Bidirectional Type Checking for Existential Types with Higher-Rank Polymorphism

Appendix

Hasti Toossi and Ningning Xie

University of Toronto, Toronto, Canada
`{hasti2c,ningningxie}@cs.toronto.edu`

Table of Contents

A The Declarative System 2
B Meta Theory of the Declarative System 4
  B.1 Typing 4
  B.2 Subtyping 8
    B.2.1 General Properties 8
    B.2.2 Reflexivity 10
    B.2.3 Transitivity 10
C The Algorithmic System 14
D Meta Theory of the Algorithmic System 19
  D.1 Algorithmic Context 19
    D.1.1 Context Extension 19
    D.1.2 Context Application 21
  D.2 Soundness 23
  D.3 Completeness of Subtyping 31
  D.4 Completeness of Typing with Restricted Existential Inference . . . 37
    D.4.1 Soundness 37
    D.4.2 Completenesss 41
  D.5 Completeness of Typing with Restricted Existential Instantiation 49
E Core Language 54
F The Elaborated Declarative System 57
G Meta Theory of the Elaborated Declarative System 60

A The Declarative System

universally quantified type	σ	$:= \epsilon \mid \forall a.\sigma$
existentially quantified type	ϵ	$:= \rho \mid \exists\, b.\epsilon$
top-level monomorphic type	ρ	$:= \tau \mid \sigma_1 \to \sigma_2$
monomorphic type	τ	$:= a \mid \mathsf{Int} \mid \tau_1 \to \tau_2 \mid \lfloor e : \exists\, a.\epsilon \rfloor$
expr	e	$:= n \mid x \mid \lambda x.\ e \mid \lambda x : \sigma.\ e \mid e_1\, e_2 \mid (e : \sigma)$
context	Γ	$:= \bullet \mid x : \sigma \mid a \mid \Gamma_1, \ldots, \Gamma_n$

$\boxed{\vdash \Gamma}$ *(Context Well-Formedness)*

d-cwf-Empty

$$\frac{}{\vdash \bullet}$$

d-cwf-Var

$$\frac{\Gamma \vdash \sigma \qquad x \notin \mathsf{dom}\,(\Gamma)}{\vdash \Gamma, x : \sigma}$$

d-cwf-TVar

$$\frac{\vdash \Gamma \qquad a \notin \mathsf{dom}\,(\Gamma)}{\vdash \Gamma, a}$$

$\boxed{\mathrm{G} \vdash \mathrm{t}}$ *(Type Well-Formedness)*

d-wf-Int

$$\frac{\vdash \Gamma}{\Gamma \vdash \mathsf{Int}}$$

d-wf-Var

$$\frac{\vdash \Gamma \qquad a \in \Gamma}{\Gamma \vdash a}$$

d-wf-Proj

$$\frac{\Gamma \vdash \exists\, a.\epsilon \qquad \mathsf{fv}\,(e) \subseteq \mathsf{dom}\,(\Gamma)}{\Gamma \vdash \lfloor e : \exists\, a.\epsilon \rfloor}$$

d-wf-Arr

$$\frac{\Gamma \vdash \sigma_1 \qquad \Gamma \vdash \sigma_2}{\Gamma \vdash \sigma_1 \to \sigma_2}$$

d-wf-Exists

$$\frac{\Gamma, a \vdash \epsilon}{\Gamma \vdash \exists\, a.\epsilon}$$

d-wf-Forall

$$\frac{\Gamma, a \vdash \sigma}{\Gamma \vdash \forall a.\sigma}$$

$\boxed{\Gamma \vdash e \Rightarrow \sigma}$ *(Inferenece)*

d-i-Int

$$\frac{\vdash \Gamma}{\Gamma \vdash n \Rightarrow \mathsf{Int}}$$

d-i-Var

$$\frac{\vdash \Gamma \qquad x : \sigma \in \Gamma}{\Gamma \vdash x \Rightarrow \sigma}$$

d-i-Abs

$$\frac{\Gamma, x : \tau \vdash e \Rightarrow \sigma \qquad \Gamma, x : \tau \vdash \sigma \rightsquigarrow_{\forall} \epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon}{\Gamma \vdash \lambda x.\ e \Rightarrow \tau \to \exists\, \overline{a}.\epsilon'}$$

d-i-AbsA

$$\frac{\begin{array}{c}\Gamma, x : \sigma_1 \vdash e \Rightarrow \sigma \\ \Gamma, x : \sigma_1 \vdash \sigma \rightsquigarrow_{\forall} \epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon \qquad \mathsf{ftv}\,(\sigma_1) = \varnothing\end{array}}{\Gamma \vdash \lambda x : \sigma_1.\ e \Rightarrow \sigma_1 \to \exists\, \overline{a}.\epsilon'}$$

d-i-App

$$\frac{\Gamma \vdash e \Rightarrow \sigma \qquad \Gamma \vdash e : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \qquad \Gamma \vdash e_1 \Leftarrow \sigma_1}{\Gamma \vdash e\, e_1 \Rightarrow \sigma_2}$$

d-i-Ann

$$\frac{\Gamma \vdash e \Leftarrow \sigma \qquad \Gamma \vdash \sigma \qquad \mathsf{ftv}\,(\sigma) = \varnothing}{\Gamma \vdash (e : \sigma) \Rightarrow \sigma}$$

$\boxed{\Gamma \vdash e \Leftarrow \sigma}$ *(Checking)*

d-c-Sub
$$\frac{\Gamma \vdash e \Rightarrow \sigma \qquad \Gamma \vdash e : \sigma <: \rho}{\Gamma \vdash e \Leftarrow \rho}$$

d-c-Abs
$$\frac{\Gamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2}{\Gamma \vdash \lambda x.\ e \Leftarrow \sigma_1 \to \sigma_2}$$

d-c-AbsA
$$\frac{\Gamma, x : \sigma_1 \vdash x : \sigma_1 <: \sigma_1' \qquad \Gamma, x : \sigma_1' \vdash e \Leftarrow \sigma_2 \qquad \mathsf{ftv}\,(\sigma_1') = \varnothing}{\Gamma \vdash \lambda x : \sigma_1'.\ e \Leftarrow \sigma_1 \to \sigma_2}$$

d-c-Exists
$$\frac{\Gamma \vdash e \Leftarrow [\tau/b]\epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash e \Leftarrow \exists\, b.\epsilon}$$

d-c-Forall
$$\frac{\Gamma, a \vdash e \Leftarrow \sigma}{\Gamma \vdash e \Leftarrow \forall a.\sigma}$$

$\boxed{\Gamma \vdash e : \sigma \rightsquigarrow \sigma_1 \to \sigma_2}$ *(Instantiation)*

d-inst-Refl
$$\frac{\vdash \Gamma}{\Gamma \vdash e : \sigma_1 \to \sigma_2 \rightsquigarrow \sigma_1 \to \sigma_2}$$

d-inst-Exists
$$\frac{\Gamma \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \rightsquigarrow \sigma_1 \to \sigma_2}{\Gamma \vdash e : \exists\, a.\epsilon \rightsquigarrow \sigma_1 \to \sigma_2}$$

d-inst-Forall
$$\frac{\Gamma \vdash e : [\tau/a]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \qquad \Gamma \vdash \tau}{\Gamma \vdash e : \forall a.\sigma \rightsquigarrow \sigma_1 \to \sigma_2}$$

$\boxed{\Gamma \vdash \sigma \rightsquigarrow_\forall \epsilon}$ *(Instantiation)*

d-instf-Refl
$$\frac{\vdash \Gamma}{\Gamma \vdash \epsilon \rightsquigarrow_\forall \epsilon}$$

d-instf-Forall
$$\frac{\Gamma \vdash [\tau/a]\sigma \rightsquigarrow_\forall \epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash \forall a.\sigma \rightsquigarrow_\forall \epsilon}$$

$\boxed{\Gamma \vdash e : \sigma_1 <: \sigma_2}$ *(Subtyping)*

d-s-Int
$$\frac{\vdash \Gamma}{\Gamma \vdash e : \mathsf{Int} <: \mathsf{Int}}$$

d-s-Var
$$\frac{\vdash \Gamma \qquad a \in \mathsf{dom}\,(\Gamma)}{\Gamma \vdash e : a <: a}$$

d-s-Proj
$$\frac{\Gamma \vdash \lfloor e : \exists\, a.\epsilon \rfloor}{\Gamma \vdash e' : \lfloor e : \exists\, a.\epsilon \rfloor <: \lfloor e : \exists\, a.\epsilon \rfloor}$$

d-s-Arr
$$\frac{\Gamma, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \qquad \Gamma, x : \sigma_1' \vdash e\, x : \sigma_2 <: \sigma_2'}{\Gamma \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2'}$$

d-s-ExistsL
$$\frac{\Gamma \vdash e : [\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: \epsilon_2}{\Gamma \vdash e : \exists\, a.\epsilon_1 <: \epsilon_2}$$

d-s-ExistsR
$$\frac{\Gamma \vdash e : \rho <: [\tau/a]\epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash e : \rho <: \exists\, a.\epsilon}$$

d-s-ForallL
$$\frac{\Gamma \vdash e : [\tau/a]\sigma <: \epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash e : \forall a.\sigma <: \epsilon}$$

d-s-ForallR
$$\frac{\Gamma, a \vdash e : \sigma_1 <: \sigma_2}{\Gamma \vdash e : \sigma_1 <: \forall a.\sigma_2}$$

Definition A.1. $\lfloor \epsilon \rfloor_x \coloneqq \{\lfloor e : \exists\, a.\epsilon' \rfloor \mid (\lfloor e : \exists\, a.\epsilon' \rfloor$ *is a sub-expression of* $\epsilon) \wedge$ $(x$ *is a free variable of* $e)\}$

B Meta Theory of the Declarative System

Lemma/Theorem	*Referenced Paper*	*Related Lemma/Thm.*
Lemma B.1 (Type Substitution)	DK	C.13
Lemma B.2 (Substitution)	DK	C.3
Theorem B.3 (Checking Subsumes Inference)	EDWL	B.3
Theorem B.4 (Subtyping Subsumes Instantiation)	–	–
Theorem B.5 (Order of Quantification Does Not Matter)	EDWL	B.4
Lemma B.7 (Occurrence)	DK	A.8
Lemma B.8 (Monotype Equality)	DK	A.9
Lemma B.9 (Reflexivity of Subtyping)	DK	A.2
Lemma B.10 (Transitivity of Subtyping (Limited))	–	–
Lemma B.11 (Subtyping Subsumes Simple Subtyping)	–	–
Lemma B.12 (Transitivity of Simple Subtyping)	DK	A.6

B.1 Typing

Lemma B.1 (Type Substitution). *Assume* $\Gamma_1 \vdash \tau$.

a) If $\Gamma_1, a, \Gamma_2 \vdash \sigma$, *then* $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]\sigma$.
b) If $\Gamma_1, a, \Gamma_2 \vdash e \Rightarrow \sigma$, *then* $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e \Rightarrow [\tau/a]\sigma$.
c) If $\Gamma_1, a, \Gamma_2 \vdash e \Leftarrow \sigma$, *then* $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e \Leftarrow [\tau/a]\sigma$.
d) If $\Gamma_1, a, \Gamma_2 \vdash e : \sigma \rightsquigarrow \rho$, *then* $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : [\tau/a]\sigma \rightsquigarrow [\tau/a]\rho$.
e) If $\Gamma_1, a, \Gamma_2 \vdash e : \sigma_1 <: \sigma_2$, *then* $\Gamma_1, [\tau/a]\Gamma_2 \vdash e : [\tau/a]\sigma_1 <: [\tau/a]\sigma_2$.

Proof. By induction on the given derivation.

- Rules D-WF-INT, D-I-INT, D-INST-REFL, D-INSTF-REFL, D-S-INT, and D-S-PROJ: Straightforward.
- Rules D-WF-PROJ, D-WF-ARR, D-I-APP, D-I-ANN, D-C-SUB, D-C-ABS, D-C-ABSA, D-INST-FORALL, D-INSTF-FORALL, D-C-FORALL, and D-S-FORALLR: Apply the induction hypothesis.
- Rule D-WF-VAR:

$$\frac{\vdash \Gamma_1, a, \Gamma_2 \qquad b \in \Gamma_1, a, \Gamma_2}{\Gamma_1, a, \Gamma_2 \vdash b}$$

  If $b = a$, then $[\tau/a]b = \tau$. We know $\Gamma_1 \vdash \tau$ by assumption, therefore $\Gamma_1, [\tau/a]\Gamma_2 \vdash \tau$.
  If $b \neq a$, then $[\tau/a]b = b$ since $\mathsf{fv}\,(b) \subseteq \Gamma_1$. Then $b \in \Gamma_1, [\tau/a]\Gamma_2$, therefore $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]b$ by rule D-WF-VAR.
- Rule D-WF-EXISTS:

$$\frac{\Gamma_1, a, \Gamma_2, b \vdash \epsilon}{\Gamma_1, a, \Gamma_2 \vdash \exists\, b.\epsilon}$$

$\Gamma_1, [\tau/a]\Gamma_2, [\tau/a]b \vdash [\tau/a]\epsilon$	i.h.
$\Gamma_1, [\tau/a]\Gamma_2, b \vdash [\tau/a]\epsilon$	$b \neq a$
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a](\exists\, b.\epsilon)$	Rule D-WF-EXISTS

- Rule d-wf-Forall: Similar to rule d-wf-Exists.
- Rule d-i-Var:

$$\frac{\vdash \Gamma_1, a, \Gamma_2 \qquad x : \sigma \in \Gamma_1, a, \Gamma_2}{\Gamma_1, a, \Gamma_2 \vdash x \Rightarrow \sigma}$$

  Since $x : \sigma \in \Gamma_1, a, \Gamma_2$, we have either $x : \sigma \in \Gamma_1$ or $x : \sigma \in \Gamma_2$. Note that $[\tau/a]x = x$.
  If $x{:}\sigma \in \Gamma_1$, then $[\tau/a]\sigma = \sigma$ since $\mathsf{fv}\,(\sigma) \subseteq \Gamma_1$. Then $x{:}[\tau/a]\sigma \in \Gamma_1, [\tau/a]\Gamma_2$, therefore $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]x \Rightarrow [\tau/a]\sigma$ by rule d-i-Var.
  If $x : \sigma \in \Gamma_2$, then $x : [\tau/a]\sigma \in \Gamma_1, [\tau/a]\Gamma_2$ by the definition of substitution, therefore $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]x \Rightarrow [\tau/a]\sigma$ by rule d-i-Var.
- Rule d-i-Abs:

$$\frac{\begin{array}{c}\Gamma_1, a, \Gamma_2, x : \tau' \vdash e \Rightarrow \sigma \\ \Gamma_1, a, \Gamma_2, x : \tau' \vdash \sigma \leadsto_\forall \epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon\end{array}}{\Gamma_1, a, \Gamma_2 \vdash \lambda x.\ e \Rightarrow \tau' \to \exists\, \overline{a}.\epsilon'}$$

  Note that τ does not refer to x since $\Gamma_1 \vdash \tau$.

$\Gamma_1, [\tau/a]\Gamma_2, [\tau/a]x : \tau' \vdash [\tau/a]e \Rightarrow [\tau/a]\sigma$	i.h.
$\Gamma_1, [\tau/a]\Gamma_2, [\tau/a]x : \tau' \vdash [\tau/a]\sigma \leadsto_\forall [\tau/a]\epsilon$	i.h.
$[\overline{a}/\lfloor [\tau/a]\epsilon \rfloor_x][\tau/a]\epsilon = [\tau/a][\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon = [\tau/a]\epsilon'$	dist. of subst.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a](\lambda x.\ e) \Rightarrow [\tau/a](\tau' \to \exists\, \overline{a}.\epsilon')$	Rule d-i-Abs

- Rule d-i-AbsA: Similar to rule d-i-Abs.
- Rule d-c-Exists:

$$\frac{\Gamma_1, a, \Gamma_2 \vdash e \Leftarrow [\tau'/b]\epsilon \qquad \Gamma_1, a, \Gamma_2 \vdash \tau'}{\Gamma_1, a, \Gamma_2 \vdash e \Leftarrow \exists\, b.\epsilon}$$

$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e \Leftarrow [\tau/a][\tau'/b]\epsilon$	i.h.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e \Leftarrow [[\tau/a]\tau'/b][\tau/a]\epsilon$	dist. of subst.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]\tau'$	i.h.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e \Leftarrow [\tau/a]\exists\, b.\epsilon$	Rule d-c-Exists

- Rule d-inst-Exists:

$$\frac{\Gamma_1, a, \Gamma_2 \vdash e : [\lfloor e : \exists\, b.\epsilon \rfloor/b]\epsilon \leadsto \rho}{\Gamma_1, a, \Gamma_2 \vdash e : \exists\, b.\epsilon \leadsto \rho}$$

$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : [\tau/a][\lfloor e : \exists\, b.\epsilon \rfloor/b]\epsilon \leadsto [\tau/a]\rho$	i.h.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : [\lfloor e : [\tau/a]\exists\, b.\epsilon \rfloor/b][\tau/a]\epsilon \leadsto [\tau/a]\rho$	dist. of subst.
$\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : [\tau/a]\exists\, b.\epsilon \leadsto [\tau/a]\rho$	Rule d-inst-Exists

- Rule d-s-Var:

$$\frac{b \in \Gamma_1, a, \Gamma_2}{\Gamma_1, a, \Gamma_2 \vdash e : b <: b}$$

  If $a = b$, then we need to show $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : \tau <: \tau$ which is true by Lemma B.9.
  If $a \neq b$, then $b \in \Gamma_1, [\tau/a]\Gamma_2$, so $\Gamma_1, [\tau/a]\Gamma_2 \vdash [\tau/a]e : b <: b$ by rule d-s-Var.

- Rule D-S-ARR:

$$\frac{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \qquad \mathit{\Gamma}_1, a, \mathit{\Gamma}_2, x : \sigma_1' \vdash e\,x : \sigma_2 <: \sigma_2'}{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2'}$$

$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2, [\tau/a]x : \sigma_1' \vdash [\tau/a]x : [\tau/a]\sigma_1' <: [\tau/a]\sigma_1$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2, [\tau/a]x : \sigma_1' \vdash [\tau/a](e\,x) : [\tau/a]\sigma_2 <: [\tau/a]\sigma_2'$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a](\sigma_1 \to \sigma_2) <: [\tau/a](\sigma_1' \to \sigma_2')$ Rule D-S-ARR

- Rule D-S-FORALLL:

$$\frac{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : [\tau'/b]\sigma <: \epsilon \qquad \mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash \tau'}{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : \forall b.\sigma <: \epsilon}$$

$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a][\tau'/b]\sigma <: [\tau/a]\epsilon$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [[\tau/a]\tau'/b][\tau/a]\sigma <: [\tau/a]\epsilon$ dist. of subst.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]\tau'$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a]\forall b.\sigma <: [\tau/a]\epsilon$ Rule D-S-FORALLL

- Rule D-S-EXISTSL:

$$\frac{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : [\lfloor e : \exists\, b.\epsilon_1 \rfloor/b]\epsilon_1 <: \epsilon_2}{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : \exists\, b.\epsilon_1 <: \epsilon_2}$$

$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a][\lfloor e : \exists\, b.\epsilon_1 \rfloor/b]\epsilon_1 <: [\tau/a]\epsilon_2$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\lfloor e : [\tau/a]\exists\, b.\epsilon_1 \rfloor/b][\tau/a]\epsilon_1 <: [\tau/a]\epsilon_2$ dist. of subst.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a]\exists\, b.\epsilon_1 <: [\tau/a]\epsilon_2$ Rule D-S-EXISTSL

- Rule D-S-EXISTSR:

$$\frac{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : \rho <: [\tau'/b]\epsilon \qquad \mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash \tau'}{\mathit{\Gamma}_1, a, \mathit{\Gamma}_2 \vdash e : \rho <: \exists\, b.\epsilon}$$

$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a]\rho <: [\tau/a][\tau'/b]\epsilon$ i.h.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a]\rho <: [[\tau/a]\tau'/b][\tau/a]\epsilon$ dist. of subst.
$\mathit{\Gamma}_1, [\tau/a]\mathit{\Gamma}_2 \vdash [\tau/a]e : [\tau/a]\rho <: [\tau/a](\exists\, b.\epsilon)$ Rule D-S-EXISTSR

□

Lemma B.2 (Substitution)**.** *Assume* $\mathit{\Gamma}_1 \vdash e' \Rightarrow \sigma'$.

a) If $\mathit{\Gamma}_1, x : \sigma', \mathit{\Gamma}_2 \vdash e \Rightarrow \sigma$, *then* $\mathit{\Gamma}_1, [e'/x]\mathit{\Gamma}_2 \vdash [e'/x]e \Rightarrow [e'/x]\sigma$.
b) If $\mathit{\Gamma}_1, x : \sigma', \mathit{\Gamma}_2 \vdash e \Leftarrow \sigma$, *then* $\mathit{\Gamma}_1, [e'/x]\mathit{\Gamma}_2 \vdash [e'/x]e \Leftarrow [e'/x]\sigma$.
c) If $\mathit{\Gamma}_1, x : \sigma', \mathit{\Gamma}_2 \vdash e : \sigma \rightsquigarrow \rho$, *then* $\mathit{\Gamma}_1, [e'/x]\mathit{\Gamma}_2 \vdash [e'/x]e : [e'/x]\sigma \rightsquigarrow [e'/x]\rho$.
d) If $\mathit{\Gamma}_1, x{:}\sigma', \mathit{\Gamma}_2 \vdash e : \sigma_1 <: \sigma_2$, *then* $\mathit{\Gamma}_1, [e'/x]\mathit{\Gamma}_2 \vdash [e'/x]e : [e'/x]\sigma_1 <: [e'/x]\sigma_2$.

Proof. Similar to the proof of Lemma B.1. □

Theorem B.3 (Checking Subsumes Inference). *If $\Gamma \vdash e \Rightarrow \rho$, then $\Gamma \vdash e \Leftarrow \rho$.*

Proof. We are given $\Gamma \vdash e \Rightarrow \rho$. By Lemma B.9, $\Gamma \vdash e : \rho <: \rho$, hence $\Gamma \vdash e \Leftarrow \rho$ by rule D-C-SUB. □

Theorem B.4 (Subtyping Subsumes Instantiation). *If $\Gamma \vdash e : \sigma \rightsquigarrow \rho$ or $\Gamma \vdash \sigma \rightsquigarrow_{\forall} \epsilon$, then $\Gamma \vdash e : \sigma <: \rho$.*

Proof. By induction on the given derivation.

- Rule D-INST-REFL:

$$\frac{\vdash \Gamma}{\Gamma \vdash e : \rho \rightsquigarrow \rho}$$

$\Gamma \vdash e : \rho <: \rho$ Lemma B.9

- Rule D-INST-FORALL:

$$\frac{\Gamma \vdash \tau \qquad \Gamma \vdash e : [\tau/a]\sigma \rightsquigarrow \rho}{\Gamma \vdash e : \forall a.\sigma \rightsquigarrow \rho}$$

$\Gamma \vdash e : [\tau/a]\sigma <: \rho$ i.h.
$\Gamma \vdash e : \forall a.\sigma <: \rho$ Rule D-S-FORALLL

- Rule D-INST-EXISTS:

$$\frac{\Gamma \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \rightsquigarrow \rho}{\Gamma \vdash e : \exists\, a.\epsilon \rightsquigarrow \rho}$$

$\Gamma \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon <: \rho$ i.h.
$\Gamma \vdash e : \exists\, a.\epsilon <: \rho$ Rule D-S-EXISTSL

- Rules D-INSTF-REFL and D-INSTF-FORALL: Similar to D-INST-.

□

Theorem B.5 (Order of Quantification Does Not Matter). *Let σ' be a type that differs from σ only by the ordering of quantified type variables. Then:*

a) $\Gamma \vdash e \Rightarrow \sigma$ *if and only if* $\Gamma \vdash e \Rightarrow \sigma'$
b) $\Gamma \vdash e \Leftarrow \sigma$ *if and only if* $\Gamma \vdash e \Leftarrow \sigma'$

Proof. We only need to consider rules that pack universal or existential quantifiers.

- Rule D-I-ABS: rule D-I-ABS packs multiple existentials at once, so the order does not matter.
- Rule D-C-FORALL:

$$\cfrac{\cfrac{\Gamma, a, b \vdash e \Leftarrow \sigma}{\Gamma, a \vdash e \Leftarrow \forall b.\sigma}}{\Gamma \vdash e \Leftarrow \forall a.\forall b.\sigma}$$

$$\begin{array}{ll}
\Gamma, a, b \vdash e \Leftarrow \sigma & \text{premise} \\
\vdash \Gamma, a, b & \text{perm. of context} \\
\vdash \Gamma, b, a & \text{defn.} \\
\Gamma, b, a \vdash e \Leftarrow \sigma & \text{perm. of context} \\
\Gamma, b \vdash e \Leftarrow \forall a.\sigma & \text{Rule } \textsc{d-c-Forall} \\
\Gamma \vdash e \Leftarrow \forall b.\forall a.\sigma & \text{Rule } \textsc{d-c-Forall}
\end{array}$$

– Rule d-c-Exists:

$$\frac{\dfrac{\Gamma \vdash e \Leftarrow [\tau_2/b][\tau_1/a]\epsilon \qquad \Gamma \vdash \tau_2}{\Gamma \vdash e \Leftarrow [\tau_1/a]\exists\, b.\epsilon} \qquad \Gamma \vdash \tau_1}{\Gamma \vdash e \Leftarrow \exists\, a.\exists\, b.\epsilon}$$

We know that $a \notin \Gamma$ and $b \in \Gamma$. Since $\Gamma \vdash \tau_1$ and $\Gamma \vdash \tau_2$, this implies $b \notin \mathsf{fv}\,(\tau_1)$ and $a \notin \mathsf{fv}\,(\tau_2)$.

$$\begin{array}{ll}
\Gamma \vdash e \Leftarrow [\tau_2/b][\tau_1/a]\epsilon & \text{premise} \\
\Gamma \vdash e \Leftarrow [\tau_1/a][\tau_2/b]\epsilon & b \notin \mathsf{fv}\,(\tau_1) \text{ and } a \notin \mathsf{fv}\,(\tau_2) \\
\Gamma \vdash e \Leftarrow [\tau_2/b]\exists\, a.\epsilon & \text{Rule } \textsc{d-c-Exists} \\
\Gamma \vdash e \Leftarrow \exists\, b.\exists\, a.\epsilon & \text{Rule } \textsc{d-c-Exists}
\end{array}$$

□

B.2 Subtyping

B.2.1 General Properties

Definition B.6 (Subterm Occurrence)**.** *Let* $\sigma_1 \preceq \sigma_2$ *iff* σ_1 *is a subterm of* σ_2*.*
Let $\sigma_1 \prec \sigma_2$ *iff* σ_1 *is a proper subterm of* σ_2*.*
Let $\sigma_1 \overset{\rightarrow}{\prec} \sigma_2$ *iff* σ_1 *occurs in* σ_2 *inside an arrow.*

Lemma B.7 (Occurrence)**.** *a) If* $\Gamma \vdash e : \sigma <: \tau$ *then* $\tau \overset{\rightarrow}{\not\prec} \sigma$*.*
b) If $\Gamma \vdash e : \tau <: \sigma$ *then* $\tau \overset{\rightarrow}{\not\prec} \sigma$*.*

Proof. By induction on the given derivation.

a) We know $\Gamma \vdash e : \sigma <: \tau$.
  - Rules d-s-Int, d-s-Var, and d-s-Proj: A type cannot be a proper subterm of itself.
  - Rule d-s-Arr:

$$\frac{\Gamma, x : \tau_1 \vdash x : \tau_1 <: \sigma_1 \qquad \Gamma, x : \tau_1 \vdash e\, x : \sigma_2 <: \tau_2}{\Gamma \vdash e : \sigma_1 \to \sigma_2 <: \tau_1 \to \tau_2}$$

$$\begin{array}{ll}
\tau_1 \overset{\rightarrow}{\not\prec} \sigma_1 & \text{i.h.} \\
\tau_2 \overset{\rightarrow}{\not\prec} \sigma_2 & \text{i.h.}
\end{array}$$

  Then $\tau_1 \to \tau_2 \not\prec \sigma_1$, since otherwise $\tau_1 \overset{\rightarrow}{\not\prec} \sigma_1$. Similarly, $\tau_1 \to \tau_2 \not\prec \sigma_2$, since otherwise $\tau_2 \overset{\rightarrow}{\not\prec} \sigma_2$. Then $\tau_1 \to \tau_2 \not\prec \sigma_1 \to \sigma_2$, which implies $\tau_1 \to \tau_2 \overset{\rightarrow}{\not\prec} \sigma_1 \to \sigma_2$.
  - Rule d-s-ArrA: Similar to rule d-s-Arr.

- Rule d-s-ForallL:

$$\frac{\mathit{\Gamma} \vdash e : [\tau'/a]\sigma <: \tau \qquad \mathit{\Gamma} \vdash \tau'}{\mathit{\Gamma} \vdash e : \forall a.\sigma <: \tau}$$

$\tau \overrightarrow{\not\preceq} [\tau'/a]\sigma$ i.h.
$\tau \overrightarrow{\not\preceq} \forall a.\sigma$ defn. of $\overrightarrow{\preceq}$

- Rule d-s-ExistsL:

$$\frac{\mathit{\Gamma} \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon <: \tau}{\mathit{\Gamma} \vdash e : \exists\, a.\epsilon <: \tau}$$

$\tau \overrightarrow{\not\preceq} [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon$ i.h.
$\tau \overrightarrow{\not\preceq} \exists\, a.\epsilon$ defn. of $\overrightarrow{\preceq}$

b) We know $\mathit{\Gamma} \vdash e : \tau <: \sigma$.

- Rules d-s-Int, d-s-Var, d-s-Proj, d-s-Arr, and d-s-Arr: Similar to the previous cases.
- Rule d-s-ForallR:

$$\frac{\mathit{\Gamma}, a \vdash e : \tau <: \sigma}{\mathit{\Gamma} \vdash e : \tau <: \forall a.\sigma}$$

$\tau \overrightarrow{\not\preceq} \sigma$ i.h.
$\tau \overrightarrow{\not\preceq} \forall a.\sigma$ defn. of $\overrightarrow{\preceq}$

- Rule d-s-ExistsL:

$$\frac{\mathit{\Gamma} \vdash e : \tau <: [\tau'/a]\epsilon \qquad \mathit{\Gamma} \vdash \tau'}{\mathit{\Gamma} \vdash e : \tau <: \exists\, a.\epsilon}$$

$\tau \overrightarrow{\not\preceq} [\tau'/a]\epsilon$ i.h.
$\tau \overrightarrow{\not\preceq} \exists\, a.\epsilon$ defn. of $\overrightarrow{\preceq}$

□

Lemma B.8 (Monotype Equality). *If* $\mathit{\Gamma} \vdash e : \tau <: \tau'$, *then* $\tau = \tau'$.

Proof. By induction on the given derivation.

Rules d-s-Int, d-s-Var, and d-s-Proj: Trivial.

- Rules d-s-ExistsL, d-s-ExistsR, d-s-ForallL, and d-s-ForallR: Not possible.
- Rule d-s-Arr:

$$\frac{\mathit{\Gamma}, x : \tau_1' \vdash x : \tau_1' <: \tau_1 \qquad \mathit{\Gamma}, x : \tau_1' \vdash e\, x : \tau_2 <: \tau_2'}{\mathit{\Gamma} \vdash e : \tau_1 \to \tau_2 <: \tau_1' \to \tau_2'}$$

$\tau_1 = \tau_1'$ i.h.
$\tau_2 = \tau_2'$ i.h.
$\tau_1 \to \tau_2 = \tau_1' \to \tau_2'$ above

□

B.2.2 Reflexivity

Lemma B.9 (Reflexivity of Subtyping)**.** *If* $\Gamma \vdash \sigma$ *and* $\mathsf{fv}\,(e) \subseteq \mathsf{dom}\,(\Gamma)$*, then* $\Gamma \vdash e : \sigma <: \sigma$*.*

Proof. By induction on the structure of σ.

- Case $\sigma = \mathsf{Int}$: Apply Rule d-s-Int.
- Case $\sigma = a$: By inversion on rule d-wf-Var, $a \in \Gamma$. Apply Rule d-s-Var.
- Case $\sigma = \lfloor e : \exists\, a.\epsilon \rfloor$: Apply Rule d-s-Proj.
- Case $\sigma = \sigma_1 \to \sigma_2$:
  $\Gamma, x : \sigma_1 \vdash x : \sigma_1 <: \sigma_1$ i.h.
  $\Gamma, x : \sigma_1 \vdash e\,x : \sigma_2 <: \sigma_2$ i.h.
  $\Gamma \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1 \to \sigma_2$ Rule d-s-Arr
- Case $\sigma = \forall a.\sigma$: We will prove the claim in the case where $\sigma = \forall a.\epsilon$. The general proof is similar.
  $\Gamma, a \vdash e : [a/a]\epsilon <: \epsilon$ i.h.
  $\Gamma, a \vdash a$ Rule d-wf-Var
  $\Gamma, a \vdash e : \forall a.\epsilon <: \epsilon$ Rule d-s-ForallL
  $\Gamma \vdash e : \forall a.\epsilon <: \forall a.\epsilon$ Rule d-s-ForallR
- Case $\sigma = \exists\, a.\epsilon$: We will prove the claim in the case where $\sigma = \exists\, a.\rho$. The general proof is similar.
  $\Gamma \vdash e : [\lfloor e : \exists\, a.\rho \rfloor / a]\rho <: [\lfloor e : \exists\, a.\rho \rfloor / a]\rho$ i.h.
  $\Gamma \vdash \lfloor e : \exists\, a.\rho \rfloor$ given
  $\Gamma \vdash e : [\lfloor e : \exists\, a.\rho \rfloor / a]\rho <: \exists\, a.\rho$ Rule d-s-ExistsR
  $\Gamma \vdash e : \exists\, a.\rho <: \exists\, a.\rho$ Rule d-s-ExistsL

□

B.2.3 Transitivity

Lemma B.10 (Transitivity of Subtyping (Limited))**.** *If* $\Gamma \vdash e_1 : \sigma_1 <: \sigma_2$ *and* $\Gamma \vdash e_2 : \sigma_2 <: \sigma_3$ *such that* $\mathsf{fv}\,(e_1) \subseteq \mathsf{dom}\,(\Gamma)$*,* $\mathsf{fv}\,(e_2) \subseteq \mathsf{dom}\,(\Gamma)$*, and* σ_2 *does not have any* $\exists$*s (of any order), then* $\Gamma \vdash e_1 : \sigma_1 <: \sigma_3$*.*

Proof. By induction on the given derivations with the metric $\langle \#\forall(\sigma_2), \# \to (\sigma_2), \mathcal{D}_1 + \mathcal{D}_2 \rangle$ as defined in Dunfield and Krishnaswami [1].

- Cases $\mathcal{D}_1 = \mathcal{D}_2 =$ rule d-s-Int, $\mathcal{D}_1 = \mathcal{D}_2 =$ rule d-s-Var, and $\mathcal{D}_1 = \mathcal{D}_2 =$ rule d-s-Proj: Apply the same rule.
- Cases $\mathcal{D}_1 = \mathcal{D}_2 =$ rule d-s-Arr:

$$\mathcal{D}_1 = \frac{\Gamma, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \qquad \Gamma, x : \sigma_1' \vdash e\,x : \sigma_2 <: \sigma_2'}{\Gamma \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2'}$$

,

$$\mathcal{D}_2 = \frac{\Gamma, y : \sigma_1'' \vdash y : \sigma_1'' <: \sigma_1' \qquad \Gamma, y : \sigma_1'' \vdash e\,y : \sigma_2' <: \sigma_2''}{\Gamma \vdash e : \sigma_1' \to \sigma_2' <: \sigma_1'' \to \sigma_2''}$$

$$\begin{array}{lr}
\Gamma, y : \sigma_1'' \vdash y : \sigma_1'' <: \sigma_1' & \text{premise} \\
\Gamma, x : \sigma_1' \vdash x : \sigma_1'' <: \sigma_1' & \text{name and bound type of } y \text{ does not matter} \\
\Gamma, x : \sigma_1' \vdash x : \sigma_1'' <: \sigma_1 & \text{induction hypothesis} \\
\Gamma, y : \sigma_1'' \vdash e\, y : \sigma_2' <: \sigma_2'' & \text{premise} \\
\Gamma, x : \sigma_1' \vdash e\, x : \sigma_2' <: \sigma_2'' & \text{name and bound type of } y \text{ does not matter} \\
\Gamma, x : \sigma_1' \vdash e\, x : \sigma_2 <: \sigma_2'' & \text{induction hypothesis} \\
\Gamma \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1'' \to \sigma_2'' & \text{Rule d-s-Arr}
\end{array}$$

- Case $\mathcal{D}_1$ = rule d-s-ForallL and $\mathcal{D}_2 \neq$ rule d-s-ForallR:

$$\mathcal{D}_1 = \frac{\Gamma \vdash e_1 : [\tau/a]\sigma_1 <: \epsilon_2 \qquad \Gamma \vdash \tau}{\Gamma \vdash e : \forall a.\sigma_1 <: \epsilon_2}$$

$$\begin{array}{lr}
\Gamma \vdash e_2 : \epsilon_2 <: \epsilon_3 & \text{result of } \mathcal{D}_2 \\
\Gamma \vdash e_1 : [\tau/a]\sigma_1 <: \epsilon_3 & \text{i.h.} \\
\Gamma \vdash e_1 : \forall a.\sigma_1 <: \epsilon_3 & \text{Rule d-s-ForallL}
\end{array}$$

- Case $\mathcal{D}_1$ = rule d-s-ExistsL and $\mathcal{D}_2 \neq$ rule d-s-ForallR:

$$\mathcal{D}_1 = \frac{\Gamma \vdash e_1 : [\lfloor e_1 : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_1 <: \epsilon_2}{\Gamma \vdash e_1 : \exists\, a.\epsilon_1 <: \epsilon_2}$$

$$\begin{array}{lr}
\Gamma \vdash e_2 : \epsilon_2 <: \epsilon_3 & \text{result of } \mathcal{D}_2 \\
\Gamma \vdash e_1 : [\lfloor e_1 : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_1 <: \epsilon_3 & \text{i.h.} \\
\Gamma \vdash e_1 : \exists\, a.\epsilon_1 <: \epsilon_3 & \text{Rule d-s-ExistsL}
\end{array}$$

- Case $\mathcal{D}_2$ = rule d-s-ForallR:

$$\mathcal{D}_2 = \frac{\Gamma, a \vdash e_2 : \sigma_2 <: \sigma_3}{\Gamma \vdash e_2 : \sigma_2 <: \forall a.\sigma_3}$$

$$\begin{array}{lr}
\Gamma \vdash e_1 : \sigma_1 <: \sigma_2 & \text{result of } \mathcal{D}_1 \\
\Gamma, a \vdash e_1 : \sigma_1 <: \sigma_2 & \text{weakening} \\
\Gamma, a \vdash e_1 : \sigma_1 <: \sigma_3 & \text{i.h.} \\
\Gamma \vdash e_1 : \sigma_1 <: \forall a.\sigma_3 & \text{Rule d-s-ForallR}
\end{array}$$

- Case $\mathcal{D}_2$ = rule d-s-ExistsR and $\mathcal{D}_1 \neq$ rule d-s-ForallL or rule d-s-ForallR:

$$\mathcal{D}_2 = \frac{\Gamma \vdash e_2 : \rho_2 <: [\tau/a]\epsilon_3 \qquad \Gamma \vdash \tau}{\Gamma \vdash e_2 : \rho_2 <: \exists\, a.\epsilon_3}$$

$$\begin{array}{lr}
\Gamma \vdash e_1 : \rho_1 <: \rho_2 & \text{result of } \mathcal{D}_1 \\
\Gamma \vdash e_2 : \rho_2 <: [\tau/a]\epsilon_3 & \text{premise of } \mathcal{D}_2 \\
\Gamma \vdash e_1 : \rho_1 <: [\tau/a]\epsilon_3 & \text{i.h.} \\
\Gamma \vdash e_1 : \rho_1 <: \exists\, a.\epsilon_3 & \text{Rule d-s-ExistsR}
\end{array}$$

- Case $\mathcal{D}_1$ = rule d-s-ForallR and $\mathcal{D}_2$ = rule d-s-ForallL:

$$\mathcal{D}_1 = \frac{\Gamma, a \vdash e_1 : \sigma_1 <: \sigma_2}{\Gamma \vdash e_1 : \sigma_1 <: \forall a.\sigma_2}$$

,

$$\mathcal{D}_2 = \frac{\Gamma \vdash e_2 : [\tau/a]\sigma_2 <: \epsilon_3 \qquad \Gamma \vdash \tau}{\Gamma \vdash e_2 : \forall a.\sigma_2 <: \epsilon_3}$$

$\Gamma \vdash e_1 : [\tau/a]\sigma_1 <: [\tau/a]\sigma_2$ Lemma B.1
$\Gamma \vdash e_1 : \sigma_1 <: [\tau/a]\sigma_2$ weakening
$\Gamma \vdash e_1 : \sigma_1 <: \epsilon_3$ i.h. since $\#\forall([\tau/a]\sigma_2) < \#\forall(\forall a.\sigma_2)$

- Case $\mathcal{D}_1$ = rule d-s-ExistsR and $\mathcal{D}_2$ = rule d-s-ExistsL: Since we do not allow $\exists$s of any order in σ_2, this case is impossible.

□

$\boxed{\Gamma \vdash \sigma_1 \leq \sigma_2}$ *(Simple Subtyping)*

$$\textsc{d-ss-Int}\ \frac{}{\Gamma \vdash \mathsf{Int} \leq \mathsf{Int}} \qquad \textsc{d-ss-Var}\ \frac{a \in \mathsf{dom}(\Gamma)}{\Gamma \vdash a \leq a} \qquad \textsc{d-ss-Proj}\ \frac{\Gamma \vdash \lfloor e : \exists\, a.\epsilon \rfloor}{\Gamma \vdash \lfloor e : \exists\, a.\epsilon \rfloor \leq \lfloor e : \exists\, a.\epsilon \rfloor}$$

$$\textsc{d-ss-Abs}\ \frac{\Gamma \vdash \sigma_1' \leq \sigma_1 \qquad \Gamma \vdash \sigma_2 \leq \sigma_2'}{\Gamma \vdash \sigma_1 \to \sigma_2 \leq \sigma_1' \to \sigma_2'} \qquad \textsc{d-ss-ExistsL}\ \frac{\Gamma, a \vdash \epsilon_1 \leq \epsilon_2}{\Gamma \vdash \exists\, a.\epsilon_1 \leq \epsilon_2}$$

$$\textsc{d-ss-ExistsR}\ \frac{\Gamma \vdash \rho \leq [\tau/a]\epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash \rho \leq \exists\, a.\epsilon} \qquad \textsc{d-ss-ForallL}\ \frac{\Gamma \vdash [\tau/a]\sigma \leq \epsilon \qquad \Gamma \vdash \tau}{\Gamma \vdash \forall a.\sigma \leq \epsilon} \qquad \textsc{d-ss-ForallR}\ \frac{\Gamma, a \vdash \sigma_1 \leq \sigma_2}{\Gamma \vdash \sigma_1 \leq \forall a.\sigma_2}$$

Lemma B.11 (Subtyping Subsumes Simple Subtyping). *If $\Gamma \vdash \sigma_1 \leq \sigma_2$ and $\mathsf{fv}(e) \subseteq \mathsf{dom}(\Gamma)$, then $\Gamma \vdash e : \sigma_1 <: \sigma_2$.*

Proof. By induction on the given derivation.

- Rule d-ss-ExistsL:

$$\frac{\Gamma, a \vdash \epsilon_1 \leq \epsilon_2}{\Gamma \vdash \exists\, a.\epsilon_1 \leq \epsilon_2}$$

$\Gamma \vdash [\lfloor e : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_1 \leq [\lfloor e : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_2$ Lemma B.1
$\Gamma \vdash [\lfloor e : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_1 \leq \epsilon_2$ well-formedness
$\Gamma \vdash e : [\lfloor e : \exists\, a.\epsilon_1 \rfloor/a]\epsilon_1 <: \epsilon_2$ i.h.
$\Gamma \vdash e : \exists\, a.\epsilon_1 <: \epsilon_2$ Rule d-ss-ExistsL

The proof for all other cases is straightforward, applying the induction hypothesis if necessary. □

Lemma B.12 (Transitivity of Simple Subtyping). *If $\Gamma \vdash \sigma_1 \leq \sigma_2$ and $\Gamma \vdash \sigma_2 \leq \sigma_3$, then $\Gamma \vdash \sigma_1 \leq \sigma_3$.*

Proof. By induction on the given derivations with the metric $\langle \#\forall(\sigma_2), |\exists(\sigma_2)|, \# \to (\sigma_2), \mathcal{D}_1 + \mathcal{D}_2 \rangle$ as defined in Dunfield and Krishnaswami [1].

- Case $\mathcal{D}_1$ = rule D-SS-EXISTSL and $\mathcal{D}_2 \neq$ rule D-SS-FORALLR:

$$\frac{\Gamma, a \vdash \epsilon_1 \le \epsilon_2}{\Gamma \vdash \exists\, a.\epsilon_1 \le \epsilon_2}$$

$\Gamma \vdash \epsilon_2 \le \epsilon_3$	result of $\mathcal{D}_2$
$\Gamma, a \vdash \epsilon_2 \le \epsilon_3$	weakening
$\Gamma, a \vdash \epsilon_1 \le \epsilon_3$	i.h.
$\Gamma \vdash \exists\, a.\epsilon_1 \le \epsilon_3$	Rule D-SS-EXISTSL

- Case $\mathcal{D}_1$ = rule D-SS-EXISTSR and $\mathcal{D}_2$ = rule D-SS-EXISTSL:

$$\frac{\Gamma \vdash \rho_1 \le [\tau/a]\epsilon_2 \qquad \Gamma \vdash \tau}{\Gamma \vdash \rho_1 \le \exists\, a.\epsilon_2}$$

,

$$\frac{\Gamma, a \vdash \epsilon_2 \le \rho_3}{\Gamma \vdash \exists\, a.\epsilon_2 \le \rho_3}$$

$\Gamma, a \vdash \rho_1 \le [\tau/a]\epsilon_2$	weakening
$\Gamma \vdash [\tau/a]\epsilon_2 \le [\tau/a]\epsilon_3$	Lemma B.1
$\Gamma \vdash \rho_1 \le [\tau/a]\rho_3$	i.h. since $\lvert\exists(\epsilon_2)\rvert < \lvert\exists(\exists\, a.\epsilon_2)\rvert$
$\Gamma \vdash \rho_1 \le \rho_3$	Rule D-SS-EXISTSR

The other cases are similar to Lemma B.10. □

C The Algorithmic System

universally quantified type	σ	$:= \epsilon \mid \forall a.\sigma$
existentially quantified type	ϵ	$:= \rho \mid \exists\, b.\epsilon$
top-level monomorphic type	ρ	$:= \tau \mid \sigma_1 \to \sigma_2$
monomorphic type	τ	$:= a \mid \hat{a} \mid \mathsf{Int} \mid \tau_1 \to \tau_2 \mid \lfloor e : \exists\, a.\epsilon \rfloor$
expr	e	$:= n \mid x \mid \lambda x.\ e \mid \lambda x : \sigma.\ e \mid e_1\, e_2 \mid (e : \sigma)$
context	Δ	$:= \bullet \mid x : \sigma \mid a \mid \hat{a} \mid \hat{a} = \tau \mid \Delta_1, \ldots, \Delta_n$
complete context	Ω	$:= \bullet \mid x : \sigma \mid a \mid \hat{a} = \tau \mid \Omega_1, \ldots, \Omega_n$

$\boxed{\vdash \Delta}$ *(Algorithmic Context Well-Formedness)*

a-cwf-Empty
$$\frac{}{\vdash \bullet}$$

a-cwf-Var
$$\frac{\Delta \vdash \sigma \qquad x \notin \mathsf{dom}\,(\Delta)}{\vdash \Delta, x : \sigma}$$

a-cwf-TVar
$$\frac{\vdash \Delta \qquad a \notin \mathsf{dom}\,(\Delta)}{\vdash \Delta, a}$$

a-cwf-UVar
$$\frac{\vdash \Delta \qquad \hat{a} \notin \mathsf{dom}\,(\Delta)}{\vdash \Delta, \hat{a}}$$

a-cwf-UVarSolved
$$\frac{\vdash \Delta \qquad \Delta \vdash \tau \qquad \hat{a} \notin \mathsf{dom}\,(\Delta)}{\vdash \Delta, \hat{a} = \tau}$$

a-cwf-Marker
$$\frac{\vdash \Delta \qquad \blacktriangleright_{\hat{a}} \notin \mathsf{dom}\,(\Delta) \qquad \hat{a} \notin \mathsf{dom}\,(\Delta)}{\vdash \Delta, \blacktriangleright_{\hat{a}}}$$

$\boxed{\Delta \vdash \sigma}$ *(Algorithmic Type Well-Formedness)*

a-wf-Int
$$\frac{\vdash \Delta}{\Delta \vdash \mathsf{Int}}$$

a-wf-Var
$$\frac{\vdash \Delta \qquad a \in \Delta}{\Delta \vdash a}$$

a-wf-UVar
$$\frac{\vdash \Delta \qquad \hat{a} \in \Delta}{\Delta \vdash \hat{a}}$$

a-wf-UVarSolved
$$\frac{\vdash \Delta \qquad \hat{a} = \tau \in \Delta}{\Delta \vdash \hat{a}}$$

a-wf-Proj
$$\frac{\Delta \vdash \exists\, a.\epsilon \qquad \mathsf{fv}\,(e) \subseteq \mathsf{dom}\,(\Delta)}{\Delta \vdash \lfloor e : \exists\, a.\epsilon \rfloor}$$

a-wf-Arr
$$\frac{\vdash \Delta \qquad \Delta \vdash \sigma_1 \qquad \Delta \vdash \sigma_2}{\Delta \vdash \sigma_1 \to \sigma_2}$$

a-wf-Exists
$$\frac{\Delta, a \vdash \epsilon}{\Delta \vdash \exists\, a.\epsilon}$$

a-wf-Forall
$$\frac{\Delta, a \vdash \sigma}{\Delta \vdash \forall a.\sigma}$$

$\boxed{\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2}$ *(Algorithmic Inference)*

a-i-Int
$$\frac{\vdash \Delta}{\Delta \vdash n \Rightarrow \mathsf{Int} \dashv \Delta}$$

a-i-Var
$$\frac{\vdash \Delta \qquad x : \sigma \in \Delta}{\Delta \vdash x \Rightarrow \sigma \dashv \Delta}$$

a-i-Abs
$$\frac{\begin{array}{c}\Delta_1, \hat{a}, x : \hat{a} \vdash e \Rightarrow \sigma \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2]\sigma \rightsquigarrow_{\forall} \epsilon \dashv \Delta_3, x : \hat{a}, \Delta_4 \\ \epsilon_1 = [\Delta_3, x : \hat{a}, \Delta_4]\epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1\end{array}}{\Delta_1 \vdash \lambda x.\ e \Rightarrow \hat{a} \to \exists\, \overline{a}.\epsilon_2 \dashv \Delta_3, \langle \Delta_4 \rangle}$$

$$\textsc{a-i-AbsA}\quad \frac{\Delta_1, x : \sigma_1 \vdash e \Rightarrow \sigma \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2]\sigma \leadsto_{\forall} \epsilon \dashv \Delta_3, x : \sigma_1, \Delta_4 \\ \epsilon_1 = [\Delta_3, x : \sigma_1, \Delta_4]\epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1 \qquad \mathsf{ftv}\,(\sigma_1) = \varnothing}{\Delta_1 \vdash \lambda x : \sigma_1.\ e \Rightarrow \sigma_1 \to \exists\, \overline{a}.\epsilon_2 \dashv \Delta_3, \langle \Delta_4 \rangle}$$

$$\textsc{a-i-App}\quad \frac{\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2 \qquad \Delta_2 \vdash e : [\Delta_2]\sigma \leadsto \sigma_1 \to \sigma_2 \dashv \Delta_3 \qquad \Delta_3 \vdash e_1 \Leftarrow \sigma_1 \dashv \Delta_4}{\Delta_1 \vdash e\, e_1 \Rightarrow \sigma_2 \dashv \Delta_4}$$

$$\textsc{a-i-Ann}\quad \frac{\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2 \qquad \Delta_1 \vdash \sigma \qquad \mathsf{ftv}\,(\sigma) = \varnothing}{\Delta_1 \vdash (e : \sigma) \Rightarrow \sigma \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2}$ *(Algorithmic Checking)*

$$\textsc{a-c-Sub}\quad \frac{\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2 \qquad \Delta_2 \vdash e : [\Delta_2]\sigma <: [\Delta_2]\rho \dashv \Delta_3}{\Delta_1 \vdash e \Leftarrow \rho \dashv \Delta_3}$$

$$\textsc{a-c-Abs}\quad \frac{\Delta_1, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \dashv \Delta_2, x : \sigma_1, \Delta_3}{\Delta_1 \vdash \lambda x.\ e \Leftarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}$$

$$\textsc{a-c-AbsA}\quad \frac{\Delta_1, x : \sigma_1 \vdash x : \sigma_1 <: \sigma_1' \dashv \Delta_2 \\ \Delta_2 \vdash e \Leftarrow [\Delta_2]\sigma_2 \dashv \Delta_3, x : \sigma_1', \Delta_4 \qquad \Delta_1 \vdash \sigma_1' \qquad \mathsf{ftv}\,(\sigma_1') = \varnothing}{\Delta_1 \vdash \lambda x : \sigma_1'.\ e \Leftarrow \sigma_1 \to \sigma_2 \dashv \Delta_3}$$

$$\textsc{a-c-AbsUVar}\quad \frac{\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1 \vdash e \Leftarrow \hat{a}_2 \dashv \Delta_2, x : \hat{a}_1, \Delta_3}{\Delta_1[\hat{a}] \vdash \lambda x.\ e \Leftarrow \hat{a} \dashv \Delta_2}$$

$$\textsc{a-c-AbsAUVar}\quad \frac{\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1 \vdash x : \hat{a}_1 <: \sigma_1 \dashv \Delta_2 \\ \Delta_2 \vdash e \Leftarrow [\Delta_2]\hat{a}_2 \dashv \Delta_3, x : \hat{a}_1, \Delta_4 \qquad \Delta_1 \vdash \sigma_1 \qquad \mathsf{ftv}\,(\sigma_1) = \varnothing}{\Delta_1 \vdash \lambda x : \sigma_1.\ e \Leftarrow \hat{a} \dashv \Delta_3}$$

$$\textsc{a-c-Exists}\quad \frac{\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \vdash e \Leftarrow [\hat{b}/b]\epsilon \dashv \Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3}{\Delta_1 \vdash e \Leftarrow \exists\, b.\epsilon \dashv \Delta_2} \qquad \textsc{a-c-Forall}\quad \frac{\Delta_1, a \vdash e \Leftarrow \sigma \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash e \Leftarrow \forall a.\sigma \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash e : \sigma \leadsto \sigma_1 \to \sigma_2 \dashv \Delta_2}$ *(Algorithmic Instantiation)*

$$\textsc{a-inst-Refl}\quad \frac{}{\Delta \vdash e : \sigma_1 \to \sigma_2 \leadsto \sigma_1 \to \sigma_2 \dashv \Delta}$$

$$\textsc{a-inst-UVar}\quad \frac{}{\Delta[\hat{a}] \vdash e : \hat{a} \leadsto \hat{a}_1 \to \hat{a}_2 \dashv \Delta[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2]}$$

a-inst-Exists
$$\frac{\Delta_1 \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}{\Delta_1 \vdash e : \exists\, a.\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}$$

a-inst-Forall
$$\frac{\Delta_1, \hat{a} \vdash e : [\hat{a}/a]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}{\Delta_1 \vdash e : \forall a.\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash \sigma \rightsquigarrow_{\forall} \epsilon \dashv \Delta_2}$ *(Algorithmic Instantiation)*

a-instf-Refl
$$\frac{}{\Delta \vdash \epsilon \rightsquigarrow_{\forall} \epsilon \dashv \Delta}$$

a-instf-Forall
$$\frac{\Delta_1, \hat{a} \vdash [\hat{a}/a]\sigma \rightsquigarrow_{\forall} \rho \dashv \Delta_2}{\Delta_1 \vdash \forall a.\sigma \rightsquigarrow_{\forall} \rho \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash e : \sigma_1 <: \sigma_2 \dashv \Delta_2}$ *(Subtyping)*

a-s-Mono
$$\frac{\Delta_1 \vdash \tau_1 \approx \tau_2 \dashv \Delta_2}{\Delta_1 \vdash e : \tau_1 <: \tau_2 \dashv \Delta_2}$$

a-s-Arr
$$\frac{\Delta_1, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \dashv \Delta_2 \qquad \Delta_2 \vdash e\,x : [\Delta_2]\sigma_2 <: [\Delta_2]\sigma_2' \dashv \Delta_3, x : \sigma_1', \Delta_4}{\Delta_1 \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2' \dashv \Delta_3}$$

a-s-ArrL
$$\frac{\begin{array}{c}\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \sigma_1 \vdash x : \sigma_1 <: \hat{a}_1 \dashv \Delta_2 \\ \Delta_2 \vdash e\,x : \hat{a}_2 <: [\Delta_2]\sigma_2 \dashv \Delta_3, x : \sigma_1, \Delta_4\end{array}}{\Delta_1[\hat{a}] \vdash e : \hat{a} <: \sigma_1 \to \sigma_2 \dashv \Delta_3}$$

a-s-ArrR
$$\frac{\begin{array}{c}\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1 \vdash x : \hat{a}_1 <: \sigma_1 \dashv \Delta_2 \\ \Delta_2 \vdash e\,x : [\Delta_2]\sigma_2 <: \hat{a}_2 \dashv \Delta_3, x : \hat{a}_1, \Delta_4\end{array}}{\Delta_1[\hat{a}] \vdash e : \sigma_1 \to \sigma_2 <: \hat{a} \dashv \Delta_3}$$

a-s-ExistsL
$$\frac{\Delta_1 \vdash e : [\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: \epsilon_2 \dashv \Delta_2}{\Delta_1 \vdash e : \exists\, a.\epsilon_1 <: \epsilon_2 \dashv \Delta_2}$$

a-s-ExistsR
$$\frac{\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : \rho <: [\hat{a}/a]\epsilon \dashv \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3}{\Delta_1 \vdash e : \rho <: \exists\, a.\epsilon \dashv \Delta_2}$$

a-s-ForallL
$$\frac{\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\hat{a}/a]\sigma <: \epsilon \dashv \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3}{\Delta_1 \vdash e : \forall a.\sigma <: \epsilon \dashv \Delta_2}$$

a-s-ForallR
$$\frac{\Delta_1, a \vdash e : \sigma_1 <: \sigma_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash e : \sigma_1 <: \forall a.\sigma_2 \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash \sigma_1 \approx \sigma_2 \dashv \Delta_2}$ *(Type Unification)*

a-ut-Refl
$$\frac{\vdash \Delta}{\Delta \vdash \sigma \approx \sigma \dashv \Delta}$$

a-ut-UVarL
$$\frac{\Delta_1 \vdash_{\hat{a}} \tau \rightsquigarrow \tau' \dashv \Delta_2}{\Delta_1[\hat{a}] \vdash \hat{a} \approx \tau \dashv \Delta_2[\hat{a} = \tau']}$$

a-ut-UVarR
$$\frac{\Delta_1 \vdash_{\hat{a}} \tau \rightsquigarrow \tau' \dashv \Delta_2}{\Delta_1[\hat{a}] \vdash \tau \approx \hat{a} \dashv \Delta_2[\hat{a} = \tau']}$$

$$\textsc{a-ut-Proj}\quad \frac{\Delta_1 \vdash e_1 \approx e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2](\exists\, a.\epsilon_1) \approx [\Delta_2](\exists\, a.\epsilon_2) \dashv \Delta_3}{\Delta_1 \vdash \lfloor e_1 : \exists\, a.\epsilon_1 \rfloor \approx \lfloor e_2 : \exists\, a.\epsilon_2 \rfloor \dashv \Delta_2}$$

$$\textsc{a-ut-Arr}\quad \frac{\Delta_1 \vdash \sigma_1 \approx \sigma_2 \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2]\sigma_1' \approx [\Delta_2]\sigma_2' \dashv \Delta_3}{\Delta_1 \vdash \sigma_1 \to \sigma_1' \approx \sigma_2 \to \sigma_2' \dashv \Delta_3}$$

$$\textsc{a-ut-Exists}\quad \frac{\Delta_1, a \vdash \epsilon_1 \approx \epsilon_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash \exists\, a.\epsilon_1 \approx \exists\, a.\epsilon_2 \dashv \Delta_2} \qquad \textsc{a-ut-Forall}\quad \frac{\Delta_1, a \vdash \sigma_1 \approx \sigma_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash \forall a.\sigma_1 \approx \forall a.\sigma_2 \dashv \Delta_2}$$

$\boxed{\Delta_1 \vdash e_1 \approx e_2 \dashv \Delta_2}$ *(Expression Unification)*

$$\textsc{a-ue-Refl}\quad \frac{\vdash \Delta}{\Delta \vdash e \approx e \dashv \Delta} \qquad \textsc{a-ue-Abs}\quad \frac{\Delta_1, x : \mathsf{Int} \vdash e_1 \approx e_2 \dashv \Delta_2, x : \mathsf{Int}, \Delta_3}{\Delta_1 \vdash \lambda x.\ e_1 \approx \lambda x.\ e_2 \dashv \Delta_2}$$

$$\textsc{a-ue-AbsA}\quad \frac{\Delta_1 \vdash \sigma_1 \approx \sigma_2 \dashv \Delta_2 \qquad \Delta_2, x : \sigma_1 \vdash [\Delta_2]e_1 \approx [\Delta_2]e_2 \dashv \Delta_3, x : \sigma_1, \Delta_4}{\Delta_1 \vdash \lambda x : \sigma_1.\ e_1 \approx \lambda x : \sigma_2.\ e_2 \dashv \Delta_3}$$

$$\textsc{a-ue-App}\quad \frac{\Delta_1 \vdash e_1 \approx e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2]e_1' \approx [\Delta_2]e_2' \dashv \Delta_3}{\Delta_1 \vdash e_1\, e_1' \approx e_2\, e_2' \dashv \Delta_3}$$

$$\textsc{a-ue-Ann}\quad \frac{\Delta_1 \vdash e_1 \approx e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash [\Delta_2]\sigma_1 \approx [\Delta_2]\sigma_2 \dashv \Delta_3}{\Delta_1 \vdash (e_1 : \sigma_1) \approx (e_2 : \sigma_2) \dashv \Delta_3}$$

$\boxed{\Delta_1 \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_2 \dashv \Delta_2}$ *(Type Promotion)*

$$\textsc{a-prt-Int}\quad \frac{\vdash \Delta}{\Delta \vdash_{\hat{a}} \mathsf{Int} \rightsquigarrow \mathsf{Int} \dashv \Delta} \qquad \textsc{a-prt-Var}\quad \frac{\vdash \Delta}{\Delta[a][\hat{a}] \vdash_{\hat{a}} a \rightsquigarrow a \dashv \Delta[a][\hat{a}]}$$

$$\textsc{a-prt-UVarL}\quad \frac{\vdash \Delta \qquad \hat{b} \neq \hat{a}}{\Delta[\hat{b}][\hat{a}] \vdash_{\hat{a}} \hat{b} \rightsquigarrow \hat{b} \dashv \Delta[\hat{b}][\hat{a}]} \qquad \textsc{a-prt-UVarR}\quad \frac{\vdash \Delta \qquad \hat{b} \neq \hat{a}}{\Delta[\hat{a}][\hat{b}] \vdash_{\hat{a}} \hat{b} \rightsquigarrow \hat{b_1} \dashv \Delta[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}]}$$

$$\textsc{a-prt-Proj}\quad \frac{\Delta_1 \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash_{\hat{a}} [\Delta_2](\exists\, a.\epsilon_1) \rightsquigarrow \exists\, a.\epsilon_2 \dashv \Delta_3}{\Delta_1 \vdash_{\hat{a}} \lfloor e_1 : \exists\, a.\epsilon_1 \rfloor \rightsquigarrow \lfloor e_2 : \exists\, a.\epsilon_2 \rfloor \dashv \Delta_3}$$

$$\textsc{a-prt-Arr}\quad \frac{\Delta_1 \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_1' \dashv \Delta_2 \qquad \Delta_2 \vdash_{\hat{a}} [\Delta_2]\sigma_2 \rightsquigarrow \sigma_2' \dashv \Delta_3}{\Delta_1 \vdash_{\hat{a}} \sigma_1 \to \sigma_2 \rightsquigarrow \sigma_1' \to \sigma_2' \dashv \Delta_3}$$

$$\textsc{a-prt-Exists}\quad \frac{\Delta_1[a, \hat{a}] \vdash_{\hat{a}} \epsilon_1 \rightsquigarrow \epsilon_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1[\hat{a}] \vdash_{\hat{a}} \exists\, a.\epsilon_1 \rightsquigarrow \exists\, a.\epsilon_2 \dashv \Delta_2, \Delta_3} \qquad \textsc{a-prt-Forall}\quad \frac{\Delta_1[a, \hat{a}] \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1[\hat{a}] \vdash_{\hat{a}} \forall a.\sigma_1 \rightsquigarrow \forall a.\sigma_2 \dashv \Delta_2, \Delta_3}$$

$\boxed{\Delta_1 \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2}$ *(Expression Promotion)*

$$\textsc{a-pre-Int}\quad \frac{\vdash \Delta}{\Delta \vdash_{\hat{a}} n \rightsquigarrow n \dashv \Delta} \qquad \textsc{a-pre-Var}\quad \frac{\vdash \Delta}{\Delta[x : \tau][\hat{a}] \vdash_{\hat{a}} x \rightsquigarrow x \dashv \Delta[x : \tau][\hat{a}]}$$

$$\textsc{a-pre-Abs}\quad \frac{\Delta_1[x : \mathsf{Int}, \hat{a}] \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2, x : \mathsf{Int}, \Delta_3}{\Delta_1[\hat{a}] \vdash_{\hat{a}} \lambda x.\ e_1 \rightsquigarrow \lambda x.\ e_2 \dashv \Delta_2, \Delta_3}$$

$$\textsc{a-pre-AbsA}\quad \frac{\Delta_1[x : \sigma_1, \hat{a}] \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_2 \dashv \Delta_2 \qquad \Delta_2 \vdash_{\hat{a}} [\Delta_2]e_1 \rightsquigarrow e_2 \dashv \Delta_2, x : \sigma_1, \Delta_3}{\Delta_1[\hat{a}] \vdash_{\hat{a}} \lambda x : \sigma_1.\ e_1 \rightsquigarrow \lambda x : \sigma_2.\ e_2 \dashv \Delta_2}$$

$$\textsc{a-pre-App}\quad \frac{\Delta_1 \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash_{\hat{a}} [\Delta_2]e_1' \rightsquigarrow e_2' \dashv \Delta_3}{\Delta_1 \vdash_{\hat{a}} e_1\, e_1' \rightsquigarrow e_2\, e_2' \dashv \Delta_3}$$

$$\textsc{a-pre-Ann}\quad \frac{\Delta_1 \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2 \qquad \Delta_2 \vdash_{\hat{a}} [\Delta_2]\sigma_1 \rightsquigarrow \sigma_2 \dashv \Delta_3}{\Delta_1 \vdash_{\hat{a}} (e_1 : \sigma_1) \rightsquigarrow (e_2 : \sigma_2) \dashv \Delta_3}$$

D Meta Theory of the Algorithmic System

Lemma/Theorem	*Related DK Lemma/Theorem*
Lemma D.3 (Properties of Context Extension)	D.20, D.21, D.16, D.25, D.22
Lemma D.4 (Extension Order)	D.24
Theorem D.5 (Typing Extension)	D.33, I.54
Lemma D.9 (Applying Contexts to Types)	H.44, D.18, D.19, H.45, H.50
Lemma D.10 (Applying Contexts to Contexts)	H.47, H.51, H.52
Lemma D.11 (Filling Completes)	H.48
Lemma D.12 (Extension of Unsolved Contexts)	–
Lemma D.13 (Soundness of Promotion)	–
Lemma D.14 (Soundness of Unification)	–
Theorem D.15 (Soundness of Algorithmic Subtyping)	H.11
Theorem D.16 (Soundness of Algorithmic Typing)	H.12
Theorem D.17 (Completeness of Algorithmic Subtyping)	K.14
Theorem D.18 (Completeness of Algorithmic Instantiation)	K.13
Lemma D.21 (Instantiation Preserves $\leq_\exists$)	–
Lemma D.22 (Global Deep $\mathsf{min}_\exists$)	–
Lemma D.23 (Global $\mathsf{min}_\exists$)	–
Lemma D.26 ($\mathsf{min}_{\lfloor\cdot\rfloor}$ of Algorithmic Typing)	–
Theorem D.27 (Soundness of Algorithmic Inference (1))	H.12
Theorem D.28 (Completeness of Algorithmic Typing (1))	K.15
Theorem D.29 (Soundness of Algorithmic Inference (2))	H.12
Lemma D.30 ($\lvert\exists\rvert$ Inequality of Compatibility)	–
Theorem D.31 (Completeness of Algorithmic Typing (2))	K.15

D.1 Algorithmic Context

D.1.1 Context Extension

$\boxed{\Delta_1 \longrightarrow \Delta_2}$ *(Context Extension)*

$$\textsc{a-ext-Empty}\quad \frac{}{\bullet \longrightarrow \bullet} \qquad \textsc{a-ext-Var}\quad \frac{\Delta_1 \longrightarrow \Delta_2 \qquad [\Delta_2]\sigma_1 = [\Delta_2]\sigma_2}{\Delta_1, x : \sigma_1 \longrightarrow \Delta_2, x : \sigma_2} \qquad \textsc{a-ext-TVar}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1, a \longrightarrow \Delta_2, a}$$

$$\textsc{a-ext-Unsolved}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1, \hat{a} \longrightarrow \Delta_2, \hat{a}} \qquad \textsc{a-ext-Solve}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1, \hat{a} \longrightarrow \Delta_2, \hat{a} = \tau} \qquad \textsc{a-ext-Solved}\quad \frac{\Delta_1 \longrightarrow \Delta_2 \qquad [\Delta_2]\tau_1 = [\Delta_2]\tau_2}{\Delta_1, \hat{a} = \tau_1 \longrightarrow \Delta_2, \hat{a} = \tau_2}$$

$$\textsc{a-ext-AddUnsolved}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1 \longrightarrow \Delta_2, \hat{a}} \qquad \textsc{a-ext-AddSolved}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1 \longrightarrow \Delta_2, \hat{a} = \tau} \qquad \textsc{a-ext-Marker}\quad \frac{\Delta_1 \longrightarrow \Delta_2}{\Delta_1, \blacktriangleright_{\hat{a}} \longrightarrow \Delta_2, \blacktriangleright_{\hat{a}}}$$

Definition D.1 (Softness)**.** *A context Δ is soft iff it consists only of $\hat{a}$ and $\hat{a} = \tau$ declarations.*

Definition D.2. *$\langle \Delta \rangle$ consists of the unsolved declarations $\hat{a}$ in the context Δ.*

Lemma D.3 (Properties of Context Extension)**.** *a)* Reflexivity*: If $\vdash \Delta$, then $\Delta \longrightarrow \Delta$.*

b) Transitivity*: If $\Delta_1 \longrightarrow \Delta_2$ and $\Delta_2 \longrightarrow \Delta_3$, then $\Delta_1 \longrightarrow \Delta_3$.*

c) Declaration Order Preservation*: If $\Delta_1 \longrightarrow \Delta_2$ and u is declared to the left of v in Δ_1, then u is declared to the left of v in Δ_2.*

d) Extension Weakening*: If $\Delta_1 \vdash \sigma$ and $\Delta_1 \longrightarrow \Delta_2$, then $\Delta_2 \vdash \sigma$.*

e) Right Softness*: If Δ_3 is soft, $\Delta_1 \longrightarrow \Delta_2$, and Δ_2, Δ_3 is well-formed, then $\Delta_1 \longrightarrow \Delta_2, \Delta_3$.*

Proof. Proof in Dunfield and Krishnaswami [1]: Lemmas D.20, D.21, D.16, D.25, D.22. ☐

Lemma D.4 (Extension Order)**.** *a) If $\Delta_1', a, \Delta_1'' \longrightarrow \Delta_2$, then $\Delta_2 = (\Delta_2', a, \Delta_2'')$ where $\Delta_1' \longrightarrow \Delta_2'$. Moreover, if Δ_1'' is soft then Δ_2'' is soft.*

b) If $\Delta_1', x : \sigma_1, \Delta_1'' \longrightarrow \Delta_2$, then $\Delta_2 = (\Delta_2', x : \sigma_2, \Delta_2'')$ where $\Delta_1' \longrightarrow \Delta_2'$ and $[\Delta_2']\sigma_1 = [\Delta_2']\sigma_2$. Moreover, if Δ_1'' is soft then Δ_2'' is soft.

c) If $\Delta_1', \hat{a}, \Delta_1'' \longrightarrow \Delta_2$, then $\Delta_2 = (\Delta_2', \Delta_3, \Delta_2'')$ where $\Delta_1' \longrightarrow \Delta_2'$ and Δ_3 is either $\hat{a}$ or $\hat{a} = \tau$.

d) If $\Delta_1', \hat{a} = \tau_1, \Delta_1'' \longrightarrow \Delta_2$, then $\Delta_2 = (\Delta_2', \hat{a} = \tau_2, \Delta_2'')$ where $\Delta_1' \longrightarrow \Delta_2'$ and $[\Delta_2']\tau_1 = [\Delta_2']\tau_2$.

Proof. Proof in Dunfield and Krishnaswami [1]: Lemma D.24. ☐

Theorem D.5 (Typing Extension)**.** *a)* Promotion Extension: *If $\Delta_1 \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_2 \dashv \Delta_2$ or $\Delta_1 \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \Delta_2$, then $\Delta_1 \longrightarrow \Delta_2$.*

b) Unification Extension: *If $\Delta_1 \vdash \sigma_1 \approx \sigma_2 \dashv \Delta_2$ or $\Delta_1 \vdash e_1 \approx e_2 \dashv \Delta_2$, then $\Delta_1 \longrightarrow \Delta_2$.*

c) Subtyping Extension: *If $\Delta_1 \vdash e : \sigma_1 <: \sigma_2 \dashv \Delta_2$.*

d) Instantiation Extension: *If $\Delta_1 \vdash e : \sigma \rightsquigarrow \rho \dashv \Delta_2$ or $\Delta_1 \vdash \sigma \rightsquigarrow_\forall \epsilon \dashv \Delta_2$, then $\Delta_1 \longrightarrow \Delta_2$.*

e) Typing Extension: *If $\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2$, $\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2$, or $\Delta_1 \vdash e : \sigma \rightsquigarrow \rho \dashv \Delta_2$, then $\Delta_1 \longrightarrow \Delta_2$.*

Proof. By induction on the given derivation.

- Rules a-i-Int, a-i-Var, a-i-App, a-i-Ann, a-c-Sub, a-c-Abs, a-c-AbsA, a-c-AbsUVar, a-c-AbsAUVar, a-c-Exists, a-c-Forall, a-inst-Refl, a-inst-Exists, a-inst-Forall, a-instf-Refl, a-instf-Forall, a-s-Mono, a-s-Arr, a-s-ArrL, a-s-ArrR, a-s-ForallL, a-s-ForallR, a-s-ExistsL, a-s-ExistsR, a-ut-Refl, a-ut-Proj, a-ut-Arr, a-ut-Exists, a-ut-Forall, a-ue-Refl, a-ue-Abs, a-ue-AbsA, a-ue-App, a-ue-Ann, a-prt-Int, a-prt-Var, a-prt-UVarL, a-prt-Proj, a-prt-Arr, a-pre-Int, a-pre-Var, a-pre-Abs, a-pre-AbsA, a-pre-App, and a-pre-Ann: Apply induction hypothesis, Lemmas D.3(a, b, e), and D.4 as appropriate.

- Rule A-I-ABS: By the induction hypothesis and Lemma D.3(b), $\Delta_1, \hat{a}, x{:}\hat{a} \longrightarrow \Delta_3, x{:}\hat{a}, \Delta_4$. By Lemma D.4, $\Delta_1 \longrightarrow \Delta_3$, therefore $\Delta_1 \longrightarrow \Delta_3, \langle\Delta_4\rangle$ by lemma D.3(e).
- Rule A-I-ABSA: Similar to rule A-I-ABS.
- Rule A-INST-UVAR: By rules A-EXT-ADDUNSOLVED and A-EXT-SOLVE, $\Delta[\hat{a}] \longrightarrow \Delta[\hat{a_1}, \hat{a_2}, \hat{a}] \longrightarrow \Delta[\hat{a_1}, \hat{a_2}, \hat{a} = \hat{a_1} \to \hat{a_2}]$.
- Rules A-UT-UVARL and A-UT-UVARR: By rule A-EXT-SOLVE, $\Delta[\hat{a}] \longrightarrow \Delta[\hat{a} = \tau]$.
- Rule A-PRT-UVARR: By rules A-EXT-ADDUNSOLVED and A-EXT-SOLVE, $\Delta[\hat{a}][\hat{b}] \longrightarrow \Delta[\hat{b_1}, \hat{a}][\hat{b}] \longrightarrow \Delta[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}]$.
- Rules A-PRT-EXISTS and A-PRT-FORALL: By the induction hypothesis, $\Delta_1[a, \hat{a}] \longrightarrow \Delta_2, a, \Delta_3$. Note that no element of $\Delta[\hat{a}]$ refers to a by well-formedness. Moreover, promotion cannot add any elements to a context which refer to a type variable, therefore no element of Δ_2 or Δ_3 refers to a, which implies $\vdash \Delta_2, \Delta_3$. Then our extension relation is preserved if we remove a from both sides, i.e. $\Delta_1[\hat{a}] \longrightarrow \Delta_2, \Delta_3$.

□

D.1.2 Context Application

Definition D.6 (Applying a complete context to a context). *The application of a complete context to a context $[\Omega]\Delta$ is defined by:*

$$
\begin{aligned}
[\bullet]\bullet &= \bullet \\
[\Omega, x : \sigma](\Delta, x : \sigma') &= [\Omega]\Delta, x : [\Omega]\sigma && \textit{if } [\Omega]\sigma = [\Omega]\sigma' \\
[\Omega, a](\Delta, a) &= [\Omega]\Delta, a \\
[\Omega, \hat{a} = \tau](\Delta, \hat{a}) &= [\Omega]\Delta \\
[\Omega, \hat{a} = \tau](\Delta, \hat{a} = \tau') &= [\Omega]\Delta && \textit{if } [\Omega]\tau = [\Omega]\tau' \\
[\Omega, \hat{a} = \tau]\Delta &= [\Omega]\Delta && \textit{if } \hat{a} \notin \mathsf{dom}\,(\Delta)
\end{aligned}
$$

Definition D.7 (Applying a context to a type). *Substitution of a context in a type $[\Omega]\sigma$ is defined by:*

$$
\begin{aligned}
[\Delta]\mathsf{Int} &= \mathsf{Int} \\
[\Delta]a &= a \\
[\Delta[\hat{a}]]\hat{a} &= \hat{a} \\
[\Delta[\hat{a} = \tau]]\hat{a} &= [\Delta[\hat{a} = \tau]]\tau \\
[\Delta]\lfloor e : \exists\, a.\epsilon \rfloor &= \lfloor [\Delta]e : [\Delta]\exists\, a.\epsilon \rfloor \\
[\Delta](\sigma_1 \to \sigma_2) &= [\Delta]\sigma_1 \to [\Delta]\sigma_2 \\
[\Delta](\exists\, b.\epsilon) &= \exists\, b.([\Delta]\epsilon) \\
[\Delta](\forall a.\sigma) &= \forall a.([\Delta]\sigma)
\end{aligned}
$$

Definition D.8 (Filling)**.** *The fillling of a context* $|\Delta|$ *solves all unsolved unification variables:*

$$\begin{aligned} |\bullet| &= \bullet \\ |\Delta, x : \sigma| &= |\Delta|, x : \sigma \\ |\Delta, a| &= |\Delta|, a \\ |\Delta, \hat{a} = \tau| &= |\Delta|, \hat{a} = \tau \\ |\Delta, \hat{a}| &= |\Delta|, \hat{a} = \mathsf{Int} \end{aligned}$$

Lemma D.9 (Applying Contexts to Types)**.** *a)* Substitution for Well-Formedness: *If* $\Omega \vdash \sigma$*, then* $\Delta \vdash [\Delta]\sigma$*.*

b) Extension Equality Preservation*: If* $\Delta_1 \vdash \sigma_1$*,* $\Delta_1 \vdash \sigma_2$*,* $[\Delta_1]\sigma_1 = [\Delta_1]\sigma_2$*, and* $\Delta_1 \longrightarrow \Delta_2$*, then* $[\Delta_2]\sigma_1 = [\Delta_2]\sigma_2$*.*

c) Substitution Extension Invariance*: If* $\Delta_1 \vdash \sigma$ *and* $\Delta_1 \longrightarrow \Delta_2$*, then* $[\Delta_2]\sigma = [\Delta_2][\Delta_1]\sigma$ *and* $[\Delta_2]\sigma = [\Delta_1][\Delta_2]\sigma$*.*

d) Substitution Stability: *For any well-formed complete context* (Ω, Ω')*, if* $\Omega \vdash \sigma$*, then* $[\Omega]\sigma = [\Omega, \Omega']\sigma$*.*

e) Finishing Types: *If* $\Omega_1 \vdash \sigma$ *and* $\Omega_1 \longrightarrow \Omega_2$ *then* $[\Omega_1]\sigma = [\Omega_2]\sigma$*.*

Proof. Proof in Dunfield and Krishnaswami [1]: Lemmas H.44, D.18, D.19, H.45, and H.50. □

Lemma D.10 (Applying Contexts to Contexts)**.** *a)* Softness Goes Away: *If* $\Delta_1, \Delta_2 \longrightarrow \Omega_1, \Omega_2$ *where* $\Delta_1 \longrightarrow \Omega_1$ *and* Δ_2 *is soft, then* $[\Omega_1, \Omega_2](\Delta_1, \Delta_2) = [\Omega_1]\Delta_1$*.*

b) Finishing Completions: *If* $\Omega_1 \longrightarrow \Omega_2$ *then* $[\Omega_1]\Omega_1 = [\Omega_2]\Omega_2$*.*

c) Confluence of Completeness: *If* $\Delta_1 \longrightarrow \Omega$ *and* $\Delta_2 \longrightarrow \Omega$*, then* $[\Omega]\Delta_1 = [\Omega]\Delta_2$*.*

Proof. Proof in Dunfield and Krishnaswami [1]: Lemmas H.47, H.51, and H.52. □

Lemma D.11 (Filling Completes)**.** *If* $\Delta_1 \longrightarrow \Omega$ *and* Δ_1, Δ_2 *is well-formed, then* $\Delta_1, \Delta_2 \longrightarrow \Omega, |\Delta_2|$*.*

Proof. Proof in Dunfield and Krishnaswami [1]: Lemmas H.48. □

Lemma D.12 (Extension of Unsolved Contexts)**.** *If* $\Delta_1, \langle\Delta_2\rangle \longrightarrow \Omega$*, then* $\Omega = (\Omega_1, \Omega_2)$ *such that* $\Delta_1 \longrightarrow \Omega_1$*.*

Proof. If $\langle\Delta_2\rangle = \bullet$, then $\Omega_1 = \Omega$ and $\Omega_2 = \bullet$.
Otherwise, $\langle\Delta_2\rangle = \hat{a}, \Delta_2'$ for some Δ_2' by definition. By Lemma D.4, we either have $\Omega = (\Delta_1', \hat{a}, \Delta_2'')$ or $\Omega = (\Delta_1', \hat{a} = \tau, \Delta_2'')$ where $\Delta_1 \longrightarrow \Delta_1'$. By definition of complete contexts, we must have $\Omega = (\Delta_1', \hat{a} = \tau, \Delta_2'')$ and Δ_1' and Δ_2'' are complete contexts. Then let $\Omega_1 = \Delta_1'$ and $\Omega_2 = (\hat{a} = \tau, \Delta_2'')$. □

D.2 Soundness

Lemma D.13 (Soundness of Promotion). *a) If* $\mathit{\Delta}_1[\hat{a}] \vdash_{\hat{a}} \sigma_1 \rightsquigarrow \sigma_2 \dashv \mathit{\Delta}_2$*, then* $[\mathit{\Delta}_2]\sigma_1 = [\mathit{\Delta}_2]\sigma_2$ *and* $\mathit{\Delta}_2 = \mathit{\Delta}_2', \hat{a}, \mathit{\Delta}_2''$ *where* $\mathit{\Delta}_2' \vdash \sigma_2$*.*

b) If $\mathit{\Delta}_1[\hat{a}] \vdash_{\hat{a}} e_1 \rightsquigarrow e_2 \dashv \mathit{\Delta}_2$*, then* $[\mathit{\Delta}_2]e_1 = [\mathit{\Delta}_2]e_2$ *and* $\mathit{\Delta}_2 = \mathit{\Delta}_2', \hat{a}, \mathit{\Delta}_2''$ *where* $\mathsf{fv}\,(e_2) \subseteq \mathsf{dom}\,(\mathit{\Delta}_2')$*.*

Proof. By induction on the given derivation.

- Rules a-prt-Int, a-prt-Var, a-prt-UVarL, a-pre-Int, and a-pre-Var: Straightforward.
- Rules a-prt-Arr, a-prt-Proj, a-pre-Abs, a-pre-AbsA, a-pre-App, and a-pre-Ann: Induction hypothesis and Lemma D.9(c).
- Rule a-prt-UVarR:

$$\frac{\vdash \mathit{\Delta} \qquad \hat{b} \neq \hat{a}}{\mathit{\Delta}[\hat{a}][\hat{b}] \vdash_{\hat{a}} \hat{b} \rightsquigarrow \hat{b_1} \dashv \mathit{\Delta}[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}]}$$

  We have $[\mathit{\Delta}[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}]]\hat{b} = \hat{b_1} = [\mathit{\Delta}[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}]]\hat{b_1}$. We can write $\mathit{\Delta}[\hat{b_1}, \hat{a}][\hat{b} = \hat{b_1}] = \mathit{\Delta}_1, \hat{b_1}, \hat{a}, \mathit{\Delta}_2, \hat{b} = \hat{b_1}, \mathit{\Delta}_3$. Then $\mathit{\Delta}_1, \hat{b_1} \vdash \hat{b_1}$ by rule a-wf-UVar.
- Rule a-prt-Exists:

$$\frac{\mathit{\Delta}_1[a, \hat{a}] \vdash_{\hat{a}} \epsilon_1 \rightsquigarrow \epsilon_2 \dashv \mathit{\Delta}_2, a, \mathit{\Delta}_3}{\mathit{\Delta}_1[\hat{a}] \vdash_{\hat{a}} \exists\, a.\epsilon_1 \rightsquigarrow \exists\, a.\epsilon_2 \dashv \mathit{\Delta}_2, \mathit{\Delta}_3}$$

  Note that promotion can only add unification variables to the context or solve unification variables with other unification variables. Therefore removing a from $\mathit{\Delta}_2, a, \mathit{\Delta}_3$ cannot break the context well-formedness, since no other element of the context can depend on a.
  $[\mathit{\Delta}_2, a, \mathit{\Delta}_3]\epsilon_1 = [\mathit{\Delta}_2, a, \mathit{\Delta}_3]\epsilon_2$ i.h.
  $[\mathit{\Delta}_2, \mathit{\Delta}_3]\epsilon_1 = [\mathit{\Delta}_2, \mathit{\Delta}_3]\epsilon_2$ defn. of subst.
  $[\mathit{\Delta}_2, \mathit{\Delta}_3](\exists\, a.\epsilon_1) = [\mathit{\Delta}_2, \mathit{\Delta}_3](\exists\, a.\epsilon_2)$ defn. of subst.
  Lastly, we can write $\mathit{\Delta}_2, a, \mathit{\Delta}_3 = \mathit{\Delta}_2, a, \mathit{\Delta}_3', \hat{a}, \mathit{\Delta}_3''$. By the induction hypothesis, $\mathit{\Delta}_2, a, \mathit{\Delta}_3' \vdash \epsilon_2$. As noted above, the well-formedness of $\mathit{\Delta}_2, a, \mathit{\Delta}_3'$ cannot depend on a, therefore we have $\mathit{\Delta}_2, \mathit{\Delta}_3', a \vdash \epsilon_2$. By rule a-wf-Exists, $\mathit{\Delta}_2, \mathit{\Delta}_3' \vdash \exists\, a.\epsilon_2$.
- Rule a-prt-Forall: Similar to rule a-prt-Exists.

□

Lemma D.14 (Soundness of Unification). *a) If* $\mathit{\Delta}_1 \vdash \sigma_1 \approx \sigma_2 \dashv \mathit{\Delta}_2$*, then* $[\mathit{\Delta}_2]\sigma_1 = [\mathit{\Delta}_2]\sigma_2$*.*

b) If $\mathit{\Delta}_1 \vdash e_1 \approx e_2 \dashv \mathit{\Delta}_2$*, then* $[\mathit{\Delta}_2]e_1 = [\mathit{\Delta}_2]e_2$*.*

Proof. By induction on the given derivation.

- Rules a-ut-Refl and a-ue-Refl: Straightforward.

- Rules a-ut-Proj, a-ut-Arr, a-ue-Abs, a-ue-AbsA, a-ue-App, and a-ue-Ann: Induction hypothesis and Lemma D.9(c).
- Rules a-ut-UVarL and a-ut-UVarR: $[[\hat{a} = \tau]\Delta_2]\hat{a} = \tau = [[\hat{a} = \tau]\Delta_2]\tau$.
- Rule a-ut-Exists:

$$\frac{\Delta_1, a \vdash \epsilon_1 \approx \epsilon_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash \exists\, a.\epsilon_1 \approx \exists\, a.\epsilon_2 \dashv \Delta_2}$$

$\mathsf{fv}\,(\exists\, a.\epsilon_1) \subseteq \mathsf{dom}\,(\Delta_1) \subseteq \mathsf{dom}\,(\Delta_2)$	given
$\mathsf{fv}\,(\exists\, a.\epsilon_2) \subseteq \mathsf{dom}\,(\Delta_1) \subseteq \mathsf{dom}\,(\Delta_2)$	given
$[\Delta_2, a, \Delta_3]\epsilon_1 = [\Delta_2, a, \Delta_3]\epsilon_2$	i.h.
$[\Delta_2, a]\epsilon_1 = [\Delta_2, a]\epsilon_2$	above
$[\Delta_2](\exists\, a.\epsilon_1) = [\Delta_2](\exists\, a.\epsilon_2)$	defn. of subst.

- Rule a-ut-Forall: Similar to rule a-ut-Exists.

□

Theorem D.15 (Soundness of Algorithmic Subtyping). *a) If* $\Delta_1 \vdash e : \sigma_1 <: \sigma_2 \dashv \Delta_2$ *where* $[\Delta_1]\sigma_1 = \sigma_1$, $[\Delta_2]\sigma_2 = \sigma_2$, *and* $\Delta_2 \longrightarrow \Omega$, *then* $[\Omega]\Delta_2 \vdash e : [\Omega]\sigma_1 <: [\Omega]\sigma_2$.

Proof. By induction on the given derivation.

- Rule a-s-Mono:

$$\frac{\Delta_1 \vdash \tau_1 \approx \tau_2 \dashv \Delta_2}{\Delta_1 \vdash e : \tau_1 <: \tau_2 \dashv \Delta_2}$$

By Lemma D.14, $[\Delta_2]\tau_1 = [\Delta_2]\tau_2$, so Lemma D.9(c) implies $[\Omega]\tau_1 = [\Omega]\tau_2$. Then, by Lemma B.9, $[\Omega]\Delta_2 \vdash e : [\Omega]\tau_1 <: [\Omega]\tau_2$.

- Rule a-s-Arr:

$$\frac{\begin{array}{c}\Delta_1, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \dashv \Delta_2 \\ \Delta_2 \vdash e\, x : [\Delta_2]\sigma_2 <: [\Delta_2]\sigma_2' \dashv \Delta_3, x : \sigma_1', \Delta_4\end{array}}{\Delta_1 \vdash e : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2' \dashv \Delta_3}$$

$\Delta_3 \longrightarrow \Omega$	given
$\Delta_3, x : \sigma_1', \Delta_4 \longrightarrow \Omega, x : \sigma_1', \lvert\Delta_4\rvert$	Lemma D.11
$\Delta_2 \longrightarrow \Delta_3, x : \sigma_1', \Delta_4$	Theorem D.5
$\Delta_2 \longrightarrow \Omega, x : \sigma_1', \lvert\Delta_4\rvert$	Lemma D.3(b)

Let $\Omega' = \Omega, x : \sigma_1', |\Delta_4|$. We have shown $\Delta_2 \longrightarrow \Omega'$ and $\Delta_3, x : \sigma_1, \Delta_4 \longrightarrow \Omega'$.

$[\Omega']\Delta_2 \vdash x : [\Omega']\sigma_1' <: [\Omega']\sigma_1$	i.h.
$[\Omega'](\Delta_3, x : \sigma_1', \Delta_4) \vdash e\, x : [\Omega'][\Delta_2]\sigma_2 <: [\Omega'][\Delta_2]\sigma_2'$	i.h.
Δ_4 is soft	Lemma D.4
$[\Omega']\Delta_2 = [\Omega'](\Delta_3, x : \sigma_1', \Delta_4) = [\Omega]\Delta_3, x : [\Omega]\sigma_1'$	Lemma D.10(a, c)
$[\Omega']\sigma_1 = [\Omega]\sigma_1$	Lemma D.9(d)
$[\Omega']\sigma_1' = [\Omega]\sigma_1'$	Lemma D.9(d)
$[\Omega'][\Delta_2]\sigma_2 = [\Omega']\sigma_2 = [\Omega]\sigma_2$	Lemma D.9(c, d)
$[\Omega'][\Delta_2]\sigma_2' = [\Omega']\sigma_2' = [\Omega]\sigma_2'$	Lemma D.9(c, d)
$[\Omega]\Delta_3, x : [\Omega]\sigma_1' \vdash x : [\Omega]\sigma_1' <: [\Omega]\sigma_1$	above
$[\Omega]\Delta_3, x : [\Omega]\sigma_1' \vdash e\, x : [\Omega]\sigma_2 <: [\Omega]\sigma_2'$	above
$[\Omega]\Delta_3 \vdash e : [\Omega](\sigma_1 \to \sigma_2) <: [\Omega](\sigma_1' \to \sigma_2')$	Rule d-s-Arr

- Rule A-S-ArrL:

$$\frac{\begin{array}{c}\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \sigma_1 \vdash x : \sigma_1 <: \hat{a}_1 \dashv \Delta_2 \\ \Delta_2 \vdash e\,x : \hat{a}_2 <: [\Delta_2]\sigma_2 \dashv \Delta_3, x : \sigma_1, \Delta_4\end{array}}{\Delta_1[\hat{a}] \vdash e : \hat{a} <: \sigma_1 \to \sigma_2 \dashv \Delta_3}$$

We must have $\hat{a} \notin \mathsf{fv}\,(\sigma_1 \to \sigma_2)$, since otherwise the premises would not be possible ($\hat{a}_1$ or $\hat{a}_2$ would have to depend on $\hat{a}$). Since $\hat{a} \notin \mathsf{fv}\,(\sigma_1 \to \sigma_2)$ and $\hat{a}_1, \hat{a}_2 \notin \mathsf{fv}\,(\sigma_1) \cup \mathsf{fv}\,(\sigma_2)$, we have $[\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \sigma_1](\sigma_1 \to \sigma_2) = [\Delta_1[\hat{a}]](\sigma_1 \to \sigma_2) = \sigma_1 \to \sigma_2$.

$\Delta_3 \longrightarrow \Omega$	given
$\Delta_3, x : \sigma_1, \Delta_4 \longrightarrow \Omega, x : \sigma_1, \lvert\Delta_4\rvert$	Lemma D.11
$\Delta_2 \longrightarrow \Delta_3, x : \sigma_1, \Delta_4$	Theorem D.5
$\Delta_2 \longrightarrow \Omega, x : \sigma_1, \lvert\Delta_4\rvert$	Lemma D.3(b)

Let $\Omega' = \Omega, x : \sigma_1, \lvert\Delta_4\rvert$. We have shown $\Delta_2 \longrightarrow \Omega'$ and $\Delta_3, x : \sigma_1, \Delta_4 \longrightarrow \Omega'$.

$[\Omega']\Delta_2 \vdash x : [\Omega']\sigma_1 <: [\Omega']\hat{a}_1$	i.h.
$[\Omega'](\Delta_3, x : \sigma_1, \Delta_4) \vdash e\,x : [\Omega']\hat{a}_2 <: [\Omega'][\Delta_2]\sigma_2$	i.h.
Δ_4 is soft	Lemma D.4
$[\Omega']\Delta_2 = [\Omega'](\Delta_3, x : \sigma_1, \Delta_4) = [\Omega]\Delta_3, x : [\Omega]\sigma_1$	Lemma D.10(a, c)
$[\Omega']\sigma_1 = [\Omega]\sigma_1$	Lemma D.9(d)
$[\Omega']\hat{a}_1 = [\Omega]\hat{a}_1$	Lemma D.9(d)
$[\Omega'][\Delta_2]\sigma_2 = [\Omega']\sigma_2 = [\Omega]\sigma_2$	Lemma D.9(c, d)
$[\Omega']\hat{a}_2 = [\Omega]\hat{a}_2$	Lemma D.9(d)
$[\Omega]\Delta_3, x : [\Omega]\sigma_1 \vdash x : [\Omega]\sigma_1 <: [\Omega]\hat{a}_1$	above
$[\Omega]\Delta_3, x : [\Omega]\sigma_1 \vdash e\,x : [\Omega]\hat{a}_2 <: [\Omega]\sigma_2$	above
$[\Omega]\Delta_3 \vdash e : [\Omega](\hat{a}_1 \to \hat{a}_2) <: [\Omega](\sigma_1 \to \sigma_2)$	Rule D-S-Arr

- Rule A-S-ArrR: Similar to rule A-S-ArrL.
- Rule A-S-ExistsL:

$$\frac{\Delta_1 \vdash e : [\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: \epsilon_2 \dashv \Delta_2}{\Delta_1 \vdash e : \exists\, a.\epsilon_1 <: \epsilon_2 \dashv \Delta_2}$$

$[\Omega]\Delta_2 \vdash e : [\Omega][\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: [\Omega]\epsilon_2$	i.h.
$[\Omega]\Delta_2 \vdash e : [\lfloor e : [\Omega]\exists\, a.\epsilon_1 \rfloor / a][\Omega]\epsilon_1 <: [\Omega]\epsilon_2$	dist. of subst.
$[\Omega]\Delta_2 \vdash e : [\Omega]\exists\, a.\epsilon_1 <: [\Omega]\epsilon_2$	Rule D-S-ExistsL

- Rule A-S-ExistsR:

$$\frac{\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : \rho <: [\hat{a}/a]\epsilon \dashv \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3}{\Delta_1 \vdash e : \rho <: \exists\, a.\epsilon \dashv \Delta_2}$$

We know $\Delta_2 \longrightarrow \Omega$, so $\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3 \longrightarrow \Omega, \blacktriangleright_{\hat{a}}, |\Delta_3|$ by Lemma D.11. Let $\Omega' = \Omega, \blacktriangleright_{\hat{a}}, |\Delta_3|$.

$[\Omega'](\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3) \vdash e : [\Omega']\rho <: [\Omega'][\hat{a}/a]\epsilon$	i.h.
Δ_3 is soft	Lemma D.4
$[\Omega'](\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3) = [\Omega]\Delta_2$	Lemma D.10(a)
$[\Omega']\rho = [\Omega]\rho$	Lemma D.9(d)
$[\Omega'][\hat{a}/a]\epsilon = [\Omega][\hat{a}/a]\epsilon$	Lemma D.9(d)
$[\Omega]\Delta_2 \vdash e : [\Omega]\rho <: [\Omega][\hat{a}/a]\epsilon$	above
$[\Omega]\Delta_2 \vdash e : [\Omega]\rho <: [[\Omega]\hat{a}/a][\Omega]\epsilon$	dist. of subst.
$[\Omega]\Delta_2 \vdash e : [\Omega]\rho <: [\Omega]\exists\, a.\epsilon$	Rule D-S-ExistsR

- Rule A-S-ForallL:

$$\frac{\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\hat{a}/a]\sigma <: \epsilon \dashv \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3}{\Delta_1 \vdash e : \forall a.\sigma <: \epsilon \dashv \Delta_2}$$

We know $\Delta_2 \longrightarrow \Omega$, so $\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3 \longrightarrow \Omega, \blacktriangleright_{\hat{a}}, |\Delta_3|$ by Lemma D.11. Let $\Omega' = \Omega, \blacktriangleright_{\hat{a}}, |\Delta_3|$.

$[\Omega'](\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3) \vdash e : [\Omega'][\hat{a}/a]\sigma <: [\Omega']\epsilon$	i.h.
Δ_3 is soft	Lemma D.4
$[\Omega'](\Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3) = [\Omega]\Delta_2$	Lemma D.10(a)
$[\Omega']\epsilon = [\Omega]\epsilon$	Lemma D.9(d)
$[\Omega'][\hat{a}/a]\sigma = [\Omega][\hat{a}/a]\sigma$	Lemma D.9(d)
$[\Omega]\Delta_2 \vdash e : [\Omega][\hat{a}/a]\sigma <: [\Omega]\epsilon$	above
$[\Omega]\Delta_2 \vdash e : [[\Omega]\hat{a}/a][\Omega]\sigma <: [\Omega]\epsilon$	dist. of subst.
$[\Omega]\Delta_2 \vdash e : [\Omega]\sigma <: [\Omega]\forall a.\epsilon$	Rule D-S-ForallL

- Rule A-S-ForallR:

$$\frac{\Delta_1, a \vdash e : \sigma_1 <: \sigma_2 \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash e : \sigma_1 <: \forall a.\sigma_2 \dashv \Delta_2}$$

Let $\Omega' = \Omega, a, |\Delta_3|$. We are given $\Delta_2 \longrightarrow \Omega$, hence $\Delta_2, a, \Delta_3 \longrightarrow \Omega'$ by Lemma D.11.

$[\Omega'](\Delta_2, a, \Delta_3) \vdash e : [\Omega']\sigma_1 <: [\Omega']\sigma_2$	i.h.
Δ_3 is soft	Lemma D.4
$[\Omega'](\Delta_2, a, \Delta_3) = [\Omega]\Delta_2, a$	Lemma D.10(a)
$[\Omega']\sigma_1 = [\Omega]\sigma_1$	Lemma D.9(d)
$[\Omega']\sigma_2 = [\Omega]\sigma_2$	Lemma D.9(d)
$[\Omega]\Delta_2, a \vdash e : [\Omega]\sigma_1 <: [\Omega]\sigma_2$	above
$[\Omega]\Delta_2 \vdash e : [\Omega]\sigma_1 <: [\Omega]\forall a.\sigma_2$	Rule D-S-ForallR

□

Theorem D.16 (Soundness of Algorithmic Typing). *Given $\Delta_2 \longrightarrow \Omega$,*

a) If $\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2$, then $[\Omega]\Delta_2 \vdash e \Rightarrow [\Omega]\sigma$.
b) If $\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2$, then $[\Omega]\Delta_2 \vdash e \Leftarrow [\Omega]\sigma$.
c) If $\Delta_1 \vdash e : \sigma \rightsquigarrow \rho \dashv \Delta_2$, then $[\Omega]\Delta_2 \vdash e : [\Omega]\sigma \rightsquigarrow [\Omega]\rho$.
d) If $\Delta_1 \vdash \sigma \rightsquigarrow_{\forall} \epsilon \dashv \Delta_2$, then $[\Omega]\Delta_2 \vdash [\Omega]\sigma \rightsquigarrow_{\forall} [\Omega]\epsilon$.

Proof. By induction on the given derivation.

- Rules A-I-Int, A-inst-Refl, and A-instf-Refl: Apply the corresponding declarative rule.
- Rule A-I-Var:

$$\frac{x : \sigma \in \mathit{\Delta}}{\mathit{\Delta} \vdash x \Rightarrow \sigma \dashv \mathit{\Delta}}$$

$x : \sigma \in \mathit{\Delta}$	premise
$x : [\mathit{\Omega}]\sigma \in [\mathit{\Omega}]\mathit{\Delta}$	defn.
$[\mathit{\Omega}]\mathit{\Delta} \vdash x \Rightarrow [\mathit{\Omega}]\sigma$	Rule D-I-Var

- Rule A-I-Abs:

$$\frac{\begin{array}{c}\mathit{\Delta}_1, \hat{a}, x : \hat{a} \vdash e \Rightarrow \sigma \dashv \mathit{\Delta}_2 \qquad \mathit{\Delta}_2 \vdash [\mathit{\Delta}_2]\sigma \rightsquigarrow_\forall \epsilon \dashv \mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4 \\ \epsilon_1 = [\mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4]\epsilon \qquad \overline{a} \ \mathsf{fresh} \qquad \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1\end{array}}{\mathit{\Delta}_1 \vdash \lambda x.\ e \Rightarrow \hat{a} \to \exists\, \overline{a}.\epsilon' \dashv \mathit{\Delta}_3, \langle \mathit{\Delta}_4 \rangle}$$

$\mathit{\Delta}_3, \langle \mathit{\Delta}_4 \rangle \longrightarrow \mathit{\Omega}$	given
$\mathit{\Omega} = (\mathit{\Omega}_1, \mathit{\Omega}_2)$ s.t. $\mathit{\Delta}_3 \longrightarrow \mathit{\Omega}_1$	Lemma D.12
$\mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4 \longrightarrow \mathit{\Omega}_1, x : \hat{a}, \lvert \mathit{\Delta}_4 \rvert$	Lemma D.11
Let $\mathit{\Omega}' = \mathit{\Omega}_1, x : \hat{a}, \lvert \mathit{\Delta}_4 \rvert$.	
$[\mathit{\Omega}']\mathit{\Delta}_2 \vdash e \Rightarrow [\mathit{\Omega}']\sigma$	i.h.
$[\mathit{\Omega}'](\mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4) \vdash [\mathit{\Omega}'][\mathit{\Delta}_2]\sigma \rightsquigarrow_\forall [\mathit{\Omega}']\epsilon$	i.h.
$\mathit{\Delta}_4$ is soft	Lemma D.4
$[\mathit{\Omega}']\mathit{\Delta}_2 = [\mathit{\Omega}'](\mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4) = [\mathit{\Omega}_1]\mathit{\Delta}_3, x : [\mathit{\Omega}']\hat{a}$	Lemma D.10(a, c)
$[\mathit{\Omega}'][\mathit{\Delta}_2]\sigma = [\mathit{\Omega}']\sigma = [\mathit{\Omega}_1]\sigma$	Lemma D.9(c, d)
$[\mathit{\Omega}']\epsilon_1 = [\mathit{\Omega}'][\mathit{\Delta}_3, x : \hat{a}, \mathit{\Delta}_4]\epsilon = [\mathit{\Omega}']\epsilon = [\mathit{\Omega}_1]\epsilon$	Lemma D.9(c, d)
$[\mathit{\Omega}_1]\mathit{\Delta}_3, x : [\mathit{\Omega}']\hat{a} \vdash e \Rightarrow [\mathit{\Omega}_1]\sigma$	above
$[\mathit{\Omega}_1]\mathit{\Delta}_3, x : [\mathit{\Omega}']\hat{a} \vdash [\mathit{\Omega}_1]\sigma \rightsquigarrow_\forall [\mathit{\Omega}_1]\epsilon$	above
$[\overline{a}/\lfloor [\mathit{\Omega}_1]\epsilon \rfloor_x][\mathit{\Omega}_1]\epsilon = [\mathit{\Omega}_1][\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon = [\mathit{\Omega}_1]\epsilon_2$	dist. of subst.
$[\mathit{\Omega}_1]\mathit{\Delta}_3 \vdash \lambda x.\ e \Rightarrow [\mathit{\Omega}_1](\hat{a} \to \exists\, \overline{a}.\epsilon_2)$	Rule D-I-Abs

  Note that $\mathit{\Omega}_1$ does not mention x, so $\lfloor [\mathit{\Omega}_1]\epsilon \rfloor_x$ and $\lfloor \epsilon \rfloor_x$ are analogous.
- Rule A-I-AbsA: Similar to rule A-I-Abs.
- Rule A-I-App:

$$\frac{\begin{array}{c}\mathit{\Delta}_1 \vdash e \Rightarrow \sigma \dashv \mathit{\Delta}_2 \\ \mathit{\Delta}_2 \vdash e : [\mathit{\Delta}_2]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \mathit{\Delta}_3 \qquad \mathit{\Delta}_3 \vdash e_1 \Leftarrow [\mathit{\Delta}_3]\sigma_1 \dashv \mathit{\Delta}_4\end{array}}{\mathit{\Delta}_1 \vdash e\, e_1 \Rightarrow \sigma_2 \dashv \mathit{\Delta}_4}$$

$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash e \Rightarrow [\mathit{\Omega}]\sigma$	i.h.
$[\mathit{\Omega}]\mathit{\Delta}_3 \vdash e : [\mathit{\Omega}][\mathit{\Delta}_2]\sigma \rightsquigarrow [\mathit{\Omega}](\sigma_1 \to \sigma_2)$	i.h.
$[\mathit{\Omega}]\mathit{\Delta}_4 \vdash e_1 \Leftarrow [\mathit{\Omega}][\mathit{\Delta}_3]\sigma_1$	i.h.
$[\mathit{\Omega}]\mathit{\Delta}_2 = [\mathit{\Omega}]\mathit{\Delta}_3 = [\mathit{\Omega}]\mathit{\Delta}_4$	Item D.10(c)
$[\mathit{\Omega}][\mathit{\Delta}_2]\sigma = [\mathit{\Omega}]\sigma$	Lemma D.9(c)
$[\mathit{\Omega}][\mathit{\Delta}_2]\sigma_1 = [\mathit{\Omega}]\sigma_1$	Lemma D.9(c)
$[\mathit{\Omega}]\mathit{\Delta}_4 \vdash e \Rightarrow [\mathit{\Omega}]\sigma$	above
$[\mathit{\Omega}]\mathit{\Delta}_4 \vdash e : [\mathit{\Omega}]\sigma \rightsquigarrow [\mathit{\Omega}]\sigma_1 \to [\mathit{\Omega}]\sigma_2$	above
$[\mathit{\Omega}]\mathit{\Delta}_4 \vdash e_1 \Leftarrow [\mathit{\Omega}]\sigma_1$	above
$[\mathit{\Omega}]\mathit{\Delta}_5 \vdash e\, e_1 \Rightarrow [\mathit{\Omega}]\sigma_2$	Rule D-I-App

- Rule A-I-Ann:

$$\frac{\mathit{\Delta}_1 \vdash e \Leftarrow \sigma \dashv \mathit{\Delta}_2 \qquad \bullet \vdash \sigma}{\mathit{\Delta}_1 \vdash (e : \sigma) \Rightarrow \sigma \dashv \mathit{\Delta}_2}$$

Note that $\mathsf{fv}\,(\sigma) = \varnothing$ since $\bullet \vdash \sigma$, therefore $[\mathit{\Omega}]\sigma = \sigma$.

$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash e \Leftarrow [\mathit{\Omega}]\sigma$	i.h.
$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash (e : [\mathit{\Omega}]\sigma) \Rightarrow [\mathit{\Omega}]\sigma$	Rule D-I-App
$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash (e : \sigma) \Rightarrow [\mathit{\Omega}]\sigma$	since $[\mathit{\Omega}]\sigma = \sigma$

- Rule A-C-Sub:

$$\frac{\mathit{\Delta}_1 \vdash e \Rightarrow \sigma \dashv \mathit{\Delta}_2 \qquad \mathit{\Delta}_2 \vdash e : [\mathit{\Delta}_2]\sigma <: [\mathit{\Delta}_2]\rho \dashv \mathit{\Delta}_3}{\mathit{\Delta}_1 \vdash e \Leftarrow \rho \dashv \mathit{\Delta}_3}$$

We have $\mathit{\Delta}_1 \longrightarrow \mathit{\Delta}_2$ by Theorem D.5, which implies $\mathit{\Delta}_1 \longrightarrow \mathit{\Omega}$ by Lemma D.3(b)

$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash e \Rightarrow [\mathit{\Omega}]\sigma$	i.h.
$[\mathit{\Omega}]\mathit{\Delta}_3 \vdash e : [\mathit{\Omega}][\mathit{\Delta}_2]\sigma <: [\mathit{\Omega}][\mathit{\Delta}_2]\rho$	Theorem D.15
$[\mathit{\Omega}]\mathit{\Delta}_2 = [\mathit{\Omega}]\mathit{\Delta}_3$	Item D.10(c)
$[\mathit{\Omega}][\mathit{\Delta}_2]\sigma = [\mathit{\Omega}]\sigma$	Lemma D.9(c)
$[\mathit{\Omega}][\mathit{\Delta}_2]\rho = [\mathit{\Omega}]\rho$	Lemma D.9(c)
$[\mathit{\Omega}]\mathit{\Delta}_3 \vdash e \Rightarrow [\mathit{\Omega}]\sigma$	above
$[\mathit{\Omega}]\mathit{\Delta}_3 \vdash e : [\mathit{\Omega}]\sigma <: [\mathit{\Omega}]\rho$	above
$[\mathit{\Omega}]\mathit{\Delta}_3 \vdash e \Leftarrow [\mathit{\Omega}]\rho$	Rule D-C-Sub

- Rule A-C-Abs:

$$\frac{\mathit{\Delta}_1, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \dashv \mathit{\Delta}_2, x : \sigma_1, \mathit{\Delta}_3}{\mathit{\Delta}_1 \vdash \lambda x.\ e \Leftarrow \sigma_1 \rightarrow \sigma_2 \dashv \mathit{\Delta}_2}$$

$\mathit{\Delta}_2 \longrightarrow \mathit{\Omega}$	given
$\mathit{\Delta}_2, x : \sigma_1, \mathit{\Delta}_3 \longrightarrow \mathit{\Omega}, x : \sigma_1, \lvert\mathit{\Delta}_3\rvert$	Lemma D.11
Let $\mathit{\Omega}' = \mathit{\Omega}, x : \sigma_1, \lvert\mathit{\Delta}_3\rvert$.	
$[\mathit{\Omega}'](\mathit{\Delta}_2, x : \sigma_1, \mathit{\Delta}_3) \vdash e \Leftarrow [\mathit{\Omega}']\sigma_2$	i.h.
$\mathit{\Delta}_3$ is soft	Lemma D.4
$[\mathit{\Omega}'](\mathit{\Delta}_2, x : \sigma_1, \mathit{\Delta}_3) = [\mathit{\Omega}]\mathit{\Delta}_2, x : [\mathit{\Omega}]\sigma_1$	Lemma D.10(a)
$[\mathit{\Omega}']\sigma_2 = [\mathit{\Omega}]\sigma_2$	Lemma D.9(d)
$[\mathit{\Omega}]\mathit{\Delta}_2, x : [\mathit{\Omega}]\sigma_1 \vdash e \Leftarrow [\mathit{\Omega}]\sigma_2$	above
$[\mathit{\Omega}]\mathit{\Delta}_2 \vdash \lambda x.\ e \Leftarrow [\mathit{\Omega}](\sigma_1 \rightarrow \sigma_2)$	Rule D-S-Arr

- Rule A-C-AbsA:

$$\frac{\begin{array}{c}\mathit{\Delta}_1, x : \sigma_1 \vdash x : \sigma_1 <: \sigma'_1 \dashv \mathit{\Delta}_2 \\ \mathit{\Delta}_2 \vdash e \Leftarrow [\mathit{\Delta}_2]\sigma_2 \dashv \mathit{\Delta}_3, x : \sigma_1, \mathit{\Delta}_4 \qquad \bullet \vdash \sigma'_1\end{array}}{\mathit{\Delta}_1 \vdash \lambda x : \sigma'_1.\ e \Leftarrow \sigma_1 \rightarrow \sigma_2 \dashv \mathit{\Delta}_3}$$

$\mathit{\Delta}_3 \longrightarrow \mathit{\Omega}$	given
$\mathit{\Delta}_3, x : \sigma_1, \mathit{\Delta}_4 \longrightarrow \mathit{\Omega}, x : \sigma_1, \lvert\mathit{\Delta}_4\rvert$	Lemma D.11
$\mathit{\Delta}_2 \longrightarrow \mathit{\Delta}_3, x : \sigma_1, \mathit{\Delta}_4$	Theorem D.5
$\mathit{\Delta}_2 \longrightarrow \mathit{\Omega}, x : \sigma_1, \lvert\mathit{\Delta}_4\rvert$	Lemma D.3(b)

Let $\mathit{\Omega}' = \mathit{\Omega}, x : \sigma_1, \lvert\mathit{\Delta}_4\rvert$. Note that $\mathsf{fv}\,(\sigma'_1) = \varnothing$ since $\bullet \vdash \sigma'_1$, therefore $[\mathit{\Omega}]\sigma'_1 =$

σ_1'.

$[\Omega']\Delta_2 \vdash x : [\Omega']\sigma_1 <: [\Omega']\sigma_1'$	i.h.
$[\Omega'](\Delta_3, x : \sigma_1, \Delta_4) \vdash e \Leftarrow [\Omega'][\Delta_2]\sigma_2$	i.h.
Δ_4 is soft	Lemma D.4
$[\Omega']\Delta_2 = [\Omega'](\Delta_3, x : \sigma_1, \Delta_4) = [\Omega]\Delta_3, x : [\Omega]\sigma_1$	Lemma D.10(a, c)
$[\Omega']\sigma_1 = [\Omega]\sigma_1$	Lemma D.9(d)
$[\Omega']\sigma_1' = [\Omega]\sigma_1'$	Lemma D.9(d)
$[\Omega'][\Delta_2]\sigma_2 = [\Omega']\sigma_2 = [\Omega]\sigma_2$	Lemma D.9(c, d)
$[\Omega]\Delta_3, x : [\Omega]\sigma_1 \vdash x : [\Omega]\sigma_1 <: [\Omega]\sigma_1'$	above
$[\Omega]\Delta_3, x : [\Omega]\sigma_1 \vdash e \Leftarrow [\Omega]\sigma_2$	above
$[\Omega]\Delta_3 \vdash \lambda x : [\Omega]\sigma_1'.\ e \Leftarrow [\Omega](\sigma_1 \to \sigma_2)$	Rule D-S-Arr
$[\Omega]\Delta_3 \vdash \lambda x : \sigma_1'.\ e \Leftarrow [\Omega](\sigma_1 \to \sigma_2)$	since $[\Omega]\sigma_1' = \sigma_1'$

- Rule A-C-AbsUVar:

$$\frac{\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1 \vdash e \Leftarrow \hat{a}_2 \dashv \Delta_2, x : \hat{a}_1, \Delta_3}{\Delta_1[\hat{a}] \vdash \lambda x.\ e \Leftarrow \hat{a} \dashv \Delta_2}$$

$\Delta_2 \longrightarrow \Omega$	given
$\Delta_2, x : \hat{a}_1, \Delta_3 \longrightarrow \Omega, x : \hat{a}_1, \lvert\Delta_3\rvert$	Lemma D.11
Let $\Omega' = \Omega, x : \hat{a}_1, \lvert\Delta_3\rvert$.	
$[\Omega'](\Delta_2, x : \hat{a}_1, \Delta_3) \vdash e \Leftarrow [\Omega']\hat{a}_2$	i.h.
Δ_3 is soft	Lemma D.4
$[\Omega'](\Delta_2, x : \hat{a}_1, \Delta_3) = [\Omega]\Delta_2, x : [\Omega]\hat{a}_1$	Lemma D.10(a)
$[\Omega']\hat{a}_2 = [\Omega]\hat{a}_2$	Lemma D.9(d)
$[\Omega]\Delta_2, x : [\Omega]\hat{a}_1 \vdash e \Leftarrow [\Omega]\hat{a}_2$	above
$[\Omega]\Delta_2 \vdash \lambda x.\ e \Leftarrow [\Omega](\hat{a}_1 \to \hat{a}_2)$	Rule D-S-Arr
$[\Omega](\hat{a}_1 \to \hat{a}_2) = [\Omega]\hat{a}$	defn.
$[\Omega]\Delta_2 \vdash \lambda x.\ e \Leftarrow [\Omega]\hat{a}$	above

- Rule A-C-AbsAUVar:

$$\frac{\begin{array}{c}\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1 \vdash x : \hat{a}_1 <: \sigma_1 \dashv \Delta_2 \\ \Delta_2 \vdash e \Leftarrow [\Delta_2]\hat{a}_2 \dashv \Delta_3, x : \hat{a}_1, \Delta_4 \qquad \bullet \vdash \sigma_1\end{array}}{\Delta_1 \vdash \lambda x : \sigma_1.\ e \Leftarrow \hat{a} \dashv \Delta_3}$$

$\Delta_3 \longrightarrow \Omega$	given
$\Delta_3, x : \hat{a}_1, \Delta_4 \longrightarrow \Omega, x : \hat{a}_1, \lvert\Delta_4\rvert$	Lemma D.11
$\Delta_2 \longrightarrow \Delta_3, x : \hat{a}_1, \Delta_4$	Theorem D.5
$\Delta_2 \longrightarrow \Omega, x : \hat{a}_1, \lvert\Delta_4\rvert$	Lemma D.3(b)

Let $\Omega' = \Omega, x : \hat{a}_1, \lvert\Delta_4\rvert$. Note that $\mathsf{fv}\,(\sigma_1) = \varnothing$ since $\bullet \vdash \sigma_1$, therefore $[\Omega]\sigma_1 =$

σ_1.

$[\Omega']\Delta_2 \vdash x : [\Omega']\hat{a}_1 <: [\Omega']\sigma_1$	i.h.
$[\Omega'](\Delta_3, x : \hat{a}_1, \Delta_4) \vdash e \Leftarrow [\Omega'][\Delta_2]\hat{a}_2$	i.h.
Δ_4 is soft	Lemma D.4
$[\Omega']\Delta_2 = [\Omega'](\Delta_3, x : \hat{a}_1, \Delta_4) = [\Omega]\Delta_3, x : [\Omega]\hat{a}_1$	Lemma D.10(a, c)
$[\Omega']\hat{a}_1 = [\Omega]\hat{a}_1$	Lemma D.9(d)
$[\Omega']\sigma_1 = [\Omega]\sigma_1$	Lemma D.9(d)
$[\Omega'][\Delta_2]\hat{a}_2 = [\Omega']\hat{a}_2 = [\Omega]\hat{a}_2$	Lemma D.9(c, d)
$[\Omega]\Delta_3, x : [\Omega]\hat{a}_1 \vdash x : [\Omega]\hat{a}_1 <: [\Omega]\sigma_1$	above
$[\Omega]\Delta_3, x : [\Omega]\hat{a}_1 \vdash e \Leftarrow [\Omega]\hat{a}_2$	above
$[\Omega]\Delta_3 \vdash \lambda x : [\Omega]\sigma_1.\ e \Leftarrow [\Omega](\hat{a}_1 \to \hat{a}_2)$	Rule D-s-Arr
$[\Omega](\hat{a}_1 \to \hat{a}_2) = [\Omega]\hat{a}$	defn.
$[\Omega]\Delta_2 \vdash \lambda x : \sigma_1.\ e \Leftarrow [\Omega]\hat{a}$	above

– Rule A-c-Exists:

$$\frac{\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \vdash e \Leftarrow [\hat{b}/b]\epsilon \dashv \Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3}{\Delta_1 \vdash e \Leftarrow \exists\, b.\epsilon \dashv \Delta_2}$$

We know $\Delta_2 \longrightarrow \Omega$, so $\Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3 \longrightarrow \Omega, \blacktriangleright_{\hat{b}}, |\Delta_3|$ by Lemma D.11. Let $\Omega' = \Omega, \blacktriangleright_{\hat{b}}, |\Delta_3|$.

$[\Omega'](\Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3) \vdash e \Leftarrow [\Omega'][\hat{b}/b]\epsilon$	i.h.
Δ_3 is soft	Lemma D.4
$[\Omega'](\Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3) = [\Omega]\Delta_2$	Lemma D.10(a)
$[\Omega'][\hat{b}/b]\epsilon = [\Omega][\hat{b}/b]\epsilon$	Lemma D.9(d)
$[\Omega]\Delta_2 \vdash e \Leftarrow [\Omega][\hat{b}/b]\epsilon$	above
$[\Omega]\Delta_2 \vdash e \Leftarrow [[\Omega]\hat{b}/b][\Omega]\epsilon$	dist. of subst.
$[\Omega]\Delta_2 \vdash e \Leftarrow [\Omega]\exists\, b.\epsilon$	Rule D-c-Exists

– Rule A-c-Forall:

$$\frac{\Delta_1, a \vdash e \Leftarrow \sigma \dashv \Delta_2, a, \Delta_3}{\Delta_1 \vdash e \Leftarrow \forall a.\sigma \dashv \Delta_2}$$

By Theorem D.5, $\Delta_1, a \longrightarrow \Delta_2, a, \Delta_3$ where Δ_3 is soft by Lemma D.4. Let $\Omega' = \Omega, a, |\Delta_3|$. Then, $\Delta_2, a, \Delta_3 \longrightarrow \Omega'$ by Lemma D.11.

$[\Omega'](\Delta_2, a, \Delta_3) \vdash e \Leftarrow [\Omega']\sigma$	i.h.
$[\Omega'](\Delta_2, a, \Delta_3) = [\Omega]\Delta_2, a$	Lemma D.10(a)
$[\Omega']\sigma = [\Omega]\sigma$	Lemma D.9(d)
$[\Omega]\Delta_2, a \vdash e \Leftarrow [\Omega]\sigma$	above
$[\Omega]\Delta_2 \vdash e \Leftarrow [\Omega]\forall a.\sigma$	Rule D-c-Forall

– Rule A-inst-UVar:

$$\frac{}{\Delta[\hat{a}] \vdash e : \hat{a} \rightsquigarrow \hat{a}_1 \to \hat{a}_2 \dashv \Delta[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2]}$$

We are given $\Delta[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2] \longrightarrow \Omega$. By the definition of extension, $\hat{a}_1$ and $\hat{a}_2$ are solved with monotypes in Ω. Then, by the definition of extension, we have $\Delta[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2] \longrightarrow \Omega[\hat{a}_1 = \tau_1, \hat{a}_2 = \tau_2, \hat{a} = \tau_1' \to \tau_2']$ where $[\Omega](\tau_1' \to \tau_2') = [\Omega](\hat{a}_1 \to \hat{a}_2)$, i.e. $[\Omega]\hat{a}_1 = [\Omega]\tau_1 = [\Omega]\tau_1'$ and $[\Omega]\hat{a}_2 = [\Omega]\tau_2 =$

$[\Omega]\tau_2'$.
$[\Omega]\hat{a} = [\Omega](\hat{a}_1 \to \hat{a}_2) = [\Omega]\tau_1 \to [\Omega]\tau_2$ defn.
$[\Omega]\Delta_2 \vdash e : [\Omega]\hat{a} \rightsquigarrow [\Omega](\hat{a}_1 \to \hat{a}_2)$ Rule d-inst-Refl

- Rule a-inst-Exists:

$$\frac{\Delta_1 \vdash e : [\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}{\Delta_1 \vdash e : \exists\, a.\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}$$

$[\Omega]\Delta_2 \vdash e : [\Omega][\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ i.h.
$[\Omega]\Delta_2 \vdash e : [\lfloor e : [\Omega]\exists\, a.\epsilon \rfloor / a][\Omega]\epsilon \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ dist. of subst.
$[\Omega]\Delta_2 \vdash e : [\Omega]\exists\, a.\epsilon \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ Rule d-inst-Exists

- Rule a-inst-Forall:

$$\frac{\Delta_1, \hat{a} \vdash e : [\hat{a}/a]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}{\Delta_1 \vdash e : \forall a.\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_2}$$

$[\Omega]\Delta_2 \vdash e : [\Omega][\hat{a}/a]\sigma \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ i.h.
$[\Omega]\Delta_2 \vdash e : [[\Omega]\hat{a}/a][\Omega]\sigma \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ dist. of subst.
$[\Omega]\Delta_2 \vdash e : [\Omega]\forall a.\sigma \rightsquigarrow [\Omega](\sigma_1 \to \sigma_2)$ Rule d-inst-Forall

- Rule a-instf-Forall: Similar to rule a-inst-Forall.

□

D.3 Completeness of Subtyping

Theorem D.17 (Completeness of Algorithmic Subtyping)**.** *Given* $\Delta_1 \longrightarrow \Omega_1$, $\Delta_1 \vdash \sigma_1$, $\Delta_1 \vdash \sigma_2$, *and* $[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\sigma_1 <: [\Omega_1]\sigma_2$, *then there exist* Δ_2 *and* Ω_2 *such that* $\Delta_2 \longrightarrow \Omega_2$, $\Omega_1 \longrightarrow \Omega_2$, *and* $\Delta_1 \vdash e : [\Delta_1]\sigma_1 <: [\Delta_1]\sigma_2 \dashv \Delta_2$.

Proof. By induction on the given derivation. We will consider separately the cases where at least one of $[\Delta_1]\sigma_1$ or $[\Delta_1]\sigma_2$ is a unification variable, i.e. $[\Delta_1]\sigma_1 \in \mathsf{unsolved}\,(\Delta_1)$ or $[\Delta_1]\sigma_2 \in \mathsf{unsolved}\,(\Delta_1)$.

a) Suppose $[\Delta_1]\sigma_1$ and $[\Delta_1]\sigma_2$ are not unification variables.
  - Rules d-s-Int, d-s-Var, and d-s-Proj: Apply rule a-s-Mono and rule a-ut-Refl.
  - Rule d-s-Arr:

$$\frac{\begin{array}{c}[\Omega_1]\Delta_1, x : [\Omega_1]\sigma_1' \vdash x : [\Omega_1]\sigma_1' <: [\Omega_1]\sigma_1 \\ [\Omega_1]\Delta_1, x : [\Omega_1]\sigma_1' \vdash e\, x : [\Omega_1]\sigma_2 <: [\Omega_1]\sigma_2'\end{array}}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\sigma_1 \to [\Omega_1]\sigma_2 <: [\Omega_1]\sigma_1' \to [\Omega_1]\sigma_2'}$$

We have $[\Omega_1, x{:}[\Omega_1]\sigma_1'](\Delta_1, x{:}[\Delta_1]\sigma_1') = [\Omega_1]\Delta_1, x{:}[\Omega_1]\sigma_1'$ since $[\Omega_1][\Delta_1]\sigma_1' = [\Omega_1]\sigma_1'$ by Lemma D.9.
$[\Omega_1, x : [\Omega_1]\sigma_1'](\Delta_1, x : [\Delta_1]\sigma_1') \vdash x : [\Omega_1, x : [\Omega_1]\sigma_1']\sigma_1' <: [\Omega_1, x : [\Omega_1]\sigma_1']\sigma_1$ premise
$\Delta_1, x : [\Delta_1]\sigma_1' \vdash x : [\Delta_1, x : [\Delta_1]\sigma_1']\sigma_1' <: [\Delta_1, x : [\Delta_1]\sigma_1']\sigma_1 \dashv \Delta_2$ i.h.
$\Omega_1, x : [\Omega_1]\sigma_1' \longrightarrow \Omega_2$ i.h.
$\Delta_2 \longrightarrow \Omega_2$ i.h.
$\Delta_1, x : [\Delta_1]\sigma_1' \vdash x : [\Delta_1]\sigma_1' <: [\Delta_1]\sigma_1 \dashv \Delta_2$ defn. of subst.

$[\varOmega_1, x : [\varOmega_1]\sigma_1'](\varDelta_1, x : [\varDelta_1]\sigma_1') = [\varOmega_1, x : [\varOmega_1]\sigma_1'](\varOmega_1, x : [\varOmega_1]\sigma_1') = [\varOmega_2]\varOmega_2 = [\varOmega_2]\varDelta_2$ Lemma D.10(b, c)
$[\varOmega_1]\sigma_2 = [\varOmega_2]\sigma_2 = [\varOmega_2][\varDelta_2]\sigma_2$ Lemma D.9(c, e)
$[\varOmega_1]\sigma_2' = [\varOmega_2]\sigma_2' = [\varOmega_2][\varDelta_2]\sigma_2'$ Lemma D.9(c, e)
$[\varOmega_2]\varDelta_2 \vdash e : [\varOmega_2][\varDelta_2]\sigma_2 <: [\varOmega_2][\varDelta_2]\sigma_2'$ premise
$\varDelta_2 \vdash e : [\varDelta_2][\varDelta_2]\sigma_2 <: [\varDelta_2][\varDelta_2]\sigma_2' \dashv \varDelta_3'$ i.h.
$\varDelta_3' \longrightarrow \varOmega_3'$ i.h.
$\varDelta_2 \longrightarrow \varOmega_3'$ i.h.

$\varDelta_1, x : [\varDelta_1]\sigma_1' \longrightarrow \varDelta_3'$ Theorem D.5
$\varDelta_3' = \varDelta_3, x : \sigma_0, \varDelta_4$ s.t. $\varDelta_1 \longrightarrow \varDelta_3$ and $[\varDelta_3]\sigma_0 = [\varDelta_3]\sigma_1'$ Lemma D.4
$\varDelta_3, x : \sigma_0, \varDelta_4 \longrightarrow \varOmega_3'$ above
$\varOmega_3' = \varOmega_3, x : \sigma_0', \varOmega_4$ s.t. $\varDelta_3 \longrightarrow \varOmega_3$ and $[\varOmega_3]\sigma_0 = [\varOmega_3]\sigma_0'$ Lemma D.4
$\varOmega_1, x : [\varOmega_1]\sigma_1' \longrightarrow \varOmega_3, x : \sigma_0', \varOmega_4$ above
$\varOmega_1 \longrightarrow \varOmega_3$ Lemma D.4
$\varDelta_1 \vdash e : [\varDelta_1]\sigma_1 \to [\varDelta_1]\sigma_2 <: [\varDelta_1]\sigma_1' \to [\varDelta_1]\sigma_2' \dashv \varDelta_3$ Rule a-s-Arr

- Rule d-s-ArrA: Similar to rule d-s-Arr.
- Rule d-s-ExistsL:

$$\frac{[\varOmega_1]\varDelta_1 \vdash e : [\lfloor e : [\varOmega_1]\exists\, a.\epsilon_1 \rfloor / a][\varOmega_1]\epsilon_1 <: [\varOmega_1]\epsilon_2}{[\varOmega_1]\varDelta_1 \vdash e : [\varOmega_1]\exists\, a.\epsilon_1 <: [\varOmega_1]\epsilon_2}$$

$[\varOmega_1]\varDelta_1 \vdash e : [\varOmega_1][\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: [\varOmega_1]\epsilon_2$ dist. of subst.
$\varDelta_1 \vdash e : [\varDelta_1][\lfloor e : \exists\, a.\epsilon_1 \rfloor / a]\epsilon_1 <: [\varDelta_1]\epsilon_2 \dashv \varDelta_2$ i.h.
$\varOmega_1 \longrightarrow \varOmega_2$ i.h.
$\varDelta_2 \longrightarrow \varOmega_2$ i.h.
$\varDelta_1 \vdash e : [\lfloor e : [\varDelta_1]\exists\, a.\epsilon_1 \rfloor / a][\varDelta_1]\epsilon_1 <: [\varDelta_1]\epsilon_2 \dashv \varDelta_2$ dist. of subst.
$\varDelta_1 \vdash e : [\varDelta_1]\exists\, a.\epsilon_1 <: [\varDelta_1]\epsilon_2 \dashv \varDelta_2$ Rule a-s-ExistsL

- Rule d-s-ExistsR:

$$\frac{[\varOmega_1]\varDelta_1 \vdash e : [\varOmega_1]\rho <: [\tau / a][\varOmega_1]\epsilon \qquad [\varOmega_1]\varDelta_1 \vdash \tau}{[\varOmega_1]\varDelta_1 \vdash e : [\varOmega_1]\rho <: [\varOmega_1]\exists\, a.\epsilon}$$

We have $\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a} \longrightarrow \varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau$ by rules a-ext-Marker and a-ext-Solve.
$[\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau](\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a}) = [\varOmega_1]\varDelta_1$ defn. of app.
$[\varOmega_1]\rho = [\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau]\rho$ defn. of subst.
$[\tau / a][\varOmega_1]\epsilon = [\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau][\hat{a} / a]\epsilon$ defn. of subst.
$[\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau](\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a}) \vdash e : [\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau]\rho$ premise
$<: [\varOmega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau][\hat{a} / a]\epsilon$
$\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a}]\rho <: [\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a}][\hat{a} / a]\epsilon \dashv \varDelta_2'$ i.h.
$\varOmega_1 \longrightarrow \varOmega_2'$ i.h.
$\varDelta_2' \longrightarrow \varOmega_2'$ i.h.
$\varDelta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\varDelta_1]\rho <: [\hat{a} / a][\varDelta_1]\epsilon \dashv \varDelta_2'$ dist. of subst.

$\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \longrightarrow \Delta'_2$ Theorem D.5
$\Delta'_2 = \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$ Lemma D.4
$\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3 \longrightarrow \Omega'_2$ above
$\Omega'_2 = \Omega_2, \blacktriangleright_{\hat{a}}, \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$ Lemma D.4
$\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau \longrightarrow \Omega_2, \blacktriangleright_{\hat{a}}, \Omega_3$ above
$\Omega_1 \longrightarrow \Omega_2$ Lemma D.4
$\Delta_1 \vdash e : [\Delta_1]\rho <: [\Delta_1]\exists\, a.\epsilon \dashv \Delta_2$ Rule A-S-ExistsR

- Rule D-S-ForallL:

$$\frac{[\Omega_1]\Delta_1 \vdash e : [\tau/a][\Omega_1]\sigma <: [\Omega_1]\epsilon \qquad [\Omega_1]\Delta_1 \vdash \tau}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\forall a.\sigma <: [\Omega_1]\epsilon}$$

We have $\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \longrightarrow \Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau$ by rules A-ext-Marker and A-ext-Solve.
$[\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau](\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a}) = [\Omega_1]\Delta_1$ defn. of app.
$[\tau/a][\Omega_1]\sigma = [\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau][\hat{a}/a]\sigma$ defn. of subst.
$[\Omega_1]\epsilon = [\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau]\epsilon$ defn. of subst.
$[\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau](\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a}) \vdash e : [\Omega_1, \hat{a}, \hat{a} = \tau][\hat{a}/a]\sigma$ premise
$<: [\Omega_1, \hat{a}, \hat{a} = \tau]\epsilon$
$\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a}][\hat{a}/a]\sigma <: [\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a}]\epsilon \dashv \Delta'_2$ i.h.
$\Omega_1 \longrightarrow \Omega'_2$ i.h.
$\Delta'_2 \longrightarrow \Omega'_2$ i.h.
$\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \vdash e : [\hat{a}/a][\Delta_1]\sigma <: [\Delta_1]\epsilon \dashv \Delta'_2$ dist. of subst.

$\Delta_1, \blacktriangleright_{\hat{a}}, \hat{a} \longrightarrow \Delta'_2$ Theorem D.5
$\Delta'_2 = \Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$ Lemma D.4
$\Delta_2, \blacktriangleright_{\hat{a}}, \Delta_3 \longrightarrow \Omega'_2$ above
$\Omega'_2 = \Omega_2, \blacktriangleright_{\hat{a}}, \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$ Lemma D.4
$\Omega_1, \blacktriangleright_{\hat{a}}, \hat{a} = \tau \longrightarrow \Omega_2, \blacktriangleright_{\hat{a}}, \Omega_3$ above
$\Omega_1 \longrightarrow \Omega_2$ Lemma D.4
$\Delta_1 \vdash e : [\Delta_1]\forall a.\sigma <: [\Delta_1]\epsilon \dashv \Delta_2$ Rule A-S-ExistsR

- Rule D-S-ForallR:

$$\frac{[\Omega_1]\Delta_1, a \vdash e : [\Omega_1]\sigma_1 <: [\Omega_1]\sigma_2}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\sigma_1 <: [\Omega_1]\forall a.\sigma_2}$$

$[\Omega_1, a](\Delta_1, a) = [\Omega_1]\Delta_1, a$ defn. of app.
$[\Omega_1, a](\Delta_1, a) \vdash e : [\Omega_1]\sigma_1 <: [\Omega_1]\sigma_2$ premise
$\Delta_1, a \vdash e : [\Delta_1, a]\sigma_1 <: [\Delta_1, a]\sigma_2 \dashv \Delta'_2$ i.h.
$\Omega_1 \longrightarrow \Omega'_2$ i.h.
$\Delta'_2 \longrightarrow \Omega'_2$ i.h.
$\Delta_1, a \vdash e : [\Delta_1]\sigma_1 <: [\Delta_1]\sigma_2 \dashv \Delta'_2$ defn. of app.

$\Delta_1, a \longrightarrow \Delta'_2$	Theorem D.5
$\Delta'_2 = \Delta_2, a, \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$	Lemma D.4
$\Delta_2, a, \Delta_3 \longrightarrow \Omega'_2$	above
$\Omega'_2 = \Omega_2, a, \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$	Lemma D.4
$\Omega_1, a \longrightarrow \Omega_2, a, \Omega_3$	above
$\Omega_1 \longrightarrow \Omega_2$	Lemma D.4
$\Delta_1 \vdash e : [\Delta_1]\sigma_1 <: [\Delta_1]\forall a.\sigma_2 \dashv \Delta_2$	Rule A-S-ForallR

b) Suppose $[\Delta_1]\sigma_1 = \hat{a}$, i.e. $\hat{a} \in \mathsf{unsolved}\,(\Delta_1)$.

- Case $[\Delta_1]\sigma_2 = \tau$: Apply rule A-S-Mono. The premise unifies $[\Delta_1]\sigma_1$ and $[\Delta_2]\sigma_2$.

  If $[\Delta_1]\sigma_2 = [\Delta_1]\sigma_1 = \hat{a}$, rule A-UT-Refl applies. Otherwise, rule A-UT-UVarL applies.
- Case $[\Delta_1]\sigma_2 = \sigma \to \sigma'$: By Lemma D.9(c), $[\Omega_1]\sigma_2 = [\Omega_1][\Delta_1]\sigma_2 = [\Omega_1]\sigma \to [\Omega_1]\sigma'$. Note that $[\Omega_1]\sigma_1 = \tau \to \tau'$, so rule D-S-Arr must have been used.

$$\frac{\begin{array}{c}[\Omega_1]\Delta_1, x : [\Omega_1]\sigma \vdash x : [\Omega_1]\sigma <: \tau \\ [\Omega_1]\Delta_1, x : [\Omega_1]\sigma \vdash e\,x : \tau' <: [\Omega_1]\sigma'\end{array}}{[\Omega_1]\Delta_1 \vdash e : \tau \to \tau' <: [\Omega_1]\sigma \to [\Omega_1]\sigma'}$$

$\Delta_1[\hat{a}] \longrightarrow \Delta_1[\hat{a}_2, \hat{a}_1, \hat{a}]$	Rule A-ext-AddUnsolved
$\longrightarrow \Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2]$	Rule A-ext-Solve
$\Omega_1[\hat{a} = \tau \to \tau']$	
$\longrightarrow \Omega_1[\hat{a}_2 = \tau', \hat{a}_1 = \tau, \hat{a} = \tau \to \tau']$	Rule A-ext-AddSolved
$\longrightarrow \Omega_1[\hat{a}_2 = \tau', \hat{a}_1 = \tau, \hat{a} = \hat{a}_1 \to \hat{a}_2]$	Rule A-ext-Solved

Let $\Delta'_1 = (\Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : [\Delta_1]\sigma)$.
Let $\Omega'_1 = (\Omega_1[\hat{a}_2 = \tau', \hat{a}_1 = \tau, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : [\Omega_1]\sigma)$.

$[\Omega_1]\Delta_1, x : [\Omega_1]\sigma = [\Omega_1, x : [\Omega_1]\sigma](\Delta_1, x : [\Delta_1]\sigma)$	defn. of app.
$[\Omega_1, x : [\Omega_1]\sigma](\Delta_1, x : [\Delta_1]\sigma) = [\Omega'_1]\Delta'_1$	Lemma D.10(b)
$[\Omega_1]\sigma = [\Omega'_1]\sigma$	Lemma D.9(e)
$[\Omega_1]\sigma' = [\Omega'_1]\sigma'$	Lemma D.9(e)
$[\Omega'_1]\hat{a}_1 = \tau$	defn.
$[\Omega'_1]\hat{a}_2 = \tau'$	defn.

$[\Omega'_1]\Delta'_1 \vdash x : [\Omega'_1]\sigma <: [\Omega'_1]\hat{a}_1$	premise
$\Delta'_1 \vdash x : [\Delta'_1]\sigma <: [\Delta'_1]\hat{a}_1 \dashv \Delta_2$	i.h.
$\Omega'_1 \longrightarrow \Omega_2$	i.h.
$\Delta_2 \longrightarrow \Omega_2$	i.h.
$\Delta'_1 \vdash x : [\Delta'_1]\sigma <: \hat{a}_1 \dashv \Delta_2$	assumption

$[\Omega_1']\Delta_1' = [\Omega_1']\Omega_1' = [\Omega_2]\Omega_2 = [\Omega_2]\Delta_2$	Lemma D.10(b, c)
$[\Omega_1']\sigma' = [\Omega_2]\sigma' = [\Omega_2][\Delta_2]\sigma'$	Lemma D.9(c, e)
$[\Omega_2]\Delta_2 \vdash e : [\Omega_2]\hat{a}_2 <: [\Omega_2][\Delta_2]\sigma'$	premise
$\Delta_2 \vdash e : [\Delta_2]\hat{a}_2 <: [\Delta_2]\sigma' \dashv \Delta_3'$	i.h.
$\Omega_2 \longrightarrow \Omega_3'$	i.h.
$\Delta_3' \longrightarrow \Omega_3'$	i.h.
$\Delta_2 \vdash e : \hat{a}_2 <: [\Delta_2]\sigma' \dashv \Delta_3'$	assumption

$\Delta_1, x : [\Delta_1]\sigma \longrightarrow \Delta_3'$	Theorem D.5
$\Delta_3' = \Delta_3, x : \sigma_0, \Delta_4$ s.t. $[\Delta_3]\sigma_0 = [\Delta_3]\sigma$	Lemma D.4
$\Delta_3, x : \sigma_0, \Delta_4 \longrightarrow \Omega_3'$	above
$\Omega_3' = \Omega_3, x : \sigma_0', \Omega_4$ s.t. $\Delta_3 \longrightarrow \Omega_3$ and $[\Omega_3]\sigma_0' = [\Omega_3][\Delta_1]\sigma$	Lemma D.4
$\Omega_1, x : [\Omega_1]\sigma \longrightarrow \Omega_3, x : \sigma_0', \Omega_4$	above
$\Omega_1 \longrightarrow \Omega_3$	Lemma D.4
$\Delta_1 \vdash e : \hat{a} <: [\Delta_1](\sigma \to \sigma') \dashv \Delta_3$	Rule A-S-ArrL

- Case $[\Delta_1]\sigma_2 = \exists\, a.\epsilon$: Rule D-S-ExistsR must have been applied. The proof given in the relevant case above will apply similarly.
- Case $[\Delta_1]\sigma_2 = \forall a.\sigma$: Rule D-S-ExistsR must have been applied. The proof given in the relevant case above will apply similarly.

c) Suppose $[\Delta_1]\sigma_2 = \hat{a}$, i.e. $\hat{a} \in \mathsf{unsolved}\,(\Delta_1)$.
The proof is similar to the previous case (using rule A-S-ArrR in the second case instead of rule A-S-ArrL).

□

Theorem D.18 (Completeness of Algorithmic Instantiation)**.** *Given* $\Delta_1 \longrightarrow \Omega_1$,

a) If $[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\sigma \rightsquigarrow \rho$, *then there exist* Δ_2, Ω_2, *and* ρ' *such that* $\Delta_2 \longrightarrow \Omega_2$, $\Omega_1 \longrightarrow \Omega_2$, $\rho = [\Omega_2]\rho'$, *and* $\Delta_1 \vdash e : [\Delta_1]\sigma \rightsquigarrow \rho' \dashv \Delta_2$.
b) If $[\Omega_1]\Delta_1 \vdash [\Omega_1]\sigma \rightsquigarrow_\forall \epsilon$, *then there exist* Δ_2, Ω_2, *and* ϵ' *such that* $\Delta_2 \longrightarrow \Omega_2$, $\Omega_1 \longrightarrow \Omega_2$, $\epsilon = [\Omega_2]\epsilon'$, *and* $\Delta_1 \vdash [\Delta_1]\sigma \rightsquigarrow_\forall \epsilon' \dashv \Delta_2$.

Proof. By induction on the given derivation.

- Rule D-inst-Refl

$$\frac{}{[\Omega_1]\Delta_1 \vdash e : \sigma_1' \to \sigma_2' \rightsquigarrow \sigma_1' \to \sigma_2'}$$

Above, we have $[\Omega_1]\rho = \sigma_1' \to \sigma_2'$. There are two possible values for $[\Delta_1]\rho$.

- $[\Delta_1]\rho = \sigma_1 \to \sigma_2$ where $\sigma_1' = [\Omega_1]\sigma_1$ and $\sigma_2' = [\Omega_1]\sigma_2$:

$$\frac{}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1](\sigma_1 \to \sigma_2) \rightsquigarrow [\Omega_1](\sigma_1 \to \sigma_2)}$$

Apply rule A-inst-Refl.

- $[\Delta_1]\rho = \hat{a}$ where $[\Omega_1]\hat{a} = \tau_1 \to \tau_2$ (with $\tau_1 = \sigma_1'$ and $\tau_2 = \sigma_2'$):

$$\frac{}{[\Omega_1]\Delta_1 \vdash e : \tau_1 \to \tau_2 \leadsto \tau_1 \to \tau_2}$$

Note that $\hat{a} \in \Delta_1$ and $\hat{a} = \tau \in \Omega_1$ such that $[\Omega_1]\tau = \tau_1 \to \tau_2$.
Let $\Delta_2 = \Delta_1[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2]$.
Let $\Omega_2 = \Omega_1[\hat{a}_1 = \tau_1, \hat{a}_2 = \tau_2, \hat{a} = \hat{a}_1 \to \hat{a}_2]$.

$$\begin{array}{lr}
\Omega_1[\hat{a}] \longrightarrow \Omega_1[\hat{a}_1 = \tau_1, \hat{a}_2 = \tau_2, \hat{a} = \tau] & \text{Rule }\textsc{a-ext-AddSolved} \\
\qquad\quad \longrightarrow \Omega_1[\hat{a}_1 = \tau_1, \hat{a}_2 = \tau_2, \hat{a} = \hat{a}_1 \to \hat{a}_2] & \text{Rule }\textsc{a-ext-Solved} \\
\qquad\quad = \Omega_2 & \text{defn.} \\
\Delta_2 = \Delta_1[\hat{a}_1, \hat{a}_2, \hat{a} = \hat{a}_1 \to \hat{a}_2] & \\
\qquad \longrightarrow \Omega_1[\hat{a}_1 = \tau_1, \hat{a}_2 = \tau_2, \hat{a} = \hat{a}_1 \to \hat{a}_2] = \Omega_2 & \text{Rule }\textsc{a-ext-Solve} \\
\Delta_1 \vdash e : \hat{a} \leadsto \hat{a}_1 \to \hat{a}_2 \dashv \Delta_2 & \text{Rule }\textsc{a-inst-UVar} \\
[\Delta_1]\hat{a} = \hat{a} & \text{assumption} \\
\Delta_1 \vdash e : [\Delta_1]\hat{a} \leadsto \hat{a}_1 \to \hat{a}_2 \dashv \Delta_2 & \text{above} \\
[\Omega_2](\hat{a}_1 \to \hat{a}_2) = \tau_1 \to \tau_2 = [\Omega_1]\hat{a} = [\Omega_1]\rho & \text{defn.}
\end{array}$$

– Rule D-INST-EXISTS

$$\frac{[\Omega_1]\Delta_1 \vdash e : [\lfloor e : [\Omega_1]\exists\, a.\epsilon \rfloor / a][\Omega_1]\epsilon \leadsto \sigma_1 \to \sigma_2}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\exists\, a.\epsilon \leadsto \sigma_1 \to \sigma_2}$$

$$\begin{array}{lr}
[\Omega_1]\Delta_1 \vdash e : [\Omega_1][\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \leadsto \sigma_1 \to \sigma_2 & \text{dist. of subst.} \\
\Delta_1 \vdash e : [\Delta_1][\lfloor e : \exists\, a.\epsilon \rfloor / a]\epsilon \leadsto \sigma_1' \to \sigma_2' \dashv \Delta_2 & \text{i.h.} \\
\Omega_1 \longrightarrow \Omega_2 & \text{i.h.} \\
\Delta_2 \longrightarrow \Omega_2 & \text{i.h.} \\
[\Omega_2](\sigma_1' \to \sigma_2') = \sigma_1 \to \sigma_2 & \text{i.h.} \\
\Delta_1 \vdash e : [\Delta_1]\exists\, a.\epsilon \leadsto \sigma_1' \to \sigma_2' \dashv \Delta_2 & \text{Rule }\textsc{a-inst-Exists}
\end{array}$$

– Rule D-INST-FORALL

$$\frac{[\Omega_1]\Delta_1 \vdash e : [\tau/a][\Omega_1]\sigma \leadsto \sigma_1 \to \sigma_2 \qquad [\Omega_1]\Delta_1 \vdash \tau}{[\Omega_1]\Delta_1 \vdash e : [\Omega_1]\forall a.\sigma \leadsto \sigma_1 \to \sigma_2}$$

$$\begin{array}{lr}
\Delta_1, \hat{a} \longrightarrow \Omega_1, \hat{a} = \tau & \text{Rule }\textsc{a-ext-Solve} \\
[\Omega_1, \hat{a} = \tau](\Delta_1, \hat{a}) = [\Omega_1]\Delta_1 & \text{defn. of app.} \\
[\tau/a][\Omega_1]\sigma = [\tau/a][\Omega_1, \hat{a} = \tau]\sigma = [\Omega_1, \hat{a} = \tau][\hat{a}/a]\sigma & \text{defn. of subst.} \\
[\Omega_1, \hat{a} = \tau](\Delta_1, \hat{a}) \vdash e : [\Omega_1, \hat{a} = \tau][\hat{a}/a]\sigma \leadsto \sigma_1 \to \sigma_2 & \text{premise} \\
\Delta_1, \hat{a} \vdash e : [\Delta_1, \hat{a}][\hat{a}/a]\sigma \leadsto \sigma_1' \to \sigma_2' \dashv \Delta_2 & \text{i.h.} \\
\Omega_1 \longrightarrow \Omega_2 & \text{i.h.} \\
\Delta_2 \longrightarrow \Omega_2 & \text{i.h.} \\
[\Omega_2](\sigma_1' \to \sigma_2') = \sigma_1 \to \sigma_2 & \text{i.h.} \\
\Delta_1, \hat{a} \vdash e : [\hat{a}/a][\Delta_1]\sigma \leadsto \sigma_1' \to \sigma_2' \dashv \Delta_2 & \text{dist. of subst.} \\
\Delta_1 \vdash e : [\Delta_1]\forall a.\sigma \leadsto \sigma_1' \to \sigma_2' \dashv \Delta_2 & \text{Rule }\textsc{a-inst-Forall}
\end{array}$$

– Rules D-INSTF-REFL and A-INSTF-FORALL: Similar to D-INST-.

□

D.4 Completeness of Typing with Restricted Existential Inference

Definition D.19. $\mathsf{min}_{\exists}(\Gamma \vdash e \Rightarrow \sigma) := \forall \sigma'.\ \Gamma \vdash e \Rightarrow \sigma' \implies |\exists(\sigma)| \leq |\exists(\sigma')|$

$\boxed{\Gamma \vdash e \Rightarrow_1 \sigma}$ *(Inference (1))*

D-I-ABS1

$$\frac{\overline{a}\ \mathsf{fresh} \qquad \begin{array}{c}\Gamma, x : \tau \vdash e \Rightarrow_1 \sigma \\ \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon\end{array} \qquad \begin{array}{c}\Gamma, x : \tau \vdash \sigma \leadsto_{\forall} \epsilon \\ \mathsf{min}_{\exists}(\Gamma \vdash \lambda x.\ e \Rightarrow \tau \to \exists\,\overline{a}.\epsilon')\end{array}}{\Gamma \vdash \lambda x.\ e \Rightarrow_1 \tau \to \exists\,\overline{a}.\epsilon'}$$

D-I-ABSA1

$$\frac{\overline{a}\ \mathsf{fresh} \qquad \begin{array}{c}\Gamma, x : \sigma_1 \vdash e \Rightarrow_1 \sigma \\ \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon\end{array} \qquad \begin{array}{c}\Gamma, x : \sigma_1 \vdash \sigma \leadsto_{\forall} \epsilon \\ \mathsf{min}_{\exists}(\Gamma \vdash \lambda x.\ e \Rightarrow \sigma_1 \to \exists\,\overline{a}.\epsilon')\end{array}}{\Gamma \vdash \lambda x : \sigma_1.\ e \Rightarrow_1 \sigma_1 \to \exists\,\overline{a}.\epsilon'}$$

D.4.1 Soundness

Definition D.20. *Define* $\sigma \leq_{\exists} \sigma'$ *as follows:*

- *If* $\sigma = \forall \overline{a}.\exists \overline{b}.\rho$ *and* $\sigma' = \forall \overline{a}'.\exists \overline{b}'.\rho'$ *such that* $\overline{a} \cup \overline{b} \cup \overline{a}' \cup \overline{b}' \neq \varnothing$*, then* $\sigma \leq_{\exists} \sigma' := (|\exists(\sigma)| \leq |\exists(\sigma')|) \wedge (\rho \leq_{\exists} \rho')$.
- *If* $\sigma = \sigma_1 \to \sigma_2$ *and* $\sigma' = \sigma_1' \to \sigma_2'$*, then* $\sigma \leq_{\exists} \sigma' := (\sigma_1 \leq_{\exists} \sigma_1') \wedge (\sigma_2 \leq_{\exists} \sigma_2')$.
- *Otherwise,* $\sigma \leq_{\exists} \sigma' := (|\exists(\sigma)| \leq |\exists(\sigma')|)$.

Lemma D.21 (Instantiation Preserves $\leq_{\exists}$)**.** *Suppose* $\sigma \leq_{\exists} \sigma'$.

a) If $\Gamma \vdash e : \sigma \leadsto \rho$ *and* $\Gamma \vdash e : \sigma' \leadsto \rho'$*, then* $\rho \leq_{\exists} \rho'$.
b) If $\Gamma \vdash \sigma \leadsto_{\forall} \epsilon$ *and* $\Gamma \vdash \sigma' \leadsto_{\forall} \epsilon'$*, then* $\epsilon \leq_{\exists} \epsilon'$.

Proof. a) There are two cases.

- Suppose $\sigma = \rho$ and $\sigma' = \rho'$.
  We have $\Gamma \vdash e : \rho \leadsto \rho$ and $\Gamma \vdash e : \rho' \leadsto \rho'$, therefore $\rho \leq_{\exists} \rho'$ by assumption.
- Suppose $\sigma = \forall \overline{a}.\exists \overline{b}.\rho_0$ and $\sigma' = \forall \overline{a}'.\exists \overline{b}'.\rho_0'$ such that $\overline{a} \cup \overline{b} \cup \overline{a}' \cup \overline{b}' \neq \varnothing$.
  We have $\sigma \leq_{\exists} \sigma'$ by assumption, which implies $\rho_0 \leq_{\exists} \rho_0'$ by the definition of $\leq_{\exists}$. Note that we have $\Gamma \vdash e : \sigma \leadsto \rho$ ($\Gamma \vdash e : \sigma' \leadsto \rho'$) such that ρ and ρ_0 (ρ' and ρ_0') have the same number and locations of $\exists$s. This is because substitution of monotypes cannot add or remove $\exists$s. Therefore, we can conclude that $\rho \leq_{\exists} \rho'$.

b) There are two cases.

- Suppose $\sigma = \epsilon$ and $\sigma' = \epsilon'$.
  We have $\Gamma \vdash \epsilon \leadsto_{\forall} \epsilon$ and $\Gamma \vdash \epsilon' \leadsto_{\forall} \epsilon'$, therefore $\epsilon \leq_{\exists} \epsilon'$ by assumption.
- Suppose $\sigma = \forall \overline{a}.\exists \overline{b}.\rho_0$ and $\sigma' = \forall \overline{a}'.\exists \overline{b}'.\rho_0'$ such that $\overline{a} \cup \overline{a}' \neq \varnothing$.
  We have $\sigma \leq_{\exists} \sigma'$ by assumption, which implies $|\exists(\exists \overline{b}.\rho_0)| \leq |\exists(\exists \overline{b}'.\rho_0')|$ and $\rho_0 \leq_{\exists} \rho_0'$ by the definition of $\leq_{\exists}$.

Similar to the above, we have $\Gamma \vdash \sigma \rightsquigarrow_{\forall} \exists\,\overline{b}.\rho$ ($\Gamma \vdash \sigma' \rightsquigarrow_{\forall} \exists\,\overline{b}'.\rho'$) such that ρ and ρ_0 (ρ' and ρ'_0) have the same number and locations of $\exists$s. This implies that $\rho \leq_{\exists} \rho'$.
Furthermore, $|\exists(\exists\,\overline{b}.\rho)| \leq |\exists(\exists\,\overline{b}'.\rho')|$ is equivalent to $\left|\overline{b}\right| \leq \left|\overline{b}'\right|$, which is equivalent to $|\exists(\exists\,\overline{b}.\rho_0)| \leq |\exists(\exists\,\overline{b}'.\rho'_0)|$ (true by assumption).
We have shown that $|\exists(\exists\,\overline{b}.\rho)| \leq |\exists(\exists\,\overline{b}'.\rho')|$.

□

Lemma D.22 (Global Deep $\mathsf{min}_{\exists}$)**.** *Suppose $\Gamma \vdash e \Rightarrow_1 \sigma$. Let Γ' be any context such that Γ and Γ' can only differ in monotype bindings of term variables, and let σ' be an arbitrary type such that $\Gamma' \vdash e \Rightarrow \sigma'$. Then we have $\sigma \leq_{\exists} \sigma'$.*

Proof. By induction on the given derivation $\Gamma \vdash e \Rightarrow_1 \sigma$.

- Rule D-I-INT:

$$\frac{\vdash \Gamma}{\Gamma \vdash n \Rightarrow_1 \mathsf{Int}}$$

  We can only have $\Gamma' \vdash e \Rightarrow \mathsf{Int}$ for any Γ', and $\sigma \leq_{\exists} \sigma$ holds vaccously.
- Rule D-I-VAR:

$$\frac{\vdash \Gamma \qquad x : \sigma \in \Gamma}{\Gamma \vdash x \Rightarrow_1 \sigma}$$

  If σ is a monotype, then $|\exists(\sigma)| = 0$ so the claim holds trivially.
  If σ is not a monotype, then we must also have $x : \sigma \in \Gamma'$ since Γ and Γ' can only differ on monotypes. Therefore we can only have $\Gamma' \vdash x \Rightarrow \sigma$, and $\sigma \leq_{\exists} \sigma$ holds vaccously.
- Rule D-I-ANN:

$$\frac{\Gamma \vdash e \Leftarrow \sigma \qquad \bullet \vdash \sigma}{\Gamma \vdash (e : \sigma) \Rightarrow_1 \sigma}$$

  We can only have $\Gamma' \vdash e \Rightarrow \sigma$ for any Γ', and $\sigma \leq_{\exists} \sigma$ holds vaccously.
- Rule D-I-ABS:

$$\frac{\overline{a}\ \mathsf{fresh} \qquad \begin{array}{c}\Gamma, x : \tau \vdash e \Rightarrow_1 \sigma \\ \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1\end{array} \qquad \begin{array}{c}\Gamma, x : \tau \vdash \sigma \rightsquigarrow_{\forall} \epsilon_1 \\ \mathsf{min}_{\exists}(\Gamma \vdash \lambda x.\ e \Rightarrow \tau \to \exists\,\overline{a}.\epsilon_2)\end{array}}{\Gamma \vdash \lambda x.\ e \Rightarrow_1 \tau \to \exists\,\overline{a}.\epsilon_2}$$

  Suppose the alternative derivation tree is as follows:

$$\frac{\Gamma', x : \tau' \vdash e \Rightarrow \sigma' \qquad \Gamma', x : \tau' \vdash \sigma' \rightsquigarrow_{\forall} \epsilon'_1 \qquad \overline{b}\ \mathsf{fresh} \qquad \epsilon'_2 = [\overline{b}/\lfloor \epsilon'_1 \rfloor_x]\epsilon'_1}{\Gamma' \vdash \lambda x.\ e \Rightarrow \tau' \to \exists\,\overline{b}.\epsilon'_2}$$

  Let $\epsilon_2 = \exists\,\overline{a}'.\rho$ and $\epsilon' = \exists\,\overline{b}.\rho'$. We need to show $\tau \to \exists\,\overline{a}.\epsilon_2 \leq_{\exists} \tau' \to \exists\,\overline{b}.\epsilon'_2$. By definition,

$$\begin{aligned}
\tau \to \exists\,\overline{a}.\epsilon_2 \leq_{\exists} \tau' \to \exists\,\overline{b}.\epsilon'_2 &= \tau \to \exists\,\overline{a}.\exists\,\overline{a}'.\rho \leq_{\exists} \tau' \to \exists\,\overline{b}.\exists\,\overline{b}'.\rho' \\
&= (\tau \leq_{\exists} \tau') \wedge (\exists\,\overline{a}.\exists\,\overline{a}'.\rho \leq_{\exists} \exists\,\overline{b}.\exists\,\overline{b}'.\rho') \\
&= (\tau \leq_{\exists} \tau') \wedge \big(|\exists(\exists\,\overline{a}.\epsilon_2)| \leq |\exists(\exists\,\overline{b}.\epsilon'_2)|\big) \wedge (\rho \leq_{\exists} \rho')
\end{aligned}$$

By the definition of $\mathsf{min}_\exists$, we can conclude $|\exists(\tau \to \exists\,\overline{a}.\epsilon)| \leq |\exists(\tau' \to \exists\,\overline{b}.\epsilon')|$ directly from the last premise of rule D-I-Abs1, which is equivalent to $|\exists(\exists\,\overline{a}.\epsilon)| \leq |\exists(\exists\,\overline{b}.\epsilon')|$. Moreover, $\tau \leq_\exists \tau'$ is true trivially. It remains to show $\rho \leq_\exists \rho'$.
Note that $\Gamma, x : \tau$ and $\Gamma', x : \tau'$ only differ in monotype bindings of term variables. Then the induction hypothesis applies to the first premise, so $\sigma \leq_\exists \sigma'$. Instantiation (Lemma D.21) and substitution of monotypes will both preserve this relation, therefore $\rho \leq_\exists \rho'$ is true.

- Rule D-I-AbsA:

$$\dfrac{\begin{array}{ccc} & \Gamma, x : \sigma_1 \vdash e \Rightarrow_1 \sigma & \Gamma, x : \sigma_1 \vdash \sigma \rightsquigarrow_\forall \epsilon_1 \\ \overline{a}\ \mathsf{fresh} & \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1 & \mathsf{min}_\exists(\Gamma \vdash \lambda x.\ e \Rightarrow \sigma_1 \to \exists\,\overline{a}.\epsilon_2) \end{array}}{\Gamma \vdash \lambda x : \sigma_1.\ e \Rightarrow_1 \sigma_1 \to \exists\,\overline{a}.\epsilon_2}$$

Suppose the alternative derivation tree is as follows:

$$\dfrac{\Gamma', x : \sigma_1 \vdash e \Rightarrow \sigma' \qquad \Gamma', x : \sigma_1 \vdash \sigma' \rightsquigarrow_\forall \epsilon_1' \qquad \overline{b}\ \mathsf{fresh} \qquad \epsilon_2' = [\overline{b}/\lfloor \epsilon_1' \rfloor_x]\epsilon_1'}{\Gamma' \vdash \lambda x : \sigma_1.\ e \Rightarrow \sigma_1 \to \exists\,\overline{b}.\epsilon_2'}$$

Let $\epsilon_2 = \exists\,\overline{a}'.\rho$ and $\epsilon' = \exists\,\overline{b}.\rho'$. We need to show $\sigma_1 \to \exists\,\overline{a}.\epsilon_2 \leq_\exists \sigma_1 \to \exists\,\overline{b}.\epsilon_2'$. By definition,

$$\begin{aligned} \sigma_1 \to \exists\,\overline{a}.\epsilon_2 \leq_\exists \sigma_1 \to \exists\,\overline{b}.\epsilon_2' &= \sigma_1 \to \exists\,\overline{a}.\exists\,\overline{a}'.\rho \leq_\exists \sigma_1 \to \exists\,\overline{b}.\exists\,\overline{b}'.\rho' \\ &= (\sigma_1 \leq_\exists \sigma_1) \land (\exists\,\overline{a}.\exists\,\overline{a}'.\rho \leq_\exists \exists\,\overline{b}.\exists\,\overline{b}'.\rho') \\ &= (\sigma_1 \leq_\exists \sigma_1) \land \big(|\exists(\exists\,\overline{a}.\epsilon_2)| \leq |\exists(\exists\,\overline{b}.\epsilon_2')|\big) \land (\rho \leq_\exists \rho') \end{aligned}$$

By the definition of $\mathsf{min}_\exists$, we can conclude $|\exists(\sigma_1 \to \exists\,\overline{a}.\epsilon)| \leq |\exists(\sigma_1 \to \exists\,\overline{b}.\epsilon')|$ directly from the last premise of rule D-I-AbsA1, which is equivalent to $|\exists(\exists\,\overline{a}.\epsilon)| \leq |\exists(\exists\,\overline{b}.\epsilon')|$. Moreover, $\sigma_1 \leq_\exists \sigma_1$ is true trivially. It remains to show $\rho \leq_\exists \rho'$.
By the induction hypothesis on the first premise, we have $\sigma \leq_\exists \sigma'$. Instantiation (Lemma D.21) and substitution of monotypes will both preserve this relation, therefore $\rho \leq_\exists \rho'$ is true.

- Rule D-I-App:

$$\dfrac{\Gamma \vdash e \Rightarrow_1 \sigma \qquad \Gamma \vdash e : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \qquad \Gamma \vdash e_1 \Leftarrow \sigma_1}{\Gamma \vdash e\, e_1 \Rightarrow_1 \sigma_2}$$

Suppose the alternative derivation tree is as follows:

$$\dfrac{\Gamma \vdash e \Rightarrow \sigma' \qquad \Gamma \vdash e : \sigma' \rightsquigarrow \sigma_1' \to \sigma_2' \qquad \Gamma \vdash e_1 \Leftarrow \sigma_1'}{\Gamma \vdash e\, e_1 \Rightarrow \sigma_2'}$$

We can write $\sigma = \forall\overline{a}.\exists\,\overline{b}.\sigma_3 \to \sigma_4$ and $\sigma' = \forall\overline{a}'.\exists\,\overline{b}'.\sigma_3' \to \sigma_4'$ by the definition of instantiation. By the induction hypothesis, we have $\sigma \leq_\exists \sigma'$, which implies $\sigma_3 \leq_\exists \sigma_3'$ and $\sigma_4 \leq_\exists \sigma_4'$. Instantiation preserves this relation by Lemma D.21, so we have $\sigma_1 \leq_\exists \sigma_1'$ and $\sigma_2 \leq_\exists \sigma_2'$.

□

Lemma D.23 (Global $\mathsf{min}_\exists$)**.** *If* $\varGamma \vdash e \Rightarrow_1 \sigma$, *then* $\varGamma \vdash e \Rightarrow \sigma$ *and* $\mathsf{min}_\exists(\varGamma \vdash e \Rightarrow \sigma)$.

Proof. Clearly $\varGamma \vdash e \Rightarrow \sigma$ holds since $\Rightarrow_1$ is a restricted form of $\Rightarrow$.

Lemma D.22 is a stronger version of the second part of this lemma. To see this, let σ' be an arbitrary type such that $\varGamma \vdash e \Rightarrow \sigma'$. Then by Lemma D.22 we have $\sigma \leq_\exists \sigma'$, which implies $|\exists(\sigma)| \leq |\exists(\sigma')|$. □

Definition D.24. $\lfloor \sigma \rfloor \coloneqq \{\lfloor e : \exists\, a.\epsilon \rfloor \mid \lfloor e : \exists\, a.\epsilon \rfloor$ *is a sub-expression of* $\sigma\}$

Definition D.25.

- $\mathsf{min}_{\lfloor\cdot\rfloor}(\varGamma, x : \tau \vdash e \Rightarrow \sigma) \coloneqq \forall \sigma', \tau'.\ \varGamma, x : \tau' \vdash e \Rightarrow \sigma' \implies (|\lfloor \sigma \rfloor_x| \leq |\lfloor \sigma' \rfloor_x|)$
- $\mathsf{min}_{\lfloor\cdot\rfloor}(\varGamma, x : \tau \vdash e : \sigma \rightsquigarrow \rho) \coloneqq \forall \rho', \tau'.\ \varGamma, x : \tau' \vdash e : \sigma \rightsquigarrow \rho' \implies (|\lfloor \rho \rfloor_x| \leq |\lfloor \rho' \rfloor_x|)$
- $\mathsf{min}_{\lfloor\cdot\rfloor}(\varGamma, x : \tau \vdash \sigma \rightsquigarrow_\forall \epsilon) \coloneqq \forall \epsilon', \tau'.\ \varGamma, x : \tau' \vdash \sigma \rightsquigarrow_\forall \epsilon' \implies (|\lfloor \epsilon \rfloor_x| \leq |\lfloor \epsilon' \rfloor_x|)$

Lemma D.26 ($\mathsf{min}_{\lfloor\cdot\rfloor}$ of Algorithmic Typing)**.** *Given* $\varDelta_2 \longrightarrow \varOmega$,

a) If $\varDelta_1 \vdash e \Rightarrow \sigma \dashv \varDelta_2$, *then* $[\varOmega]\varDelta_2 \vdash e \Rightarrow [\varOmega]\sigma$ *and* $\mathsf{min}_{\lfloor\cdot\rfloor}([\varOmega]\varDelta_2 \vdash e \Rightarrow [\varOmega]\sigma)$.
b) If $\varDelta_1 \vdash e : \sigma \rightsquigarrow \rho \dashv \varDelta_2$, *then* $[\varOmega]\varDelta_2 \vdash e : [\varOmega]\sigma \rightsquigarrow [\varOmega]\rho$, *and* $\mathsf{min}_{\lfloor\cdot\rfloor}([\varOmega]\varDelta_2 \vdash e : [\varOmega]\sigma \rightsquigarrow [\varOmega]\rho)$.
c) If $\varDelta_1 \vdash \sigma \rightsquigarrow_\forall \rho \dashv \varDelta_2$, *then* $[\varOmega]\varDelta_2 \vdash [\varOmega]\sigma \rightsquigarrow_\forall [\varOmega]\epsilon$, *and* $\mathsf{min}_{\lfloor\cdot\rfloor}([\varOmega]\varDelta_2 \vdash [\varOmega]\sigma \rightsquigarrow_\forall [\varOmega]\epsilon)$.

Proof. The corresponding declarative judgements hold by Theorem D.16.

The only declarative rules which are non-deterministic in terms of the number of projections in the output are rules D-I-ABS, D-INST-FORALL, and D-INSTF-FORALL. In our case (i.e. in the declarative judgements corresponding to algorithmic judgements), the guessed τs are equal to the solution of some unifiction variable $\hat{a}$ in $\varOmega$, which is unsolved in $\varDelta_2$. Then $[\varOmega]\hat{a}$ must not contain any x-projetions by the definition of $\longrightarrow$. Therefore no other guessed τ can lead to a lower number of x-projections. □

Theorem D.27 (Soundness of Algorithmic Inference (1))**.** *Given* $\varDelta_2 \longrightarrow \varOmega$, *if* $\varDelta_1 \vdash e \Rightarrow \sigma \dashv \varDelta_2$, *then* $[\varOmega]\varDelta_2 \vdash e \Rightarrow_1 [\varOmega]\sigma$.

Proof. By induction on the given derivation. We will omit some details provided in the proof of Theorem D.16.

- Rules A-I-INT, A-I-VAR, and A-I-ANN: Apply the corresponding declarative rule. The declarative rule is deterministic, therefore $\mathsf{min}_\exists$ is true trivially.
- Rule A-I-ABS:

$$\frac{\begin{array}{c}\varDelta_1, \hat{a}, x : \hat{a} \vdash e \Rightarrow \sigma \dashv \varDelta_2 \qquad \varDelta_2 \vdash \sigma \rightsquigarrow_\forall \epsilon \dashv \varDelta_3, x : \hat{a}, \varDelta_4 \\ \epsilon_1 = [\varDelta_3, x : \hat{a}, \varDelta_4]\epsilon \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1\end{array}}{\varDelta_1 \vdash \lambda x.\ e \Rightarrow \hat{a} \rightarrow \exists\, \overline{a}.\epsilon_2 \dashv \varDelta_3, \langle \varDelta_4 \rangle}$$

Let $\varOmega' = \varOmega_1, x : \hat{a}, |\varDelta_4|$ where $\varDelta_3, \langle \varDelta_4 \rangle \longrightarrow \varDelta_1, \varDelta_2$. Similarly to the proof of Theorem D.16:

$[\varOmega_1]\varDelta_3, x : [\varOmega']\hat{a} \vdash e \Rightarrow_1 [\varOmega_1]\sigma$ Theorem D.16
$[\varOmega_1]\varDelta_3, x : [\varOmega']\hat{a} \vdash [\varOmega_1]\sigma \rightsquigarrow_\forall [\varOmega_1]\epsilon$
$[\overline{a}/\lfloor [\varOmega_1]\epsilon \rfloor_x][\varOmega_1]\epsilon = [\varOmega_1]\epsilon_2$

We need to show $\mathsf{min}_{\exists}([\Omega_1]\Delta_3 \vdash \lambda x.\ e \Rightarrow [\Omega_1](\hat{a} \to \exists\,\overline{a}.\epsilon'))$. Consider the following arbitrary derivation tree for $\lambda x.\ e$:

$$\frac{\begin{array}{c}[\Omega_1]\Delta_3, x : \tau \vdash e \Rightarrow \sigma' \\ [\Omega_1]\Delta_3, x : \tau \vdash \sigma' \rightsquigarrow_{\forall} \epsilon'_1 \qquad \overline{b}\ \mathsf{fresh} \qquad \epsilon'_2 = [\overline{b}/\lfloor \epsilon'_1 \rfloor_x]\epsilon'_1\end{array}}{[\Omega_1]\Delta_3 \vdash \lambda x.\ e \Rightarrow \tau \to \exists\,\overline{b}.\epsilon'_2}$$

We need to show $|\exists([\Omega_1](\hat{a} \to \exists\,\overline{a}.\epsilon_2))| \leq |\exists(\tau \to \exists\,\overline{b}.\epsilon'_2)|$, which is equivalent to $|\exists([\Omega_1]\exists\,\overline{a}.\epsilon_2)| \leq |\exists(\exists\,\overline{b}.\epsilon'_2)|$.
By Lemma D.26, $\mathsf{min}_{\lfloor\cdot\rfloor}([\Omega_1]\Delta_3, x : [\Omega']\hat{a} \vdash [\Omega_1]\sigma \rightsquigarrow_{\forall} [\Omega_1]\epsilon)$, so $|\lfloor [\Omega_1]\epsilon \rfloor_x| \leq |\lfloor \epsilon'_1 \rfloor_x|$. This implies that $|\overline{a}| \leq |\overline{b}|$.
By Lemma D.22 applied to the first premise, we have $[\Omega_1]\sigma \leq_{\exists} \sigma'$, so $|\exists([\Omega_1]\epsilon)| \leq |\exists(\epsilon'_1)|$ by Lemma D.21. Substitution doesn't change the number of $\exists$s, so $|\exists([\Omega_1]\epsilon_2)| \leq |\exists(\epsilon'_2)|$.
We can conclude that $|\exists([\Omega_1]\exists\,\overline{a}.\epsilon_2)| = |\overline{a}| + |\exists([\Omega_1]\epsilon_2)| \leq |\overline{b}| + |\exists(\epsilon'_2)| = |\exists(\exists\,\overline{b}.\epsilon'_2)|$.

- Rule A-I-AbsA: Similar to rule A-I-Abs.
- Rule A-I-App:

$$\frac{\begin{array}{c}\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2 \\ \Delta_2 \vdash e : [\Delta_2]\rho \rightsquigarrow \sigma_1 \to \sigma_2 \dashv \Delta_3 \qquad \Delta_3 \vdash e_1 \Leftarrow [\Delta_3]\sigma_1 \dashv \Delta_4\end{array}}{\Delta_1 \vdash e\,e_1 \Rightarrow \sigma_2 \dashv \Delta_4}$$

Similarly to the proof of Theorem D.16:

$[\Omega]\Delta_4 \vdash e \Rightarrow_1 [\Omega]\sigma$	
$[\Omega]\Delta_4 \vdash e : [\Omega]\sigma \rightsquigarrow [\Omega]\sigma_1 \to [\Omega]\sigma_2$	
$[\Omega]\Delta_4 \vdash e_1 \Leftarrow [\Omega]\sigma_1$	
$[\Omega]\Delta_5 \vdash e\,e_1 \Rightarrow_1 [\Omega]\sigma_2$	Rule D-I-App

□

D.4.2 Completenesss

Theorem D.28 (Completeness of Algorithmic Typing (1))**.** *Given* $\Delta_1 \longrightarrow \Omega_1$,

a) If $[\Omega_1]\Delta_1 \vdash e \Rightarrow_1 \sigma$*, then there exist* Δ_2*,* Ω_2*, and* σ' *such that* $\Delta_2 \longrightarrow \Omega_2$*,* $\Omega_1 \longrightarrow \Omega_2$*,* $\sigma = [\Omega_2]\sigma'$*, and* $\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$.
b) If $[\Omega_1]\Delta_1 \vdash e \Leftarrow [\Omega_1]\sigma$ *where* $\Delta_1 \vdash \sigma$*, then there exist* Δ_2 *and* Ω_2 *such that* $\Delta_2 \longrightarrow \Omega_2$*,* $\Omega_1 \longrightarrow \Omega_2$*, and* $\Delta_1 \vdash e \Leftarrow [\Delta_1]\sigma \dashv \Delta_2$.

Proof. By induction on the given derivation.

- Rule D-I-Int: Apply A-I-Int.
- Rule D-I-Var:

$$\frac{x : \sigma \in [\Omega_1]\Delta_1}{[\Omega_1]\Delta_1 \vdash x \Rightarrow_1 \sigma}$$

$x : \sigma \in [\Omega_1]\Delta_1$	premise
$x : \sigma' \in \Delta_1$ s.t. $[\Omega_1]\sigma' = \sigma$	defn. of context application
$\Delta_1 \vdash x \Rightarrow \sigma' \dashv \Delta_1$	Rule A-I-Var

- Rule D-I-ABS:

$$\frac{\overline{a}\ \mathsf{fresh} \qquad \begin{array}{c}[\Omega_1]\Delta_1, x:\tau \vdash e \Rightarrow_1 \sigma \\ \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1\end{array} \qquad \begin{array}{c}[\Omega_1]\Delta_1, x:\tau \vdash \sigma \rightsquigarrow_\forall \epsilon_1 \\ \mathsf{min}_\exists([\Omega_1]\Delta_1 \vdash \lambda x.\ e \Rightarrow \tau \to \exists\,\overline{a}.\epsilon_2)\end{array}}{[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Rightarrow_1 \tau \to \exists\,\overline{a}.\epsilon_2}$$

$[\Omega_1, \hat{a} = \tau, x:\tau](\Delta_1, \hat{a}, x:\hat{a}) = [\Omega_1]\Delta_1, x:\tau$	defn. of app.
$[\Omega_1, \hat{a} = \tau, x:\tau](\Delta_1, \hat{a}, x:\hat{a}) \vdash e \Rightarrow_1 \sigma$	premise
$\Delta_1, \hat{a}, x:\hat{a} \vdash e \Rightarrow \sigma' \dashv \Delta_2$	i.h.
$\Delta_2 \longrightarrow \Omega_2$	i.h.
$\Omega_1, \hat{a} = \tau, x:\tau \longrightarrow \Omega_2$	i.h.
$\sigma = [\Omega_2]\sigma'$	i.h.

$[\Omega_1, \hat{a} = \tau, x:\tau](\Delta_1, \hat{a}, x:\hat{a}) = [\Omega_2]\Delta_2$	Lemma D.10(b, c)
$[\Omega_2]\Delta_2 \vdash [\Omega_2]\sigma' \rightsquigarrow_\forall \epsilon_1$	premise
$\Delta_2 \vdash [\Delta_2]\sigma' \rightsquigarrow_\forall \epsilon_0 \dashv \Delta_3'$	Theorem D.18
$\Delta_3' \longrightarrow \Omega_3'$	Theorem D.18
$\Omega_2 \longrightarrow \Omega_3'$	Theorem D.18
$\epsilon_1 = [\Omega_3']\epsilon_0$	Theorem D.18

$\Delta_1, \hat{a}, x:\hat{a} \longrightarrow \Delta_3'$	Theorem D.5
$\Delta_3' = \Delta_3, x:\tau_0, \Delta_4$ s.t. $\Delta_1 \longrightarrow \Delta_3$ and $[\Delta_3]\tau_0 = [\Delta_3]\hat{a}$	Lemma D.4
$\Delta_3, x:\tau_0, \Delta_4 \longrightarrow \Omega_3'$	above
$\Omega_3' = \Omega_3, x:\tau_0', \Omega_4$ s.t. $\Delta_3 \longrightarrow \Omega_3$	Lemma D.4
$\Omega_1, \hat{a} = \tau, x:\tau \longrightarrow \Omega_3, x:\tau_0', \Omega_4$	above
$\Omega_1 \longrightarrow \Omega_3$	Lemma D.4

$\Delta_1 \longrightarrow \Delta_3 \longrightarrow \Delta_3, \langle\Delta_4\rangle$	Rule A-EXT-ADDUNSOLVED
$\Delta_3, \langle\Delta_4\rangle \longrightarrow \Omega_3, \lvert\langle\Delta_4\rangle\rvert$	Lemma D.11
$\Omega_1 \longrightarrow \Omega_3, \lvert\langle\Delta_4\rangle\rvert$	Rule A-EXT-ADDSOLVED

Let $\epsilon_1' = [\Delta_3']\epsilon_0$.
Let $\epsilon_2' = [\overline{b}/\lfloor \epsilon_1' \rfloor_x]\epsilon_1'$.

$\Delta_1 \vdash \lambda x.\ e \Rightarrow \hat{a} \to \exists\,\overline{b}.\epsilon_2' \dashv \Delta_3, \langle\Delta_4\rangle$	Rule A-I-ABS

We want to show $\tau \to \exists\,\overline{a}.\epsilon_2 = [\Omega_3, \lvert\langle\Delta_4\rangle\rvert](\hat{a} \to \exists\,\overline{b}.\epsilon_2')$. By Lemma D.9(e), we know that $[\Omega_3, \lvert\langle\Delta_4\rangle\rvert]\hat{a} = [\Omega_1, \hat{a} = \tau, x:\tau]\hat{a} = \tau$. Then we need to show $\exists\,\overline{a}.\epsilon_2 = [\Omega_3, \lvert\langle\Delta_4\rangle\rvert](\exists\,\overline{b}.\epsilon_2')$.
We claim that ϵ_1 and ϵ_1' have the same set and locations of x-existential projections, since otherwise the $\mathsf{min}_\exists$ constraint would be contradicted.
Suppose for contradiction that ϵ_1 and ϵ_1' have different x-existential projections. From above, we know that $\epsilon_1 = [\Omega_3']\epsilon_0$ and $\epsilon_1' = [\Delta_3']\epsilon_0$. Therefore, there must exist (at least one) unification variable $\hat{b}$ such that $\hat{b}$ is unsolved in $\Delta_3' = \Delta_3, x:\tau_0, \Delta_4$ and solved in $\Omega_3' = \Omega_3, x:\tau, \Omega_4$ with a solution $\hat{b} = \tau'$ such that τ' contains an x-existential projection. Note that $\hat{b}$ is in Δ_4 and Ω_4 respectively, since it must appear after x.

Let Ω_4' be Ω_4 but with solution of variables appearing in $\langle\Delta_4\rangle$ replaced by Int. Note that $\Delta_3, x : \hat{a}, \Delta_4 \longrightarrow \Omega_3, x : \tau, \Omega_4'$ holds.

$\Delta_1, \hat{a}, x : \hat{a} \vdash e \Rightarrow \sigma' \dashv \Delta_2$ above
$\Delta_2 \longrightarrow \Omega_3, x : \tau, \Omega_4'$ Lemma D.3(b)
$[\Omega_3, x : \tau, \Omega_4']\Delta_2 \vdash e \Rightarrow_1 [\Omega_3, x : \tau, \Omega_4']\sigma'$ Theorem D.16

$\Delta_2 \vdash [\Delta_2]\sigma' \rightsquigarrow_\forall \epsilon_0 \dashv \Delta_3'$ above
$\Delta_3' \longrightarrow \Omega_3, x : \tau, \Omega_4'$ Lemma D.3(b)
$[\Omega_3, x : \tau, \Omega_4']\Delta_3' \vdash [\Omega_3, x : \tau, \Omega_4']\sigma' \rightsquigarrow_\forall [\Omega_3, x : \tau, \Omega_4']\epsilon_0$ Theorem D.16

Let $\epsilon_1'' = [\Omega_3, x : \tau, \Omega_4']\epsilon_0$.
Let $\epsilon_2'' = [\overline{c}/\lfloor \epsilon_1'' \rfloor_x]\epsilon_1''$.
$[\Omega_3, x{:}\tau, \Omega_4'](\Delta_3, x{:}\hat{a}, \Delta_4) = [\Omega_3, x{:}\tau, \Omega_4']\Delta_2 = [\Omega_3]\Delta_1 =$ Lemma D.10(b, c)
$[\Omega_1]\Delta_1$
$[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Rightarrow \tau \to \exists\,\overline{c}.\epsilon_2''$ Rule D-I-ABS

We have $\epsilon_1'' = [\Omega_3, x : \tau, \Omega_4']\epsilon_0$ and $\epsilon_1' = [\Delta_3, x : \tau_0, \Delta_4]\epsilon_0$. We know that all unsolved unifications variables in Δ_4 are solved by Int in Ω_4'. Therefore, it is not possible for ϵ_1'' and ϵ_1' to have different x-existential projections.
Then, by assumption, we have $|\lfloor \epsilon_1'' \rfloor_x| = |\lfloor \epsilon_1' \rfloor_x| < |\lfloor \epsilon_1 \rfloor_x|$, which implies $|\exists(\tau \to \exists\,\overline{c}.\epsilon_2'')| < |\exists(\tau \to \exists\,\overline{a}.\epsilon_2)|$. This contradicts the assumption that $\tau \to \exists\,\overline{a}.\epsilon_2$ has the minimum number of $\exists$s among the possible inferred types of $\lambda x.\ e$. We have shown that ϵ_1 and ϵ_1' have the same set and locations of x-existential projections.

$$\begin{aligned}
[\Omega_3, \Omega_5]\epsilon_2' &= [\Omega_3']\epsilon_2' && \Omega_3, \Omega_5 \subseteq \Omega_3' \\
&= [\Omega_3'][\overline{b}/\lfloor \epsilon_1' \rfloor_x]\epsilon_1' && \text{defn. of } \epsilon_2' \\
&= [\overline{b}/\lfloor \epsilon_1' \rfloor_x][\Omega_3']\epsilon_1' && (\star) \\
&= [\overline{b}/\lfloor \epsilon_1' \rfloor_x][\Omega_3'][\Delta_3']\epsilon_0 && \text{defn. of } \epsilon_1' \\
&= [\overline{b}/\lfloor \epsilon_1' \rfloor_x][\Omega_3']\epsilon_0 && \text{Lemma D.9(c)} \\
&= [\overline{b}/\lfloor \epsilon_1' \rfloor_x]\epsilon_1 && \text{above} \\
&= [\overline{b}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1 && (\star) \\
&= \epsilon_2 && \text{defn. of } \epsilon_2
\end{aligned}$$

Here $(\star)$ refers to the fact that ϵ_1 and ϵ_1' have the same set and locations of x-existential projections. Therefore, we have shown that $\tau \to \exists\,\overline{a}.\epsilon_2 = [\Omega_3, \Omega_5](\hat{a} \to \exists\,\overline{b}.\epsilon_2')$.

- Rule D-I-ABSA: Similar to rule D-I-ABS.
- Rule D-I-APP:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Rightarrow_1 \sigma \qquad [\Omega_1]\Delta_1 \vdash e : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \qquad [\Omega_1]\Delta_1 \vdash e_1 \Leftarrow \sigma_1}{[\Omega_1]\Delta_1 \vdash e\, e_1 \Rightarrow_1 \sigma_2}$$

$\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$	i.h.
$\Omega_1 \longrightarrow \Omega_2$	i.h.
$\Delta_2 \longrightarrow \Omega_2$	i.h.
$\sigma = [\Omega_2]\sigma'$	i.h.

$[\Omega_1]\Delta_1 = [\Omega_1]\Omega_1 = [\Omega_2]\Omega_2 = [\Omega_2]\Delta_2$	Lemma D.10(b, c)
$\sigma = [\Omega_2]\sigma' = [\Omega_2][\Delta_2]\sigma'$	Lemma D.9(c)
$[\Omega_2]\Delta_2 \vdash e : [\Omega_2][\Delta_2]\sigma' \rightsquigarrow \sigma_1 \to \sigma_2$	premise
$\Delta_2 \vdash e : [\Delta_2]\sigma' \rightsquigarrow \sigma_1' \to \sigma_2' \dashv \Delta_3$	Theorem D.18
$\Omega_2 \longrightarrow \Omega_3$	Theorem D.18
$\Delta_3 \longrightarrow \Omega_3$	Theorem D.18
$\sigma_1 \to \sigma_2 = [\Omega_3](\sigma_1' \to \sigma_2')$	Theorem D.18

$[\Omega_1]\Delta_1 = [\Omega_1]\Omega_1 = [\Omega_3]\Omega_3 = [\Omega_3]\Delta_3$	Lemma D.10(b, c)
$\sigma_1 = [\Omega_3]\sigma_1' = [\Omega_3][\Delta_3]\sigma_1'$	Lemma D.9(c)
$[\Omega_3]\Delta_3 \vdash e_1 \Leftarrow [\Omega_3][\Delta_3]\sigma_1'$	premise
$\Delta_3 \vdash e_1 \Leftarrow [\Delta_3]\sigma_1 \dashv \Delta_4$	i.h.
$\Omega_3 \longrightarrow \Omega_4$	i.h.
$\Delta_4 \longrightarrow \Omega_4$	i.h.

$\Omega_1 \longrightarrow \Omega_4$	Lemma D.3(b)
$\Delta_1 \vdash e\, e_1 \Rightarrow \sigma_2' \dashv \Delta_4$	Rule A-I-App
$\sigma_2 = [\Omega_3]\sigma_2' = [\Omega_4]\sigma_2'$	Lemma D.9(e)

- Rule D-I-Ann:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Leftarrow \sigma \qquad \bullet \vdash \sigma}{[\Omega_1]\Delta_1 \vdash (e : \sigma) \Rightarrow_1 \sigma}$$

σ doesn't contain any type variables	2nd premise
$\sigma = [\Delta]\sigma = [\Omega]\sigma$ for any Δ and Ω	above
$[\Omega_1]\Delta_1 \vdash e \Leftarrow [\Omega_1]\sigma$	1st premise

$\Delta_1 \vdash e \Leftarrow [\Delta_1]\sigma \dashv \Delta_2$	i.h.
$\Omega_1 \longrightarrow \Omega_2$	i.h.
$\Delta_2 \longrightarrow \Omega_2$	i.h.

$\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2$	above
$\Delta_1 \vdash (e : \sigma) \Rightarrow \sigma \dashv \Delta_2$	Rule A-I-Ann
$[\Omega_2]\sigma = \sigma$	above

- Rule D-C-Sub:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Rightarrow_1 \sigma \qquad [\Omega_1]\Delta_1 \vdash e : \sigma <: [\Omega_1]\rho}{[\Omega_1]\Delta_1 \vdash e \Leftarrow [\Omega_1]\rho}$$

$\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$ i.h.
$\Omega_1 \longrightarrow \Omega_2$ i.h.
$\Delta_2 \longrightarrow \Omega_2$ i.h.
$\sigma = [\Omega_2]\sigma'$ i.h.

$[\Omega_1]\Delta_1 = [\Omega_1]\Omega_1 = [\Omega_2]\Omega_2 = [\Omega_2]\Delta_2$ Lemma D.10(b, c)
$\sigma_1 = [\Omega_2]\sigma' = [\Omega_2][\Delta_2]\sigma'$ Lemma D.9(c)
$[\Omega_2]\rho = [\Omega_2][\Delta_2]\rho$ Lemma D.9(c)
$[\Omega_2]\Delta_2 \vdash e : [\Omega_2][\Delta_2]\sigma' <: [\Omega_2][\Delta_2]\rho$ premise
$\Delta_2 \vdash e : [\Delta_2]\sigma' <: [\Delta_2]\rho \dashv \Delta_3$ Theorem D.17
$\Omega_2 \longrightarrow \Omega_3$ Theorem D.17
$\Delta_3 \longrightarrow \Omega_3$ Theorem D.17
$\Delta_1 \vdash e \Leftarrow \rho \dashv \Delta_3$ Rule A-c-Sub

– Rule D-c-Abs:

$$\frac{[\Omega_1]\Delta_1, x : \sigma_1' \vdash e \Leftarrow \sigma_2'}{[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Leftarrow \sigma_1' \to \sigma_2'}$$

Above, we have $[\Omega_1]\rho = \sigma_1' \to \sigma_2'$. There are two possible values for $[\Delta_1]\rho$.

- $[\Delta_1]\rho = \sigma_1 \to \sigma_2$ where $\sigma_1' = [\Omega_1]\sigma_1$ and $\sigma_2' = [\Omega_1]\sigma_2$:

$$\frac{[\Omega_1]\Delta_1, x : [\Omega_1]\sigma_1 \vdash e \Leftarrow [\Omega_1]\sigma_2}{[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Leftarrow [\Omega_1]\sigma_1 \to [\Omega_1]\sigma_2}$$

$\Delta_1, x : [\Delta_1]\sigma_1 \longrightarrow \Omega_1, x : [\Omega_1]\sigma_1$ Rule A-ext-Solved
$[\Omega_1, x : [\Omega_1]\sigma_1](\Delta_1, x : [\Delta_1]\sigma_1) = [\Omega_1]\Delta_1, x : [\Omega_1]\sigma_1$ defn. of app.
$[\Omega_1]\sigma_2 = [\Omega_1, x : [\Omega_1]\sigma_1]\sigma_2$ defn. of subst.
$[\Omega_1, x : [\Omega_1]\sigma_1](\Delta_1, x : [\Delta_1]\sigma_1) \vdash e \Leftarrow [\Omega_1, x : [\Omega_1]\sigma_1]\sigma_2$ premise
$\Delta_1, x : [\Delta_1]\sigma_1 \vdash e \Leftarrow [\Delta_1, x : [\Delta_1]\sigma_1]\sigma_2 \dashv \Delta_2'$ i.h.
$\Omega_1, x : [\Omega_1]\sigma_1 \longrightarrow \Omega_2'$ i.h.
$\Delta_2' \longrightarrow \Omega_2'$ i.h.
$\Delta_1, x : [\Delta_1]\sigma_1 \vdash e \Leftarrow [\Delta_1]\sigma_2 \dashv \Delta_2'$ above

$\Delta_1, x : [\Delta_1]\sigma_1 \longrightarrow \Delta_2'$ Theorem D.5
$\Delta_2' = \Delta_2, x : \sigma_1', \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$ and $[\Delta_2]\sigma_1 = [\Delta_2]\sigma_1'$ Lemma D.4
$\Delta_2, x : \sigma_1', \Delta_3 \longrightarrow \Omega_2'$ above
$\Omega_2' = \Omega_2, x : \sigma_1'', \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$ and $[\Omega_2]\sigma_1' = [\Omega_2]\sigma_1''$ Lemma D.4
$\Omega_1, x : [\Omega_1]\sigma_1 \longrightarrow \Omega_2, x : \sigma_1'', \Omega_3$ above
$\Omega_1 \longrightarrow \Omega_2$ Lemma D.4
$\Delta_1 \vdash \lambda x.\ e \Leftarrow [\Delta_1](\sigma_1 \to \sigma_2) \dashv \Delta_2$ Rule A-c-Abs

- $[\Delta_1]\rho = \hat{a}$ where $[\Omega_1]\hat{a} = \tau_1 \to \tau_2$ (with $\tau_1 = \sigma_1'$ and $\tau_2 = \sigma_2'$):

$$\frac{[\Omega_1]\Delta_1, x : \tau_1 \vdash e \Leftarrow \tau_2}{[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Leftarrow \tau_1 \to \tau_2}$$

Let $\mathit{\Delta}_1' = \mathit{\Delta}_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1$.
Let $\mathit{\Omega}_1' = \mathit{\Omega}_1[\hat{a}_2 = \tau_2, \hat{a}_1 = \tau_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \tau_1$.
$[\mathit{\Omega}_1']\mathit{\Delta}_1' = [\mathit{\Omega}_1]\mathit{\Delta}_1$ defn.
$\tau_2 = [\mathit{\Omega}_1']\hat{a}_2$ defn.
$[\mathit{\Omega}_1']\mathit{\Delta}_1' \vdash e \Leftarrow [\mathit{\Omega}_1']\hat{a}_2$ above
$\mathit{\Delta}_1' \vdash e \Leftarrow \hat{a}_2 \dashv \mathit{\Delta}_2'$ i.h.
$\mathit{\Omega}_1' \longrightarrow \mathit{\Omega}_2'$ i.h.
$\mathit{\Delta}_2' \longrightarrow \mathit{\Omega}_2'$ i.h.

$\mathit{\Delta}_1' = \mathit{\Delta}_1, x : \hat{a}_1 \longrightarrow \mathit{\Delta}_2'$ Theorem D.5
$\mathit{\Delta}_2' = \mathit{\Delta}_2, x : \sigma_0, \mathit{\Delta}_3$ s.t. $\mathit{\Delta}_1 \longrightarrow \mathit{\Delta}_2$ and $[\mathit{\Delta}_2]\sigma_0 = [\mathit{\Delta}_2]\hat{a}_1$ Lemma D.4
$\mathit{\Delta}_2, x : \sigma_0, \mathit{\Delta}_3 \longrightarrow \mathit{\Omega}_2'$ above
$\mathit{\Omega}_2' = \mathit{\Omega}_2, x : \sigma_0', \mathit{\Omega}_3$ s.t. $\mathit{\Delta}_2 \longrightarrow \mathit{\Omega}_2$ and $[\mathit{\Omega}_2]\sigma_0' = [\mathit{\Omega}_2]\sigma_0$ Lemma D.4
$\mathit{\Omega}_1' = \mathit{\Omega}_1, x : \tau_1 \longrightarrow \mathit{\Omega}_2, x : \sigma_0', \mathit{\Omega}_3$ above
$\mathit{\Omega}_1 \longrightarrow \mathit{\Omega}_2$ Lemma D.4
$\mathit{\Delta}_1 \vdash e \Leftarrow \hat{a} \dashv \mathit{\Delta}_2$ Rule a-i-AbsUVar

- Rule d-c-AbsA:

$$\frac{[\mathit{\Omega}_1]\mathit{\Delta}_1, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_0 \qquad [\mathit{\Omega}_1]\mathit{\Delta}_1, x : \sigma_1' \vdash e \Leftarrow \sigma_2' \qquad \bullet \vdash \sigma_0}{[\mathit{\Omega}_1]\mathit{\Delta}_1 \vdash \lambda x : \sigma_0.\ e \Leftarrow \sigma_1' \to \sigma_2'}$$

Above, we have $[\mathit{\Omega}_1]\rho = \sigma_1' \to \sigma_2'$. There are two possible values for $[\mathit{\Delta}_1]\rho$.

- $[\mathit{\Delta}_1]\rho = \sigma_1 \to \sigma_2$ where $\sigma_1' = [\mathit{\Omega}_1]\sigma_1$ and $\sigma_2' = [\mathit{\Omega}_1]\sigma_2$:

$$\frac{\begin{array}{c}[\mathit{\Omega}_1]\mathit{\Delta}_1, x : [\mathit{\Omega}_1]\sigma_1 \vdash x : [\mathit{\Omega}_1]\sigma_1 <: \sigma_0 \\ [\mathit{\Omega}_1]\mathit{\Delta}_1, x : [\mathit{\Omega}_1]\sigma_1 \vdash e \Leftarrow [\mathit{\Omega}_1]\sigma_2 \qquad \bullet \vdash \sigma_0\end{array}}{[\mathit{\Omega}_1]\mathit{\Delta}_1 \vdash \lambda x : \sigma_0.\ e \Leftarrow [\mathit{\Omega}_1]\sigma_1 \to [\mathit{\Omega}_1]\sigma_2}$$

$\mathit{\Delta}_1, x : \sigma_1 \longrightarrow \mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1$ Rule a-ext-Solved
$[\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1](\mathit{\Delta}_1, x : \sigma_1) = [\mathit{\Omega}_1]\mathit{\Delta}_1, x : [\mathit{\Omega}_1]\sigma_1$ defn. of app.
$[\mathit{\Omega}_1]\sigma_1 = [\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1]\sigma_1$ defn. of subst.
$[\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1](\mathit{\Delta}_1, x : \sigma_1) \vdash x : [\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1]\sigma_1 <: [\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1]\sigma_0$ premise
$\mathit{\Delta}_1, x : \sigma_1 \vdash x : [\mathit{\Delta}_1, x : \sigma_1]\sigma_1 <: [\mathit{\Delta}_1, x : \sigma_1]\sigma_0 \dashv \mathit{\Delta}_2'$ Theorem D.17
$\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1 \longrightarrow \mathit{\Omega}_2'$ i.h.
$\mathit{\Delta}_2' \longrightarrow \mathit{\Omega}_2'$ i.h.
$\mathit{\Delta}_1, x : \sigma_1 \vdash x : [\mathit{\Delta}_1]\sigma_1 <: \sigma_0 \dashv \mathit{\Delta}_2'$ above

$[\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1](\mathit{\Delta}_1, x : \sigma_1) = [\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1](\mathit{\Omega}_1, x : [\mathit{\Omega}_1]\sigma_1)$
$= [\mathit{\Omega}_2']\mathit{\Omega}_2' = [\mathit{\Omega}_2']\mathit{\Delta}_2'$ Lemma D.10(b, c)
$[\mathit{\Omega}_1]\sigma_2 = [\mathit{\Omega}_2']\sigma_2 = [\mathit{\Omega}_2'][\mathit{\Delta}_2']\sigma_2$ Lemma D.9(c)
$[\mathit{\Omega}_2']\mathit{\Delta}_2' \vdash e \Leftarrow [\mathit{\Omega}_2'][\mathit{\Delta}_2']\sigma_2$ premise
$\mathit{\Delta}_2' \vdash e \Leftarrow [\mathit{\Delta}_2']\sigma_2 \dashv \mathit{\Delta}_3'$ i.h.
$\mathit{\Omega}_2' \longrightarrow \mathit{\Omega}_3'$ i.h.
$\mathit{\Delta}_3' \longrightarrow \mathit{\Omega}_3'$ i.h.

$\Delta_1, x : \sigma_1 \longrightarrow \Delta'_3$ Theorem D.5
$\Delta'_3 = \Delta_3, x : \sigma''_1, \Delta_4$ s.t. $\Delta_1 \longrightarrow \Delta_3$ and $[\Delta_3]\sigma_1 = [\Delta_3]\sigma''_1$ Lemma D.4
$\Delta_3, x : \sigma''_1, \Delta_3 \longrightarrow \Omega'_3$ above
$\Omega'_3 = \Omega_3, x : \sigma'''_1, \Omega_4$ s.t. $\Delta_3 \longrightarrow \Omega_3$ and $[\Omega_3]\sigma''_1 = [\Omega_3]\sigma'''_1$ Lemma D.4
$\Omega_1, x : [\Omega_1]\sigma_1 \longrightarrow \Omega_3, x : \sigma'''_1, \Omega_4$ above
$\Omega_1 \longrightarrow \Omega_3$ Lemma D.4
$\Delta_1 \vdash \lambda x : \sigma_0.\ e \Leftarrow [\Delta_1](\sigma_1 \to \sigma_2) \dashv \Delta_3$ Rule A-C-AbsA

- $[\Delta_1]\rho = \hat{a}$ where $[\Omega_1]\hat{a} = \tau_1 \to \tau_2$ (with $\tau_1 = \sigma'_1$ and $\tau_2 = \sigma'_2$):

$$\frac{[\Omega_1]\Delta_1, x : \tau_1 \vdash x : \tau_1 <: \sigma_0 \qquad [\Omega_1]\Delta_1, x : \tau_1 \vdash e \Leftarrow \tau_2 \qquad \bullet \vdash \sigma_0}{[\Omega_1]\Delta_1 \vdash \lambda x : \sigma_0.\ e \Leftarrow \tau_1 \to \tau_2}$$

Let $\Delta'_1 = \Delta_1[\hat{a}_2, \hat{a}_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \hat{a}_1$.
Let $\Omega'_1 = \Omega_1[\hat{a}_2 = \tau_2, \hat{a}_1 = \tau_1, \hat{a} = \hat{a}_1 \to \hat{a}_2], x : \tau_1$.
$[\Omega'_1]\Delta'_1 = [\Omega_1]\Delta_1$ defn.
$\tau_1 = [\Omega'_1]\hat{a}_1$ defn.
$\tau_2 = [\Omega'_1]\hat{a}_2$ defn.
σ_0 doesn't contain any type variables 3rd premise
$\sigma_0 = [\Delta]\sigma_0 = [\Omega]\sigma_0$ for any Δ and Ω above
$[\Omega'_1]\Delta'_1 \vdash x : [\Omega'_1]\hat{a}_1 <: [\Omega'_1]\sigma_0$ premise
$\Delta'_1 \vdash x : [\Delta'_1]\hat{a}_1 <: [\Delta'_1]\sigma_0 \dashv \Delta'_2$ i.h.
$\Omega'_1 \longrightarrow \Omega'_2$ i.h.
$\Delta'_2 \longrightarrow \Omega'_2$ i.h.
$\Delta'_1 \vdash x : \hat{a}_1 <: \sigma_0 \dashv \Delta'_2$ above

$[\Omega'_1]\Delta'_1 = [\Omega'_1]\Omega'_1 = [\Omega'_2]\Omega'_2 = [\Omega'_2]\Delta'_2$ Lemma D.10(b, c)
$\tau_2 = [\Omega'_1]\hat{a}_1 = [\Omega'_2]\hat{a}_2 = [\Omega'_2][\Delta'_2]\hat{a}_2$ Lemma D.9(c)
$[\Omega'_2]\Delta'_2 \vdash e \Leftarrow [\Omega'_2][\Delta'_2]\hat{a}_2$ premise
$\Delta'_2 \vdash e \Leftarrow [\Delta'_2]\hat{a}_2 \dashv \Delta'_3$ i.h.
$\Omega'_2 \longrightarrow \Omega'_3$ i.h.
$\Delta'_3 \longrightarrow \Omega'_3$ i.h.

$\Delta'_1 = \Delta_1, x : \hat{a}_1 \longrightarrow \Delta'_3$ Theorem D.5
$\Delta'_3 = \Delta_3, x : \tau'_1, \Delta_4$ s.t. $\Delta_1 \longrightarrow \Delta_3$ and $[\Delta_3]\hat{a}_1 = [\Delta_3]\tau'_1$ Lemma D.4
$\Delta_3, x : \tau'_1, \Delta_3 \longrightarrow \Omega'_3$ above
$\Omega'_3 = \Omega_3, x : \tau''_1, \Omega_4$ s.t. $\Delta_3 \longrightarrow \Omega_3$ and $[\Omega_3]\tau'_1 = [\Omega_3]\tau''_1$ Lemma D.4
$\Omega_1, x : \tau_1 \longrightarrow \Omega_3, x : \tau''_1, \Omega_4$ above
$\Omega_1 \longrightarrow \Omega_3$ Lemma D.4
$\Delta_1 \vdash \lambda x : \sigma_0.\ e \Leftarrow [\Delta_1]\hat{a} \dashv \Delta_3$ Rule A-C-AbsAUVar

– Rule D-C-Exists:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Leftarrow [\tau/b][\Omega_1]\epsilon \qquad [\Omega_1]\Delta_1 \vdash \tau}{[\Omega_1]\Delta_1 \vdash e \Leftarrow [\Omega_1]\exists\, b.\epsilon}$$

$\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \longrightarrow \Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau$	Rule A-EXT-SOLVE
$[\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau](\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b}) = [\Omega_1]\Delta_1$	defn. of app.
$[\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau]\epsilon = [\Omega_1]\epsilon$	defn. of subst.
$[\tau/b][\Omega_1]\epsilon = [\tau/b][\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau]\epsilon = [\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau][\hat{b}/b]\epsilon$	defn. of subst.
$[\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau](\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b}) \vdash e \Leftarrow [\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau][\hat{b}/b]\epsilon$	premise
$\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \vdash e \Leftarrow [\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b}][\hat{b}/b]\epsilon \dashv \Delta_2'$	i.h.
$\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau \longrightarrow \Omega_2'$	i.h.
$\Delta_2' \longrightarrow \Omega_2'$	i.h.
$[\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b}][\hat{b}/b]\epsilon = [\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b}][\hat{b}/b]\epsilon = [\hat{b}/b][\Delta_1]\epsilon$	defn. of subst.
$\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \vdash e \Leftarrow [\hat{b}/b][\Delta_1]\epsilon \dashv \Delta_2'$	above

$\Delta_1, \blacktriangleright_{\hat{b}}, \hat{b} \longrightarrow \Delta_2'$	Theorem D.5
$\Delta_2' = \Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$	Lemma D.4
$\Delta_2, \blacktriangleright_{\hat{b}}, \Delta_3 \longrightarrow \Omega_2'$	above
$\Omega_2' = \Omega_2, \blacktriangleright_{\hat{b}}, \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$	Lemma D.4
$\Omega_1, \blacktriangleright_{\hat{b}}, \hat{b} = \tau \longrightarrow \Omega_2, \blacktriangleright_{\hat{b}}, \Omega_3$	above
$\Omega_1 \longrightarrow \Omega_2$	Lemma D.4
$\Delta_1 \vdash e \Leftarrow [\Delta_1]\exists\, b.\epsilon \dashv \Delta_2$	Rule A-C-EXISTS

- Rule D-C-FORALL:

$$\frac{[\Omega_1]\Delta_1, a \vdash e \Leftarrow [\Omega_1]\sigma}{[\Omega_1]\Delta_1 \vdash e \Leftarrow [\Omega_1]\forall a.\sigma}$$

$\Delta_1, a \longrightarrow \Omega_1, a$	Rule A-EXT-TVAR
$[\Omega_1, a](\Delta_1, a) = [\Omega_1]\Delta_1$	defn. of app.
$[\Omega_1, a]\sigma = [\Omega_1]\sigma$	defn. of subst.
$[\Omega_1, a](\Delta_1, a) \vdash e \Leftarrow [\Omega_1, a]\sigma$	premise
$\Delta_1, a \vdash e \Leftarrow [\Delta_1, a]\sigma \dashv \Delta_2'$	i.h.
$\Omega_1, a \longrightarrow \Omega_2'$	i.h.
$\Delta_2' \longrightarrow \Omega_2'$	i.h.
$\Delta_1, a \vdash e \Leftarrow [\Delta_1]\sigma \dashv \Delta_2'$	above

$\Delta_1, a \longrightarrow \Delta_2'$	Theorem D.5
$\Delta_2' = \Delta_2, a, \Delta_3$ s.t. $\Delta_1 \longrightarrow \Delta_2$	Lemma D.4
$\Delta_2, a, \Delta_3 \longrightarrow \Omega_2'$	above
$\Omega_2' = \Omega_2, a, \Omega_3$ s.t. $\Delta_2 \longrightarrow \Omega_2$	Lemma D.4
$\Omega_1, a \longrightarrow \Omega_2, a, \Omega_3$	above
$\Omega_1 \longrightarrow \Omega_2$	Lemma D.4
$\Delta_1 \vdash e \Leftarrow [\Delta_1]\forall a.\sigma \dashv \Delta_2$	Rule A-C-FORALL

□

D.5 Completeness of Typing with Restricted Existential Instantiation

$\boxed{\Gamma \vdash e \Rightarrow_2 \sigma}$ *(Inference (2))*

D-I-App2

$$\frac{\Gamma \vdash e \Rightarrow_2 \sigma \qquad \mathsf{min}_\exists(\Gamma \vdash e \Rightarrow \sigma) \qquad \Gamma \vdash e : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \qquad \Gamma \vdash e_1 \Leftarrow_2 \sigma_1}{\Gamma \vdash e\, e_1 \Rightarrow_2 \sigma_2}$$

D-I-Ann2

$$\frac{\Gamma \vdash e \Leftarrow_2 \sigma \qquad \bullet \vdash \sigma}{\Gamma \vdash (e : \sigma) \Rightarrow_2 \sigma}$$

$\boxed{\Gamma \vdash e \Leftarrow_2 \sigma}$ *(Checking (2))*

D-C-Sub2

$$\frac{\Gamma \vdash e \Rightarrow_2 \sigma \qquad \mathsf{min}_\exists(\Gamma \vdash e \Rightarrow \sigma) \qquad \Gamma \vdash e : \sigma <: \rho}{\Gamma \vdash e \Leftarrow_2 \rho}$$

Theorem D.29 (Soundness of Algorithmic Inference (2))**.** *Given $\Delta_2 \longrightarrow \Omega$, if $\Delta_1 \vdash e \Rightarrow \sigma \dashv \Delta_2$, then $[\Omega]\Delta_2 \vdash e \Rightarrow_2 [\Omega]\sigma$.*

Proof. By Theorem D.27, we have $[\Omega]\Delta_2 \vdash e \Rightarrow_1 [\Omega]\sigma$. Then, by Lemma D.23, we have $\mathsf{min}_\exists([\Omega]\Delta_2 \vdash e \Rightarrow [\Omega]\sigma)$. This ensures that the $\mathsf{min}_\exists$ conditions in rules D-I-App2 and D-C-Sub is satisfied whenever encountered in the derivation of $[\Omega]\Delta_2 \vdash e \Rightarrow_2 [\Omega]\sigma$. The remaining parts of the proof are the same as Theorem D.16. ☐

$\boxed{\overline{c} \vdash \sigma_1 \lesssim \sigma_2}$ *(Compatibility)*

D-cmt-Refl

$$\frac{}{\overline{c} \vdash \sigma \lesssim \sigma}$$

D-cmt-Arr

$$\frac{\overline{c} \vdash \sigma_2 \lesssim \sigma_2'}{\overline{c} \vdash \sigma_1 \to \sigma_2 \lesssim \sigma_1 \to \sigma_2'}$$

D-cmt-Exists

$$\frac{\overline{c} \vdash [c/a]\epsilon_1 \lesssim [c/b]\epsilon_2}{c, \overline{c} \vdash \exists\, a.\epsilon_1 \lesssim \exists\, b.\epsilon_2}$$

D-cmt-ExistsR

$$\frac{\overline{c} \vdash \epsilon_1 \lesssim [c/a]\epsilon_2}{c, \overline{c} \vdash \epsilon_1 \lesssim \exists\, a.\epsilon_2}$$

Lemma D.30 ($|\exists|$ Inequality of Compatibility)**.** *If $\overline{c} \vdash \sigma_1 \lesssim \sigma_2$, then $|\exists(\sigma_1)| \leq |\exists(\sigma_2)|$. Moreover, $|\exists(\sigma_1)| = |\exists(\sigma_2)|$ if and only if $\sigma_1 = \sigma_2$.*

Proof. By induction on the given derivation.

- Rule D-cmt-Mono: We have $|\exists(\sigma_1)| = |\exists(\sigma_2)|$ and $\sigma_1 = \sigma_2$.
- Rules D-cmt-Arr and D-cmt-ExistsR: Apply induction hypothesis.
- Rule D-cmt-Exists:

$$\frac{\overline{c} \vdash \epsilon_1 \lesssim [c/a]\epsilon_2}{c, \overline{c} \vdash \epsilon_1 \lesssim \exists\, a.\epsilon_2}$$

  By the induction hypothesis, $|\exists(\epsilon_1)| \leq |\exists(\epsilon_2)|$, so $|\exists(\epsilon_1)| \leq |\exists(\epsilon_2)| < |\exists(\epsilon_2)| + 1 = |\exists(\exists\, a.\epsilon_2)|$. We have $|\exists(\exists\, a.\epsilon_1)| \neq |\exists(\epsilon_2)|$ and $\exists\, a.\epsilon_1 \neq \epsilon_2$.

□

Theorem D.31 (Completeness of Algorithmic Typing (2)). *Given* $\Delta_1 \longrightarrow \Omega_1$,

a) If $[\Omega_1]\Delta_1 \vdash e \Rightarrow_2 \sigma$, *then there exist* Δ_2, Ω_2, *and* σ' *and* $\overline{c} \notin \mathsf{dom}\,(\Omega_1) \cup \mathsf{dom}\,(\Delta_2)$, *such that* $\overline{c}, \Delta_2 \longrightarrow \Omega_2$, $\overline{c}, \Omega_1 \longrightarrow \Omega_2$, $\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$, *and* $\overline{c} \vdash [\Omega_2]\sigma' \lesssim \sigma$.

b) If $[\Omega_1]\Delta_1 \vdash e \Leftarrow_2 [\Omega_1]\sigma$ *where* $\Delta_1 \vdash \sigma$, *then there exist* Δ_2 *and* Ω_2 *such that* $\Omega_1 \longrightarrow \Omega_2$, $\Delta_2 \longrightarrow \Omega_2$, *and* $\Delta_1 \vdash e \Leftarrow [\Delta_1]\sigma \dashv \Delta_2$.

Proof. By induction on the given derivation.

- Rule D-I-INT: Apply A-I-INT. Rule D-CMT-REFL implies $\overline{c} \vdash \mathsf{Int} \lesssim \mathsf{Int}$.
- Rule D-I-VAR:

$$\frac{x : \sigma \in [\Omega_1]\Delta_1}{[\Omega_1]\Delta_1 \vdash x \Rightarrow_2 \sigma}$$

$$\begin{aligned}
&x : \sigma \in [\Omega_1]\Delta_1 && \text{premise}\\
&x : \sigma' \in \Delta_1 \text{ s.t. } [\Omega_1]\sigma' = \sigma && \text{defn. of context application}\\
&\Delta_1 \vdash x \Rightarrow \sigma' \dashv \Delta_1 && \text{Rule A-I-VAR}\\
&[\overline{c}, \Omega_1]\sigma' = [\Omega_1]\sigma' = \sigma && \text{defn. of subst.}\\
&\overline{c} \vdash [\overline{c}, \Omega_1]\sigma' \lesssim \sigma && \text{Rule D-CMT-VAR}
\end{aligned}$$

- Rule D-I-ABS2:

$$\frac{[\Omega_1]\Delta_1, x : \tau \vdash e \Rightarrow_2 \sigma \qquad [\Omega_1]\Delta_1, x : \tau \vdash \sigma \rightsquigarrow_\forall \epsilon_1 \qquad \overline{a}\ \mathsf{fresh} \qquad \epsilon_2 = [\overline{a}/\lfloor \epsilon_1 \rfloor_x]\epsilon_1}{[\Omega_1]\Delta_1 \vdash \lambda x.\ e \Rightarrow_2 \tau \rightarrow \exists\overline{a}.\epsilon_2}$$

$$\begin{aligned}
&[\Omega_1, \hat{a} = \tau, x : \tau](\Delta_1, \hat{a}, x : \hat{a}) = [\Omega_1]\Delta_1, x : \tau && \text{defn. of substitution}\\
&[\Omega_1, \hat{a} = \tau, x : \tau](\Delta_1, \hat{a}, x : \hat{a}) \vdash e \Rightarrow_2 \sigma && \text{premise}\\
&\Delta_1, \hat{a}, x : \hat{a} \vdash e \Rightarrow \sigma' \dashv \Delta_2 && \text{i.h.}\\
&\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau \longrightarrow \Omega_2 && \text{i.h.}\\
&\overline{c}, \Delta_2 \longrightarrow \Omega_2 && \text{i.h.}\\
&\overline{c} \vdash [\Omega_2]\sigma' \lesssim \sigma && \text{i.h.}
\end{aligned}$$

$$\begin{aligned}
&[\Omega_2](\overline{c}, \Delta_2) = [\Omega_2]\Omega_2 && \text{Lemma D.10(b, c)}\\
&\qquad = [\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau](\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau) &&\\
&\qquad = [\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau](\overline{c}, \Delta_1, \hat{a}, x : \hat{a}) &&\\
&[\Omega_1, \hat{a} = \tau, x : \tau](\Delta_1, \hat{a}, x : \hat{a}) \vdash \sigma \rightsquigarrow_\forall \epsilon_1 && \text{premise}\\
&[\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau](\overline{c}, \Delta_1, \hat{a}, x : \hat{a}) \vdash \sigma \rightsquigarrow_\forall \epsilon_1 && \text{weakening}\\
&[\Omega_2](\overline{c}, \Delta_2) \vdash \sigma \rightsquigarrow_\forall \epsilon_1 && \text{above}\\
&\overline{c}, \Delta_2 \vdash [\overline{c}, \Delta_2]\sigma' \rightsquigarrow_\forall \epsilon_1' \dashv \overline{c}, \Delta_3' && \text{i.h.}\\
&\Omega_2 \longrightarrow \Omega_3' && \text{i.h.}\\
&\overline{c}, \Delta_3' \longrightarrow \Omega_3' && \text{i.h.}\\
&\overline{c} \vdash [\Omega_3']\epsilon_1' \lesssim \epsilon_1 && \text{i.h.}\\
&\Delta_2 \vdash [\Delta_2]\sigma' \rightsquigarrow_\forall \epsilon_1' \dashv \Delta_3' && \overline{c} \notin \mathsf{dom}\,(\Delta_2)
\end{aligned}$$

$\Delta_1, \hat{a}, x : \hat{a} \longrightarrow \Delta_3'$	Theorem D.5
$\Delta_3' = \Delta_3, x : \tau_0, \Delta_4$ s.t. $\Delta_1 \longrightarrow \Delta_1, \hat{a} \longrightarrow \Delta_3$ and $[\Delta_3]\tau_0 = [\Delta_3]\hat{a}$	Lemma D.4
$\overline{c}, \Delta_3, x : \tau_0, \Delta_4 \longrightarrow \Omega_3'$	above
$\Omega_3' = \overline{c}, \Omega_3, x : \tau_0', \Omega_4$ s.t. $\Delta_3 \longrightarrow \Omega_3$ and $[\Delta_3]\tau_0 = [\Delta_3]\tau_0'$	Lemma D.4
$\overline{c}, \Omega_1, \hat{a} = \tau, x : \tau \longrightarrow \Omega_3, x : \tau_0', \Omega_4$	above
$\overline{c}, \Omega_1 \longrightarrow \overline{c}, \Omega_1, \hat{a} = \tau \longrightarrow \Omega_3$	Lemma D.4

$\Delta_1 \longrightarrow \Delta_1, \hat{a} \longrightarrow \Delta_3 \longrightarrow \Delta_3, \langle\Delta_4\rangle$	Rule A-EXT-AddUnsolved
Let Ω_4' consist of the the solutions of $\langle\Delta_4\rangle$ in Ω_4.	
$\overline{c}, \Delta_3, \langle\Delta_4\rangle \longrightarrow \Omega_3, \Omega_4'$	defn.
$\overline{c}, \Omega_1 \longrightarrow \Omega_1, \hat{a} = \tau \longrightarrow \Omega_3, \Omega_4'$	Rule A-EXT-AddSolved

Let $\epsilon_1'' = [\Delta_3']\epsilon_1'$.	
$\overline{c} \vdash [\Omega_3']\epsilon_1'' \lesssim \epsilon_1$	above
Let $\epsilon_2' = [\overline{b}/\lfloor\epsilon_1''\rfloor_x]\epsilon_1''$.	
$\Delta_1 \vdash \lambda x.\ e \Rightarrow \hat{a} \rightarrow \exists\,\overline{b}.\epsilon_2' \dashv \Delta_3, \langle\Delta_4\rangle$	Rule A-I-Abs

It remains to show that $\Omega_3, \Omega_4'' \vdash [\Omega_3, \Omega_4''](\hat{a} \rightarrow \exists\,\overline{b}.\epsilon_2') <: \tau \rightarrow \exists\,\overline{a}.\epsilon_2$ for some Ω_4'' such that $\overline{c}, \Delta_3, \langle\Delta_4\rangle \longrightarrow \Omega_3, \Omega_4''$ and $\overline{c}, \Omega_1 \longrightarrow \Omega_3, \Omega_4''$.

Given $\Omega_3' \vdash [\Omega_3']\epsilon_1'' <: \epsilon_1$, we know $[\Omega_3']\epsilon_1''$ has the same set and locations of x existential projections as ϵ_1. ϵ_1'' has a (non-strict) subset of the occurrences and locations of x-existential projections of $[\Omega_3']\epsilon_1''$ (and of ϵ_1). By alpha equivalence, WLOG we can assume that $\overline{b} \subseteq \overline{a}$. Let $\overline{a}'$ be the remaining variables in $\overline{a}$. In the $\lesssim$ judgmenet, $\overline{a}$ are replaced by type variables $\overline{c}$. Let $\overline{c}' \subseteq \overline{c}$ be the variables corresponding to $\overline{a}'$.

Recall that $\Omega_3' = \Delta_3, x : \tau_0, \Omega_4$, so if there are any additional x-projections in $[\Omega_3']\epsilon_1''$ as compared to ϵ_1'' (i.e. projections corresponding to $\overline{a}'$), they must be introduced in Ω_4. Let Ω_4'' be Ω_4' with these projections substituted by the corresponding variable in $\overline{c}'$ (which also corresponds to a variable in $\overline{a}'$). Note that $\overline{c}, \Delta_3, \langle\Delta_4\rangle \longrightarrow \Omega_3, \Omega_4''$ and $\overline{c}, \Omega_1 \longrightarrow \Omega_3, \Omega_4''$ hold.

Given the above, by starting from $\overline{c} \vdash [\Omega_3']\epsilon_1'' \lesssim \epsilon_1$ and applying rules A-CMT-Exists and A-CMT-ExistsR repeatedly as appropriate, we can conclude that $\overline{c} \vdash [\Omega_3, \Omega_4''](\exists\,\overline{b}.\epsilon_2') \lesssim \exists\,\overline{a}.\epsilon_2$.

By the definition of extension, $[\Omega_3, \Omega_4'']\hat{a} = [\Omega_1, \hat{a} = \tau]\hat{a} = \tau$. Then, by Rule D-CMT-Arr, we have $\overline{c} \vdash [\Omega_3, \Omega_4''](\hat{a} \rightarrow \exists\,\overline{b}.\epsilon_2') \lesssim \tau \rightarrow \exists\,\overline{a}.\epsilon_2$.

- Rule D-I-App2:

$$\frac{\begin{array}{c}[\Omega_1]\Delta_1 \vdash e \Rightarrow_2 \sigma \qquad \mathsf{min}_\exists([\Omega_1]\Delta_1 \vdash e \Rightarrow \sigma) \\ [\Omega_1]\Delta_1 \vdash e : \sigma \rightsquigarrow \sigma_1 \rightarrow \sigma_2 \qquad [\Omega_1]\Delta_1 \vdash e_1 \Leftarrow_2 \sigma_1\end{array}}{[\Omega_1]\Delta_1 \vdash e\,e_1 \Rightarrow_2 \sigma_2}$$

$\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$	i.h.
$\bar{c}, \Omega_1 \longrightarrow \Omega_2$	i.h.
$\bar{c}, \Delta_2 \longrightarrow \Omega_2$	i.h.
$\bar{c} \vdash [\Omega_2]\sigma' \lesssim \sigma$	i.h.
$\lvert\exists(\sigma)\rvert \geq \lvert\exists([\Omega_2]\sigma')\rvert$	Lemma D.30

$[\Omega_2]\Delta_2 \vdash e \Rightarrow [\Omega_2]\sigma'$	Theorem D.16
$[\bar{c}, \Omega_1](\bar{c}, \Delta_1) = [\bar{c}, \Omega_1](\bar{c}, \Omega_1) = [\Omega_2]\Omega_2 = [\Omega_2](\bar{c}, \Delta_2)$	Lemma D.10(b, c)
$[\bar{c}, \Omega_1](\bar{c}, \Delta_1) \vdash e \Rightarrow [\Omega_2]\sigma'$	above
$[\Omega_1]\Delta_1 \vdash e \Rightarrow [\Omega_2]\sigma'$	$\bar{c} \notin \mathsf{dom}\,(\Delta_1)$
$\mathsf{min}_\exists([\Omega_1]\Delta_1 \vdash e \Rightarrow \sigma)$	premise
$\lvert\exists(\sigma)\rvert \leq \lvert\exists([\Omega_2]\sigma')\rvert$	defn. of $\mathsf{min}_\exists$
$\lvert\exists(\sigma)\rvert = \lvert\exists([\Omega_2]\sigma')\rvert$	above
$\sigma = [\Omega_2]\sigma'$	Lemma D.30

$[\Omega_1]\Delta_1 \vdash e : [\Omega_2]\sigma' \rightsquigarrow \sigma_1 \to \sigma_2$	premise
$[\bar{c}, \Omega_1](\bar{c}, \Delta_1) \vdash e : [\Omega_2]\sigma' \rightsquigarrow \sigma_1 \to \sigma_2$	weakening
$[\Omega_2](\bar{c}, \Delta_2) \vdash e : [\Omega_2]\sigma' \rightsquigarrow \sigma_1 \to \sigma_2$	above
$\bar{c}, \Delta_2 \vdash e : [\Delta_2]\sigma' \rightsquigarrow \sigma_1' \to \sigma_2' \dashv \bar{c}, \Delta_3$	Theorem D.18
$\Omega_2 \longrightarrow \Omega_3$	Theorem D.18
$\bar{c}, \Delta_3 \longrightarrow \Omega_3$	Theorem D.18
$\sigma_1 \to \sigma_2 = [\Omega_3](\sigma_1' \to \sigma_2')$	Theorem D.18
$\Delta_2 \vdash e : [\Delta_2]\sigma' \rightsquigarrow \sigma_1' \to \sigma_2' \dashv \Delta_3$	$\bar{c} \notin \mathsf{dom}\,(\Delta_2)$

$[\bar{c}, \Omega_1](\bar{c}, \Delta_1) = [\bar{c}, \Omega_1](\bar{c}, \Omega_1) = [\Omega_3]\Omega_3 = [\Omega_3](\bar{c}, \Delta_3)$	Lemma D.10(b, c)
$\sigma_1 = [\Omega_3]\sigma_1' = [\Omega_3][\bar{c}, \Delta_3]\sigma_1' = [\Omega_3][\Delta_3]\sigma_1'$	Lemma D.9(c)
$[\Omega_1]\Delta_1 \vdash e_1 \Leftarrow_2 [\Omega_3][\Delta_3]\sigma_1'$	premise
$[\bar{c}, \Omega_1](\bar{c}, \Delta_1) \vdash e_1 \Leftarrow_2 [\Omega_3][\Delta_3]\sigma_1'$	weakening
$[\Omega_3](\bar{c}, \Delta_3) \vdash e_1 \Leftarrow_2 [\Omega_3][\Delta_3]\sigma_1'$	above
$\bar{c}, \Delta_3 \vdash e_1 \Leftarrow [\Delta_3]\sigma_1' \dashv \bar{c}, \Delta_4$	i.h.
$\Omega_3 \longrightarrow \Omega_4$	i.h.
$\bar{c}, \Delta_4 \longrightarrow \Omega_4$	i.h.
$\Delta_3 \vdash e_1 \Leftarrow [\Delta_3]\sigma_1' \dashv \Delta_4$	$\bar{c} \notin \mathsf{dom}\,(\Delta_3)$

$\Delta_1 \vdash e\, e_1 \Rightarrow \sigma_2' \dashv \Delta_4$	Rule A-I-App
$[\Omega_4]\sigma_2' = [\Omega_3]\sigma_2' = \sigma_2$	Lemma D.9(e)
$\bar{c} \vdash [\Omega_4]\sigma_2' \lesssim \sigma_2$	Rule D-cmt-Refl

- Rule D-I-Ann2:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Leftarrow_2 \sigma \qquad \bullet \vdash \sigma}{[\Omega_1]\Delta_1 \vdash (e : \sigma) \Rightarrow_2 \sigma}$$

σ doesn't contain any type variables	2nd premise
$\sigma = [\Delta_1]\sigma = [\Omega_1]\sigma$	above
$[\Omega_1]\Delta_1 \vdash e \Leftarrow_2 [\Omega_1]\sigma$	1st premise
$\Delta_1 \vdash e \Leftarrow [\Delta_1]\sigma \dashv \Delta_2$	i.h.
$\Omega_1 \longrightarrow \Omega_2$	i.h.
$\Delta_2 \longrightarrow \Omega_2$	i.h.
$\Delta_1 \vdash e \Leftarrow \sigma \dashv \Delta_2$	above
$\Delta_1 \vdash (e : \sigma) \Rightarrow \sigma \dashv \Delta_2$	Rule A-I-Ann
$[\overline{c}, \Omega_2]\sigma = \sigma$	above
$\overline{c} \vdash [\Omega_2]\sigma \lesssim \sigma$	Rule D-CMT-Refl

- Rule D-C-Sub2:

$$\frac{[\Omega_1]\Delta_1 \vdash e \Rightarrow_2 \sigma \qquad \mathsf{min}_\exists([\Omega_1]\Delta_1 \vdash e \Rightarrow \sigma) \qquad [\Omega_1]\Delta_1 \vdash e : \sigma <: [\Omega_1]\rho}{[\Omega_1]\Delta_1 \vdash e \Leftarrow_2 [\Omega_1]\rho}$$

$\Delta_1 \vdash e \Rightarrow \sigma' \dashv \Delta_2$	i.h.
$\overline{c}, \Delta_2 \longrightarrow \Omega_2$	i.h.
$\overline{c}, \Omega_1 \longrightarrow \Omega_2$	i.h.
$\overline{c} \vdash [\Omega_2]\sigma' \lesssim \sigma$	i.h.
$\lvert\exists(\sigma)\rvert \geq \lvert\exists([\Omega_2]\sigma')\rvert$	Lemma D.30
$[\Omega_2]\Delta_2 \vdash e \Rightarrow [\Omega_2]\sigma'$	Theorem D.16
$[\overline{c}, \Omega_1](\overline{c}, \Delta_1) = [\overline{c}, \Omega_1](\overline{c}, \Omega_1) = [\Omega_2]\Omega_2 = [\Omega_2](\overline{c}, \Delta_2)$	Lemma D.10(b, c)
$[\overline{c}, \Omega_1](\overline{c}, \Delta_1) \vdash e \Rightarrow [\Omega_2]\sigma'$	above
$[\Omega_1]\Delta_1 \vdash e \Rightarrow [\Omega_2]\sigma'$	$\overline{c} \notin \mathsf{dom}\,([\Omega_1]\Delta_1)$
$\mathsf{min}_\exists([\Omega_1]\Delta_1 \vdash e \Rightarrow \sigma)$	premise
$\lvert\exists(\sigma)\rvert \leq \lvert\exists([\Omega_2]\sigma')\rvert$	defn. of $\mathsf{min}_\exists$
$\lvert\exists(\sigma)\rvert = \lvert\exists([\Omega_2]\sigma')\rvert$	above
$\sigma = [\Omega_2]\sigma'$	Lemma D.30
$[\Omega_1]\rho = [\Omega_2]\rho$	Lemma D.9(e)
$[\Omega_1]\Delta_1 \vdash e : [\Omega_2]\sigma' <: [\Omega_2]\rho$	premise
$[\overline{c}, \Omega_1](\overline{c}, \Delta_1) \vdash e : [\Omega_2]\sigma' <: [\Omega_2]\rho$	weakening
$[\Omega_2](\overline{c}, \Delta_2) \vdash e : [\Omega_2]\sigma' <: [\Omega_2]\rho$	above
$\overline{c}, \Delta_2 \vdash e : [\Delta_2]\sigma' <: [\Delta_2]\rho \dashv \overline{c}, \Delta_3$	Theorem D.17
$\overline{c}, \Delta_3 \longrightarrow \Omega_3$	Theorem D.17
$\Omega_2 \longrightarrow \Omega_3$	Theorem D.17
$\Delta_2 \vdash e : [\Delta_2]\sigma' <: [\Delta_2]\rho \dashv \Delta_3$	$\overline{c} \notin \mathsf{dom}\,(\Delta_2)$
$[\Omega_2]\rho = [\Omega_2][\Delta_1]\rho$	Lemma D.9(c)
$\Delta_2 \vdash e : [\Delta_2]\sigma' <: [\Delta_2][\Delta_1]\rho \dashv \Delta_3$	above
$\Delta_1 \vdash e \Leftarrow [\Delta_1]\rho \dashv \Delta_3$	rule A-C-Sub

- Rules D-C-Abs, D-C-Exists, and D-C-Forall: Proof similar to Theorem D.28.

□

E Core Language

The core calculus, presented below, is the same as the one in Eisenberg et al. [2], which is proven to be type-safe through syntactic type soundness [3]. Expressions e are explicitly typed, and thus include type abstractions $\Lambda a.\ \mathrm{e}$ and applications $\mathrm{e}\,\mathrm{t}$, explicit packing $\mathsf{pack}\,\mathrm{t}_1, \mathrm{e}\ \mathsf{as}\ \mathrm{t}_2$ and opening $\mathsf{open}\,\mathrm{e}$, and casted expressions $\mathrm{e} \rhd \gamma$ with type coercions γ. Types t include, among other forms, polymorphic and existential types.[1] We refer interested readers to Eisenberg et al. [2] for more details.

$$
\begin{array}{lll}
\text{type} & \mathrm{t} & := a \mid \mathsf{Int} \mid \mathrm{t}_1 \to \mathrm{t}_2 \mid \forall a.\mathrm{t} \mid \exists\, a.\mathrm{t} \mid \lfloor \mathrm{e} \rfloor \\
\text{expression} & \mathrm{e} & := n \mid x \mid \lambda x : \mathrm{t}.\ \mathrm{e} \mid \mathrm{e}_1\,\mathrm{e}_2 \mid \Lambda a.\ \mathrm{e} \mid \mathrm{e}\,\mathrm{t} \mid \mathsf{pack}\,\mathrm{t}_1, \mathrm{e}\ \mathsf{as}\ \mathrm{t}_2 \\
 & & \quad \mid \mathsf{open}\,\mathrm{e} \mid \mathrm{e} \rhd \gamma \\
\text{value} & v & := n \mid \lambda x : \mathrm{t}.\ \mathrm{e} \mid \Lambda a.\ v \mid \mathsf{pack}\,\mathrm{t}_1, v\ \mathsf{as}\ \mathrm{t}_2 \mid \mathsf{pack}\,\mathrm{t}_1, v \rhd \gamma\ \mathsf{as}\ \mathrm{t}_2 \\
\text{type coercion} & \gamma & := \langle \mathrm{t} \rangle \mid \mathsf{sym}\,\gamma \mid \gamma_1 ; ; \gamma_2 \mid \lfloor \eta \rfloor \mid \gamma_1 @@ \gamma_2 \mid \mathsf{projpack}\,\mathrm{t}_1, \mathrm{e}\ \mathsf{as}\ \mathrm{t}_2 \\
 & & \quad \mid \gamma_1 \to \gamma_2 \mid \forall a.\gamma \mid \exists\, a.\gamma \mid \mathsf{fst}\,\gamma \mid \mathsf{snd}\,\gamma \\
\text{expression coercion} & \eta & := \mathrm{e} \rhd \gamma \mid \mathsf{step}\,\mathrm{e} \\
\text{context} & \mathrm{G} & := \bullet \mid x : \mathrm{t} \mid a \mid \mathrm{G}_1, \ldots, \mathrm{G}_n
\end{array}
$$

$\boxed{\vdash \mathrm{G}}$ *(Context Well-Formedness)*

$$
\textsc{c-cwf-Empty}\ \frac{}{\vdash \bullet}
\qquad
\textsc{c-cwf-Var}\ \frac{\mathrm{G} \vdash \mathrm{t} \qquad x \notin \mathsf{dom}\,(\mathrm{G})}{\vdash \mathrm{G}, x : \mathrm{t}}
\qquad
\textsc{c-cwf-TVar}\ \frac{\vdash \mathrm{G} \qquad a \notin \mathsf{dom}\,(\mathrm{G})}{\vdash \mathrm{G}, a}
$$

$\boxed{\mathrm{G} \vdash \mathrm{t}}$ *(Type Well-Formedness)*

$$
\textsc{c-wf-Int}\ \frac{\vdash \mathrm{G}}{\mathrm{G} \vdash \mathsf{Int}}
\qquad
\textsc{c-wf-Var}\ \frac{\vdash \mathrm{G} \qquad a \in \mathrm{G}}{\mathrm{G} \vdash a}
\qquad
\textsc{c-wf-Arr}\ \frac{\mathrm{G} \vdash \mathrm{t}_1 \qquad \mathrm{G} \vdash \mathrm{t}_2}{\mathrm{G} \vdash \mathrm{t}_1 \to \mathrm{t}_2}
\qquad
\textsc{c-wf-Exists}\ \frac{\mathrm{G}, a \vdash \mathrm{t}}{\mathrm{G} \vdash \exists\, a.\mathrm{t}}
$$

$$
\textsc{c-wf-Forall}\ \frac{\mathrm{G}, a \vdash \mathrm{t}}{\mathrm{G} \vdash \forall a.\mathrm{t}}
\qquad
\textsc{c-wf-Proj}\ \frac{\vdash \mathrm{G} \qquad \mathsf{fv}\,(\mathrm{e}) \subseteq \mathsf{dom}\,(\mathrm{G})}{\mathrm{G} \vdash \lfloor \mathrm{e} \rfloor}
$$

---

[1] As Eisenberg et al. [2] noted, elaboration of source types can be non-deterministic: expressions inside the existential projection $\lfloor e : \epsilon \rfloor$ may have different elaborations due to non-deterministic type choices. As a result, equivalent types in the source may not remain equivalent in the core. We follow EDWL to use core expressions instead of source expressions within existential projections when establishing elaboration soundness.

$\boxed{\mathrm{G} \vdash \mathrm{e} : \mathrm{t}}$ *(Expression Typing)*

$$\textsc{c-e-Int}\quad \frac{\vdash \mathrm{G}}{\mathrm{G} \vdash n : \mathsf{Int}}
\qquad
\textsc{c-e-Var}\quad \frac{\vdash \mathrm{G} \qquad x : \mathrm{t} \in \mathrm{G}}{\mathrm{G} \vdash x : \mathrm{t}}
\qquad
\textsc{c-e-Abs}\quad \frac{\mathrm{G}, x : \mathrm{t}_1 \vdash \mathrm{e} : \mathrm{t}_2 \qquad x \notin \mathsf{fv}\,(\mathrm{t}_2)}{\mathrm{G} \vdash \lambda x : \mathrm{t}_1.\ \mathrm{e} : \mathrm{t}_1 \to \mathrm{t}_2}$$

$$\textsc{c-e-App}\quad \frac{\mathrm{G} \vdash \mathrm{e}_1 : \mathrm{t}_1 \to \mathrm{t}_2 \qquad \mathrm{G} \vdash \mathrm{e}_2 : \mathrm{t}_1}{\mathrm{G} \vdash \mathrm{e}_1\, \mathrm{e}_2 : \mathrm{t}_2}
\qquad
\textsc{c-e-TAbs}\quad \frac{\mathrm{G}, a \vdash \mathrm{e} : \mathrm{t}}{\mathrm{G} \vdash \Lambda a.\ \mathrm{e} : \forall a.\mathrm{t}}$$

$$\textsc{c-e-TApp}\quad \frac{\mathrm{G} \vdash \mathrm{e} : \forall a.\mathrm{t}_1 \qquad \mathrm{G} \vdash \mathrm{t}_2}{\mathrm{G} \vdash \mathrm{e}\, \mathrm{t}_2 : [\mathrm{t}_2/a]\mathrm{t}_1}
\qquad
\textsc{c-e-Pack}\quad \frac{\mathrm{G} \vdash \mathrm{t}_1 \qquad \mathrm{G} \vdash \exists\, a.\mathrm{t}_2 \qquad \mathrm{G} \vdash \mathrm{e} : [\mathrm{t}_1/a]\mathrm{t}_2}{\mathrm{G} \vdash \mathsf{pack}\, \mathrm{t}_1, \mathrm{e}\, \mathsf{as} \exists\, a.\mathrm{t}_2 : \exists\, a.\mathrm{t}_2}$$

$$\textsc{c-e-Open}\quad \frac{\mathrm{G} \vdash \mathrm{e} : \exists\, a.\mathrm{t}}{\mathrm{G} \vdash \mathsf{open}\, \mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\mathrm{t}}
\qquad
\textsc{c-e-Cast}\quad \frac{\mathrm{G} \vdash \mathrm{e} : \mathrm{t}_1 \qquad \mathrm{G} \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}{\mathrm{G} \vdash \mathrm{e} \rhd \gamma : \mathrm{t}_2}$$

$\boxed{\mathrm{G} \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}$ *(Coercion Typing)*

$$\textsc{c-g-Refl}\quad \frac{\mathrm{G} \vdash \mathrm{t}}{\mathrm{G} \vdash \langle \mathrm{t} \rangle : \mathrm{t} \approx \mathrm{t}}
\qquad
\textsc{c-g-Sym}\quad \frac{\mathrm{G} \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}{\mathrm{G} \vdash \mathsf{sym}\, \gamma : \mathrm{t}_2 \approx \mathrm{t}_1}$$

$$\textsc{c-g-Trans}\quad \frac{\mathrm{G} \vdash \gamma_1 : \mathrm{t}_1 \approx \mathrm{t}_2 \qquad \mathrm{G} \vdash \gamma_2 : \mathrm{t}_2 \approx \mathrm{t}_3}{\mathrm{G} \vdash \gamma_1 ;; \gamma_2 : \mathrm{t}_1 \approx \mathrm{t}_3}
\qquad
\textsc{c-g-Arr}\quad \frac{\mathrm{G} \vdash \gamma_1 : \mathrm{t}_1 \approx \mathrm{t}_1' \qquad \mathrm{G} \vdash \gamma_2 : \mathrm{t}_2 \approx \mathrm{t}_2'}{\mathrm{G} \vdash \gamma_1 \to \gamma_2 : \mathrm{t}_1 \to \mathrm{t}_2 \approx \mathrm{t}_1' \to \mathrm{t}_2'}$$

$$\textsc{c-g-Exists}\quad \frac{\mathrm{G}, a \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}{\mathrm{G} \vdash \exists\, a.\gamma : \exists\, a.\mathrm{t}_1 \approx \exists\, a.\mathrm{t}_2}
\qquad
\textsc{c-g-Forall}\quad \frac{\mathrm{G}, a \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}{\mathrm{G} \vdash \forall a.\gamma : \forall a.\mathrm{t}_1 \approx \forall a.\mathrm{t}_2}$$

$$\textsc{c-g-Proj}\quad \frac{\mathrm{G} \vdash \eta : \mathrm{e}_1 \approx \mathrm{e}_2}{\mathrm{G} \vdash \lfloor \eta \rfloor : \lfloor \mathrm{e}_1 \rfloor \approx \lfloor \mathrm{e}_2 \rfloor}
\qquad
\textsc{c-g-ProjPack}\quad \frac{\mathrm{G} \vdash \mathsf{pack}\, \mathrm{t}_1, \mathrm{e}\, \mathsf{as}\, \mathrm{t}_2 : \mathrm{t}_2}{\mathrm{G} \vdash \mathsf{projpack}\, \mathrm{t}_1, \mathrm{e}\, \mathsf{as}\, \mathrm{t}_2 : \lfloor \mathsf{pack}\, \mathrm{t}_1, \mathrm{e}\, \mathsf{as}\, \mathrm{t}_2 \rfloor \approx \mathrm{t}_1}$$

$$\textsc{c-g-InstForall}\quad \frac{\mathrm{G} \vdash \gamma_1 : \forall a.\mathrm{t}_1 \approx \forall a.\mathrm{t}_2 \qquad \mathrm{G} \vdash \gamma_2 : \mathrm{t}_3 \approx \mathrm{t}_4}{\mathrm{G} \vdash \gamma_1 @@ \gamma_2 : [\mathrm{t}_3/a]\mathrm{t}_1 \approx [\mathrm{t}_4/a]\mathrm{t}_2}$$

$$\textsc{c-g-InstExists}\quad \frac{\mathrm{G} \vdash \gamma_1 : \exists\, a.\mathrm{t}_1 \approx \exists\, a.\mathrm{t}_2 \qquad \mathrm{G} \vdash \gamma_2 : \mathrm{t}_3 \approx \mathrm{t}_4}{\mathrm{G} \vdash \gamma_1 @@ \gamma_2 : [\mathrm{t}_3/a]\mathrm{t}_1 \approx [\mathrm{t}_4/a]\mathrm{t}_2}
\qquad
\textsc{c-g-Fst}\quad \frac{\mathrm{G} \vdash \gamma : \mathrm{t}_1 \to \mathrm{t}_2 \approx \mathrm{t}_1' \to \mathrm{t}_2'}{\mathrm{G} \vdash \mathsf{fst}\, \gamma : \mathrm{t}_1 \approx \mathrm{t}_1'}$$

$$\textsc{c-g-Snd}\quad \frac{\mathrm{G} \vdash \gamma : \mathrm{t}_1 \to \mathrm{t}_2 \approx \mathrm{t}_1' \to \mathrm{t}_2'}{\mathrm{G} \vdash \mathsf{snd}\, \gamma : \mathrm{t}_2 \approx \mathrm{t}_2'}$$

$\boxed{\mathrm{G} \vdash \eta : \mathrm{e}_1 \approx \mathrm{e}_2}$ *(Expression Coercion Typing)*

$$\textsc{c-h-Coherence}\quad \frac{\mathrm{G} \vdash \mathrm{e} : \mathrm{t}_1 \qquad \mathrm{G} \vdash \gamma : \mathrm{t}_1 \approx \mathrm{t}_2}{\mathrm{G} \vdash \mathrm{e} \rhd \gamma : \mathrm{e} \approx \mathrm{e} \rhd \gamma}
\qquad
\textsc{c-h-Step}\quad \frac{\mathrm{G} \vdash \mathrm{e} : \mathrm{t} \qquad \mathrm{G} \vdash \mathrm{e}' : \mathrm{t} \qquad \mathrm{G} \vdash \mathrm{e} \longrightarrow \mathrm{e}'}{\mathrm{G} \vdash \mathsf{step}\, \mathrm{e} : \mathrm{e} \approx \mathrm{e}'}$$

$\boxed{G \vdash e \longrightarrow e'}$ *(Operational Semantics)*

c-s-Beta

$$\frac{}{G \vdash (\lambda x : t.\ e_1)\, e_2 \longrightarrow [e_2/x]e_1}$$

c-s-AppCong

$$\frac{G \vdash e_1 \longrightarrow e_1'}{G \vdash e_1\, e_2 \longrightarrow e_1'\, e_2}$$

c-s-AppPull

$$\frac{v = \lambda x : t.\ e_0 \qquad \gamma_1 = \mathsf{sym}\,(\mathsf{fst}\,\gamma) \qquad \gamma_2 = \mathsf{snd}\,\gamma}{G \vdash (v \rhd \gamma)\, e \longrightarrow (v\,(e \rhd \gamma_1)) \rhd \gamma_2}$$

c-s-TAbsCong

$$\frac{G, a \vdash e \longrightarrow e'}{G \vdash \mathit{\Lambda} a.\ e \longrightarrow \mathit{\Lambda} a.\ e'}$$

c-s-TAbsPull

$$\frac{}{G \vdash \mathit{\Lambda} a.\ (v \rhd \gamma) \longrightarrow (\mathit{\Lambda} a.\ v) \rhd \forall a.\gamma}$$

c-s-TBeta

$$\frac{}{G \vdash (\mathit{\Lambda} a.\ v)\, t \longrightarrow [t/a]v}$$

c-s-TAppCong

$$\frac{G \vdash e \longrightarrow e'}{G \vdash e\, t \longrightarrow e'\, t}$$

c-s-TAppPull

$$\frac{G \vdash v : \forall a.t_0}{G \vdash (v \rhd \gamma)\, t \longrightarrow v\, t \rhd (\gamma @@ \langle t \rangle)}$$

c-s-PackCong

$$\frac{G \vdash e \longrightarrow e'}{G \vdash \mathsf{pack}\, t_1, e\, \mathsf{as}\, t_2 \longrightarrow \mathsf{pack}\, t_1, e'\, \mathsf{as}\, t_2}$$

c-s-OpenPack

$$\frac{}{G \vdash \mathsf{open}\,(\mathsf{pack}\, t_1, v\, \mathsf{as}\, t_2) \longrightarrow v \rhd \langle t_2 \rangle @@ (\mathsf{sym}\,(\mathsf{projpack}\, t_1, v\, \mathsf{as}\, t_2))}$$

c-s-OpenPackCasted

$$\frac{}{G \vdash \mathsf{open}\,(\mathsf{pack}\, t_1, (v \rhd \gamma)\, \mathsf{as}\, t_2) \longrightarrow (v \rhd \gamma) \rhd \langle t_2 \rangle @@ (\mathsf{sym}\,(\mathsf{projpack}\, t_1, (v \rhd \gamma)\, \mathsf{as}\, t_2))}$$

c-s-OpenCong

$$\frac{G \vdash e : t \qquad G \vdash e \longrightarrow e'}{G \vdash \mathsf{open}\, e \longrightarrow \mathsf{open}\, e' \rhd \langle t \rangle @@ (\mathsf{sym}\, \lfloor \mathsf{step}\, e \rfloor)}$$

c-s-OpenPull

$$\frac{v = \mathsf{pack}\, t_1, v_0\, \mathsf{as}\, \exists\, a.t_0}{G \vdash \mathsf{open}\,(v \rhd \gamma) \longrightarrow (\mathsf{open}\, v) \rhd \gamma @@ \lfloor v \rhd \gamma \rfloor}$$

c-s-CastCong

$$\frac{G \vdash e \longrightarrow e'}{G \vdash e \rhd \gamma \longrightarrow e' \rhd \gamma}$$

c-s-CastTrans

$$\frac{}{G \vdash (v \rhd \gamma_1) \rhd \gamma_2 \longrightarrow v \rhd (\gamma_1 ;; \gamma_2)}$$

F The Elaborated Declarative System

This section presents a type-directed elaboration process from our declarative type system to the type-safe core calculus, thus establish type soundness of the declarative type system. Since the elaboration is mostly the same as Eisenberg et al. [2], we keep the discussion brief. We refer interested readers to Eisenberg et al. [2] for more details.

The type-directed elaboration rules presented below elaborate a source expression e to a core expression e. We write $\sigma \rightrightarrows \mathrm{t}$ to elaborate a source type σ to a core type t. We write $\varGamma \rightrightarrows \mathrm{G}$ for context elaboration that elaborates all types in a context. Rule E-I-ABS inserts an explicit pack for inferred existential types.

The subtyping judgment takes an input core expression e_1 and produces an output e_2. Rule E-S-FORALLL recurses with $\mathrm{e}\,\mathrm{t}$, while rule E-S-FORALLR adds a type abstraction to the output expression. Rule E-S-EXISTSL recurses with $\mathsf{open}\,\mathrm{e}$. Rule E-S-EXISTSR inserts an explicit pack.

universally quantified type	σ	$:= \epsilon \mid \forall a.\sigma$
existentially quantified type	ϵ	$:= \rho \mid \exists\, b.\epsilon$
top-level monomorphic type	ρ	$:= \tau \mid \sigma_1 \to \sigma_2$
monomorphic type	τ	$:= a \mid \mathsf{Int} \mid \tau_1 \to \tau_2 \mid \lfloor \mathrm{e} \rfloor$
expr	e	$:= n \mid x \mid \lambda x.\ e \mid e_1\, e_2 \mid (e : \sigma)$
context	$\varGamma$	$:= \bullet \mid x : \sigma \mid a \mid \varGamma_1, \ldots, \varGamma_n$

$\boxed{\varGamma \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e}}$ *(Inferenece)*

$$\textsc{e-i-Int}\quad \frac{\vdash \varGamma}{\varGamma \vdash n \Rightarrow \mathsf{Int} \rightrightarrows n} \qquad\qquad \textsc{e-i-Var}\quad \frac{\vdash \varGamma \qquad x : \sigma \in \varGamma}{\varGamma \vdash x \Rightarrow \sigma \rightrightarrows x}$$

$$\textsc{e-i-Abs}\quad \frac{\begin{array}{c} \varGamma, x : \tau \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e} \qquad \varGamma, x : \tau \vdash \mathrm{e} : \sigma \rightsquigarrow_{\forall} \epsilon \rightrightarrows \mathrm{e}' \\ \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon \qquad \tau \rightrightarrows \mathrm{t}_1 \qquad \epsilon \rightrightarrows \mathrm{t}_2 \qquad \epsilon' \rightrightarrows \mathrm{t}_2' \end{array}}{\varGamma \vdash \lambda x.\ e \Rightarrow \tau \to \exists\, \overline{a}.\epsilon' \rightrightarrows \lambda x : \mathrm{t}_1.\ \mathsf{pack}\ \lfloor \mathrm{t}_2 \rfloor_x, \mathrm{e}'\ \mathsf{as}\ \exists\, \overline{a}.\mathrm{t}_2'}$$

$$\textsc{e-i-AbsA}\quad \frac{\begin{array}{c} \varGamma, x : \sigma_1 \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e} \qquad \varGamma, x : \sigma_1 \vdash \mathrm{e} : \sigma \rightsquigarrow_{\forall} \epsilon \rightrightarrows \mathrm{e}' \\ \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon \qquad \mathsf{ftv}\,(\sigma_1) = \varnothing \qquad \sigma_1 \rightrightarrows \mathrm{t}_1 \qquad \epsilon \rightrightarrows \mathrm{t}_2 \qquad \epsilon' \rightrightarrows \mathrm{t}_2' \end{array}}{\varGamma \vdash \lambda x : \sigma_1.\ e \Rightarrow \sigma_1 \to \exists\, \overline{a}.\epsilon' \rightrightarrows \lambda x : \mathrm{t}_1.\ \mathsf{pack}\ \lfloor \mathrm{t}_2 \rfloor_x, \mathrm{e}'\ \mathsf{as}\ \exists\, \overline{a}.\mathrm{t}_2'}$$

$$\textsc{e-i-App}\quad \frac{\varGamma \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e} \qquad \varGamma \vdash \mathrm{e} : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}' \qquad \varGamma \vdash e_1 \Leftarrow \sigma_1 \rightrightarrows \mathrm{e}_1}{\varGamma \vdash e\, e_1 \Rightarrow \sigma_2 \rightrightarrows \mathrm{e}'\, \mathrm{e}_1}$$

$$\textsc{e-i-Ann}\quad \frac{\varGamma \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e} \qquad \varGamma \vdash \sigma \qquad \mathsf{ftv}\,(\sigma) = \varnothing}{\varGamma \vdash (e : \sigma) \Rightarrow \sigma \rightrightarrows \mathrm{e}}$$

$$\boxed{\varGamma \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e}} \qquad \textit{(Checking)}$$

$$\textsc{e-c-Sub}\ \frac{\varGamma \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e} \qquad \varGamma \vdash \mathrm{e} : \sigma <: \rho \rightrightarrows \mathrm{e}'}{\varGamma \vdash e \Leftarrow \rho \rightrightarrows \mathrm{e}'}
\qquad
\textsc{e-c-Abs}\ \frac{\varGamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathrm{e} \qquad \sigma_1 \rightrightarrows \mathrm{t}}{\varGamma \vdash \lambda x.\ e \Leftarrow \sigma_1 \to \sigma_2 \rightrightarrows \lambda x : \mathrm{t}.\ \mathrm{e}}$$

$$\textsc{e-c-AbsA}\ \frac{\begin{array}{c}\varGamma, x : \sigma_1 \vdash x : \sigma_1 <: \sigma_1' \rightrightarrows \mathrm{e}_1 \\ \varGamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathrm{e} \qquad \varGamma, x : \sigma_1 \vdash \sigma_1' \qquad \mathsf{ftv}\,(\sigma_1') = \varnothing \qquad \sigma_1 \rightrightarrows \mathrm{t}\end{array}}{\varGamma \vdash \lambda x : \sigma_1'.\ e \Leftarrow \sigma_1 \to \sigma_2 \rightrightarrows \lambda x : \mathrm{t}.\ \mathrm{e}}$$

$$\textsc{e-c-Exists}\ \frac{\varGamma \vdash e \Leftarrow [\tau / b]\epsilon \rightrightarrows \mathrm{e} \qquad \varGamma \vdash \tau \qquad \tau \rightrightarrows \mathrm{t} \qquad \epsilon \rightrightarrows \mathrm{t}'}{\varGamma \vdash e \Leftarrow \exists\, b.\epsilon \rightrightarrows \mathsf{pack}\,\mathrm{t}, \mathrm{e}\,\mathsf{as}\,\exists\, b.\mathrm{t}'}
\qquad
\textsc{e-c-Forall}\ \frac{\varGamma, a \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e}}{\varGamma \vdash e \Leftarrow \forall a.\sigma \rightrightarrows \varLambda a.\ \mathrm{e}}$$

$$\boxed{\varGamma \vdash \mathrm{e}_1 : \sigma \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}_2} \qquad \textit{(Instantiation)}$$

$$\textsc{e-inst-Refl}\ \frac{\vdash \varGamma}{\varGamma \vdash \mathrm{e} : \sigma_1 \to \sigma_2 \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}}
\qquad
\textsc{e-inst-Exists}\ \frac{\varGamma \vdash \mathsf{open}\,\mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}{\varGamma \vdash \mathrm{e} : \exists\, a.\epsilon \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}$$

$$\textsc{e-inst-Forall}\ \frac{\varGamma \vdash \mathrm{e}\,\mathrm{t} : [\tau / a]\sigma \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}' \qquad \varGamma \vdash \tau \qquad \tau \rightrightarrows \mathrm{t}}{\varGamma \vdash \mathrm{e} : \forall a.\sigma \leadsto \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}$$

$$\boxed{\varGamma \vdash \mathrm{e}_1 : \sigma \leadsto_{\forall} \epsilon \rightrightarrows \mathrm{e}_2} \qquad \textit{(Instantiation)}$$

$$\textsc{e-instf-Refl}\ \frac{\vdash \varGamma}{\varGamma \vdash \mathrm{e} : \epsilon \leadsto_{\forall} \epsilon \rightrightarrows \mathrm{e}}
\qquad
\textsc{e-instf-Forall}\ \frac{\varGamma \vdash \mathrm{e}\,\mathrm{t} : [\tau / a]\sigma \leadsto_{\forall} \epsilon \rightrightarrows \mathrm{e}' \qquad \varGamma \vdash \tau \qquad \tau \rightrightarrows \mathrm{t}}{\varGamma \vdash \mathrm{e} : \forall a.\sigma \leadsto_{\forall} \epsilon \rightrightarrows \mathrm{e}'}$$

$$\boxed{\varGamma \vdash \mathrm{e}_1 : \sigma_1 <: \sigma_2 \rightrightarrows \mathrm{e}_2} \qquad \textit{(Subtyping)}$$

$$\textsc{e-s-Int}\ \frac{\vdash \varGamma}{\varGamma \vdash \mathrm{e} : \mathsf{Int} <: \mathsf{Int} \rightrightarrows \mathrm{e}}
\qquad
\textsc{e-s-Var}\ \frac{\vdash \varGamma \qquad a \in \mathsf{dom}\,(\varGamma)}{\varGamma \vdash \mathrm{e} : a <: a \rightrightarrows \mathrm{e}}
\qquad
\textsc{e-s-Proj}\ \frac{\vdash \varGamma}{\varGamma \vdash \mathrm{e}' : \lfloor \mathrm{e} \rfloor <: \lfloor \mathrm{e} \rfloor \rightrightarrows \mathrm{e}'}$$

$$\textsc{e-s-Arr}\ \frac{\varGamma, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \rightrightarrows \mathrm{e}_1 \qquad \varGamma, x : \sigma_1' \vdash \mathrm{e}\,\mathrm{e}_1 : \sigma_2 <: \sigma_2' \rightrightarrows \mathrm{e}' \qquad \sigma_1' \rightrightarrows \mathrm{t}_1'}{\varGamma \vdash \mathrm{e} : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2' \rightrightarrows \lambda x : \mathrm{t}_1'.\ \mathrm{e}'}$$

$$\textsc{e-s-ExistsL}\ \frac{\varGamma \vdash \mathsf{open}\,\mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon_1 <: \epsilon_2 \rightrightarrows \mathrm{e}'}{\varGamma \vdash \mathrm{e} : \exists\, a.\epsilon_1 <: \epsilon_2 \rightrightarrows \mathrm{e}'}$$

$$\textsc{e-s-ExistsR}\ \frac{\varGamma \vdash \mathrm{e} : \rho <: [\tau / a]\epsilon \rightrightarrows \mathrm{e}' \qquad \varGamma \vdash \tau \qquad \tau \rightrightarrows \mathrm{t} \qquad \epsilon \rightrightarrows \mathrm{t}'}{\varGamma \vdash \mathrm{e} : \rho <: \exists\, a.\epsilon \rightrightarrows \mathsf{pack}\,\mathrm{t}, \mathrm{e}'\,\mathsf{as}\,\exists\, a.\mathrm{t}'}$$

$$\textsc{e-s-ForallL}\quad \frac{\Gamma \vdash \mathrm{e}\,\mathrm{t} : [\tau/a]\sigma <: \epsilon \rightrightarrows \mathrm{e}' \qquad \Gamma \vdash \tau \qquad \tau \rightrightarrows \mathrm{t}}{\Gamma \vdash \mathrm{e} : \forall a.\sigma <: \epsilon \rightrightarrows \mathrm{e}'}$$

$$\textsc{e-s-ForallR}\quad \frac{\Gamma, a \vdash \mathrm{e} : \sigma_1 <: \sigma_2 \rightrightarrows \mathrm{e}'}{\Gamma \vdash \mathrm{e} : \sigma_1 <: \forall a.\sigma_2 \rightrightarrows \Lambda a.\ \mathrm{e}'}$$

$\boxed{\sigma \rightrightarrows \mathrm{t}}$ *(Type Elaboration)*

$$\textsc{e-t-Forall}\quad \frac{\sigma \rightrightarrows \mathrm{t}}{\forall a.\sigma \rightrightarrows \forall a.\mathrm{t}} \qquad \textsc{e-t-Exists}\quad \frac{\epsilon \rightrightarrows \mathrm{t}}{\exists\, a.\epsilon \rightrightarrows \exists\, a.\mathrm{t}} \qquad \textsc{e-t-Arrow}\quad \frac{\sigma_1 \rightrightarrows \mathrm{t}_1 \qquad \sigma_2 \rightrightarrows \mathrm{t}_2}{\sigma_1 \to \sigma_2 \rightrightarrows \mathrm{t}_1 \to \mathrm{t}_2} \qquad \textsc{e-t-Proj}\quad \frac{}{\lfloor \mathrm{e} \rfloor \rightrightarrows \lfloor \mathrm{e} \rfloor}$$

$$\textsc{e-t-Var}\quad \frac{}{a \rightrightarrows a} \qquad \textsc{e-t-Int}\quad \frac{}{\mathsf{Int} \rightrightarrows \mathsf{Int}}$$

$\boxed{\Gamma \rightrightarrows \mathrm{G}}$ *(Context Elaboration)*

$$\textsc{e-ctx-Empty}\quad \frac{}{\bullet \rightrightarrows \bullet} \qquad \textsc{e-ctx-Var}\quad \frac{\Gamma \rightrightarrows \mathrm{G} \qquad \sigma \rightrightarrows \mathrm{t}}{\Gamma, x : \sigma \rightrightarrows \mathrm{G}, x : \mathrm{t}} \qquad \textsc{e-ctx-TVar}\quad \frac{\Gamma \rightrightarrows \mathrm{G}}{\Gamma, a \rightrightarrows \mathrm{G}, a}$$

G Meta Theory of the Elaborated Declarative System

Lemma/Theorem	*Related Lemma/Theorem in EDWL*
Theorem G.1 (Elaboration Soundness)	B.1

We prove that our elaboration preserves typing. Combined with type soundness of the core calculus, this theorem implies that the source calculus is also type sound.

Theorem G.1 (Elaboration Soundness)**.** *a) If* $\Gamma \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e}$*, then* $G \vdash \mathrm{e} : t$*, where* $\Gamma \rightrightarrows G$ *and* $\sigma \rightrightarrows t$*.*

b) If $\Gamma \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e}$*, then* $G \vdash \mathrm{e} : t$*, where* $\Gamma \rightrightarrows G$ *and* $\sigma \rightrightarrows t$*.*

c) If $\Gamma \vdash \mathrm{e} : \sigma \rightsquigarrow \sigma_1 \rightarrow \sigma_2 \rightrightarrows \mathrm{e}'$ *and* $G \vdash \mathrm{e} : t$*, then* $G \vdash \mathrm{e}' : t'$*, where* $\Gamma \rightrightarrows G$*,* $\sigma \rightrightarrows t$*, and* $\sigma_1 \rightarrow \sigma_2 \rightrightarrows t'$*.*

d) If $\Gamma \vdash \mathrm{e} : \sigma \rightsquigarrow_{\forall} \epsilon \rightrightarrows \mathrm{e}'$ *and* $G \vdash \mathrm{e} : t$*, then* $G \vdash \mathrm{e}' : t'$*, where* $\Gamma \rightrightarrows G$*,* $\sigma \rightrightarrows t$*, and* $\epsilon \rightrightarrows t'$*.*

e) If $\Gamma \vdash \mathrm{e}_1 : \sigma_1 <: \sigma_2 \rightrightarrows \mathrm{e}_2$ *and* $G \vdash \mathrm{e}_1 : t_1$*, then* $G \vdash \mathrm{e}_2 : t_2$*, where* $\Gamma \rightrightarrows G$*,* $\sigma_1 \rightrightarrows t_1$ *and* $\sigma_2 \rightrightarrows t_2$*.*

Proof. By induction on the typing rule.

- Rule E-I-Int: Apply rule C-E-Int.
- Rule E-I-Ann: Apply the induction hypothesis.
- Rules E-inst-Refl, E-instF-Refl, E-S-Int, E-S-Var, and E-S-Proj: Holds by assumption.
- Rule E-I-Var:

$$\frac{\vdash \Gamma \qquad x : \sigma \in \Gamma}{\Gamma \vdash x \Rightarrow \sigma \rightrightarrows x}$$

Let $\sigma \rightrightarrows \mathrm{t}$.

$x : \sigma \in \Gamma$	premise
$x : \mathrm{t} \in \mathrm{G}$	context elaboration
$\mathrm{G} \vdash x : \mathrm{t}$	Rule C-E-Var

- Rule E-I-Abs:

$$\frac{\begin{array}{c}\Gamma, x : \tau \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e} \qquad \Gamma, x : \tau \vdash \mathrm{e} : \sigma \rightsquigarrow_{\forall} \epsilon \rightrightarrows \mathrm{e}' \\ \overline{a}\ \mathsf{fresh} \qquad \epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon \qquad \tau \rightrightarrows \mathrm{t}_1 \qquad \epsilon \rightrightarrows \mathrm{t}_2 \qquad \epsilon' \rightrightarrows \mathrm{t}_2'\end{array}}{\Gamma \vdash \lambda x.\ e \Rightarrow \tau \rightarrow \exists\,\overline{a}.\epsilon' \rightrightarrows \lambda x : \mathrm{t}_1.\ \mathsf{pack}\ \lfloor \mathrm{t}_2 \rfloor_x, \mathrm{e}'\ \mathsf{as}\ \exists\,\overline{a}.\mathrm{t}_2'}$$

Let $\sigma \rightrightarrows \mathrm{t}_0$.

$\Gamma, x : \tau \vdash e \Rightarrow \sigma \rightrightarrows \mathrm{e}$	premise
$\mathrm{G}, x : \mathrm{t}_1 \vdash \mathrm{e} : \mathrm{t}_0$	i.h.
$\Gamma, x : \tau \vdash \mathrm{e} : \sigma \rightsquigarrow_{\forall} \epsilon \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G}, x : \mathrm{t}_1 \vdash \mathrm{e}' : \mathrm{t}_2$	i.h.
$\epsilon' = [\overline{a}/\lfloor \epsilon \rfloor_x]\epsilon$	premise
$\mathrm{t}_2' = [\overline{a}/\lfloor \mathrm{t}_2 \rfloor_x]\mathrm{t}_2$	type elaboration
$\mathrm{G}, x : \mathrm{t}_1 \vdash \mathrm{e}' : [\lfloor \mathrm{t}_2 \rfloor_x/\overline{a}]\mathrm{t}_2'$	defn. of t_2'
$\mathrm{G}, x : \mathrm{t}_1 \vdash \mathsf{pack}\ \lfloor \mathrm{t}_2 \rfloor_x, \mathrm{e}\ \mathsf{as}\ \exists\,\overline{a}.\mathrm{t}_2' : \exists\,\overline{a}.\mathrm{t}_2'$	Rule C-E-Pack
$\mathrm{G} \vdash \lambda x : \mathrm{t}_1.\ \mathsf{pack}\ \lfloor \mathrm{t}_2 \rfloor_x, \mathrm{e}'\ \mathsf{as}\ \exists\,\overline{a}.\mathrm{t}_2' : \mathrm{t}_1 \rightarrow \exists\,\overline{a}.\mathrm{t}_2'$	Rule C-E-Abs

- Rule e-i-AbsA: Similar to rule e-i-Abs.
- Rule e-i-App:

$$\frac{\Gamma \vdash e \Rightarrow \sigma \rightrightarrows \mathsf{e} \qquad \Gamma \vdash \mathsf{e} : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathsf{e}' \qquad \Gamma \vdash e_1 \Leftarrow \sigma_1 \rightrightarrows \mathsf{e}_1}{\Gamma \vdash e\, e_1 \Rightarrow \sigma_2 \rightrightarrows \mathsf{e}'\, \mathsf{e}_1}$$

Let $\sigma \rightrightarrows \mathsf{t}$, $\sigma_1 \rightrightarrows \mathsf{t}_1$, and $\sigma_2 \rightrightarrows \mathsf{t}_2$.

$\Gamma \vdash e \Rightarrow \sigma \rightrightarrows \mathsf{e}$	premise
$\mathsf{G} \vdash \mathsf{e} : \mathsf{t}$	i.h.
$\Gamma \vdash \mathsf{e} : \sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathsf{e}'$	premise
$\mathsf{G} \vdash \mathsf{e}' : \mathsf{t}_1 \to \mathsf{t}_2$	i.h.
$\Gamma \vdash e_1 \Leftarrow \sigma_1 \rightrightarrows \mathsf{e}_1$	premise
$\mathsf{G} \vdash \mathsf{e}_1 : \mathsf{t}_1$	i.h.
$\mathsf{G} \vdash \mathsf{e}'\, \mathsf{e}_1 : \mathsf{t}_2$	Rule c-e-App

- Rule e-c-Sub:

$$\frac{\Gamma \vdash e \Rightarrow \sigma \rightrightarrows \mathsf{e} \qquad \Gamma \vdash \mathsf{e} : \sigma <: \rho \rightrightarrows \mathsf{e}'}{\Gamma \vdash e \Leftarrow \rho \rightrightarrows \mathsf{e}'}$$

Let $\sigma \rightrightarrows \mathsf{t}$ and $\rho \rightrightarrows \mathsf{t}'$.

$\Gamma \vdash e \Rightarrow \sigma \rightrightarrows \mathsf{e}$	premise
$\mathsf{G} \vdash \mathsf{e} : \mathsf{t}$	i.h.
$\Gamma \vdash \mathsf{e} : \sigma <: \rho \rightrightarrows \mathsf{e}'$	premise
$\mathsf{G} \vdash \mathsf{e}' : \mathsf{t}'$	i.h.

- Rule e-c-Abs:

$$\frac{\Gamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathsf{e} \qquad \sigma_1 \rightrightarrows \mathsf{t}}{\Gamma \vdash \lambda x.\ e \Leftarrow \sigma_1 \to \sigma_2 \rightrightarrows \lambda x : \mathsf{t}.\ \mathsf{e}}$$

Let $\sigma_2 \rightrightarrows \mathsf{t}_2$.

$\Gamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathsf{e}$	premise
$\mathsf{G} \vdash \mathsf{e} : \mathsf{t}'$	i.h.
$\mathsf{G} \vdash \lambda x : \mathsf{t}.\ \mathsf{e} : \mathsf{t} \to \mathsf{t}'$	Rule c-e-Abs

- Rule e-c-AbsA:

$$\frac{\begin{array}{c}\Gamma, x : \sigma_1 \vdash x : \sigma_1 <: \sigma_1' \rightrightarrows \mathsf{e}_1 \\ \Gamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathsf{e} \qquad \mathsf{ftv}\,(\sigma_1') = \varnothing \qquad \sigma_1 \rightrightarrows \mathsf{t}\end{array}}{\Gamma \vdash \lambda x : \sigma_1'.\ e \Leftarrow \sigma_1 \to \sigma_2 \rightrightarrows \lambda x : \mathsf{t}.\ \mathsf{e}}$$

Let $\sigma_2 \rightrightarrows \mathsf{t}'$.

$\Gamma, x : \sigma_1 \vdash e \Leftarrow \sigma_2 \rightrightarrows \mathsf{e}$	premise
$\mathsf{G} \vdash \mathsf{e} : \mathsf{t}'$	i.h.
$\mathsf{G} \vdash \lambda x : \mathsf{t}.\ \mathsf{e} : \mathsf{t} \to \mathsf{t}'$	Rule c-e-Abs

- Rule e-c-Exists:

$$\frac{\Gamma \vdash e \Leftarrow [\tau/b]\epsilon \rightrightarrows \mathsf{e} \qquad \Gamma \vdash \tau \qquad \tau \rightrightarrows \mathsf{t} \qquad \epsilon \rightrightarrows \mathsf{t}'}{\Gamma \vdash e \Leftarrow \exists\, b.\epsilon \rightrightarrows \mathsf{pack}\,\mathsf{t}, \mathsf{e}\,\mathsf{as}\,\exists\, b.\mathsf{t}'}$$

$\Gamma \vdash e \Leftarrow [\tau/b]\epsilon \rightrightarrows \mathsf{e}$	premise
$\mathsf{G} \vdash \mathsf{e} : [\mathsf{t}/b]\mathsf{t}'$	i.h.
$\mathsf{G} \vdash \mathsf{pack}\,\mathsf{t}, \mathsf{e}\,\mathsf{as}\,\exists\, b.\mathsf{t}' : \exists\, a.\mathsf{t}'$	Rule c-e-Pack

- Rule E-C-Forall:

$$\frac{\Gamma, a \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e}}{\Gamma \vdash e \Leftarrow \forall a.\sigma \rightrightarrows \Lambda a.\ \mathrm{e}}$$

Since $\forall a.\sigma \rightrightarrows \mathrm{t}$, we must have $\mathrm{t} = \forall a.\mathrm{t}'$ such that $\sigma \rightrightarrows \mathrm{t}'$ by rule E-T-Forall.

$\Gamma, a \vdash e \Leftarrow \sigma \rightrightarrows \mathrm{e}$	premise
$\mathrm{G}, a \vdash \mathrm{e} : \mathrm{t}'$	i.h.
$\mathrm{G} \vdash \Lambda a.\ \mathrm{e} : \forall a.\mathrm{t}'$	Rule C-E-TApp

- Rule E-inst-Exists:

$$\frac{\Gamma \vdash \mathsf{open}\, \mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}{\Gamma \vdash \mathrm{e} : \exists\, a.\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}$$

Let $\epsilon \rightrightarrows \mathrm{t}$ and $\sigma_1 \to \sigma_2 \rightrightarrows \mathrm{t}'$.

$\mathrm{G} \vdash \mathrm{e} : \exists\, a.\mathrm{t}$	assumption
$\mathrm{G} \vdash \mathsf{open}\, \mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\mathrm{t}$	Rule C-E-Open
$\Gamma \vdash \mathsf{open}\, \mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G} \vdash \mathrm{e}' : \mathrm{t}'$	i.h.

- Rule E-inst-Forall:

$$\frac{\tau \rightrightarrows \mathrm{t} \qquad \mathsf{fv}\,(\tau) \subseteq \mathsf{dom}\,(\Gamma) \qquad \Gamma \vdash \mathrm{e}\,\mathrm{t} : [\tau / a]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}{\Gamma \vdash \mathrm{e} : \forall a.\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'}$$

Let $\sigma \rightrightarrows \mathrm{t}'$ and $\sigma_1 \to \sigma_2 \rightrightarrows \mathrm{t}''$.

$\mathrm{G} \vdash \mathrm{e} : \forall a.\mathrm{t}'$	assumption
$\mathrm{G} \vdash \mathrm{e}\,\mathrm{t} : [\mathrm{t}/a]\mathrm{t}'$	Rule C-E-TApp
$\Gamma \vdash \mathrm{e}\,\mathrm{t} : [\tau/a]\sigma \rightsquigarrow \sigma_1 \to \sigma_2 \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G} \vdash \mathrm{e}' : \mathrm{t}''$	i.h.

- Rule E-instf-Forall: Similar to rule E-inst-Forall.
- Rule E-s-Arr:

$$\frac{\Gamma, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \rightrightarrows \mathrm{e}_1 \qquad \Gamma, x : \sigma_1' \vdash \mathrm{e}\,\mathrm{e}_1 : \sigma_2 <: \sigma_2' \rightrightarrows \mathrm{e}' \qquad \sigma_1' \rightrightarrows \mathrm{t}_1'}{\Gamma \vdash \mathrm{e} : \sigma_1 \to \sigma_2 <: \sigma_1' \to \sigma_2' \rightrightarrows \lambda x : \mathrm{t}_1'.\ \mathrm{e}'}$$

Let $\sigma_1 \rightrightarrows \mathrm{t}_1$, $\sigma_2 \rightrightarrows \mathrm{t}_2$, $\sigma_1' \rightrightarrows \mathrm{t}_1'$, and $\sigma_2' \rightrightarrows \mathrm{t}_2'$.

$\mathrm{G} \vdash \mathrm{e} : \mathrm{t}_1 \to \mathrm{t}_2$	assumption
$\mathrm{G}, x : \mathrm{t}_1' \vdash \mathrm{e} : \mathrm{t}_1 \to \mathrm{t}_2$	weakening
$\mathrm{G}, x : \mathrm{t}_1' \vdash x : \mathrm{t}_1'$	Rule C-E-Var
$\Gamma, x : \sigma_1' \vdash x : \sigma_1' <: \sigma_1 \rightrightarrows \mathrm{e}_1$	premise
$\mathrm{G}, x : \mathrm{t}_1' \vdash \mathrm{e}_1 : \mathrm{t}_1$	i.h.
$\mathrm{G}, x : \mathrm{t}_1' \vdash \mathrm{e}\,\mathrm{e}_1 : \mathrm{t}_2$	Rule C-E-App
$\Gamma, x : \sigma_1' \vdash \mathrm{e}\,\mathrm{e}_1 : \sigma_2 <: \sigma_2' \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G}, x : \mathrm{t}_1' \vdash \mathrm{e}' : \mathrm{t}_2'$	i.h.
$\mathrm{G} \vdash \lambda x : \mathrm{t}_1'.\ \mathrm{e}' : \mathrm{t}_1' \to \mathrm{t}_2'$	Rule C-E-Abs

- Rule E-s-ExistsL:

$$\frac{\Gamma \vdash \mathsf{open}\, \mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon_1 <: \epsilon_2 \rightrightarrows \mathrm{e}'}{\Gamma \vdash \mathrm{e} : \exists\, a.\epsilon_1 <: \epsilon_2 \rightrightarrows \mathrm{e}'}$$

Let $\epsilon_1 \rightrightarrows \mathrm{t}_1$ and $\epsilon_2 \rightrightarrows \mathrm{t}_2$.

$\mathrm{G} \vdash \mathrm{e} : \exists\, a.\mathrm{t}_1$	assumption
$\mathrm{G} \vdash \mathsf{open}\,\mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\mathrm{t}_1$	Rule c-e-Open
$\Gamma \vdash \mathsf{open}\,\mathrm{e} : [\lfloor \mathrm{e} \rfloor / a]\epsilon_1 <: \epsilon_2 \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G} \vdash \mathrm{e}' : \mathrm{t}_2$	i.h.

- Rule e-s-ExistsR:

$$\frac{\Gamma \vdash \mathrm{e} : \rho <: [\tau/a]\epsilon \rightrightarrows \mathrm{e}' \qquad \mathsf{fv}\,(\tau) \subseteq \mathsf{dom}\,(\Gamma) \qquad \tau \rightrightarrows \mathrm{t} \qquad \epsilon \rightrightarrows \mathrm{t}'}{\Gamma \vdash \mathrm{e} : \rho <: \exists\, a.\epsilon \rightrightarrows \mathsf{pack}\,\mathrm{t}, \mathrm{e}'\,\mathsf{as}\,\exists\, a.\mathrm{t}'}$$

Let $\rho \rightrightarrows \mathrm{t}''$.

$\mathrm{G} \vdash \mathrm{e}' : \mathrm{t}''$	assumption
$\Gamma \vdash \mathrm{e} : \rho <: [\tau/a]\epsilon \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G} \vdash \mathrm{e}' : [\mathrm{t}/a]\mathrm{t}'$	i.h.
$\mathrm{G} \vdash \mathsf{pack}\,\mathrm{t}, \mathrm{e}'\,\mathsf{as}\,\exists\, a.\mathrm{t}' : \exists\, a.\mathrm{t}'$	Rule c-e-Pack

- Rule e-s-ForallL:

$$\frac{\Gamma \vdash \mathrm{e}\,\mathrm{t} : [\tau/a]\sigma <: \epsilon \rightrightarrows \mathrm{e}' \qquad \mathsf{fv}\,(\tau) \subseteq \mathsf{dom}\,(\Gamma) \qquad \tau \rightrightarrows \mathrm{t}}{\Gamma \vdash \mathrm{e} : \forall a.\sigma <: \epsilon \rightrightarrows \mathrm{e}'}$$

Let $\epsilon \rightrightarrows \mathrm{t}'$ and $\sigma \rightrightarrows \mathrm{t}''$.

$\mathrm{G} \vdash \mathrm{e} : \forall a.\mathrm{t}''$	assumption
$\mathrm{G} \vdash \mathrm{e}\,\mathrm{t} : [\mathrm{t}/a]\mathrm{t}''$	Rule c-e-TApp
$\Gamma \vdash \mathrm{e}\,\mathrm{t} : [\tau/a]\sigma <: \epsilon \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G} \vdash \mathrm{e}' : \mathrm{t}'$	i.h.

- Rule e-s-ForallR:

$$\frac{\Gamma, a \vdash \mathrm{e} : \sigma_1 <: \sigma_2 \rightrightarrows \mathrm{e}'}{\Gamma \vdash \mathrm{e} : \sigma_1 <: \forall a.\sigma_2 \rightrightarrows \Lambda a.\ \mathrm{e}'}$$

Let $\sigma_1 \rightrightarrows \mathrm{t}_1$ and $\sigma_2 \rightrightarrows \mathrm{t}_2$.

$\mathrm{G} \vdash \mathrm{e} : \mathrm{t}_1$	assumption
$\mathrm{G}, a \vdash \mathrm{e} : \mathrm{t}_1$	weakening
$\Gamma, a \vdash \mathrm{e} : \sigma_1 <: \sigma_2 \rightrightarrows \mathrm{e}'$	premise
$\mathrm{G}, a \vdash \mathrm{e}' : \mathrm{t}_2$	i.h.
$\mathrm{G} \vdash \Lambda a.\ \mathrm{e}' : \forall a.\mathrm{t}_2$	Rule c-e-TAbs

$\square$

Bibliography

[1] Dunfield, J., Krishnaswami, N.R.: Complete and easy bidirectional typechecking for higher-rank polymorphism. In: Proceedings of the 18th ACM SIGPLAN international conference on Functional programming, pp. 429–442 (2013)

[2] Eisenberg, R.A., Duboc, G., Weirich, S., Lee, D.: An existential crisis resolved: Type inference for first-class existential types. Proceedings of the ACM on Programming Languages **5**(ICFP), 1–29 (2021)

[3] Wright, A.K., Felleisen, M.: A syntactic approach to type soundness. Information and computation **115**(1), 38–94 (1994)