  $A'$  (note  $A = \forall \gamma. A'$ ). By the IH, (1)  $\Gamma, \alpha^I, \gamma, \beta^J \vdash M : B$  for some  $\beta^J$  and  $B$  and (2) for any  $y \notin \text{dom}(\Gamma)$ ,  $\Gamma, y : \forall \alpha^I. \forall \gamma. \forall \beta^J. B, \alpha^I, \gamma \vdash E[y] : A'$ .

By (T\_GEN),  $\Gamma, y : \forall \alpha^I. \forall \gamma. \forall \beta^J. B, \alpha^I \vdash E[y] : \forall \gamma. A'$ . Since  $A = \forall \gamma. A'$ , we finish.

Otherwise: By the IH(s) and the corresponding typing rule, as the case for (T\_APP).

□

**Lemma 22.** *Suppose that  $\Gamma_1 \vdash A \sqsubseteq B$  and  $\Gamma_1 \vdash A$ .*

1. *If  $\Gamma_1, x : B, \Gamma_2 \vdash M : C$ , then  $\Gamma_1, x : A, \Gamma_2 \vdash M : C$ .*
2. *If  $\Gamma_1, x : B, \Gamma_2 \vdash H : C \Rightarrow D$ , then  $\Gamma_1, x : A, \Gamma_2 \vdash H : C \Rightarrow D$ .*

*Proof.* Straightforward by mutual induction on the typing derivations.

□

**Lemma 23.** *If  $\text{ty}(\text{op}) = \forall \alpha^I. A \hookrightarrow B$  and  $\Gamma \vdash E[\# \text{op}(v)] : C$ , then*

- $\Gamma, \beta^J \vdash D^I$ ,
- $\Gamma, \beta^J \vdash v : A[D^I/\alpha^I]$ , and
- *for any  $y \notin \text{dom}(\Gamma)$ ,  $\Gamma, y : \forall \beta^J. B[D^I/\alpha^I] \vdash E[y] : C$*

*for some  $\beta^J$  and  $D^I$ .*

*Proof.* By Lemma 21,

- $\Gamma, \beta_1^{J_1} \vdash \# \text{op}(v) : C'$  and
- $\Gamma, y : \forall \beta_1^{J_1}. C' \vdash E[y] : C$  for any  $y \notin \text{dom}(\Gamma)$

for some  $\beta_1^{J_1}$  and  $C'$ . By Lemma 20,

- $\Gamma, \beta_1^{J_1}, \beta_2^{J_2} \vdash D^I$ ,
- $\Gamma, \beta_1^{J_1}, \beta_2^{J_2} \vdash v : A[D^I/\alpha^I]$ , and
- $\Gamma, \beta_1^{J_1} \vdash \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq C'$

for some  $\beta_2^{J_2}$  and  $D^I$ .

We show the conclusion by letting  $\beta^J = \beta_1^{J_1}, \beta_2^{J_2}$ . It suffices to show that, for any  $y \notin \text{dom}(\Gamma)$ ,

$$\Gamma, y : \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \vdash E[y] : C.$$

Since  $\Gamma, \beta_1^{J_1} \vdash \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq C'$ , we have

$$\Gamma \vdash \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq \forall \beta_1^{J_1}. C'$$

by (C\_POLY). Since  $\Gamma, y : \forall \beta_1^{J_1}. C' \vdash E[y] : C$ , we have

$$\Gamma, y : \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \vdash E[y] : C.$$

by Lemma 22.

□

**Lemma 24** (Type containment inversion: product types). *If  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 \times A_2 \sqsubseteq \forall \alpha_2^{I_2}. B_1 \times B_2$ , then there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, \beta^J$ , and  $C^{I_{11}}$  such that*

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta^J \vdash C^{I_{11}}$ ,

- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_1$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_2[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$ , and
- type variables in  $\{\beta^J\}$  do not appear free in  $A_1$  and  $A_2$ .

*Proof.* By induction on the type containment derivation. The proof is similar to that of Lemma 10.  $\square$

**Lemma 25** (Type containment inversion: sum types). *If  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 + A_2 \sqsubseteq \forall \alpha_2^{I_2}. B_1 + B_2$ , then there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, \beta^J$ , and  $C^{I_{11}}$  such that*

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta^J \vdash C^{I_{11}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_1$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_2[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$ , and
- type variables in  $\{\beta^J\}$  do not appear free in  $A_1$  and  $A_2$ .

*Proof.* By induction on the type containment derivation. The proof is similar to that of Lemma 10.  $\square$

**Lemma 26** (Type containment inversion: list types). *If  $\Gamma \vdash \forall \alpha_1^{I_1}. A \text{ list} \sqsubseteq \forall \alpha_2^{I_2}. B \text{ list}$ , then there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, \beta^J$ , and  $C^{I_{11}}$  such that*

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta^J \vdash C^{I_{11}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B$ , and
- type variables in  $\{\beta^J\}$  do not appear free in  $A$ .

*Proof.* By induction on the type containment derivation. The proof is similar to that of Lemma 10.  $\square$

**Lemma 27.** *Assume that  $\Gamma \vdash B \sqsubseteq C$ .*

1. *If  $\alpha \notin \text{ftv}(A)^+$ , then  $\Gamma \vdash A[C/\alpha] \sqsubseteq A[B/\alpha]$ .*
2. *If  $\alpha \notin \text{ftv}(A)^-$ , then  $\Gamma \vdash A[B/\alpha] \sqsubseteq A[C/\alpha]$ .*

*Proof.* By structural induction on  $A$ . In what follows, we assume  $\vdash \Gamma$  because it can be shown easily by induction on the derivation of  $\Gamma \vdash B \sqsubseteq C$ .

Case  $A = \beta$ : If  $\beta = \alpha$ , then we have to show that  $\Gamma \vdash B \sqsubseteq C$ , which is assumed. Note that we do not need to consider the first, negative case, i.e., to show  $\Gamma \vdash C \sqsubseteq B$ , because no occurrence of type variable  $\alpha$  in type  $\alpha$  is negative.

Otherwise, if  $\beta \neq \alpha$ , then it suffices to show that  $\Gamma \vdash \beta \sqsubseteq \beta$ , which is derived by (C\_REFL).

Case  $A = \iota$ : By (C\_REFL).

Case  $A = \forall \beta. A'$ : By Lemma 1 (3), the IH, and (C\_POLY) for each case.

Case  $A = A_1 \rightarrow A_2$ : By the IHs and (C\_FUN) for each case.

Case  $A = A_1 \times A_2$ : By the IH and (C\_PROD) for each case.

Case  $A = A_1 + A_2$ : By the IH and (C\_SUM) for each case.

Case  $A = A' \text{ list}$ : By the IH and (C\_LIST) for each case.

$\square$

**Lemma 28.** Assume that  $\vdash \Gamma$  and  $\alpha \notin \text{ftv}(A)$ .

1. If  $\beta \notin \text{ftv}(A)_{\text{ns}}^+$ , then  $\Gamma \vdash \forall \alpha. A[B/\beta] \sqsubseteq A[\forall \alpha. B/\beta]$ .
2. If  $\beta \notin \text{ftv}(A)^-$ , then  $\Gamma \vdash A[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. A[B/\beta]$ .

*Proof.* We first show case (2). By (C\_TRANS), it suffices to show that  $\Gamma \vdash A[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. A[\forall \alpha. B/\beta]$  and  $\Gamma \vdash \forall \alpha. A[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. A[B/\beta]$ . The former is derived by (C\_GEN). The latter is derived as follows, where we have  $\vdash \Gamma, \alpha$  because we can assume that  $\alpha \notin \text{dom}(\Gamma)$  without loss of generality:

$$\frac{\frac{\frac{\vdash \Gamma, \alpha}{\Gamma, \alpha \vdash \forall \alpha. B \sqsubseteq B} \text{ (C_INST)}}{\Gamma, \alpha \vdash A[\forall \alpha. B/\beta] \sqsubseteq A[B/\beta]} \text{ by Lemma 27}}{\Gamma \vdash \forall \alpha. A[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. A[B/\beta]} \text{ (C_POLY)}$$

Next, we show case (1) by induction on  $A$ .

Case  $A = \gamma$ : If  $\gamma = \beta$ , then we have to show that  $\Gamma \vdash \forall \alpha. B \sqsubseteq \forall \alpha. B$ , which is shown by (C\_REFL). Otherwise, if  $\gamma \neq \beta$ , then we have to show that  $\Gamma \vdash \forall \alpha. \gamma \sqsubseteq \gamma$ , which is derived by (C\_INST) (the type used for instantiation can be arbitrary, e.g.,  $\forall \alpha. \alpha$ ).

Case  $A = \iota$ : By (C\_INST).

Case  $A = C \rightarrow D$ : The occurrences of  $\beta$  in  $C \rightarrow D$  are only negative or strictly positive. By definition, the occurrences of  $\beta$  in  $C$  are only positive. Thus, by case (2),  $\Gamma \vdash C[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. C[B/\beta]$ . By definition, the occurrences of  $\beta$  in  $D$  are only negative or strictly positive. Thus, by the IH,  $\Gamma \vdash \forall \alpha. D[B/\beta] \sqsubseteq D[\forall \alpha. B/\beta]$ . By (C\_FUN),

$$\Gamma \vdash (\forall \alpha. C[B/\beta]) \rightarrow \forall \alpha. D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow D[\forall \alpha. B/\beta].$$

By (C\_DFUN) and (C\_TRANS),

$$\Gamma \vdash \forall \alpha. (\forall \alpha. C[B/\beta]) \rightarrow D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow D[\forall \alpha. B/\beta]. \quad (1)$$

By (C\_INST),

$$\Gamma, \alpha \vdash \forall \alpha. C[B/\beta] \sqsubseteq C[B/\beta]. \quad (2)$$

By (C\_FUN) and (C\_POLY) with (2),

$$\Gamma \vdash \forall \alpha. C[B/\beta] \rightarrow D[B/\beta] \sqsubseteq \forall \alpha. (\forall \alpha. C[B/\beta]) \rightarrow D[B/\beta].$$

Thus, by (C\_TRANS) with (1),

$$\Gamma \vdash \forall \alpha. C[B/\beta] \rightarrow D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow D[\forall \alpha. B/\beta].$$

Case  $A = \forall \gamma. C$ : By the IH, (C\_POLY), Lemma 9, and (C\_TRANS).

Case  $A = C \times D$ : The occurrences of  $\beta$  in  $C \times D$  are only negative or strictly positive. By definition, the occurrences of  $\beta$  in  $C$  are only negative or strictly positive. Thus, by the IH,  $\Gamma \vdash \forall \alpha. C[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta]$ . Similarly, we also have  $\Gamma \vdash \forall \alpha. D[B/\beta] \sqsubseteq D[\forall \alpha. B/\beta]$ . By (C\_PROD),

$$\Gamma \vdash (\forall \alpha. C[B/\beta]) \times \forall \alpha. D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \times D[\forall \alpha. B/\beta].$$

By (C\_DPROD) and (C\_TRANS),

$$\Gamma \vdash \forall \alpha. (C[B/\beta] \times D[B/\beta]) \sqsubseteq C[\forall \alpha. B/\beta] \times D[\forall \alpha. B/\beta].$$

Case  $A = C + D$ : Similarly to the case that  $A$  is a product type; this case uses (C\_SUM) and (C\_DSUM) instead of (C\_PROD) and (C\_DPROD).

Case  $A = C \text{ list}$ : Similarly to the case that  $A$  is a product type; this case uses (C\_LIST) and (C\_DLIST) instead of (C\_PROD) and (C\_DPROD).

□

**Lemma 29** (Subject reduction). *Assume that all operations satisfy the signature restriction.*

1. If  $\Delta \vdash M_1 : A$  and  $M_1 \rightsquigarrow M_2$ , then  $\Delta \vdash M_2 : A$ .
2. If  $\Delta \vdash M_1 : A$  and  $M_1 \longrightarrow M_2$ , then  $\Delta \vdash M_2 : A$ .

*Proof.* 1. Suppose that  $\Delta \vdash M_1 : A$  and  $M_1 \rightsquigarrow M_2$ . By induction on the typing derivation for  $M_1$ .

Case (T\_VAR), (T\_OP), (T\_PAIR), (T\_INL), (T\_INR), and (T\_CONS): Contradictory because there are no reduction rules that can be applied to  $M_1$ .

Case (T\_CONST), (T\_ABS), and (T\_NIL): Contradictory since  $M_1$  is a value and no reduction rules can be applied to values.

Case (T\_APP): We have two reduction rules which can be applied to function applications.

Case (R\_CONST): We are given

- $M_1 = c_1 c_2$ ,
- $M_2 = \zeta(c_1, c_2)$ ,
- $\Delta \vdash c_1 c_2 : A$ ,
- $\Delta \vdash c_1 : B \rightarrow A$ , and
- $\Delta \vdash c_2 : B$

for some  $c_1$ ,  $c_2$ , and  $B$ . By Lemma 12,  $\Delta \vdash ty(c_1) \sqsubseteq B \rightarrow A$ . By Lemma 6 and Assumption 1,  $ty(c_1) = \iota \rightarrow C$  for some  $\iota$  and  $C$ . Since  $\zeta(c_1, c_2)$  is defined, it is found that  $ty(c_2) = \iota$  and  $ty(\zeta(c_1, c_2)) = C$ . Since  $\vdash \Delta$  by Lemma 14, we have  $\Delta \vdash \zeta(c_1, c_2) : ty(\zeta(c_1, c_2))$ . Since  $\Delta \vdash \iota \rightarrow ty(\zeta(c_1, c_2)) \sqsubseteq B \rightarrow A$  (recall that  $C = ty(\zeta(c_1, c_2))$ ), we have  $\Delta \vdash ty(\zeta(c_1, c_2)) \sqsubseteq A$  by Lemma 11. By (T\_INST), we have  $\Delta \vdash \zeta(c_1, c_2) : A$ .

Case (R\_BETA): We are given

- $M_1 = (\lambda x.M) v$ ,
- $M_2 = M[v/x]$ ,
- $\Delta \vdash (\lambda x.M) v : A$ ,
- $\Delta \vdash \lambda x.M : B \rightarrow A$ , and
- $\Delta \vdash v : B$

for some  $x$ ,  $M$ ,  $v$ , and  $B$ . By Lemma 15  $\Delta, \alpha^I, x : B' \vdash M : A'$  and  $\Delta \vdash \forall \alpha^I. B' \rightarrow A' \sqsubseteq B \rightarrow A$  for some  $\alpha^I$ ,  $A'$ , and  $B'$ . By Lemma 10, there exist  $\alpha_1^{I_1}$ ,  $\alpha_2^{I_2}$ ,  $\beta^J$ , and  $C^{I_1}$  such that

- $\{\alpha^I\} = \{\alpha_1^{I_1}\} \uplus \{\alpha_2^{I_2}\}$ ,
- $\Delta, \beta^J \vdash C^{I_1}$ ,
- $\Delta \vdash B \sqsubseteq \forall \beta^J. B' [C^{I_1} / \alpha_1^{I_1}]$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. A' [C^{I_1} / \alpha_1^{I_1}] \sqsubseteq A$ , and
- type variables in  $\beta^J$  do not appear free in  $A'$  and  $B'$ .

By Lemma 1,  $\Delta, \beta^J, \alpha^I, x : B' \vdash M : A'$  and  $\Delta, \beta^J, \alpha_2^{I_2} \vdash C^{I_1}$ . Thus, by Lemma 2 (4),

$$\Delta, \beta^J, \alpha_2^{I_2}, x : B' [C^{I_1} / \alpha_1^{I_1}] \vdash M : A' [C^{I_1} / \alpha_1^{I_1}]. \quad (3)$$

Because  $\Delta \vdash v : B$  and  $\Delta \vdash B \sqsubseteq \forall \beta^J. B' [C^{I_1} / \alpha_1^{I_1}]$  and  $\Delta, \alpha_2^{I_2} \vdash \forall \beta^J. B' [C^{I_1} / \alpha_1^{I_1}]$  (which can be easily shown with Lemma 14), we have

$$\Delta, \alpha_2^{I_2} \vdash v : \forall \beta^J. B' [C^{I_1} / \alpha_1^{I_1}]$$

by Lemma 1 and (T\_INST). Then, by Lemma 1 (4), (C\_INST), and (T\_INST), we have

$$\Delta, \beta^J, \alpha_2^{I_2} \vdash v : B' [C^{I_1} / \alpha_1^{I_1}].$$

By Lemma 4 (1) with (3),

$$\Delta, \beta^J, \alpha_2^{I_2} \vdash M[v/x] : A'[C^{I_1}/\alpha_1^{I_1}] .$$

By (T\_GEN) (with the permutation of the bindings in the typing context),

$$\Delta \vdash M[v/x] : \forall \alpha_2^{I_2}. \forall \beta^J. A'[C^{I_1}/\alpha_1^{I_1}] .$$

Since  $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. A'[C^{I_1}/\alpha_1^{I_1}] \sqsubseteq A$ , we have  $\Delta \vdash M[v/x] : A$  by (T\_INST).

Case (T\_GEN): By the IH and (T\_GEN).

Case (T\_INST): By the IH and (T\_INST).

Case (T\_HANDLE): We have two reduction rules which can be applied to `handle`-with expressions.

Case (R\_RETURN): We are given

- $M_1 = \text{handle } v \text{ with } H$ ,
- $H^{\text{return}} = \text{return } x \rightarrow M$ ,
- $M_2 = M[v/x]$ ,
- $\Delta \vdash \text{handle } v \text{ with } H : A$ ,
- $\Delta \vdash v : B$ ,
- $\Delta \vdash H : B \Rightarrow A$

for some  $v, H, x, M$ , and  $B$ . By inversion of the derivation of  $\Delta \vdash H : B \Rightarrow A$ , we have  $\Delta, x : B \vdash M : A$ .

By Lemma 4 (1),  $\Delta \vdash M[v/x] : A$ , which is the conclusion we have to show.

Case (R\_HANDLE): We are given

- $M_1 = \text{handle } E[\#op(v)] \text{ with } H$ ,
- $op \notin E$ ,
- $H(op) = op(x, k) \rightarrow M$ ,
- $M_2 = M[v/x][\lambda y. \text{handle } E[y] \text{ with } H/k]$ ,
- $\Delta \vdash \text{handle } E[\#op(v)] \text{ with } H : A$ ,
- $\Delta \vdash E[\#op(v)] : B$ ,
- $\Delta \vdash H : B \Rightarrow A$

for some  $E, op, v, H, x, y, k, M$ , and  $B$ . Suppose that  $ty(op) = \forall \alpha. C \hookrightarrow D$ . By inversion of the derivation of  $\Delta \vdash H : B \Rightarrow A$ , we have  $\Delta, \alpha, x : C, k : D \rightarrow A \vdash M : A$ .

By Lemma 23,  $\Delta, \beta^J \vdash C_0$  and  $\Delta, \beta^J \vdash v : C[C_0/\alpha]$  for some  $\beta^J$  and  $C_0$ . Since  $\Delta \vdash \forall \beta^J. C_0$ ,

$$\Delta, x : C[\forall \beta^J. C_0/\alpha], k : D[\forall \beta^J. C_0/\alpha] \rightarrow A \vdash M : A \quad (4)$$

by Lemma 2 (4) (note that type variables in  $\alpha$  do not appear free in  $A$ ).

Since  $\Delta, \beta^J \vdash v : C[C_0/\alpha]$ , we have  $\Delta \vdash v : \forall \beta^J. C[C_0/\alpha]$  by (T\_GEN). By Definition 5,  $\{\alpha\} \cap ftv(C)_{\text{ns}}^+ = \emptyset$ . Thus, we have  $\Delta \vdash v : C[\forall \beta^J. C_0/\alpha]$  by Lemma 28 (1) and (T\_INST) (note that  $\vdash \Delta$  by Lemma 14 and we can suppose that  $\beta^J$  do not appear free in  $C$ ). Thus, by applying Lemma 4 (1) to (4), we have

$$\Delta, k : D[\forall \beta^J. C_0/\alpha] \rightarrow A \vdash M[v/x] : A . \quad (5)$$

We show that

$$\Delta \vdash \lambda y. \text{handle } E[y] \text{ with } H : D[\forall \beta^J. C_0/\alpha] \rightarrow A .$$

By Definition 5,  $\{\alpha\} \cap ftv(D)^- = \emptyset$ . Thus, we have

$$\Delta \vdash D[\forall \beta^J. C_0/\alpha] \sqsubseteq \forall \beta^J. D[C_0/\alpha]$$

by Lemma 28 (2) (note that  $\vdash \Delta$  by Lemma 14 and we can suppose that  $\beta^J$  do not appear free in  $D$ ).

By Lemma 23,

$$\Delta, y : \forall \beta^J. D[C_0/\alpha] \vdash E[y] : B .$$

By Lemma 22,

$$\Delta, y : D[\forall \beta^J. C_0/\alpha] \vdash E[y] : B .$$

Thus, we have

$$\Delta, y : D[\forall \beta^J. C_0/\alpha] \vdash \text{handle } E[y] \text{ with } H : A$$

by Lemma 1 (5) and (T\_HANDLE). By (T\_ABS),

$$\Delta \vdash \lambda y. \text{handle } E[y] \text{ with } H : D[\forall \beta^J. C_0/\alpha] \rightarrow A.$$

By applying Lemma 4 (1) to (5), we have

$$\Delta \vdash M[v/x][\lambda y. \text{handle } E[y] \text{ with } H/k] : A,$$

which is what we have to show.

Case (T\_PROJ1): We have one reduction rule (R\_PROJ1) which can be applied to projection  $\pi_1$ . Thus, we are given

- $M_1 = \pi_1(v_1, v_2)$ ,
- $M_2 = v_1$ ,
- $\Delta \vdash \pi_1(v_1, v_2) : A$ ,
- $\Delta \vdash (v_1, v_2) : A \times B$

for some  $v_1, v_2$ , and  $B$ . By Lemma 16,  $\Delta, \alpha^I \vdash v_1 : C_1$  and  $\Delta, \alpha^I \vdash v_2 : C_2$  and  $\Delta \vdash \forall \alpha^I. C_1 \times C_2 \sqsubseteq A \times B$  for some  $\alpha^I, C_1$ , and  $C_2$ . By Lemma 24, there exist  $\alpha_1^{I_1}, \alpha_2^{I_2}, \beta^J$ , and  $D^{I_1}$  such that

- $\{\alpha^I\} = \{\alpha_1^{I_1}\} \uplus \{\alpha_2^{I_2}\}$ ,
- $\Delta, \beta^J \vdash D^{I_1}$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_1[D^{I_1}/\alpha_1^{I_1}] \sqsubseteq A$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_2[D^{I_1}/\alpha_1^{I_1}] \sqsubseteq B$ , and
- type variables in  $\beta^J$  do not appear in  $C_1$  and  $C_2$ .

We have to show that

$$\Delta \vdash v_1 : A.$$

Since  $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_1[D^{I_1}/\alpha_1^{I_1}] \sqsubseteq A$ , it suffices to show that

$$\Delta \vdash v_1 : \forall \alpha_2^{I_2}. \forall \beta^J. C_1[D^{I_1}/\alpha_1^{I_1}]$$

by (T\_INST). We have  $\Delta, \beta^J, \alpha^I \vdash v_1 : C_1$  by Lemma 1 (4). By Lemma 2 (4), we have  $\Delta, \beta^J, \alpha_2^{I_2} \vdash v_1 : C_1[D^{I_1}/\alpha_1^{I_1}]$ . By (T\_GEN) (and swapping  $\beta^J$  and  $\alpha_2^{I_2}$  in the typing context  $\Delta, \beta^J, \alpha_2^{I_2}$ ), we have

$$\Delta \vdash v_1 : \forall \alpha_2^{I_2}. \forall \beta^J. C_1[D^{I_1}/\alpha_1^{I_1}].$$

Case (T\_PROJ2): Similar to the case for (T\_PROJ1).

Case (T\_CASE): We have two reduction rules which can be applied to case expressions.

Case (R\_CASEL): We are given

- $M_1 = \text{case } (\text{inl } v) \text{ of } \text{inl } x \rightarrow M'_1; \text{inr } y \rightarrow M'_2$ ,
- $M_2 = M'_1[v/x]$ ,
- $\Delta \vdash \text{case } (\text{inl } v) \text{ of } \text{inl } x \rightarrow M'_1; \text{inr } y \rightarrow M'_2 : A$ ,
- $\Delta \vdash \text{inl } v : B_1 + B_2$ ,
- $\Delta, x : B_1 \vdash M'_1 : A$ , and
- $\Delta, x : B_2 \vdash M'_2 : A$

for some  $v, x, y, M'_1, M'_2, B_1$ , and  $B_2$ . By Lemma 17,  $\Delta, \alpha^I \vdash v : C_1$  and  $\Delta \vdash \forall \alpha^I. C_1 + C_2 \sqsubseteq B_1 + B_2$  for some  $\alpha^I, C_1$ , and  $C_2$ . By Lemma 25, there exist  $\alpha_1^{I_1}, \alpha_2^{I_2}, \beta^J$ , and  $D^{I_1}$  such that

- $\{\alpha^I\} = \{\alpha_1^{I_1}\} \uplus \{\alpha_2^{I_2}\}$ ,
- $\Delta, \beta^J \vdash D^{I_1}$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_1[D^{I_1}/\alpha_1^{I_1}] \sqsubseteq B_1$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_2[D^{I_1}/\alpha_1^{I_1}] \sqsubseteq B_2$ , and

- type variables in  $\beta^J$  do not appear in  $C_1$  and  $C_2$ .

We first show that

$$\Delta \vdash v : B_1 .$$

Since  $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C_1[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \sqsubseteq B_1$ , it suffices to show that

$$\Delta \vdash v : \forall \alpha_2^{I_2}. \forall \beta^J. C_1[\mathbf{D}^{I_1}/\alpha_1^{I_1}]$$

by (T\_INST). We have  $\Delta, \beta^J, \alpha^I \vdash v_1 : C_1$  by Lemma 1 (4). By Lemma 2 (4), we have  $\Delta, \beta^J, \alpha_2^{I_2} \vdash v_1 : C_1[\mathbf{D}^{I_1}/\alpha_1^{I_1}]$ . By (T\_GEN) (and swapping  $\beta^J$  and  $\alpha_2^{I_2}$  in the typing context  $\Delta, \beta^J, \alpha_2^{I_2}$ ), we have

$$\Delta \vdash v_1 : \forall \alpha_2^{I_2}. \forall \beta^J. C_1[\mathbf{D}^{I_1}/\alpha_1^{I_1}] .$$

Since  $\Delta, x : B_1 \vdash M'_1 : A$ , we have

$$\Delta \vdash M'_1[v/x] : A$$

by Lemma 4 (1).

Case (R\_CASER): Similar to the case for (R\_CASER), using Lemma 18 instead of Lemma 17.

Case (T\_CASELIST): We have two reduction rules which can be applied to case expressions for lists.

Case (R\_NIL): Obvious.

Case (R\_CONS): We are given

- $M_1 = \text{case}(\text{cons } v) \text{ of nil} \rightarrow M'_1; \text{cons } x \rightarrow M'_2$ ,
- $M_2 = M'_2[v/x]$ ,
- $\Delta \vdash \text{case}(\text{cons } v) \text{ of nil} \rightarrow M'_1; \text{cons } y \rightarrow M'_2 : A$ ,
- $\Delta \vdash \text{cons } v : B \text{ list}$ , and
- $\Delta, x : B \times B \text{ list} \vdash M'_2 : A$

for some  $v, x, M'_1, M'_2$ , and  $B$ . By Lemma 19,  $\Delta, \alpha^I \vdash v : C \times C \text{ list}$  and  $\Delta \vdash \forall \alpha^I. C \text{ list} \sqsubseteq B \text{ list}$  for some  $\alpha^I$  and  $C$ . By Lemma 26, there exist  $\alpha_1^{I_1}, \alpha_2^{I_2}, \beta^J$ , and  $\mathbf{D}^{I_1}$  such that

- $\{\alpha^I\} = \{\alpha_1^{I_1}\} \uplus \{\alpha_2^{I_2}\}$ ,
- $\Delta, \beta^J \vdash \mathbf{D}^{I_1}$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \sqsubseteq B$ , and
- type variables in  $\beta^J$  do not appear in  $C$ .

We first show that

$$\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \times C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \text{ list} \sqsubseteq B \times B \text{ list} .$$

Since  $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \sqsubseteq B$ , we have

$$\Delta \vdash (\forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}]) \text{ list} \sqsubseteq B \text{ list}$$

by (C\_LIST). We also have

$$\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \text{ list} \sqsubseteq (\forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}]) \text{ list}$$

by (C\_DLIST). Thus, by (C\_TRANS), we have

$$\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \text{ list} \sqsubseteq B \text{ list} .$$

By (C\_PROD),

$$\Delta \vdash (\forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}]) \times (\forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \text{ list}) \sqsubseteq B \times B \text{ list} .$$

By (C\_DPROD) and (C\_TRANS), we have

$$\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \times C[\mathbf{D}^{I_1}/\alpha_1^{I_1}] \text{ list} \sqsubseteq B \times B \text{ list} \quad (6)$$

Next, we show that

$$\Delta \vdash v : B \times B \text{ list} .$$

By (T\_INST) with (6), it suffices to show that

$$\Delta \vdash v : \forall \alpha_2^{I_2}. \forall \beta^J. C[D^{I_1}/\alpha_1^{I_1}] \times C[D^{I_1}/\alpha_1^{I_1}] \text{ list} .$$

We have  $\Delta, \beta^J, \alpha^I \vdash v : C \times C \text{ list}$  by Lemma 1 (4). By Lemma 2 (4), we have  $\Delta, \beta^J, \alpha_2^{I_2} \vdash v : C[D^{I_1}/\alpha_1^{I_1}] \times C[D^{I_1}/\alpha_1^{I_1}] \text{ list}$ . By (T\_GEN) (and swapping  $\beta^J$  and  $\alpha_2^{I_2}$  in the typing context  $\Delta, \beta^J, \alpha_2^{I_2}$ ), we have

$$\Delta \vdash v : \forall \alpha_2^{I_2}. \forall \beta^J. C[D^{I_1}/\alpha_1^{I_1}] \times C[D^{I_1}/\alpha_1^{I_1}] \text{ list} .$$

Since  $\Delta, x : B \times B \text{ list} \vdash M'_2 : A$ , we have

$$\Delta \vdash M'_2[v/x] : A$$

by Lemma 4 (1).

Case (T\_FIX): We have one reduction rule (R\_FIX) which can be applied to the fixed-point operator. The proof is straightforward with Lemma 4 (1) and (T\_ABS).

2. Suppose that  $\Delta \vdash M_1 : A$  and  $M_1 \longrightarrow M_2$ . By definition, there exist some  $E$ ,  $M'_1$ , and  $M'_2$  such that  $M_1 = E[M'_1]$ ,  $M_2 = E[M'_2]$ , and  $M'_1 \rightsquigarrow M'_2$ . The proof proceeds by induction on the typing derivation of  $M_1 = E[M'_1]$ . If  $E = []$ , then we have the conclusion by the first case. In what follows, we suppose that  $E \neq []$ . By case analysis on the typing rule applied last to derive  $\Delta \vdash E[M'_1] : A$ .

Case (T\_VAR), (T\_CONST), (T\_ABS), (T\_NIL), and (T\_FIX): Contradictory because  $E$  has to be  $[]$ .

Case (T\_APP): By case analysis on  $E$ .

Case  $E = E' M$ : We are given

- $\Delta \vdash E'[M'_1] : B \rightarrow A$  and
- $\Delta \vdash M : B$

for some  $B$ . By the IH,  $\Delta \vdash E'[M'_2] : B \rightarrow A$ . Since  $M_2 = E'[M'_2] M$ , we have the conclusion by (T\_APP).

Case  $E = v E'$ : By the IH.

Case (T\_GEN): By the IH.

Case (T\_INST): By the IH.

Case (T\_OP): By the IH.

Case (T\_HANDLE): By the IH.

Case (T\_PAIR): By the IH.

Case (T\_PROJ1): By the IH.

Case (T\_PROJ2): By the IH.

Case (T\_INL): By the IH.

Case (T\_INR): By the IH.

Case (T\_CASE): By the IH.

Case (T\_CONS): By the IH.

Case (T\_CASELIST): By the IH.

□

**Theorem 1** (Type Soundness). *Assume that all operations satisfy the signature restriction. If  $\Delta \vdash M : A$  and  $M \longrightarrow^* M'$  and  $M' \not\rightarrow$ , then:*

- $M'$  is a value; or
- $M' = E[\#op(v)]$  for some  $E$ ,  $op$ , and  $v$  such that  $op \notin E$ .

*Proof.* By Lemmas 29 and 13.

□



## 2.2 Soundness of the Type-and-Effect System

This section show soundness of the type-and-effect system. We may reuse the lemmas proven in Section 2.1 if their statements and proofs do not need change.

**Lemma 30** (Weakening). *Suppose that  $\vdash \Gamma_1, \Gamma_2$ . Let  $\Gamma_3$  be a typing context such that  $\text{dom}(\Gamma_2) \cap \text{dom}(\Gamma_3) = \emptyset$ .*

1. *If  $\vdash \Gamma_1, \Gamma_3$ , then  $\vdash \Gamma_1, \Gamma_2, \Gamma_3$ .*
2. *If  $\Gamma_1, \Gamma_3 \vdash A$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash A$ .*
3. *If  $\Gamma_1, \Gamma_3 \vdash A \sqsubseteq B$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash A \sqsubseteq B$ .*
4. *If  $\Gamma_1, \Gamma_3 \vdash M : A \mid \epsilon$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash M : A \mid \epsilon$ .*
5. *If  $\Gamma_1, \Gamma_3 \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon'$ , then  $\Gamma_1, \Gamma_2, \Gamma_3 \vdash H : A \mid \epsilon \Rightarrow B \mid \epsilon'$ .*

*Proof.* By mutual induction on the derivations of the judgments. □

**Lemma 31** (Type substitution). *Suppose that  $\Gamma_1 \vdash A$ .*

1. *If  $\vdash \Gamma_1, \alpha, \Gamma_2$ , then  $\vdash \Gamma_1, \Gamma_2 [A/\alpha]$ .*
2. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash B$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash B[A/\alpha]$ .*
3. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash B \sqsubseteq C$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash B[A/\alpha] \sqsubseteq C[A/\alpha]$ .*
4. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash M : B \mid \epsilon$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash M : B[A/\alpha] \mid \epsilon$ .*
5. *If  $\Gamma_1, \alpha, \Gamma_2 \vdash H : B \mid \epsilon \Rightarrow C \mid \epsilon'$ , then  $\Gamma_1, \Gamma_2 [A/\alpha] \vdash H : B[A/\alpha] \mid \epsilon \Rightarrow C[A/\alpha] \mid \epsilon'$ .*

*Proof.* Straightforward by mutual induction on the derivations of the judgments, as in Lemma 2. □

**Lemma 32** (Term substitution). *Suppose that  $\Gamma_1 \vdash M : A \mid \epsilon$  for any  $\epsilon$ .*

1. *If  $\Gamma_1, x : A, \Gamma_2 \vdash M' : B \mid \epsilon$ , then  $\Gamma_1, \Gamma_2 \vdash M'[M/x] : B \mid \epsilon$ .*
2. *If  $\Gamma_1, x : A, \Gamma_2 \vdash H : B \mid \epsilon \Rightarrow C \mid \epsilon'$ , then  $\Gamma_1, \Gamma_2 \vdash H[M/x] : B \mid \epsilon \Rightarrow C \mid \epsilon'$ .*

*Proof.* By mutual induction on the typing derivations as in Lemma 4. □

**Lemma 33** (Canonical forms). *Suppose that  $\Gamma \vdash v : A \mid \epsilon$ .*

1. *If  $\text{unqualify}(A) = \iota$ , then  $v = c$  for some  $c$ .*
2. *If  $\text{unqualify}(A) = B \rightarrow^{\epsilon'} C$ , then  $v = c$  for some  $c$ , or  $v = \lambda x. M$  for some  $x$  and  $M$ .*
3. *If  $\text{unqualify}(A) = B \times C$ , then  $v = (v_1, v_2)$  for some  $v_1$  and  $v_2$ .*
4. *If  $\text{unqualify}(A) = B + C$ , then  $v = \text{inl } v'$  or  $v = \text{inr } v'$  for some  $v'$ .*
5. *If  $\text{unqualify}(A) = B \text{ list}$ , then  $v = \text{nil}$  or  $v = \text{cons } v'$  for some  $v'$ .*

*Proof.* Similarly to Lemma 8. □

**Lemma 34** (Type containment inversion: polymorphic function types). *If  $\Gamma \vdash \forall \alpha_1^{I_1}. A_1 \rightarrow^{\epsilon_1} A_2 \sqsubseteq \forall \alpha_2^{I_2}. B_1 \rightarrow^{\epsilon_2} B_2$ , then  $\epsilon_1 = \epsilon_2$  and there exist  $\alpha_{11}^{I_{11}}, \alpha_{12}^{I_{12}}, \beta^J$ , and  $C^{I_{11}}$  such that*

- $\{\alpha_1^{I_1}\} = \{\alpha_{11}^{I_{11}}\} \uplus \{\alpha_{12}^{I_{12}}\}$ ,
- $\Gamma, \alpha_2^{I_2}, \beta^J \vdash C^{I_{11}}$ ,
- $\Gamma, \alpha_2^{I_2} \vdash B_1 \sqsubseteq \forall \beta^J. A_1[C^{I_{11}}/\alpha_{11}^{I_{11}}]$ ,
- $\Gamma, \alpha_2^{I_2} \vdash \forall \alpha_{12}^{I_{12}}. \forall \beta^J. A_2[C^{I_{11}}/\alpha_{11}^{I_{11}}] \sqsubseteq B_2$ ,

- type variables in  $\{\beta^J\}$  do not appear free in  $A_1$  and  $A_2$ , and
- if  $\alpha_{12}^{I_{12}}$  or  $\beta^J$  is not the empty sequence,  $SR(\epsilon_1)$ .

*Proof.* Similarly to Lemma 10. □

**Lemma 35.** *If  $\Gamma \vdash A_1 \rightarrow^{\epsilon_1} A_2 \sqsubseteq B_1 \rightarrow^{\epsilon_2} B_2$ , then  $\epsilon_1 = \epsilon_2$  and  $\Gamma \vdash B_1 \sqsubseteq A_1$  and  $\Gamma \vdash A_2 \sqsubseteq B_2$ .*

*Proof.* Similarly to Lemma 11 with Lemma 34. □

**Lemma 36** (Value inversion: constants). *If  $\Gamma \vdash c : A \mid \epsilon$ , then  $\Gamma \vdash ty(c) \sqsubseteq A$ .*

*Proof.* Similarly to Lemma 12. □

**Lemma 37** (Progress). *If  $\Delta \vdash M : A \mid \epsilon$ , then:*

- $M \longrightarrow M'$  for some  $M'$ ;
- $M$  is a value; or
- $M = E[\#op(v)]$  for some  $E$ ,  $op$ , and  $v$  such that  $op \notin E$  and  $op \in \epsilon$ .

*Proof.* Similarly to Lemma 13 with the lemmas proven in this section. The case for (TE-WEAK) is also straightforward. □

**Lemma 38** (Value inversion: lambda abstractions). *If  $\Gamma \vdash \lambda x.M : A \mid \epsilon$ , then  $\Gamma, \alpha, x : B \vdash M : C \mid \epsilon'$  and  $\Gamma \vdash \forall \alpha. B \rightarrow^{\epsilon'} C \sqsubseteq A$  for some  $\alpha$ ,  $B$ ,  $C$ , and  $\epsilon'$ .*

*Proof.* Similarly to Lemma 15. □

**Lemma 39** (Value inversion: pairs). *If  $\Gamma \vdash (M_1, M_2) : A \mid \epsilon$ , then  $\Gamma, \alpha \vdash M_1 : B_1 \mid \epsilon$  and  $\Gamma, \alpha \vdash M_2 : B_2 \mid \epsilon$  and  $\Gamma \vdash \forall \alpha. B_1 \times B_2 \sqsubseteq A$  for some  $\alpha$ ,  $B_1$ , and  $B_2$ .*

*Proof.* Similarly to Lemma 16. □

**Lemma 40** (Value inversion: left injections). *If  $\Gamma \vdash \text{inl } M : A \mid \epsilon$ , then  $\Gamma, \alpha \vdash M : B \mid \epsilon$  and  $\Gamma \vdash \forall \alpha. B + C \sqsubseteq A$  for some  $\alpha$ ,  $B$ , and  $C$ .*

*Proof.* Similarly to Lemma 17. □

**Lemma 41** (Value inversion: right injections). *If  $\Gamma \vdash \text{inr } M : A \mid \epsilon$ , then  $\Gamma, \alpha \vdash M : C \mid \epsilon$  and  $\Gamma \vdash \forall \alpha. B + C \sqsubseteq A$  for some  $\alpha$ ,  $B$ , and  $C$ .*

*Proof.* Similarly to the proof of Lemma 18. □

**Lemma 42** (Value inversion: cons). *If  $\Gamma \vdash \text{cons } M : A \mid \epsilon$ , then  $\Gamma, \alpha \vdash M : B \times B \text{ list} \mid \epsilon$  and  $\Gamma \vdash \forall \alpha. B \text{ list} \sqsubseteq A$  for some  $\alpha$  and  $B$ .*

*Proof.* Similarly to Lemma 19. □

**Lemma 43.** *If  $ty(\text{op}) = \forall \alpha^I. A \hookrightarrow B$  and  $\Gamma \vdash \#op(v) : C \mid \epsilon$ , then*

- $\Gamma, \beta^J \vdash D^I$ ,
- $\Gamma, \beta^J \vdash v : A[D^I/\alpha^I] \mid \epsilon'$ ,
- $\epsilon' \subseteq \epsilon$ ,
- $op \in \epsilon'$ , and
- $\Gamma \vdash \forall \beta^J. B[D^I/\alpha^I] \sqsubseteq C$ ; or

for some  $\beta^J$ ,  $D^I$ , and  $\epsilon'$ . Furthermore, if  $\beta^J$  is not the empty sequence,  $SR(\epsilon')$  holds.

*Proof.* By induction on the typing derivation. There are only five typing rules that can be applied to  $\#op(v)$ .

Case (TE\_GEN): Straightforward by the IH. Note that  $SR(\epsilon)$  by inversion.

Case (TE\_INST): Straightforward by the IH and (C\_TRANS).

Case (TE\_OP): Trivial.

Case (TE\_WEAK): By the IH.

□

**Lemma 44.** *If  $\Gamma, \alpha^I \vdash E[\#op(v)] : A \mid \epsilon$  and  $op \notin E$ , then*

- $\Gamma, \alpha^I, \beta^J \vdash \#op(v) : B \mid \epsilon'$  and
- $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E[y] : A \mid \epsilon$  for any  $y \notin \text{dom}(\Gamma)$ , and
- $op \in \epsilon$

for some  $\beta^J$ ,  $B$ , and  $\epsilon'$ . Furthermore, if  $\beta^J$  is not the empty sequence, then  $SR(\{op\})$  holds.

*Proof.* By induction on the typing derivation.

Case (TE\_VAR), (TE\_CONST), (TE\_ABS), (TE\_NIL), and (TE\_FIX): Contradictory.

Case (TE\_APP): By case analysis on  $E$ .

Case  $E = E' M_2$ : By inversion of the typing derivation, we have  $\Gamma, \alpha^I \vdash E'[\#op(v)] : C \rightarrow^{\epsilon''} A \mid \epsilon$  and  $\Gamma, \alpha^I \vdash M_2 : C \mid \epsilon$  and  $\epsilon'' \subseteq \epsilon$  for some  $C$  and  $\epsilon''$ . By the IH,

- $\Gamma, \alpha^I, \beta^J \vdash \#op(v) : B \mid \epsilon'$ ,
- $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E'[y] : C \rightarrow^{\epsilon''} A \mid \epsilon$  for any  $y \notin \text{dom}(\Gamma)$ , and
- $op \in \epsilon$ ,
- If  $\beta^J$  is not the empty sequence, then  $SR(\{op\})$  holds.

for some  $\beta^J$ ,  $B$ , and  $\epsilon'$ . By Lemma 30 (4) and (TE\_APP),  $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E'[y] M_2 : A \mid \epsilon$ , i.e.,  $\Gamma, y : \forall \alpha^I. \forall \beta^J. B, \alpha^I \vdash E[y] : A \mid \epsilon$ .

Case  $E = v_1 E'$ : Similarly to the above case.

Case (TE\_GEN): By the IH. We find  $SR(\{op\})$  by  $op \in \epsilon$  and  $SR(\epsilon)$ .

Case (TE\_INST): By the IH.

Case (TE\_OP): If  $E = []$ , the proof is straightforward by letting  $\beta^J$  be the empty sequence,  $B = A$ , and  $\epsilon' = \epsilon$ ;  $op \in \epsilon$  is found by Lemma 43.

Otherwise, the proof is similar to the case for (TE\_APP).

Case (TE\_HANDLE): By the IH. We find  $op \in \epsilon$  because the handler does not have an operation clause for  $op$  ( $op \notin E$ ).

Case (TE\_WEAK): By the IH.

Otherwise: Similarly to the case for (TE\_APP).

□

**Lemma 45.** *Suppose that  $\Gamma_1 \vdash A \sqsubseteq B$  and  $\Gamma_1 \vdash A$ .*

1. *If  $\Gamma_1, x : B, \Gamma_2 \vdash M : C \mid \epsilon$ , then  $\Gamma_1, x : A, \Gamma_2 \vdash M : C \mid \epsilon$ .*
2. *If  $\Gamma_1, x : B, \Gamma_2 \vdash H : C \mid \epsilon \Rightarrow D \mid \epsilon'$ , then  $\Gamma_1, x : A, \Gamma_2 \vdash H : C \mid \epsilon \Rightarrow D \mid \epsilon'$ .*

*Proof.* By mutual induction on the typing derivations.

□

**Lemma 46.** *If  $ty(op) = \forall \alpha^I. A \hookrightarrow B$  and  $\Gamma \vdash E[\#op(v)] : C \mid \epsilon$  and  $op \notin E$ , then*

- $\Gamma, \beta^J \vdash D^I$ ,
- $\Gamma, \beta^J \vdash v : A[D^I/\alpha^I] \mid \epsilon'$ , and
- for any  $y \notin \text{dom}(\Gamma)$ ,  $\Gamma, y : \forall \beta^J. B[D^I/\alpha^I] \vdash E[y] : C \mid \epsilon$

for some  $\beta^J$ ,  $D^I$ , and  $\epsilon'$ . Furthermore, if  $\beta^J$  is not the empty sequence,  $SR(\{\text{op}\})$  holds.

*Proof.* By Lemma 44,

- $\Gamma, \beta_1^{J_1} \vdash \# \text{op}(v) : C' \mid \epsilon''$  and
- $\Gamma, y : \forall \beta_1^{J_1}. C' \vdash E[y] : C \mid \epsilon$  for any  $y \notin \text{dom}(\Gamma)$ , and
- if  $\beta_1^{J_1}$  is not the empty sequence, then  $SR(\{\text{op}\})$  holds

for some  $\beta_1^{J_1}$  and  $C'$ . By Lemma 43,

- $\Gamma, \beta_1^{J_1}, \beta_2^{J_2} \vdash D^I$ ,
- $\Gamma, \beta_1^{J_1}, \beta_2^{J_2} \vdash v : A[D^I/\alpha^I] \mid \epsilon'$ ,
- $\Gamma, \beta_1^{J_1} \vdash \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq C'$ , and
- if  $\beta_2^{J_2}$  is not the empty sequence,  $SR(\{\text{op}\})$  holds

for some  $\beta_2^{J_2}$ ,  $D^I$ , and  $\epsilon'$ .

We show the conclusion by letting  $\beta^J = \beta_1^{J_1}, \beta_2^{J_2}$ . It suffices to show that, for any  $y \notin \text{dom}(\Gamma)$ ,

$$\Gamma, y : \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \vdash E[y] : C \mid \epsilon.$$

Since  $\Gamma, \beta_1^{J_1} \vdash \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq C'$ , we have

$$\Gamma \vdash \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \sqsubseteq \forall \beta_1^{J_1}. C'$$

by (C-POLY). Since  $\Gamma, y : \forall \beta_1^{J_1}. C' \vdash E[y] : C \mid \epsilon$ , we have

$$\Gamma, y : \forall \beta_1^{J_1}. \forall \beta_2^{J_2}. B[D^I/\alpha^I] \vdash E[y] : C \mid \epsilon.$$

by Lemma 45. □

**Lemma 47.** If  $\Gamma \vdash v : A \mid \epsilon$ , then  $\Gamma \vdash v : A \mid \epsilon'$  for any  $\epsilon'$ .

*Proof.* Straightforward by induction on the typing derivation. □

**Lemma 48.** Assume that  $\vdash \Gamma$  and  $\alpha \notin \text{ftv}(A)$ .

1. Suppose that (1)  $\beta \notin \text{ftv}(A)_{\text{ns}}^+$  and (2) for any function type  $C \rightarrow^\epsilon D$  occurring at a strictly positive position of  $A$ , if  $\beta \in \text{ftv}(D)$ , then  $SR(\epsilon)$ . Then  $\Gamma \vdash \forall \alpha. A[B/\beta] \sqsubseteq A[\forall \alpha. B/\beta]$ .
2. If  $\beta \notin \text{ftv}(A)^-$ , then  $\Gamma \vdash A[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. A[B/\beta]$ .

*Proof.* Case (2) can be proven similarly to Lemma 28 (2).

We show case (1) by induction on  $A$ . We consider the case that  $A = C \rightarrow^\epsilon D$  for some  $C$ ,  $D$ , and  $\epsilon$ ; the other cases are shown similarly to those in Lemma 28 (1). By case (2) with  $C$ , we have  $\Gamma \vdash C[\forall \alpha. B/\beta] \sqsubseteq \forall \alpha. C[B/\beta]$ .

Now, we show that

$$\Gamma \vdash \forall \alpha. (\forall \alpha. C[B/\beta]) \rightarrow^\epsilon D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow^\epsilon D[\forall \alpha. B/\beta]. \quad (7)$$

If  $\beta \in \text{ftv}(D)$ , then  $SR(\epsilon)$  by the assumption. By the IH on  $D$ ,  $\Gamma \vdash \forall \alpha. D[B/\beta] \sqsubseteq D[\forall \alpha. B/\beta]$ . By (C-FUNEFF),

$$\Gamma \vdash (\forall \alpha. C[B/\beta]) \rightarrow^\epsilon \forall \alpha. D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow^\epsilon D[\forall \alpha. B/\beta].$$

Since  $SR(\epsilon)$ , we have (7) by (C\_DFUNEFF) and (C\_TRANS). Otherwise, if  $\beta \notin \text{ftv}(D)$ , then  $\Gamma, \alpha \vdash D[B/\beta] \sqsubseteq D[\forall \alpha. B/\beta]$  by (C\_REFL) because  $D[B/\beta] = D[\forall \alpha. B/\beta] = D$ . Thus,

$$\Gamma \vdash \forall \alpha. (\forall \alpha. C[B/\beta]) \rightarrow^\epsilon D[B/\beta] \sqsubseteq \forall \alpha. C[\forall \alpha. B/\beta] \rightarrow^\epsilon D[\forall \alpha. B/\beta]$$

by (C\_POLY) and Lemma 30 (3). Since  $\alpha \notin \text{ftv}(A)$  and  $A = C \rightarrow^\epsilon D$ , we can have (7) by eliminating the outermost  $\forall$  on the RHS type with (C\_INST).

By (C\_INST),

$$\Gamma, \alpha \vdash \forall \alpha. C[B/\beta] \sqsubseteq C[B/\beta]. \quad (8)$$

By (C\_REFL), (C\_FUNEFF), and (C\_POLY) with (8),

$$\Gamma \vdash \forall \alpha. C[B/\beta] \rightarrow^\epsilon D[B/\beta] \sqsubseteq \forall \alpha. (\forall \alpha. C[B/\beta]) \rightarrow^\epsilon D[B/\beta].$$

Thus, by (C\_TRANS) with (7),

$$\Gamma \vdash \forall \alpha. C[B/\beta] \rightarrow^\epsilon D[B/\beta] \sqsubseteq C[\forall \alpha. B/\beta] \rightarrow^\epsilon D[\forall \alpha. B/\beta].$$

□

**Lemma 49** (Subject reduction).

1. If  $\Delta \vdash M_1 : A \mid \epsilon$  and  $M_1 \rightsquigarrow M_2$ , then  $\Delta \vdash M_2 : A \mid \epsilon$ .
2. If  $\Delta \vdash M_1 : A \mid \epsilon$  and  $M_1 \longrightarrow M_2$ , then  $\Delta \vdash M_2 : A \mid \epsilon$ .

*Proof.* 1. By induction on the typing derivation. Most of the cases are similar to Lemma 29. We here focus on the cases that need a treatment specific to the type-and-effect system.

Case (TE\_APP)/(R\_BETA): We are given

- $M_1 = (\lambda x. M) v$ ,
- $M_2 = M[v/x]$ ,
- $\Delta \vdash (\lambda x. M) v : A \mid \epsilon$ ,
- $\Delta \vdash \lambda x. M : B \rightarrow^{\epsilon_0} A \mid \epsilon$ ,
- $\Delta \vdash v : B \mid \epsilon$ , and
- $\epsilon_0 \subseteq \epsilon$

for some  $x, M, v, B$ , and  $\epsilon_0$ . By Lemma 38  $\Delta, \alpha^I, x : B' \vdash M : A' \mid \epsilon'$  and  $\Delta \vdash \forall \alpha^I. B' \rightarrow^{\epsilon'} A' \sqsubseteq B \rightarrow^{\epsilon_0} A$  for some  $\alpha^I, A', B'$ , and  $\epsilon'$ . By Lemma 34, we find  $\epsilon' = \epsilon_0$ , and there exist  $\alpha_1^{I_1}, \alpha_2^{I_2}, \beta^J$ , and  $C^{I_1}$  such that

- $\{\alpha^I\} = \{\alpha_1^{I_1}\} \uplus \{\alpha_2^{I_2}\}$ ,
- $\Delta, \beta^J \vdash C^{I_1}$ ,
- $\Delta \vdash B \sqsubseteq \forall \beta^J. B'[C^{I_1}/\alpha_1^{I_1}]$ ,
- $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. A'[C^{I_1}/\alpha_1^{I_1}] \sqsubseteq A$ , and
- type variables in  $\beta^J$  do not appear free in  $A'$  and  $B'$ , and
- If  $\alpha_2^{I_2}$  or  $\beta^J$  is not the empty sequence,  $SR(\epsilon_0)$ .

By Lemma 30,  $\Delta, \beta^J, \alpha^I, x : B' \vdash M : A' \mid \epsilon'$  and  $\Delta, \beta^J, \alpha_2^{I_2} \vdash C^{I_1}$ . Thus, by Lemma 31 (4),

$$\Delta, \beta^J, \alpha_2^{I_2}, x : B' [C^{I_1}/\alpha_1^{I_1}] \vdash M : A' [C^{I_1}/\alpha_1^{I_1}] \mid \epsilon' \quad (9)$$

Since  $\Delta \vdash v : B \mid \epsilon$  and  $\Delta \vdash B \sqsubseteq \forall \beta^J. B'[C^{I_1}/\alpha_1^{I_1}]$ , we have

$$\Delta \vdash v : \forall \beta^J. B'[C^{I_1}/\alpha_1^{I_1}] \mid \epsilon$$

by (TE\_INST) (note that  $\Delta \vdash \forall \beta^J. B'[C^{I_1}/\alpha_1^{I_1}]$  is shown easily with Lemma 14). By Lemma 30 (4), (C\_INST), and (TE\_INST), we have

$$\Delta, \beta^J, \alpha_2^{I_2} \vdash v : B' [C^{I_1}/\alpha_1^{I_1}] \mid \epsilon.$$

By Lemmas 47 and 32 (1) with (9),

$$\Delta, \beta^J, \alpha_2^{I_2} \vdash M[v/x] : A'[C^{I_1}/\alpha_1^{I_1}] \mid \epsilon'.$$

By (TE\_GEN) (with the permutation of the bindings in the typing context),

$$\Delta \vdash M[v/x] : \forall \alpha_2^{I_2}. \forall \beta^J. A'[C^{I_1}/\alpha_1^{I_1}] \mid \epsilon'$$

(note that If  $\alpha_2^{I_2}$  or  $\beta^J$  is not the empty sequence,  $SR(\epsilon')$ ). Since  $\Delta \vdash \forall \alpha_2^{I_2}. \forall \beta^J. A'[C^{I_1}/\alpha_1^{I_1}] \sqsubseteq A$ , we have  $\Delta \vdash M[v/x] : A \mid \epsilon'$  by (TE\_INST). Since  $\epsilon' \subseteq \epsilon$ , we have

$$\Delta \vdash M[v/x] : A \mid \epsilon$$

by (TE\_WEAK).

Case (TE\_GEN): By the IH and (TE\_GEN).

Case (TE\_HANDLE)/(R\_HANDLE): We are given

- $M_1 = \text{handle } E[\#op(v)] \text{ with } H$ ,
- $op \notin E$ ,
- $H(op) = op(x, k) \rightarrow M$ ,
- $M_2 = M[v/x][\lambda y. \text{handle } E[y] \text{ with } H/k]$ ,
- $\Delta \vdash \text{handle } E[\#op(v)] \text{ with } H : A \mid \epsilon$ ,
- $\Delta \vdash E[\#op(v)] : B \mid \epsilon'$ ,
- $\Delta \vdash H : B \mid \epsilon' \Rightarrow A \mid \epsilon$

for some  $E, op, v, H, x, y, k, M, B$ , and  $\epsilon'$ . Suppose that  $ty(op) = \forall \alpha. C \hookrightarrow D$ . By inversion of the derivation of  $\Delta \vdash H : B \mid \epsilon' \Rightarrow A \mid \epsilon$ , we have  $\Delta, \alpha, x : C, k : D \rightarrow^\epsilon A \vdash M : A \mid \epsilon$ .

By Lemma 46,

- $\Delta, \beta^J \vdash C_0$ ,
- $\Delta, \beta^J \vdash v : C[C_0/\alpha] \mid \epsilon_0$ ,
- $\Gamma, y : \forall \beta^J. D[C_0/\alpha] \vdash E[y] : B \mid \epsilon'$ , and
- if  $\beta^J$  is not the empty sequence,  $SR(\{op\})$

for some  $\beta^J, C_0$ , and  $\epsilon_0$ . Since  $\Delta \vdash \forall \beta^J. C_0$ ,

$$\Delta, x : C[\forall \beta^J. C_0/\alpha], k : D[\forall \beta^J. C_0/\alpha] \rightarrow^\epsilon A \vdash M : A \mid \epsilon \quad (10)$$

by Lemma 31 (4) (note that type variables in  $\alpha$  do not appear free in  $A$ ). Since  $\Delta, \beta^J \vdash v : C[C_0/\alpha] \mid \epsilon_0$ , we have  $\Delta \vdash v : \forall \beta^J. C[C_0/\alpha] \mid \epsilon_0$  by Lemma 47 and (TE\_GEN).

We show that  $\Delta \vdash v : C[\forall \beta^J. C_0/\alpha] \mid \epsilon_0$ . If  $\beta^J$  is not empty, then  $SR(\{op\})$ . Thus, we have the derivation by Lemma 48 (1) and (TE\_INST) (note that  $\vdash \Delta$  by Lemma 14 and we can suppose that  $\beta^J$  do not appear free in  $C$ ). Otherwise, if  $\beta^J$  is empty, we also have it.

By applying Lemmas 47 and 32 (1) to (10), we have

$$\Delta, k : D[\forall \beta^J. C_0/\alpha] \rightarrow^\epsilon A \vdash M[v/x] : A \mid \epsilon. \quad (11)$$

We show that

$$\Delta \vdash \lambda y. \text{handle } E[y] \text{ with } H : D[\forall \beta^J. C_0/\alpha] \rightarrow^\epsilon A \mid \epsilon''$$

for any  $\epsilon''$ .

For that, we first show that  $\Delta \vdash D[\forall \beta^J. C_0/\alpha] \sqsubseteq \forall \beta^J. D[C_0/\alpha]$ . If  $\beta^J$  is not empty, then  $SR(\{op\})$ . Thus, we have the derivation by Lemma 48 (2) (note that  $\vdash \Delta$  by Lemma 14 and we can suppose that  $\beta^J$  do not appear free in  $D$ ). Otherwise, if  $\beta^J$  is empty, we also have it by (C\_REFL).

Thus, since  $\Gamma, y : \forall \beta^J. D[C_0/\alpha] \vdash E[y] : B \mid \epsilon'$ , we have

$$\Delta, y : D[\forall \beta^J. C_0/\alpha] \vdash E[y] : B \mid \epsilon'$$

by Lemma 45. Thus, we have

$$\Delta, y : D[\forall \beta^J. C_0/\alpha] \vdash \text{handle } E[y] \text{ with } H : A \mid \epsilon$$

by Lemma 30 (5) and (TE\_HANDLE). By (TE\_ABS),

$$\Delta \vdash \lambda y. \text{handle } E[y] \text{ with } H : D[\forall \beta^J. C_0/\alpha] \rightarrow^\epsilon A \mid \epsilon''$$

for any  $\epsilon''$ .

By applying Lemma 32 (1) to (11), we have

$$\Delta \vdash M[v/x][\lambda y. \text{handle } E[y] \text{ with } H/k] : A \mid \epsilon,$$

which is what we have to show.

Case (TE\_FIX)/(R\_FIX): By Lemma 32. Note that the fixed-point operator can be given any effect.

2. Straightforward by induction on the typing derivation. □

**Theorem 2** (Type Soundness). *If  $\Delta \vdash M : A \mid \emptyset$  and  $M \longrightarrow^* M'$  and  $M' \not\rightarrow$ , then  $M'$  is a value.*

*Proof.* By Lemmas 49 and 37. □