Spine-local Type Inference: Proof Appendix

Christopher Jenkins

February 12, 2019

Contents

1 Type Inference Rules

1.1 Syntax

Types	$S, T, U, V ::= X, Y, Z \mid S \rightarrow T \mid \forall X. T$
Contexts	$\Gamma ::= \cdot \mid \Gamma, X \mid \Gamma, x\!:\!T$
Terms (Internal)	$e, p ::= x \mid \lambda x\!:\!T. e \mid \Lambda X. e \mid e\ e' \mid e[T]$
Terms (External)	$t ::= x \mid \lambda x\!:\!T. t \mid \lambda x. t \mid \Lambda X. t \mid t\ t' \mid t[T]$
Prototypes	$P ::= ? \mid T \mid ? \rightarrow P$
Decorated Types	$W ::= T \mid S \rightarrow W \mid \forall X\!=\!X. W \mid \forall X\!=\!S. W \mid (X, ? \rightarrow P)$

1.2 Terminology

In both the internal and external languages, we say that the *applicand* of a term or type application is the term in the function position. A *head* is either a variable or λ-abstraction (bare or annotated), and an *application spine* (or just *spine*) is a view of an application as consisting of some head (called the *spine head*) followed by a sequence of (term and type) arguments. The *maximal application* of a sub-expression is the spine in which it occurs as an applicand, or just the sub-expression itself if it does not. For example, spine $x[S]\ y\ z$ is the maximal application of itself and its applicand sub-expressions x, $x[S]$, and $x[S]\ y$, with x as head of the spine. Predicate $App(t)$ indicates term t is some term or type application (in either language) and we define it formally as $(\exists\, t_1, t_2.\ t = t_1\ t_2)\ \vee\ (\exists\, t', S.\ t = t'[S])$. Finally, for any application $e_1\ e_2$ we shall call a *term applicand** any applicand occuring in the spine of e_1.

Turning to definitions for types and contexts, function $DTV(\Gamma)$ calculates the set of *declared type variables* of context Γ and is defined recursively by the following set of equations:

$$\begin{array}{rcl} DTV(\cdot) & = & \varnothing \\ DTV(\Gamma, X) & = & DTV(\Gamma) \cup \{X\} \\ DTV(\Gamma, x\!:\!T) & = & DTV(\Gamma) \end{array}$$

Predicate $WF(\Gamma, T)$ indicates that type T is *well-formed* under Γ – that is, all free type variables of T occur as declared type variables in Γ (formally $FV(T) \subseteq DTV(\Gamma)$).

1.3 Meta-language Definitions

$$
\begin{array}{rcl}
\mathit{TmApp}(t) & = & (\exists t_1, t_2.\ t = t_1\ t_2) \\
\mathit{TpApp}(t) & = & (\exists t', S.\ t = t'[S]) \\
\mathit{App}(t) & = & \mathit{TmApp}(t) \vee \mathit{TpApp}(t)
\end{array}
$$

$$
\begin{array}{rcl}
\mathit{WF}(\Gamma, T) & = & (\mathit{FV}(T) - \mathit{DTV}(\Gamma) = \varnothing) \\
\mathit{WF}(\Gamma, ?) & = & \mathit{True} \\
\mathit{WF}(\Gamma, ?\!\to P) & = & \mathit{WF}(\Gamma, P)
\end{array}
$$

$$
\begin{array}{rcl}
\mathit{DTV}(\cdot) & = & \varnothing \\
\mathit{DTV}(\Gamma, X) & = & \mathit{DTV}(\Gamma) \cup \{X\} \\
\mathit{DTV}(\Gamma, x\!:\!T) & = & \mathit{DTV}(\Gamma)
\end{array}
$$

$$
\begin{array}{rcl}
\mathit{MV}(\Gamma, p) & = & \varnothing \text{ when } \neg\mathit{App}(p) \\
\mathit{MV}(\Gamma, p[X]) & = & \mathit{MV}(\Gamma, p) \cup \{X\} \text{ when } X \notin \mathit{DTV}(\Gamma) \\
\mathit{MV}(\Gamma, p[S]) & = & \mathit{MV}(\Gamma, p) \text{ when } \mathit{WF}(\Gamma, S) \\
\mathit{MV}(\Gamma, p\ e) & = & \mathit{MV}(\Gamma, p)
\end{array}
$$

$$
\begin{array}{rcl}
\lfloor \lambda\, x\!:\!T.\, e \rfloor & = & \{\lambda\, x\!:\!T.\, t \mid t \in \lfloor e \rfloor\} \cup \{\lambda\, x.\, t \mid t \in \lfloor e \rfloor\} \\
\lfloor \Lambda\, X.\, e \rfloor & = & \{\Lambda\, X.\, t \mid t \in \lfloor e \rfloor\} \\
\lfloor e\ e' \rfloor & = & \{t\ t' \mid t \in \lfloor e \rfloor_a \wedge t' \in \lfloor e' \rfloor\} \\
\lfloor e[S] \rfloor & = & \{t[S] \mid t \in \lfloor e \rfloor\}
\end{array}
$$

$$
\begin{array}{rcl}
\lfloor e[S] \rfloor_a & = & \{t \mid t \in \lfloor e \rfloor_a\} \cup \{t[S] \mid t \in \lfloor e \rfloor\} \\
\lfloor e \rfloor_a & = & \lfloor e \rfloor \text{ otherwise}
\end{array}
$$

$$
\begin{array}{rcl}
\lfloor S \to W \rfloor & = & S \to \lfloor W \rfloor \\
\lfloor \forall\, X\!=\!R.\, W \rfloor & = & \forall\, X.\, \lfloor W \rfloor \\
\lfloor (X, ?\!\to P) \rfloor & = & X
\end{array}
$$

$$
\begin{array}{rcl}
\mathit{arr}_P(?) = \mathit{arr}_P(T) & = & 0 \\
\mathit{arr}_P(?\!\to P) & = & 1 + \mathit{arr}_P(P) \\
\mathit{arr}_W((X, ?\!\to P)) = \mathit{arr}_W(T) & = & 0 \\
\mathit{arr}_W(\forall\, X\!=\!R.\, W) & = & \mathit{arr}_W(W) \\
\mathit{arr}_W(S \to W) & = & 1 + \mathit{arr}_W(W)
\end{array}
$$

1.4 Bidirectional Rules

$$\boxed{\Gamma \vdash_\delta t : T \rightsquigarrow e} \qquad \frac{}{\Gamma \vdash_\delta x : \Gamma(x) \rightsquigarrow x}\ Var \qquad \frac{\Gamma, x{:}T \vdash_\Downarrow t : S \rightsquigarrow e}{\Gamma \vdash_\Downarrow \lambda\, x.\, t : T \to S \rightsquigarrow \lambda\, x{:}T.\, e}\ Abs$$

$$\frac{\Gamma, x{:}T \vdash_\delta t : S \rightsquigarrow e}{\Gamma \vdash_\delta \lambda\, x{:}T.\, t : T \to S \rightsquigarrow \lambda\, x{:}T.\, e}\ AAbs \qquad \frac{\Gamma, X \vdash_\delta t : T \rightsquigarrow e}{\Gamma \vdash_\delta \Lambda\, X.\, t : \forall X.T \rightsquigarrow \Lambda\, X.\, e}\ TAbs \qquad \frac{\Gamma \vdash_\Uparrow t : \forall X.T \rightsquigarrow e}{\Gamma \vdash_\delta t[S] : [S/X]T \rightsquigarrow e[S]}\ TApp$$

$$\frac{\Gamma; ? \vdash^{\mathrm{I}} t\ t' : T \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, e) = \varnothing}{\Gamma \vdash_\Uparrow t\ t' : T \rightsquigarrow e}\ AppSyn \qquad \frac{\Gamma; \sigma\ T \vdash^{\mathrm{I}} t\ t' : T \rightsquigarrow (p, \sigma) \quad MV(\Gamma, p) = dom(\sigma)}{\Gamma \vdash_\Downarrow t\ t' : \sigma\ T \rightsquigarrow \sigma\ p}\ AppChk$$

Figure 1: Bidirectional inference rules with elaboration

1.5 Specificational Rules

(a) Shim (specification)

$$T_? ::= T \mid ? \qquad \frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T \leadsto (p, \sigma) \quad MV(\Gamma, T) = dom(\sigma)}{\Gamma; T_? \vdash^{\mathrm{I}} t\ t' : T \leadsto (p, \sigma)}$$

(b) $\boxed{\Gamma \vdash^{\mathrm{P}} t : T \leadsto (p, \sigma)}$

$$\frac{\neg App(t) \quad \Gamma \vdash_{\Uparrow} t : T \leadsto e}{\Gamma \vdash^{\mathrm{P}} t : T \leadsto (e, \sigma_{id})}\ PHead \qquad \frac{\Gamma \vdash^{\mathrm{P}} t : \forall X.T \leadsto (p, \sigma)}{\Gamma \vdash^{\mathrm{P}} t[S] : [S/X]T \leadsto (p[S], \sigma)}\ PTApp$$

$$\frac{\Gamma \vdash^{\mathrm{P}} t : T \leadsto (p, \sigma) \quad \Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \leadsto (p', \sigma')}{\Gamma \vdash^{\mathrm{P}} t\ t' : T' \leadsto (p', \sigma')}\ PApp$$

(c) $\boxed{\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \leadsto (p', \sigma')}$

$$\frac{\sigma'' \in \{\sigma, [S/X] \circ \sigma\} \quad WF(\Gamma, S) \quad \Gamma \vdash^{\cdot} (p[X]{:}T, \sigma'') \cdot t' : T' \leadsto (p', \sigma')}{\Gamma \vdash^{\cdot} (p{:}\forall X.T, \sigma) \cdot t' : T' \leadsto (p', \sigma')}\ PForall \qquad \frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \vdash_{\Downarrow} t' : \sigma\ S \leadsto e'}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : T \leadsto (p\ e', \sigma)}\ PChk$$

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \overline{[U/Y]}\ \sigma\ S \leadsto e'}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : \overline{[U/Y]}\ T \leadsto ((\overline{[U/Y]}\ p)\ e', \sigma)}\ PSyn$$

Figure 2: Specification for contextual type-argument inference

1.6 Algorithmic Rules

(a) Shim (algorithm)

$$T_? ::= T \mid ? \qquad \frac{\Gamma; T_? \Vdash^? t\ t' : T \rightsquigarrow (p, \sigma)}{\Gamma; T_? \vdash^{\mathrm{I}} t\ t' : T \rightsquigarrow (p, \sigma)}$$

(b) $\boxed{\Gamma; P \Vdash^? t : W \rightsquigarrow (p, \sigma)}$

$$\frac{\neg App(t) \quad \Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e \quad \varnothing \Vdash^{:=} T := ? \to P \Rightarrow (\sigma_{id}, W)}{\Gamma; ? \to P \Vdash^? t : W \rightsquigarrow (e, \sigma_{id})}\ ?Head \qquad \frac{\Gamma; ? \to P \Vdash^? t : \forall X{=}R.\, W \rightsquigarrow (p, \sigma) \quad R \in \{X, S\}}{\Gamma; ? \to P \Vdash^? t[S] : [S/X]W \rightsquigarrow (p[S], \sigma)}\ ?TApp$$

$$\frac{\Gamma; ? \to P \Vdash^? t : W \rightsquigarrow (p, \sigma) \quad \Gamma \Vdash^{\cdot} (p{:}W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')}{\Gamma; P \Vdash^? t\ t' : W' \rightsquigarrow (p', \sigma')}\ ?App$$

(c) $\boxed{\Gamma \Vdash^{\cdot} (p{:}W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')}$

$$\frac{\sigma'' = \text{ if } R{=}X \text{ then } \sigma \text{ else } [R/X]{\circ}\sigma \quad \Gamma \Vdash^{\cdot} (p[X]{:}W, \sigma'') \cdot t' : W' \rightsquigarrow (p', \sigma')}{\Gamma \Vdash^{\cdot} (p{:}\forall X{=}R.\, W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')}\ ?Forall \qquad \frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \Vdash_{\Downarrow} t' : S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p{:}S \to W, \sigma) \cdot t' : W \rightsquigarrow (p\ e', \sigma)}\ ?Chk$$

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \Vdash_{\Uparrow} t : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p{:}S \to W) \cdot t' : \overline{[U/Y]}\ W \rightsquigarrow ((\overline{[U/Y]}\ p)\ e', \sigma)}\ ?Syn$$

(d) $\boxed{\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)}$

$$\frac{\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)}{\overline{X} \Vdash^{:=} S \to T := ?{\to} P \Rightarrow (\sigma, S \to W)}\ MArr \qquad \frac{\overline{X}, X \Vdash^{:=} T := ?{\to} P \Rightarrow (\sigma, W)}{\overline{X} \Vdash^{:=} \forall X.\, T := ?{\to} P \Rightarrow (\sigma - X, \forall X{=}\sigma(X).\, W)}\ MForall$$

$$\frac{\overline{Y} = FV(T) \cap \overline{X} \quad FV(\overline{U}) \cap (BTV(S) \cup \overline{X}) = \varnothing \quad \overline{[U/Y]}\ T = S}{\overline{X} \Vdash^{:=} T := S \Rightarrow (\overline{[U/Y]}, T)}\ MType$$

$$\frac{}{\overline{X} \Vdash^{:=} T := ? \Rightarrow (\sigma_{id}, T)}\ M? \qquad \frac{X \in \overline{X}}{\overline{X} \Vdash^{:=} X := ?{\to} P \Rightarrow (\sigma_{id}, (X, ?{\to} P))}\ MCurr$$

Figure 3: Algorithm for contextual type argument inference

2 Termination of Algorithmic Rules

The inference rules presented in 1.6 are terminating and deriving these judgments is decidable

Theorem 1. *(Decidability of Typing):*

1. *For any context* Γ *and term* t*, it is decidable whether* $\Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e$ *for some* T *and* e
2. *For any context* Γ*, term* t*, and type* T*, it is decidable whether* $\Gamma \Vdash_{\Downarrow} t : T \rightsquigarrow e$ *for some* e
3. *For any context* Γ*, prototype* P*, and term* t*, it is decidable whether* $\Gamma; P \Vdash^{?} t : W \rightsquigarrow (p, \sigma)$ *for some* W*,* p*, and* σ
4. *For any context* Γ*, terms* p *and* t'*, decorated type* W*, and substitution* σ*, it is decidable whether* $\Gamma \Vdash^{\cdot} (p\!:\!W, \sigma)\cdot t' : W' \rightsquigarrow (p', \sigma')$ *for some* W'*,* p'*, and* σ'
5. *For any set of meta-variables* $\overline{X}$*, type* T*, and prototype* P*, it is decidable whether* $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ *for some* σ *and* W

Proof . The proof is a straightforward mutual induction resp. on the size of

1. the subject of typing t
2. the subject of typing t
3. the subject of typing t
4. the decorated type W (that annotates p)
5. the prototype P

3 Soundness of $\vdash_\delta$ wrt $\vdash$

Our soundness statement for the external language is that every well-typed term of the external language elaborates to a well-typed term of the internal language, and it is proven using mutual induction on the following three theorems.

1. If $\Gamma \vdash_\delta t : T \rightsquigarrow e$ then $\Gamma \vdash e : T$
2. If $\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (p, \sigma)$ then $\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$
3. If $\Gamma \vdash^{\cdot} (p\!:\!T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ and $\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$ where
   - $dom(\sigma) \subseteq MV(\Gamma, p)$
   - For all $X \in dom(\sigma), WF(\Gamma, \sigma(X))$
   - $dom(\sigma) \cap BTV(cod(\sigma)) = \varnothing$

   then $\Gamma, MV(\Gamma, \sigma'\ p') \vdash \sigma'\ p' : \sigma'\ T'$

3.1 Bidirectional Rules

Theorem 2. *(Soundness of* $\vdash_\delta$*):*
If $\Gamma \vdash_\delta t : T \rightsquigarrow e$ *then* $\Gamma \vdash e : T$

Proof. By mutual induction of the assumed derivation.

Case Var**:** Directly from assumption

$$\frac{}{\Gamma \vdash x : \Gamma(x)}\ Var$$

Case $AAbs$**:** Our assumed derivation is

$$\frac{\Gamma, x\!:\!T \vdash_\delta t : S \rightsquigarrow e}{\Gamma \vdash_\delta \lambda\, x\!:\!T.\, t : T \to S \rightsquigarrow \lambda\, x\!:\!T.\, e}\ AAbs$$

Invoking the IH on the premise we get $\Gamma, x\!:\!T \vdash e : S$ so we can conclude with

$$\frac{\Gamma, x\!:\!T \vdash e : S}{\Gamma \vdash \lambda\, x\!:\!T.\, e : T \to S}\ FAbs$$

Case *Abs***:** Similar to *AAbs*, invokding the IH on the premise (specialized to $\vdash_{\Downarrow}$) and using *FAbs*.

Case *TAbs***:** Similar to *AAbs*, invoking the IH on the premise and using *FTAbs*

Case *TApp***:** Similar to *AAbs* and *TAbs*, invoking the IH on the premise (specialied to $\vdash_{\Uparrow}$) and using *FTApp*.

Case *AppSyn***:** Our assumed derivation (after in-lining judgment $\vdash^{\mathrm{I}}$) is

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t : T \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, e) = MV(\Gamma, T) = \varnothing}{\Gamma \vdash_{\Uparrow} t\ t' : T \rightsquigarrow e}\ AppSyn$$

By mutual induction on Theorem 3 (soundness of $\vdash^{\mathrm{P}}$) on the first premise, we have

$\Gamma, MV(\Gamma, e) \vdash \sigma_{id}\ e : \sigma_{id}\ T$

which after a little re-writing gives us

$\Gamma \vdash e : T$

which is what we need.

Caase *AppChk***:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (p, \sigma) \quad MV(\Gamma, p) = MV(\Gamma, T) = dom(\sigma)}{\Gamma \vdash_{\Downarrow} t\ t' : \sigma\ T \rightsquigarrow \sigma\ p}\ AppChk$$

By mutual induction on Theorem 3 (soundness of $\vdash^{\mathrm{P}}$) on the first premise, we have

$\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$

Since we know (from the second premise of our assumed derivation) that $MV(\Gamma, p) = dom(\sigma)$, we know that $MV(\Gamma, \sigma\ p) = \varnothing$, so we can rewrite to

$\Gamma \vdash \sigma\ p : \sigma\ T$

which is what we need. □

3.2 Partial Synthesis Rules

Theorem 3. *(Soundness of $\vdash^{P}$):*
If $\Gamma \vdash^{P} t : T \rightsquigarrow (p, \sigma)$ then $\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$

Proof. By mutual induction on the assumed derivation.

Case *PHead***:** Our assumed derivation is

$$\frac{\neg App(t) \qquad \Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e}{\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (e, \sigma_{id})}\ PHead$$

By mutual induction on the soundness of $\vdash_{\Uparrow}$ on the second premise we get

$\Gamma \vdash e : T$

Since e is well-typed under Γ using the internal typing rules it has no metavariables. Therefore, $MV(\Gamma, e) = \varnothing$, and we conclude

$\Gamma, \varnothing \vdash \sigma_{id}\ e : \sigma_{id}\ T$

Case *PTApp***:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t : \forall X.T \rightsquigarrow (p, \sigma)}{\Gamma \vdash^{\mathrm{P}} t[S] : [S/X]T \rightsquigarrow (p[S], \sigma)}\ ITApp$$

By the IH on our premise we get

$\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ \forall X.T$

Implicit here is that $WF(\Gamma, S)$, so $MV(\Gamma, p[S]) = MV(\Gamma, p)$, and bound X is fresh w.r.t. Γ, p, and σ, so $\sigma\ \forall X.T = \forall X.\sigma\ T$. We conclude

$$\frac{\Gamma, MV(\Gamma, \sigma\ p[S]) \vdash \sigma\ p : \forall X.\sigma\ T}{\Gamma, MV(\Gamma, \sigma\ p[S]) \vdash \sigma\ p[S] : [S/X]\ \sigma\ T}\ FTApp$$

Case $PApp$**:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (p, \sigma) \quad \Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')}{\Gamma \vdash^{\mathrm{P}} t\ t' : T' \rightsquigarrow (p', \sigma')}\ IApp$$

By the IH on the premise we have

$\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$

With this and with the second premise of our assumed derivation, we need to invoke mutual induction on Theorem 4 (soundness of $\vdash^{\cdot}$) to get

$\Gamma, MV(\Gamma, \sigma'\ p') \vdash \sigma'\ p' : \sigma'\ T'$.

To do so, we must meet the pre-requisite of Theorem 4: $dom(\sigma) \subseteq MV(\Gamma, p)$ for all X, $WF(\Gamma, \sigma(X))$, and that $dom(\sigma) \cap BVT(cod(\sigma)) = \varnothing$. The first two of these we have from Lemma 1, and the last of these we have from Lemma 2. $\square$

3.3 Partial Application Rules

Theorem 4. *(Soundness of $\vdash^{\cdot}$ wrt $\vdash$): If $\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ and $\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ T$ where*

- $dom(\sigma) \subseteq MV(\Gamma, p)$

  *Our solution set σ really solves **meta-variables**.*

- *For all $X \in dom(\sigma), WF(\Gamma, \sigma(X))$*

  *Our solution set σ **really solves** meta-variables.*

- $dom(\sigma) \cap BTV(cod(\sigma)) = \varnothing$

  No meta-variables are ever generated by solutions in σ

then $\Gamma, MV(\Gamma, \sigma'\ p') \vdash \sigma'\ p' : \sigma'\ T'$

Proof. By mutual induction on the assumed derivation of $\vdash^{\cdot}$.

Case $PForall$**:** Our assumed derivations are

$$\frac{\sigma'' \in \{\sigma, [S/X] \circ \sigma\}, WF(\Gamma, S) \quad \Gamma \vdash^{\cdot} (p[X]{:}T, \sigma'') \cdot t' : T' \rightsquigarrow (p', \sigma')}{\Gamma \vdash^{\cdot} (p{:}\forall X.T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')}\ PForall \text{ and } \Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ \forall X.T$$

We perform case analysis on σ'': either $\sigma'' = \sigma$ or $\sigma'' = \sigma \circ [S/X]$. If it is the former, then since X is fresh wrt σ we have $MV(\Gamma, \sigma''\ p[X]) = MV(\Gamma, \sigma\ p) \cup \{X\}$ and $\sigma''\ p[X] = \sigma\ p[X]$. We have by weakening

$$\frac{\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ \forall X.T}{\Gamma, MV(\Gamma, \sigma''\ p[X]) \vdash \sigma''\ p : \sigma\ \forall X.T}\ Weaken$$

If $\sigma'' = [S/X] \circ \sigma$ them we have $MV(\Gamma, \sigma''\ p[X]) = MV(\Gamma, \sigma\ p[S]) = MV(\Gamma, \sigma\ p)$ and we need only rewrite our second assumed derivation to $\Gamma, MV(\Gamma, \sigma''\ p[X]) \vdash \sigma''\ p : \sigma\ \forall X.T$

In both cases, we can derive

$$\frac{\Gamma, MV(\Gamma, \sigma''\ p) \vdash \sigma''\ p : \sigma\ \forall X.T}{\Gamma, MV(\Gamma, \sigma''\ p[X]) \vdash \sigma''\ p[X] : \sigma''\ T}\ TApp^F$$

We are now ready to invoke the IH with this and with the second premise of our assumed derivation of $\vdash^{\cdot}$ to derive

$\Gamma, MV(\Gamma, \sigma'\ p' \vdash \sigma'\ p' : \sigma'\ T')$

which is what we need to conclude. (Note that our third condition is satisfied for σ'' since bound variable X occurs before applying substitution σ.)

Case *PChk***:** Our assumed derivations are

$$\dfrac{MV(\Gamma, \sigma\ S) = \varnothing \qquad \Gamma \vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : T \rightsquigarrow (p\ e', \sigma)}\ PChk \text{ and } \Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ S \to T$$

By mutual induction on Theorem 2 (soundness of $\vdash_\delta$ *wrt* $\vdash$) and by weakening we have

$$\dfrac{\dfrac{\Gamma \vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow e'}{\Gamma \vdash e' : \sigma\ S}\ Sound \vdash_{\Downarrow}}{\Gamma, MV(\Gamma, \sigma\ p) \vdash e' : \sigma\ S}\ Weaken$$

With this and our second assumption, the derivation of $\vdash$, we can conclude

$$\dfrac{\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ S \to T \quad \Gamma, MV(\Gamma, \sigma\ p) \vdash e' : \sigma\ S}{\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ (p\ e') : \sigma\ T}\ App$$

noting that since e' is well-typed under Γ, $\sigma\ e' = e'$.

Case *PSyn***:** Our assumed derivations are

$$\dfrac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : \overline{[U/Y]}\ T \rightsquigarrow (\overline{[U/Y]}\ (p\ e'), \sigma)}\ PSyn \quad \text{and } \Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ S \to T$$

By mutual induction on Theorem 2 (soundness of $\vdash_{\Uparrow}$) and weakening on the second premise of our assumed derivation of $\vdash^{\cdot}$ we have

$$\dfrac{\dfrac{\Gamma \vdash_{\Uparrow} t' : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e'}{\Gamma \vdash e' : \overline{[U/Y]}\ \sigma\ S}\ Sound \vdash_{\Uparrow}}{\Gamma, MV(\Gamma, \overline{[U/Y]}\ \sigma\ p) \vdash e' : \overline{[U/Y]}\ \sigma\ S}\ Weaken$$

Let $\sigma'' = \overline{[U/Y]} \circ \sigma$. By appeal to Lemma 3 on the typeability of substituting solutions in for meta-variables (whose pre-conditions that $dom(\sigma) \subseteq MV(\Gamma, p)$ and that for all X, $WF(\Gamma, \sigma(X))$ we are able to satisfy by assumption) we have

$$\dfrac{\Gamma, MV(\Gamma, \sigma\ p) \vdash \sigma\ p : \sigma\ S \to T}{\Gamma, MV(\Gamma, \sigma''\ p) \vdash \sigma''\ p : \sigma''\ S \to T}\ Lemma3$$

From this and rule *App* from $\vdash$ we can derive

$$\dfrac{\Gamma, MV(\Gamma, \sigma''\ p) \vdash \sigma''\ p : \sigma''\ S \to T \quad \Gamma, MV(\Gamma, \sigma''\ p) \vdash e' : \sigma''\ S}{\Gamma, MV(\Gamma, \sigma\ \overline{[U/Y]}\ (p\ e')) \vdash \sigma\ \overline{[U/Y]}\ (p\ e') : \sigma\ \overline{[U/Y]}T}\ App$$

Which is what we need to conclude. Note that the re-arrangment of $\overline{[U/Y]}\ \sigma\ T$ to $\sigma\ \overline{[U/Y]}\ T$ is justified by the assumption that $dom(\sigma) \cap BTV(cod(\sigma)) = \varnothing$, as no meta-variables (including any in $\overline{Y}$) can be introduced by the bound type variables of some solution in σ. □

3.4 Lemma: Well-formed and well-scoped solutions

Lemma 1.

- *If* $\Gamma \vdash^P t : T \rightsquigarrow (p, \sigma)$ *then* $dom(\sigma) \subseteq MV(\Gamma, p)$ *and for all* $X \in dom(\sigma), WF(\Gamma, \sigma(X))$.
- *If* $dom(\sigma) \subseteq MV(\Gamma, p)$ *and for all* $X \in dom(\sigma), WF(\Gamma, \sigma(X))$*, and if* $\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$*, then* $dom(\sigma') \subseteq MV(\Gamma, p')$ *and for all* $X \in dom(\sigma'), WF(\Gamma, \sigma'(X))$

Proof. Straightfoward induction on the assumed derivation where the first invokes the second. □

3.5 Lemma: σ introduces no meta-variables

Lemma 2.

- *If* $\Gamma \vdash^{P} t : T \rightsquigarrow (p, \sigma)$ *then* $dom(\sigma) \cap BTV(cod(\sigma)) = \varnothing$
- *If* $\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ *and* $dom(\sigma) \cap BTV(cod(\sigma)) = \varnothing$ *then* $dom(\sigma') \cap BTV(cod(\sigma')) = \varnothing$

Proof. By induction on the assumed derivations where the first invokes the second. Note that only rule $PForall$ adds any meta-variable solutions and that these are generated from a type that does not have access to solutions in the input substitution σ. ☐

3.6 Lemma: Sound use of σ on $\vdash$

Lemma 3. *If* $\Gamma, MV(\Gamma, p) \vdash p : T$ *and* $dom(\sigma) \subseteq MV(\Gamma, p)$ *and for all* $X \in dom(\sigma), WF(\Gamma, \sigma(X))$*, then* $\Gamma, (\overline{X} - dom(\sigma)) \vdash \sigma\, p : \sigma\, T$

Proof. By induction on the assumed derivation. ☐

3.7 Lemma: Well-formed Partial Types

Lemma 4.

- *If* $\Gamma \vdash^{P} t : T \rightsquigarrow (p, \sigma)$ *then* $WF(\Gamma', T)$ *where* $\Gamma' = \Gamma, MV(\Gamma, p)$.
- *If* $\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ *and* $WF(\Gamma', T)$ *(where* $\Gamma' = \Gamma, MV(\Gamma, p)$*),*
  then $WF(\Gamma', T')$.

Proof. By a similar argument to Theorem 3 and Theorem 4, we can strengthen the two theorems above to yield:

- If $\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (p, \sigma)$ then $\Gamma, MV(\Gamma, p) \vdash p : T$
- If $\Gamma \vdash^{\cdot} (p{:}T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ and $\Gamma, MV(\Gamma, p) \vdash p : T$ then $\Gamma, MV(\Gamma, p') \vdash p' : T'$

and from there, reason that any term well-typed by the internal typing rules was typed with a well-formed type. ☐

4 Soundness of $\Vdash_\delta$ wrt $\vdash_\delta$

Soundness of the algorithmic rules means that any external term typeable with the algorithmic rules is also typeable with the specificational rules, and is shown by mutual induction on the following three theorems:

1. If $\Gamma \Vdash_\delta t : T \rightsquigarrow e$ then $\Gamma \vdash_\delta t : T \rightsquigarrow e$
2. If $\Gamma; P \Vdash^{?} t : W \rightsquigarrow (p, \sigma)$ then $\Gamma \vdash^{\mathrm{P}} t : \lfloor W \rfloor \rightsquigarrow (p, \sigma)$
3. If $\Gamma \Vdash^{\cdot} (p{:}W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')$
   and $MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor :=\, ? \rightarrow P \Rightarrow (\sigma, W)$ with $WF(\Gamma, ? \rightarrow P)$
   then $\Gamma \vdash^{\cdot} (p{:}\lfloor W \rfloor, \sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p', \sigma')$

Where $\Vdash_\delta$ indicates the bidirectional rules using the shim judgment defined in Figure 3a.

4.1 Bidirectional Rules

Theorem 5. *If* $\Gamma \Vdash_\delta t : T \rightsquigarrow e$ *then* $\Gamma \vdash_\delta t : T \rightsquigarrow e$

Proof. By straightforward induction on the assumed derivation. The rules of the two systems are identical except for *AppSyn* and *AppChk*, so only these are shown.

Case ***AppSyn*:** Our assumed derivation is

$$\frac{\Gamma; ? \Vdash^{?} t\ t' : T \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, e) = \varnothing}{\Gamma \Vdash_{\Uparrow} t\ t' : T \rightsquigarrow e}\ AppSyn$$

By mutual induction of sound $\Vdash^{?}$ on the first premise, we have

$\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (e', \sigma_{id})$ (since $\lfloor T \rfloor = T$).

We now need to satisfy the specificational condition that $MV(\Gamma, T) = dom(\sigma_{id}) = \varnothing$. We have via Lemma 4 (well-formedness of synthesized partial types) that $WF(\Gamma, T)$ which guarantees this. We can now conclude

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, T) = MV(\Gamma, e) = \varnothing}{\Gamma \vdash_{\Uparrow} t\ t' : T \rightsquigarrow e}\ AppSyn$$

Case *AppChk***:** Our assumed derivation is

$$\frac{\Gamma; T \Vdash^{?} t\ t' : T \rightsquigarrow (p, \sigma) \qquad MV(\Gamma, p) = dom(\sigma)}{\Gamma \Vdash_{\Downarrow} t\ t' : \sigma\ T \rightsquigarrow \sigma\ p}\ AppChk$$

By mutual induction on Theorem 6 (Soundness of prototype synthesis) on the first premise, we have

$$\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (p, \sigma)$$

The last condition we need to meet for the specificational version of *AppChk* is that $MV(\Gamma, T) = dom(\sigma)$. We first note that by Lemma 4 that $WF(\Gamma', T)$ (where $\Gamma' = \Gamma, MV(\Gamma, p)$). Next, we invoke Lemma 6 (prototype synthesis preserves matching) to get

$$MV(\Gamma, p) \Vdash^{:=} T := \sigma\ T \Rightarrow (\sigma, T)$$

By inversion, the only rule that could form this match is *MType*, which after a little rewriting in terms of meta-variables and σ gives us:

$$\frac{dom(\sigma) = FV(T) \cap MV(\Gamma, p)}{MV(\Gamma, p) \Vdash^{:=} T := \sigma\ T \Rightarrow (\sigma, T)}\ MType$$

and this premise is equivalent to saying $dom(\sigma) = MV(\Gamma, T)$ (since by Lemma 1 we have that $WF(\Gamma', T)$ where $\Gamma' = \Gamma, MV(\Gamma, p)$). We can conclude

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (p, \sigma) \quad MV(\Gamma, p) = MV(\Gamma, T) = dom(\sigma)}{\Gamma \vdash_{\Downarrow} t\ t' : \sigma\ T \rightsquigarrow \sigma\ p}\ AppChk$$

□

4.2 Prototype Rules

Theorem 6. *If* $\Gamma; P \Vdash^{?} t : W \rightsquigarrow (p, \sigma)$ *then* $\Gamma \vdash^{P} t : \lfloor W \rfloor \rightsquigarrow (p, \sigma)$

Proof. By induction on the assumed derivation.

Case $?Head$**:** Our assumed derivation is

$$\frac{\neg App(t) \qquad \Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e \qquad \varnothing \Vdash^{:=} T := ? \Rightarrow P \Rightarrow (\sigma_{id}, W)}{\Gamma; ? \Rightarrow P \Vdash^{?} t : W \rightsquigarrow (e, \sigma_{id})}\ ?Head$$

By mutual induction on the soundness of $\Vdash_{\Uparrow}$ we have

$$\frac{\Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e}{\Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e}\ Theorem2$$

We now to construct

$$\frac{\neg App(t) \quad \Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e}{\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (e, \sigma_{id})}\ IHead$$

Case $?TApp$**:** Our assumed derivation is

$$\frac{\Gamma; ? \rightarrow P \Vdash^{?} t : \forall\, X{=}R.\, W \rightsquigarrow (p, \sigma) \quad R \in \{X, S\}}{\Gamma; ? \rightarrow P \Vdash^{?} t[S] : [S/X]W \rightsquigarrow (p[S], \sigma)}\ ?TApp$$

By the IH on the first premise we have

$$\Gamma \vdash^{\mathrm{P}} t : \forall X. \lfloor W \rfloor \rightsquigarrow (p, \sigma)$$

We can conclude with

$$\frac{\Gamma \vdash^{\mathrm{P}} t : \forall X. \lfloor W \rfloor \rightsquigarrow (p, \sigma)}{\Gamma \vdash^{\mathrm{P}} t[S] : \lfloor [S/X] W \rfloor \rightsquigarrow (p[S], \sigma)}\ ITApp$$ where its clear that $[S/X]\ \lfloor W \rfloor = \lfloor [S/X]\ W \rfloor$

Case $?App$ Our assumed derivation is

$$\frac{\Gamma; ? \to P \Vdash^{?} t : W \rightsquigarrow (p,\sigma) \quad \Gamma \Vdash^{\cdot} (p\!:\!W,\sigma) \cdot t' : W' \rightsquigarrow (p',\sigma)'}{\Gamma; P \Vdash^{?} t\ t' : W' \rightsquigarrow (p',\sigma)'}\ ?App$$

By the IH on the first premise we have

$\Gamma \vdash^{\mathrm{P}} t : \lfloor W \rfloor \rightsquigarrow (p,\sigma)$

by Lemma 6 we can derive

$MV(\Gamma,p) \Vdash^{:=} \lfloor W \rfloor :=? \to P \Rightarrow (\sigma, W)$

This match lets us invoke mutual induction on Theorem 7 (soundness of $\Vdash^{\cdot}$) on the second premise, and we have

$\Gamma \vdash^{\cdot} (p\!:\!\lfloor W \rfloor,\sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma)'$

We can conclude with

$$\frac{\Gamma \vdash^{\mathrm{P}} t : \lfloor W \rfloor \rightsquigarrow (p,\sigma) \qquad \Gamma \vdash^{\cdot} (p\!:\!\lfloor W \rfloor,\sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma)'}{\Gamma \vdash^{\mathrm{P}} t\ t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma)'}\ IApp$$

□

4.3 Prototype Application Rules

Theorem 7. *If* $\Gamma \Vdash^{\cdot} (p\!:\!W,\sigma) \cdot t' : W' \rightsquigarrow (p',\sigma')$ *and* $MV(\Gamma,p) \Vdash^{:=} \lfloor W \rfloor :=? \to P \Rightarrow (\sigma, W)$ *with* $WF(\Gamma, ? \to P)$ *then* $\Gamma \vdash^{\cdot} (p\!:\!\lfloor W \rfloor,\sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma')$

Proof. By induction on the assumed derivation of $\Vdash^{\cdot}$

Case $?Forall$**:** Our assumed derivation for $\Vdash^{\cdot}$ is

$$\frac{\sigma'' = \text{if } R\!=\!X \text{ then } \sigma \text{ else } [R/X] \circ \sigma \quad \Gamma \Vdash^{\cdot} (p[X]\!:\!W,\sigma'') \cdot t' : W' \rightsquigarrow (p',\sigma')}{\Gamma \Vdash^{\cdot} (p\!:\!\forall X\!=\!R.\, W,\sigma) \cdot t' : W' \rightsquigarrow (p',\sigma')}\ ?Forall$$

and our assumed match is

$MV(\Gamma,p) \Vdash^{:=} \forall X.\, \lfloor W \rfloor :=? \to P \Rightarrow (\sigma, \forall X\!=\!R.\, W)$

The only rule that could result in this conclusion is $MForall$, whose premise is

$MV(\Gamma,p), X \Vdash^{:=} \lfloor W \rfloor := ? \to P \Rightarrow ([R/X] \circ \sigma, W)$

We appeal to Lemma 7 on the well-formedness of solutions in $[R/X] \circ \sigma$ to get $R = X$ or $WF(\Gamma,R)$. This makes R a legal guess for our specificational system. Now we invoke the IH (using the match directly above and the second premise of our assumed derivation of $\vdash^{\cdot}$) to get

$\Gamma \vdash^{\cdot} (p[X]\!:\!\lfloor W \rfloor, [R/X] \circ \sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma)'$

allowing us to conclude $\Gamma \vdash^{\cdot} (p\!:\!\forall X.\, \lfloor W \rfloor,\sigma) \cdot t' : \lfloor W' \rfloor \rightsquigarrow (p',\sigma')$

Case $?Chk$ Our assumed derivation is

$$\frac{MV(\Gamma,\sigma\ S) = \varnothing \quad \Gamma \Vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p\!:\!S \to W,\sigma) \cdot t' : W \rightsquigarrow (p\ e',\sigma)}\ ?Chk$$

and our assumed match is

$MV(\Gamma,p) \Vdash^{:=} S \to \lfloor W \rfloor :=? \to P \Rightarrow (\sigma, S \to W)$

By mutual induction on Theorem 5 (soundness of $\Vdash_\delta$) on the second premise we have $\Gamma \vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow e$. We can conclude

$$\frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow p'}{\Gamma \vdash^{\cdot} (p, S \rightarrow \lfloor W \rfloor) \cdot t' : \lfloor W \rfloor \rightsquigarrow (p\ e', \sigma)}\ PChk$$

Case $?Syn$ Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \Vdash_{\Uparrow} t : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p{:}S \rightarrow W, \sigma) \cdot t' : \overline{[U/Y]}\ W \rightsquigarrow ((\overline{[U/Y]}\ p)\ e', \sigma)}\ ?Syn$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} S \rightarrow \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, S \rightarrow W)$

By mutual induction on the soundness of $\Vdash_{\Uparrow}$ we have

$\Gamma \vdash_{\Uparrow} t : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e$

which allows us to conclude

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \overline{[U/Y]} \circ \sigma\ S \rightsquigarrow e}{\Gamma \vdash^{\cdot} (p{:}S \rightarrow \lfloor W \rfloor) \cdot t' : \lfloor \overline{[U/Y]}\ W \rfloor \rightsquigarrow (\overline{[U/Y]}\ (p\ e'), \sigma)}\ PSyn$$ □

4.4 Lemma: Sound decoration erasure

Lemma 5. *If* $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ *then* $\lfloor W \rfloor = T$

Proof. Straightforward induction on the assumed derivation. □

4.5 Lemma: Sound $\Vdash^{?}$ wrt $\Vdash^{:=}$

Lemma 6. *If* $\Gamma; P \Vdash^{?} t : W \rightsquigarrow (p, \sigma)$ *then* $MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor := P \Rightarrow (\sigma, W)$

Proof. By induction on the assumed derivation

Case $?Head$ Our assumed derivation is

$$\frac{\neg App(t) \quad \Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e \quad \varnothing \Vdash^{:=} T := ? \rightarrow P \Rightarrow (\sigma_{id}, W)}{\Gamma; ? \rightarrow P \Vdash^{?} t : W \rightsquigarrow (e, \sigma_{id})}\ ?Head$$

We apply Lemma 5 on the third hypothesis to get

$MV(\Gamma, e) = \varnothing \Vdash^{:=} \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma_{id}, W)$

which is what we need.

Case $?TApp$ Our assumed derivation is

$$\frac{\Gamma; ? \rightarrow P \Vdash^{?} t : \forall X{=}R.W \rightsquigarrow (p, \sigma) \quad R \in \{X, S\}}{\Gamma; ? \rightarrow P \Vdash^{?} t[S] : [S/X]\ W \rightsquigarrow (p[S], \sigma)}\ ?TApp$$

We invoke the IH on the first premise, yielding

$MV(\Gamma, p) \Vdash^{:=} \forall X. \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, W)$

The only rule which allows us to form this conclusion is $MForall$, with premise

$MV(\Gamma, p), X \Vdash^{:=} \lfloor W \rfloor := ? \rightarrow P \Rightarrow (\sigma \circ [R/X], W)$

The derivation of $\Vdash^{\cdot}$ implies (implicitly) that $[S/X]\ W$ is defined, and it is clear that $\sigma \circ [R/X](X) \in \{X, S\}$, so by Lemma 8 (validity of using substitutions on matches) we have

$$MV(\Gamma, p[S]) \Vdash^{:=} \lfloor [S/X]W \rfloor :=? \rightarrow P \Rightarrow (\sigma, [S/X]W)$$

allowing us to complete the proof.

Case $?App$ Our assumed derivation is

$$\frac{\Gamma; ?\rightarrow P \Vdash^{?} t : W \rightsquigarrow (p, \sigma) \quad \Gamma \Vdash^{\cdot} (p\!:\!W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')}{\Gamma; P \Vdash^{?} t\ t' : W' \rightsquigarrow (p', \sigma')}\ ?App$$

By the IH on the first premise, we have

$$MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, W)$$

With this and with the second premise, we appeal to Lemma 9 (algorithmic application preserves matching) to conclude

$$MV(\Gamma, p') \Vdash^{:=} \lfloor W' \rfloor := P \Rightarrow (\sigma, W')$$

□

4.6 Lemma: Matches generate well-formed Decorations

Lemma 7. *If $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ with $WF(\Gamma, P)$ and $WF((\Gamma, \overline{X}), T)$ then for all $X \in \overline{X}$, $\sigma(X) = X$ or $WF(\Gamma, \sigma(X))$*

Proof. By a simple inductive argument on the assumed derivation. First, note that after a base-case is formed using rules $MType$, $M?$, or $MCurr$, the generated solution decreases in its domain with each inductive use of $MForall$, so we need only consider the base cases. Next, base cases $M?$ and $MCurr$ produce σ_{id}, and the property we are trying to prove holds trivially for the empty solution. The only case of interest, then, is $MType$.

$MType$ tells us that our assumed prototype P is some type S, so by assumption $WF(\Gamma, S)$. This means that in the substitution we produce, $\overline{[U/Y]}$, the free type variables in the codomain ($FV(\overline{U})$) do not overlap with any meta-variables. Furthermore, free type variables in $\overline{U}$ cannot be confused with bound type variables in S thanks to the second condition, so the only free type variables in $\overline{U}$ must be those declared in Γ – giving us $WF(\Gamma, \overline{U})$. □

4.7 Lemma: Substitutions on Matches

Lemma 8. *If $\overline{X}, X \Vdash^{:=} T := P \Rightarrow (\sigma, W)$, $[S/X]\ W$ is defined, and $\sigma(X) \in \{X, S\}$, and there is some Γ such that $WF(\Gamma, S)$ and $WF(\Gamma, P)$, then $\overline{X} \Vdash^{:=} [S/X]\ T := P \Rightarrow (\sigma - X, [S/X]\ W)$*

Proof. By a simple inductive arugment on the assumed derivation. The only interesting cases are the two base cases of the match $MCurr$ (which works because by assumption $[S/X]\ W$ is defined) and $MType$. For $MType$ we have

$$\frac{\overline{Y} = FT(T) \cap (\overline{X} \cup \{X\}) \quad FV(\overline{U}) \cap (BTV(T') \cup \overline{X} \cup \{X\}) = \varnothing \quad \overline{[U/Y]}\ T = T'}{\overline{X}, X \Vdash^{:=} T := S \Rightarrow (\overline{[U/Y]}, W)}\ MType$$

We have two subcases to consider, corresponding the assumption that $\overline{[U/Y]}(X) \in \{X, S\}$. If $\overline{[U/Y]}(X) = X$ then clearly $X \notin \overline{Y}$ and $[S/X]\ T = T$, and we can easily modify the above derivation to

$$\frac{\overline{Y} = FT([S/X]\ T) \cap (\overline{X}) \quad FV(\overline{U}) \cap (BTV(T') \cup \overline{X}) = \varnothing \quad \overline{[U/Y]}\ [S/X]\ T = T'}{\overline{X} \Vdash^{:=} [S/X]\ T := T' \Rightarrow (\overline{[U/Y]}, [S/X]\ W)}\ MType$$

If $\overline{[U/Y]}(X) = S$ then we can easily factor out the mapping $[S/X]$ and similarly get the same derivation. □

4.8 Lemma: $\Vdash^{\cdot}$ is sound wrt Matching

Lemma 9. *If $MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, W)$ and $\Gamma \Vdash^{\cdot} (p\!:\!W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')$ then $MV(\Gamma, p') \Vdash^{:=} \lfloor W' \rfloor := P \Rightarrow (\sigma, W')$*

Proof. By induction on the assumed derivation of $\Vdash^{\cdot}$.

Case $?Forall$**:** Our assumed derivation is

$$\frac{\sigma'' = \text{ if } R\!=\!X \text{ then } \sigma \text{ else } [R/X]\circ\sigma \quad \Gamma \Vdash^{\cdot} (p[X]\!:\!W, \sigma'') \cdot t' : W' \rightsquigarrow (p', \sigma')}{\Gamma \Vdash^{\cdot} (p\!:\!\forall X\!=\!R.\, W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')}\ ?Forall$$

And our assumed match is

$MV(\Gamma, p) \Vdash^{:=} \forall X. \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, \forall X = R. W)$

Now, the only rule that could have formed this match (by inversion) is rule $MForall$, whose premise is

$MV(\Gamma, p[X]) \Vdash^{:=} \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma'', W)$

This is the match we need to invoke the IH on the derivation of $\Vdash^{\cdot}$ in the premise of our assumption – the IH gives us

$MV(\Gamma, p) \Vdash^{:=} \lfloor W' \rfloor := P \Rightarrow (\sigma', W')$

which is what we need to conclude.

Case $?Chk$**:** Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \Vdash_{\Downarrow} t' : S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p{:}S \rightarrow W, \sigma) \cdot t' : W \rightsquigarrow (p\ e', \sigma)}\ ?Chk$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} S \rightarrow \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, S \rightarrow W)$

By inversion, the only rule introducing this match is $MArr$ whose premise is

$MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor := P \Rightarrow (\sigma, W)$

which is what we need to conclude!

Case $?Syn$**:** Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \Vdash_{\Uparrow} t : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p{:}S \rightarrow W, \sigma) \cdot t' : \overline{[U/Y]}\ W \rightsquigarrow ((\overline{[U/Y]}\ p)\ e', \sigma)}\ ?Syn$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} S \rightarrow \lfloor W \rfloor :=? \rightarrow P \Rightarrow (\sigma, S \rightarrow W)$

By inversion, the only rule introducing this match is $MArr$ whose premise is

$MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor := P \Rightarrow (\sigma, W)$

By Lemma 8 (substitutions on matches) we can rewrite this match to

$MV(\Gamma, \overline{[U/Y]}\ p) \Vdash^{:=} \lfloor \overline{[U/Y]}\ W \rfloor :=\Rightarrow (\sigma, \overline{[U/Y]}\ W)$ since

- $\overline{[U/Y]}\ W$ is defined
- For all $Y \in \overline{Y}$, $\sigma(Y) = Y$
- $WF(\Gamma, S)$ and $WF(\Gamma, P)$ by (implicit) assumption
- $\sigma - \overline{Y} = \sigma$, by the definition of $\overline{Y}$

This is precisely the match we need to conclude. □

5 Completeness of $\Vdash_\delta$ *wrt* $\vdash_\delta$

1. If $\Gamma \vdash_\delta t : T \rightsquigarrow e$ then $\Gamma \Vdash_\delta t : T \rightsquigarrow e$
2. If $\Gamma \vdash^{P} t : T^+ \rightsquigarrow (p^+, \sigma)$
   and $MV(\Gamma, p^+) \Vdash^{:=} T^+ := P \Rightarrow (\sigma^A, W^+)$, where[1]

[1]The superscript $^+$ denotes only that the terms and types of the declarative system have some additional substitutions σ^+ in them that the algorithmic rules would not have made.

- $\sigma \subseteq \sigma^A$
- $arr_P(P) \geq 1$ when $\neg App(t)$

then there exists (p, T, W, σ^+) where

- $MV(\Gamma, p) \Vdash^{:=} T := P \Rightarrow (\sigma^A \circ \sigma^+, W)$
- $\sigma^+(p, T, W) = (p^+, T^+, W^+)$ and $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p^+)$
- $\Gamma; P \Vdash^? t : W \rightsquigarrow (p, \sigma^A \circ \sigma^+)$

3. If $\Gamma \vdash^{\cdot} (p^+ \!:\! T^+, \sigma) \cdot t' : T'^+ \rightsquigarrow (p'^+, \sigma')$ where

- $MV(\Gamma, p'^+) \Vdash^{:=} T'^+ := P \Rightarrow (\sigma'^A, W'^+)$ with $\sigma' \subseteq \sigma'^A$
- and $MV(\Gamma, p) \Vdash^{:=} T := ? \to P \Rightarrow (\sigma^A \circ \sigma^+, W)$ with $\sigma \subseteq \sigma^A$
- and $\sigma^+(p, T) = (p^+, T^+)$

then exists (p', T', W', σ'^+) where

- $MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma'^A \circ \sigma'^+, W')$ with $\sigma' \subseteq \sigma'^A$
- $\sigma'^+(p', T', W') = (p'^+, T'^+, W'^+)$, $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'^+)$
- and $\Gamma \Vdash^{\cdot} (p \!:\! W, \sigma^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

5.1 Complete $\Vdash_\delta$ wrt $\vdash_\delta$

Theorem 8. *If $\Gamma \vdash_\delta t : T \rightsquigarrow e$ then $\Gamma \Vdash_\delta t : T \rightsquigarrow e$.*

Proof. By induction on the assumed derivation. The rules for the two systems are identical except for *AppSyn* and *AppChk*, so only these are shown.

Case *AppSyn* Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, T) = MV(\Gamma, e) = \varnothing}{\Gamma \vdash_{\Uparrow} t\ t' : T \rightsquigarrow e}\ AppSyn$$

To invoke mutual induction on the completeness of $\Vdash^?$ we must provide a match. This would be

$$\frac{}{\varnothing \Vdash^{:=} T := ? \Rightarrow (\sigma_{id}, T)}\ M?$$

It is immediate that the two precondition hold for the substitution – $\sigma_{id} \subseteq \sigma_{id}$. We now invoke complete $\Vdash^?$ (instantiating σ_{id} for σ^A) to get (p, T, W, σ^+) where

- $MV(\Gamma, p) \Vdash^{:=} T := ? \Rightarrow (\sigma^+, W)$

  The only rule which forms a match like this (by inversion) is $M?$. From this we know that $\sigma^+ = \sigma_{id}$.

- $\sigma^+(p, T, W) = (e, T', T')$

  This gives us $(p, T, W) = (e, T', T')$

- $\Gamma; ? \Vdash^? t : W \rightsquigarrow (p, \sigma^A \circ \sigma^+)$

  which we rewrite to $\Gamma; ? \Vdash^? t : T' \rightsquigarrow (e, \sigma_{id})$

We conclude

$$\frac{\Gamma \Vdash^? t\ t' : T \rightsquigarrow (p, \sigma_{id}) \quad MV(\Gamma, p) = \varnothing}{\Gamma \Vdash_{\Uparrow} t\ t' : T \rightsquigarrow e}\ AppSyn$$

Case *AppChk***:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T^+ \rightsquigarrow (p^+, \sigma) \quad MV(\Gamma, p^+) = MV(\Gamma, T^+) = dom(\sigma)}{\Gamma \vdash_{\Downarrow} t\ t' : \sigma\ T^+ \rightsquigarrow \sigma\ p^+}\ AppChk$$

To invoke mutual induction on the completeness of $\Vdash^?$ we must provide a match. This would be

$$\frac{dom(\sigma) = MV(\Gamma, T^+) = FV(T) \cap MV(\Gamma, p^+) \quad FV(cod(\sigma)) \cap (BTV(\sigma\ T^+) \cup MV(\Gamma, p^+)) = \varnothing}{MV(\Gamma, p^+) \Vdash^{:=} T^+ := \sigma\ T^+ \Rightarrow (\sigma, T^+)}\ MType$$

(The second condition we get from the stronger result of Lemma 1, that solutions in σ are well-formed wrt Γ). It is clear that this match satisfies the precondition for Theorem 9 (the completeness of $\Vdash^?$), so we invoke it (mutually-inductively) to get (p, T, W, σ^+) where

- $MV(\Gamma, p) \Vdash^{:=} T := \sigma\ \sigma^+\ T \Rightarrow (\sigma \circ \sigma^+, W)$
  The only matching rule that could give us this conclusion is $MType$, which tells us $W = T$
- $\sigma^+(p, T, W) = (p^+, T^+, T^+)$ and $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p^+)$
  This allows us to derive $MV(\Gamma, p) = MV(\Gamma, T) = dom(\sigma \circ \sigma^+)$
- $\Gamma; T \Vdash^? t : T \rightsquigarrow (p, \sigma \circ \sigma^+)$

We can therefore conclude

$$\frac{\Gamma; T \Vdash^? t\ t' : T' \rightsquigarrow (p, \sigma \circ \sigma^+) \quad MV(\Gamma, p) = dom(\sigma \circ \sigma^+)}{\Gamma \Vdash_{\Downarrow} t\ t' : T \rightsquigarrow \sigma\ \sigma^+\ p}\ AppChk$$

□

5.2 Complete $\Vdash^?$ *wrt* $\vdash^{\mathbf{P}}$

Theorem 9.
If $\Gamma \vdash^P t : T^+ \rightsquigarrow (p^+, \sigma)$ *and* $MV(\Gamma, p^+) \Vdash^{:=} T^+ := P \Rightarrow (\sigma^A, W^+)$*, where*

- $\sigma \subseteq \sigma^A$
- $arr_P(P) \geq 1$ *when* $\neg App(t)$

then there exists (p, T, W, σ^+) *where*

- $MV(\Gamma, p) \Vdash^{:=} T := P \Rightarrow (\sigma^A \circ \sigma^+, W)$
- $\sigma^+(p, T, W) = (p^+, T^+, W^+)$*,* $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p^+)$
- $\Gamma; P \Vdash^? t : W \rightsquigarrow (p, \sigma^A \circ \sigma^+)$

Proof. By induction on the assumed derivation.

Case $?Head$**:** Our assumed derivation is

$$\frac{\neg App(t) \quad \Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e}{\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (e, \sigma_{id})}\ ?Head$$ and our assumed match is $MV(\Gamma, e) \Vdash^{:=} T :=?\!\rightarrow P \Rightarrow (\sigma^A, W)$

Since we know e is well-typed under Γ, $MV(\Gamma, e) = \varnothing$. Appealing to Lemma 10 we get $dom(\sigma^A) \subseteq \varnothing$, so $\sigma^A = \sigma_{id}$. We rewrite our match to

$\varnothing \Vdash^{:=} T :=?\!\rightarrow P \Rightarrow (\sigma_{id}, W)$

Now, invoke mutual induction on the completeness of $\Vdash_{\Uparrow}$ (Theorem 8) to get

$\Gamma \Vdash_{\Uparrow} t : T \rightsquigarrow e$

and choose (e, T, W, σ_{id}) to meet the desired derivation and conditions

- $MV(\Gamma, e) \Vdash^{:=} T :=?\!\rightarrow P \Rightarrow (\sigma_{id}, W)$
- $\sigma_{id}(e, T, W) = (e, T, W)$,
- $\Gamma; ?\!\rightarrow P \Vdash^? t : W \rightsquigarrow (e, \sigma_{id})$

Case $?TApp$**:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t : \forall X.\, T^{+} \rightsquigarrow (p^{+}, \sigma)}{\Gamma \vdash^{\mathrm{P}} t[S] : [S/X]T^{+} \rightsquigarrow (p^{+}[S], \sigma)}\ ITApp$$

and our assumed match is

$$MV(\Gamma, p^{+}[S]) \Vdash^{:=} [S/X]T^{+} := ? \to P \Rightarrow (\sigma^{A}, [S/X]W^{+})$$

We appeal to Lemma 11 (invertible substitutions in matching) to get

$MV(\Gamma, p^{+}[S]), X \Vdash^{:=} T^{+} := ? \to P \Rightarrow (\sigma^{A} \circ \sigma_{X}, W^{+}_{X})$ where

- $\sigma_X \subseteq [S/X]$
  That is, $\sigma_X(X) \in \{X, S\}$
- $[S/X]W^{+}_{X} = W^{+}$

With that match, we now can apply matching rule $MForall$ to get

$$MV(\Gamma, p[S]) \Vdash^{:=} \forall X.\, T^{+} := ? \to P \Rightarrow (\sigma^{A}, \forall X = \sigma_X(X).\, W^{+})$$

Lastly, we note that $MV(\Gamma, p[S]) = MV(\Gamma, p)$, so we are able to invoke the IH to get (p, T, W, σ^{+}) where

- $MV(\Gamma, p) \Vdash^{:=} T := ? \to P \Rightarrow (\sigma^{A} \circ \sigma^{+}, W)$
- $\sigma^{+}(p, T, W) = (p^{+}, \forall X.\, T^{+}, \forall X = \sigma_X(X).\, W^{+})$
  and $dom(\sigma^{+}) = MV(\Gamma, p) - MV(\Gamma, p^{+})$
- $\Gamma; ? \to P \Vdash^{?} t : W \rightsquigarrow (p, \sigma^{A} \circ \sigma^{+})$

Before we can finish the derivation of $?TApp$ we must deal with a subtle issue – what if $T = Y$ and $W = (Y, ? \to P)$, with $\sigma^{+}(Y) = \forall X = \sigma_X(X).\, W^{+}$? This would prevent the algorithmic rules from inferring a type application, and we'd be stuck!
Fortunately, we need only look at the match and equality produced by the the result of calling the IH to sort this out. If $T = Y$ then it could only be formed by $MCurr$, yielding

$$MV(\Gamma, p) \Vdash^{:=} Y := ? \to P \Rightarrow (\sigma^{A} \circ \sigma^{+} = \sigma_{id}, (Y, ? \to P))$$

But now it's impossible that $\sigma^{+}(Y) = Y = \forall X = \sigma_X(X).\, W^{+}$. Therefore, we know that T has the form $\forall X.\, T$ (we shadow the original T from here on out) and revisit our conclusions

- $MV(\Gamma, p) \Vdash^{:=} \forall X.\, T := ? \to P \Rightarrow (\sigma^{A} \circ \sigma^{+}, \forall X = R.\, W)$
- $\sigma^{+}(p, \forall X.\, T, \forall X = R.\, W) = (p^{+}, \forall X.\, T^{+}, \forall X = \sigma_X(X).\, W^{+})$
  and $dom(\sigma^{+}) = MV(\Gamma, p) - MV(\Gamma, p^{+})$
  We therefore know that $R = \sigma_X(X)$
- $\Gamma; ? \to P \Vdash^{?} t : \forall X = R.\, W \rightsquigarrow (p, \sigma^{A} \circ \sigma^{+})$

This allows us to conclude

$$\Gamma; ? \to P \Vdash^{?} t[S] : [S/X]W \rightsquigarrow (p[S], \sigma^{A} \circ \sigma^{+})$$

Case $?App$**:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t : T^{+} \rightsquigarrow (p^{+}, \sigma) \quad \Gamma \vdash^{\cdot} (p^{+} : T^{+}, \sigma) \cdot t' : T'^{+} \rightsquigarrow (p'^{+}, \sigma')}{\Gamma \vdash^{\mathrm{P}} t\ t' : T'^{+} \rightsquigarrow (p'^{+}, \sigma')}\ ?App$$

Our assumed match is

$$MV(\Gamma, p'^{+}) \Vdash^{:=} T'^{+} := P \Rightarrow (\sigma'^{A}, W'^{+})$$

By Lemma 13 (application of partially synthesized applicands preserves matching backwards) we get from this and the second premise of the derivation

$MV(\Gamma, p^+) \Vdash^{:=} T^+ :=? \rightarrow P \Rightarrow (\sigma^A, W^+)$, with $\sigma \subseteq \sigma^A$.

This allows us to invoke the IH on the first premise of the derivation to get (p, T, σ^+) where

- $MV(\Gamma, p) \Vdash^{:=} T :=? \rightarrow P \Rightarrow (\sigma^A \circ \sigma^+, W)$
- $\sigma^+(p, T, W) = (p^+, T^+, W^+)$ and $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p^+)$
- $\Gamma; ? \rightarrow P \Vdash^? t : W \rightsquigarrow (p, \sigma^A \circ \sigma^+)$

The first two of these conditions, and the match we assumed, satisfy the preconditions Theorem 10, allowing us to use mutual induction to get (p', T', W', σ'^+) where

- $MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma'^A \circ \sigma'^+, W')$ with $\sigma' \subseteq \sigma'^A$
- $\sigma'^+(p', T', W') = (p'^+, T'^+, W'^+)$, $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'+)$
- and $\Gamma \Vdash^{\cdot} (p{:}W, \sigma^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

This allows us to conclude $\Gamma; P \Vdash^? t\ t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

□

5.3 Complete $\Vdash^{\cdot}$ *wrt* $\vdash^{\cdot}$

Theorem 10. *Completeness of the algorithm wrt the specification (applications):*
If $\Gamma \vdash^{\cdot} (p^+{:}T^+, \sigma) \cdot t' : T'^+ \rightsquigarrow (p'^+, \sigma')$ *where*

- $MV(\Gamma, p'^+) \Vdash^{:=} T'^+ := P \Rightarrow (\sigma'^A, W'^+)$ *with* $\sigma' \subseteq \sigma'^A$
- *and* $MV(\Gamma, p) \Vdash^{:=} T :=? \rightarrow P \Rightarrow (\sigma^A \circ \sigma^+, W)$ *with* $\sigma \subseteq \sigma^A$
- *and* $\sigma^+(p, T) = (p^+, T^+)$, $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p^+)$

then exists (p', T', W', σ'^+) *where*

- $MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma^A \circ \sigma^+, W')$ *with* $\sigma' \subseteq \sigma'^A$
- $\sigma'^+(p', T', W') = (p'^+, T'^+, W'^+)$, $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'^+)$
- *and* $\Gamma \Vdash^{\cdot} (p{:}W, \sigma^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

Proof. By a (not-so-easy) induction on the assumed derivation.

Case $PForall$ Our assumed derivation is

$$\frac{\sigma'' \in \{\sigma, [S/X] \circ \sigma\}, WF(\Gamma, S) \quad \Gamma \vdash^{\cdot} (p^+[X]{:}T^+, \sigma'') \cdot t' : T'^+ \rightsquigarrow (p', \sigma')}{\Gamma \vdash^{\cdot} (p{:}\forall X. T^+, \sigma) \cdot t' : T'^+ \rightsquigarrow (p'^+, \sigma')} \; PForall$$

Our assumed conditions are

- $MV(\Gamma, p'^+) \Vdash^{:=} T'^+ := P \Rightarrow (\sigma'^A, W'^+)$ with $\sigma' \subseteq \sigma'^A$
- and $MV(\Gamma, p) \Vdash^{:=} T :=? \rightarrow P \Rightarrow (\sigma^A \circ \sigma^+, W)$ with $\sigma \subseteq \sigma^A$
- and $\sigma^+(p, T) = (p^+, \forall X. T^+)$, $dom(\sigma^+) = MV(\Gamma, p) - MV(\Gamma, p+)$

To make progress we need some way to reveal that $T \neq Y$ for some $Y \in MV(\Gamma, p)$ – because if it were, then we would not be able to apply the algorithmic rule $?Forall$. First, we note that it is easy to show (Lemma 18) that $\forall X. T^+$ being in the application position of a judgment of $\vdash^{\cdot}$, it must really have the following form

$\forall X, \overline{X}. S^+ \rightarrow T'^+_{\overline{Y}}$.

(The reason for subscript $\overline{Y}$ will become apparent a little later). By a similar observation (a kind of "peek-ahead" assumed derivation of $\vdash^{\cdot}$, Lemma 19) tells us that the base-case for our assumed derivation of $\vdash^{\cdot}$ generates some substitution $\sigma_{\overline{Y}}$, where $dom(\sigma_{\overline{Y}}) = \overline{Y} \subseteq MV(\Gamma, \sigma\ p[X][\overline{X}])$, such that $\sigma_{\overline{Y}}\ T'^+_{\overline{Y}} = T'^+$

Returning to our troubles, if $T = Y$ then the second of our assumed matches must have been formed by rule $MCurr$, which tells us

$MV(\Gamma, p) \Vdash^{:=} Y := ? \rightarrow P \Rightarrow (\sigma^A \circ \sigma^+ = \sigma_{id}, (Y, ? \rightarrow P) = W)$

However, our *third* condition tells us that $\sigma^+(Y) = \sigma_{id}(Y) = \forall X, \overline{X}. S^+ \rightarrow T'^+_{\overline{Y}}$ – which is impossible! We can iterate this argument over each bound variable in $\overline{X}$ to get, finally, that T looks like $\forall X, \overline{X}. S \rightarrow T'_{\overline{Y}}$ for some S and $T'_{\overline{Y}}$. Knowing this, we revisit the second and third assumed conditions on our derivation:

- $MV(\Gamma, p) \Vdash^{:=} \forall X, \overline{X}. S \to T'_{\overline{Y}} :=? \to P \Rightarrow (\sigma^A \circ \sigma^+, \forall X, \overline{X} = R, \overline{R}. W'_{\overline{Y}})$
  with $\sigma \subseteq \sigma^A$, and for some $W'_{\overline{Y}}$, R, and $\overline{R}$
- and $\sigma^+(p, \forall X, \overline{X}. S \to T'_{\overline{Y}}) = (p^+, \forall X, \overline{X}. S^+ \to T'^+_{\overline{Y}})$,
  $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'^+)$

We next need some way to relate the specificational system's "guess" $\sigma''(X)$ with the match-generated decoration R. Our algorithmic rules will first want to define $\sigma''^A \ = \ \text{if } R{=}X \text{ then } \sigma^A \text{ else } [R/X] \circ \sigma^A$. To satisfy the precondition on the IH, we need to show that $\sigma'' \subseteq \sigma''^A$. As $\sigma \subseteq \sigma^A$, this reduces to showing that if $\sigma''(X) = S$ then $\sigma''^A(X) = S$.

σ'' and σ''^A: By an easy inductive argument we know that $\vdash^{\cdot}$ grows its generated solutions monotonically, so the derivation in the premise of our assumption,

$$\Gamma \vdash^{\cdot} (p^+[X]{:}T^+, \sigma'') \cdot t' : T'^+ \rightsquigarrow (p', \sigma')$$

tells us that $\sigma'' \subseteq \sigma'$, and furthermore by assumption $\sigma' \subseteq \sigma'^A$ If the specificational rules guessed S, then it is clear that $\sigma'^A(X) = S$. Next, since we have that $T'^+ \ = \ \sigma^+ \ \sigma_{\overline{Y}} \ T'_{\overline{Y}}$, the match in our first condition is

$$MV(\Gamma, p'^+) \Vdash^{:=} \sigma^+ \ \sigma_{\overline{Y}} \ T'_{\overline{Y}} := P \Rightarrow (\sigma'^A, W'^+)$$

We invoke Lemma 11 (invertible substitutions in matches) to get

$$MV(\Gamma, p'), \overline{Y} \Vdash^{:=} T'_{\overline{Y}} := P \Rightarrow (\sigma'^A \circ \sigma''', W'_{\overline{Y}}) \text{ (for some } \sigma''' \subseteq \sigma^+ \circ \sigma_{\overline{Y}}).$$

which we can repack into (successively using rule $MForall$)

$$MV(\Gamma, p) \Vdash^{:=} \forall X, \overline{X}. S \to T'_{\overline{Y}} :=? \to P \Rightarrow (\sigma^A \circ \sigma^+, \forall X, \overline{X} = R, \overline{R}. S \to W'_{\overline{Y}})$$

(re-use of meta-variables σ^A, σ^+ is justified by Lemma 15 and the match from our second assumed condition). Well, if $\sigma''(X) = S$, then $\sigma'^A(X) = R = S$. And since $R = S$, the algorithmic rules must choose $\sigma''^A = [S/X] \circ \sigma^A$, meaning that $\sigma''^A(X) = S$ as well, giving us that $\sigma'' \subseteq \sigma''^A$.

IH: To recap, we now meet the desired preconditions to invoke the IH

- $MV(\Gamma, p'^+) \Vdash^{:=} T'^+ := P \Rightarrow (\sigma'^A, W'^+)$ with $\sigma' \subseteq \sigma'^A$
  This remains unmodified from our assumption
- and $MV(\Gamma, p[X]) \Vdash^{:=} \forall X, \overline{X}. S \to T'_{\overline{Y}} :=? \to P \Rightarrow (\sigma''^A \circ \sigma^+, \forall \overline{X} = \overline{R}. S \to W'_{\overline{Y}})$ with $\sigma'' \subseteq \sigma''^A$
- and $\sigma^+(p[R], T) = (p^+[R], T^+)$

We invoke the IH to get (p', T', W', σ'^+) where

- $MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma'^A \circ \sigma'^+, W')$ with $\sigma' \subseteq \sigma'^A$
- $\sigma'^+(p', T', W') = (p'^+, T'^+, W'^+)$, $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'^+)$
- and $\Gamma \Vdash^{\cdot} (p[X]{:}W, \sigma''^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

which is what we need to derive

$$\Gamma \Vdash^{\cdot} (p{:}\forall X, \overline{X} = \sigma''^A(X), \overline{R}. S \to W'_{\overline{Y}}) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$$

Case $PChk$ Our assumed derivation is

$$\frac{MV(\Gamma, \sigma \ S^+) = \varnothing \quad \Gamma \vdash_{\Downarrow} t' : \sigma \ S^+ \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p^+{:}S^+ \to T^+, \sigma) \cdot t' : T^+ \rightsquigarrow (p^+ \ e', \sigma)} \ PChk$$

Our assumed conditions are

- $MV(\Gamma, p^+ e) \Vdash^{:=} T^+ := P \Rightarrow (\sigma'^A, W'^+)$ with $\sigma \subseteq \sigma'^A$
- and $MV(\Gamma, p) \Vdash^{:=} S \to T :=? \to P \Rightarrow (\sigma^A \circ \sigma^+, S \to W)$ with $\sigma \subseteq \sigma^A$
  Again, reasoning (using the equality directly below) that the subject of this match could not be some $Y \in MV(\Gamma, p)$, it must be of the form $S \to T$

- and $\sigma^+(p, S \to T) = (p^+, S^+ \to T^+)$

We must now pick out a suitable (σ'^+, p', T', W') to provide as the result of this case. Pick $(\sigma^+, p\ e, T, W)$. Now we show the post-conditions of the theorem hold.

- $MV(\Gamma, p\ e') \Vdash^{:=} T := P \Rightarrow (\sigma'^A \circ \sigma^+, W)$ with $\sigma \subseteq \sigma'^A$
  From the match given in our second assumed condition it is clear that

  $$MV(\Gamma, p\ e') \Vdash^{:=} T := P \Rightarrow (\sigma^A \circ \sigma^+, W)$$

  (because the only rule that could form it, $MArr$, would have this as its premise). Aligning this with our requirement reduces to showing that $\sigma'^A = \sigma^A$. By Lemma 14 on the soundness of solutions *by* the matching, *for* the matching, we have

  $$MV(\Gamma, p^+\ e') \Vdash^{:=} T^+ := P \Rightarrow (\sigma^A, \sigma^+\ W),$$

  and comparing to our first condition, uniquess of matching solutions (Lemma 15) gives us

  $$\sigma^A = \sigma'^A \text{ and } \sigma^+\ W = W'^+$$

- $\sigma^+(p\ e', T, W) = (p^+\ e', T^+, W'^+)$, $dom(\sigma^+) = MV(\Gamma, p\ \ e') - MV(\Gamma, p^+\ e')$,
  Directly from assumptions and the equation in the point above, and from the fact that $MV(\Gamma, e') = \varnothing$
- and $\Gamma \Vdash^{\cdot} (p{:}S \to W, \sigma^A \circ \sigma^+) \cdot t' : W \rightsquigarrow (p\ e', \sigma'^A \circ \sigma^+)$
  For this we invoke mutual induction on the completeness of $\Vdash_{\Downarrow}$ (Theorem 8) to get
  $\Gamma \Vdash_{\Downarrow} t' : \sigma^A\ \sigma^+\ S \rightsquigarrow e'$, noting that $\sigma^+\ S = S^+$ and $MV(\Gamma, \sigma^A\ S^+) = \varnothing$

which is what we need to conclude

Case $PSyn$ Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S^+) = \overline{Y^+} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \overline{[U/Y^+]}\ \sigma\ S^+ \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p^+{:}S^+ \to T^+, \sigma) \cdot t' : \overline{[U/Y^+]}\ T^+ \rightsquigarrow ((\overline{[U/Y^+]}\ p^+)\ e', \sigma)}\ PSyn$$

Our assumed conditions are

- $MV(\Gamma, \overline{[U/Y^+]}p^+\ e) \Vdash^{:=} \overline{[U/Y^+]}T^+ := P \Rightarrow (\sigma'^A, W'^+)$ with $\sigma \subseteq \sigma'^A$
- and $MV(\Gamma, p) \Vdash^{:=} S \to T :=?\!\to P \Rightarrow (\sigma^A \circ \sigma^+, S \to W)$ with $\sigma \subseteq \sigma^A$
  Again, reasoning that the subject of this match must be $S \to T$ and not some $Y \in MV(\Gamma, p)$
- and $\sigma^+(p, S \to T) = (p^+, S^+ \to T^+)$

We must pick a suitable (p', T', W', σ'^+) for which we can derive a judgment formed by $\Vdash^{\cdot}$ with the needed properties. To do this, we must first ask what we know about any unsolved meta-variables $\overline{Y}$ that the *algorithm* will encounter – $MV(\Gamma, \sigma^A\ \sigma^+\ S) = \overline{Y}$ – given that $MV(\Gamma, \sigma\ S^+) = MV(\Gamma, \sigma\ \sigma^+ S) = \overline{Y^+}$.

First, it is clear that $\overline{Y} \subseteq \overline{Y^+}$ because $\sigma \subseteq \sigma^A$. So, consider the match from our first assumed condition. By Lemma 11 on inverting substitutions in the subject of a match, we get

$$MV(\Gamma, p^+\ e) \Vdash^{:=} T^+ := P \Rightarrow (\sigma'^A \circ \sigma_{\overline{Y}}, W'^+_{\overline{Y}}), \text{ where } \sigma_{\overline{Y}}(W'^+_{\overline{Y}}) = W'^+$$

Now, consider the match from our second assumed condition. By inversion we know it can only have been formed by $MArr$, whose premise we further transform by Lemma 14 (re-substituting solutions in matches – in this case σ^+) to get

$$MV(\Gamma, p^+\ e) \Vdash^{:=} T^+ := P \Rightarrow (\sigma^A, \sigma^+\ W)$$

And now, by uniqueness of solutions of matching (Lemma 15) we get $(\sigma^A, \sigma^+\ W) = (\sigma'^A \circ \sigma_{\overline{Y}}, W'^+_{\overline{Y}})$. Let us call, for the sake of simplicity, the second component of both pairs W^+, and let $\sigma_{\overline{Y'}} = \overline{[U/Y]} - \sigma_{\overline{Y}}$

We return to the task of selecting (p', T', W', σ'^+). We pick

$$((\sigma_{\overline{Y'}}\ p)\ e', \sigma_{\overline{Y'}}\ T, \sigma_{\overline{Y'}}\ W, \sigma^+ \circ \sigma_{\overline{Y}})^2,$$

and now witness the following post-conditions:

[2] If you were wondering what the purpose was of σ^+ in these proofs, now you know – the specificational rules may opt to discover from synthetic type-argument inference what the algorithm would know from contextual type-argument inference

- $MV(\Gamma, p') \Vdash^{:=} \sigma_{\overline{Y'}}\ T := P \Rightarrow (\sigma'^A \circ \sigma^+ \circ \sigma_{\overline{Y}}, \sigma_{\overline{Y'}}\ W)$ with $\sigma \subseteq \sigma'^A$
  This comes from the equational reasoning above, taking the match

  $MV(\Gamma, p') \Vdash^{:=} T := P \Rightarrow (\sigma^A \circ \sigma^+, W)$

  noting that $\sigma^A = \sigma'^A \circ \sigma_{\overline{Y}}$ and deploying solution $\sigma_{\overline{Y'}}$ to T (Lemma 8), while then reasoning that $dom(\sigma_{\overline{Y'}}) \cap dom(\sigma^A \circ \sigma_{\overline{Y}}) = \varnothing$ (solution $\sigma_{\overline{Y'}}$ doesn't interfere with the solutions the match generates) from the definition of $\sigma_{\overline{Y'}}$.
- $\sigma'^+(p', T', W') = (p'^+, T'^+, W'^+)$, $dom(\sigma'^+) = MV(\Gamma, p') - MV(\Gamma, p'^+)$
  Again with some equational reasoning. For example, $\sigma'^+\ T' = \sigma^+\ \sigma_{\overline{Y}}\ \sigma_{\overline{Y'}}\ T = \overline{[U/Y]}\ T^+ = T'^+$
- and $\Gamma \Vdash^{\cdot} (p{:}W, \sigma^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$

This last piece requires some care – the algorithm might use *more* contextual information than the specificational derivation, meaning that we might need to derive $?Chk$ even though our assumed case is $PSyn$. If $\overline{Y} \neq \varnothing$, we know that the algorithm will try to derive

$\Gamma \Vdash_{\Uparrow} t : \sigma_{\overline{Y'}}\ \sigma^A\ \sigma^+\ S \rightsquigarrow e'$.

By an invocation of mutual induction on the completeness of $\Vdash_{\Uparrow}$ (Theorem 8) on the second premise of our assumed derivation of $\vdash^{\cdot}$, we know that the algorithm can derive

$\Gamma \Vdash_{\Uparrow} t' : \overline{[U/Y^+]}\ \sigma\ S^+ \rightsquigarrow e'$

which (by some equational reasoning) is what we need.

However, if $\overline{Y} = \varnothing$, the algorithm will actually try to *check* the term t' against a fully known type. We need

$\Gamma \Vdash_{\Downarrow} t' : \sigma^A\ \sigma^+\ S \rightsquigarrow e'$

By Lemma 16 (checking mode extends synthesizing mode) on the second premise of our assumed derivation we have

$\Gamma \vdash_{\Downarrow} t' : \overline{[U/Y^+]}\ \sigma\ S^+ \rightsquigarrow e'$

By mutual induction on the completeness of $\Vdash_{\Downarrow}$ (Theorem 8) we get

$\Gamma \Vdash_{\Downarrow} t' : \overline{[U/Y^+]}\ \sigma\ S^+ \rightsquigarrow e'$

which, after a bit of equational reasoning on the substitutions, is what we need. So in either case, we are able to conclude

$\Gamma \Vdash^{\cdot} (p{:}S \to W, \sigma^A \circ \sigma^+) \cdot t' : W' \rightsquigarrow (p', \sigma'^A \circ \sigma'^+)$ ☐

5.4 Lemma: Match solutions solve match meta-variables

Lemma 10.
If $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ then $dom(\sigma) \subseteq \overline{X}$

Proof. Straightforward induction on the assumed derivation. ☐

5.5 Lemma: Invertible substitutions in matches

Lemma 11.
If $\overline{X} \Vdash^{:=} \overline{[U/Y]}T := P \Rightarrow (\sigma, W)$ and $\overline{X} \cap FV(\overline{U}) = \varnothing$
then $\overline{X}, \overline{Y} \Vdash^{:=} T := P \Rightarrow (\sigma \circ \sigma_{\overline{Y}}, W_{\overline{Y}})$ where

- $\sigma_{\overline{Y}} \subseteq \overline{[U/Y]}$
- $\overline{[U/Y]} W_{\overline{Y}} = W$

Proof. By a straightforward inductive argument on the assumed derivation. ☐

5.6 Lemma: $\Vdash^{\cdot}$ preserves $\Vdash^{:=}$ (forwards)

Lemma 12.
If $\Gamma \Vdash^{\cdot} (p{:}W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')$ and $MV(\Gamma, p) \Vdash^{:=} \lfloor W \rfloor :=? \to P \Rightarrow (\sigma, W)$
then $MV(\Gamma, p') \Vdash^{:=} \lfloor W' \rfloor :=? \to P \Rightarrow (\sigma', W')$

Proof. By induction on the assumed derivation.

Case $PForall$ Our assumed derivation is

$$\frac{\sigma'' = \text{if } R\!=\!X \text{ then } \sigma \text{ else } [R/X] \circ \sigma \quad \Gamma \Vdash^{\cdot} (p[X]\!:\!W, \sigma'') \cdot t' : W' \rightsquigarrow (p', \sigma')}{\Gamma \Vdash^{\cdot} (p\!:\!\forall X\!=\!R. W, \sigma) \cdot t' : W' \rightsquigarrow (p', \sigma')} \; PForall$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} \forall X. \lfloor W \rfloor :=?\!\rightarrow P \Rightarrow (\sigma, \forall X\!=\!R. W)$

The only rule giving us this match (by inversion) is $MForall$, with premise

$MV(\Gamma, p[X]) \Vdash^{:=} \lfloor W \rfloor :=?\!\rightarrow P \Rightarrow (\sigma'', W)$

We can now invoke the IH on the second premise of our assumed derivation to get

$MV(\Gamma, p') \Vdash^{:=} \lfloor W' \rfloor := P \Rightarrow (\sigma', W')$

which is what we need to conclude.

Case $PChk$**:** Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \Vdash_{\Downarrow} t' : S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p\!:\!S \rightarrow W, \sigma) \cdot t' : W \rightsquigarrow (p\ e', \sigma)} \; PChk$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} S \rightarrow \lfloor W \rfloor :=?\!\rightarrow P \Rightarrow (\sigma, W)$

By inversion the only rule we can use to form this match is $MArr$ with premise

$MV(\Gamma, p\ e') \Vdash^{:=} \lfloor W \rfloor := P \Rightarrow (\sigma, W)$.
Since $MV(\Gamma, e') = \varnothing$ this is what we need to conclude.

Case $PSyn$**:** Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \Vdash_{\Uparrow} t : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e}{\Gamma \Vdash^{\cdot} (p\!:\!S \rightarrow W) \cdot t' : \overline{[U/Y]}\ W \rightsquigarrow ((\overline{[U/Y]}\ p)\ e', \sigma)} \; PSyn$$

and our assumed match is

$MV(\Gamma, p) \Vdash^{:=} S \rightarrow \lfloor W \rfloor :=?\!\rightarrow P \Rightarrow (\sigma, W)$

The only rule allowing us to form this match is $MArr$, with premise

$MV(\Gamma, p\ e') \Vdash^{:=} \lfloor W \rfloor := P \Rightarrow (\sigma, W)$

By Lemma 8, and by noting that $\overline{Y} \cap dom(\sigma) = \varnothing$ from our first premise, we have

$MV(\Gamma, \overline{[U/Y]}p\ e') \Vdash^{:=} \lfloor \overline{[U/Y]}W \rfloor := P \Rightarrow (\sigma, \overline{[U/Y]}W)$ which allows us to conclude. □

5.7 Lemma: $\vdash^{\cdot}$ preserves $\Vdash^{:=}$ (backwards)

Lemma 13.
If $\Gamma \vdash^{\cdot} (p\!:\!T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma)$ *and* $MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma', W')$
then there exists (σ^A, W) *where* $MV(\Gamma, p) \Vdash^{:=} T :=?\!\rightarrow P \Rightarrow (\sigma^A, W)$ *and* $\sigma \subseteq \sigma^A$

Proof. By induction on the assumed derivation of $\vdash^{\cdot}$.

Case $PForall$ Our assumed derivation is

$$\frac{\sigma'' \in \{\sigma, \sigma \circ [S/X]\}, WF(\Gamma, S) \quad \Gamma \vdash^{\cdot} (p[X]\!:\!T, \sigma'') \cdot t' : T' \rightsquigarrow (p', \sigma')}{\Gamma \vdash^{\cdot} (p\!:\!\forall X. T, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')} \; PForall$$

Our assumed match is

$MV(\Gamma, p') \Vdash^{:=} T' := P \Rightarrow (\sigma'^A, W')$ where $\sigma' \subseteq \sigma'^A$

We invoke the IH on this match and the second premise to get

$MV(\Gamma, p[X]) \Vdash^{:=} T := ? \to P \Rightarrow (\sigma^A, W)$ where $\sigma'' \subseteq \sigma^A$

Applying matching rule $MForall$ gives us the desired result.

$MV(\Gamma, p) \Vdash^{:=} \forall X. T := ? \to P \Rightarrow (\sigma'^A - X, \forall X = \sigma''(X). W)$

Case $PChk$**:** Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \varnothing \quad \Gamma \vdash_{\Downarrow} t' : \sigma\ S \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : T \rightsquigarrow (p\ e', \sigma)}\ PChk$$

Our assumed match is

$MV(\Gamma, p\ e') \Vdash^{:=} T := P \Rightarrow (\sigma'^A, W), \sigma \subseteq \sigma'^A$.

We invoke matching rule $MArr$ to conlcude (noting $MV(\Gamma, e') = \varnothing$)

$MV(\Gamma, p) \Vdash^{:=} S \to T := ? \to P \Rightarrow (\sigma'^A, S \to W), \sigma \subseteq \sigma'^A$

Case $PSyn$ Our assumed derivation is

$$\frac{MV(\Gamma, \sigma\ S) = \overline{Y} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \overline{[U/Y]}\ \sigma\ S \rightsquigarrow e}{\Gamma \vdash^{\cdot} (p{:}S \to T, \sigma) \cdot t' : \overline{[U/Y]}\ T \rightsquigarrow ((\overline{[U/Y]}\ p)\ e', \sigma)}\ PSyn$$

Our assumed match is

$MV(\Gamma, \overline{[U/Y]} p\ e') \Vdash^{:=} \overline{[U/Y]} T := P \Rightarrow (\sigma'^A, W')$

By Lemma 11 (invertibility of substitutions in a match), we have

$MV(\Gamma, p\ e') \Vdash^{:=} T := P \Rightarrow (\sigma'^A \circ \sigma_{\overline{Y}}, W'_{\overline{Y}}),\ \sigma_{\overline{Y}} \subseteq \overline{[U/Y]}$

Noting that $\sigma \subseteq \sigma'^A \subseteq \sigma'^A \circ \sigma_{\overline{Y}}$, we apply rule $MArr$ to conclude

$MV(\Gamma, p\ e') \Vdash^{:=} S \to T := ? \to P \Rightarrow (\sigma'^A \circ \sigma_{\overline{Y}}, S \to W'_{\overline{Y}})$

□

5.8 Lemma: Match Solutions are Match Sound

Lemma 14.
If $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma \circ \sigma', W)$ *then* $\overline{X} - dom(\sigma') \Vdash^{:=} \sigma'\ T := P(\sigma, \sigma'\ W)$

Proof. By an easy inductive argument on the assumed derivation. □

5.9 Lemma: Function-ness of Matching

Lemma 15.
If $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ *and* $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma', W')$,
then $(\sigma, W) = (\sigma', W')$

Proof. By an easy inductive argument on the assumed derivation. □

5.10 Lemma: Checking extends Synthesizing

Lemma 16.
If $\Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e$ *then* $\Gamma \vdash_{\Downarrow} t : T \rightsquigarrow e$

Proof. A mostly easy induction, given that many rules are "direction-polymorphic." The only interesting case is *AppSyn*, which we look more closely at now

Case *AppSyn***:** Our assumed derivation is

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T' \rightsquigarrow (e', \sigma_{id}) \quad MV(\Gamma, e') = MV(\Gamma, T') = \varnothing}{\Gamma \vdash_{\Uparrow} t\ t' : T' \rightsquigarrow e'}\ AppSyn$$

But $dom(\sigma_{id}) = \varnothing$ so we already have what we need:

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ t' : T' \rightsquigarrow (e, \sigma_{id}) \quad MV(\Gamma, e') = MV(\Gamma, T') = dom(\sigma_{id})}{\Gamma \vdash_{\Downarrow} t\ t' : \sigma_{id}\ T' \rightsquigarrow \sigma_{id}\ e'}\ AppChk$$

□

5.11 Lemma: Matching Arrows of P and W:

Lemma 17.

Let $arr_P(P)$ be the number of prototype arrows prefixing P, and $arr_W(W)$ the number of decorated-type arrows preceding W.

- *If* $\Gamma; P \Vdash^{?} t : W \rightsquigarrow (p, \sigma)$ *then* $arr_W(W) \leq arr_P(P)$
- *If* $\overline{X} \Vdash^{:=} T := P \Rightarrow (\sigma, W)$ *then* $arr_P(P) \leq arr_W(W)$

Proof. Straightforward:

- The first point is a special case of the third, by invoking Lemma 6.
- By an easy inductive argument on the assumed derivation of $\Vdash^{:=}$, noting that the number of prototype and decorated arrows encountered during the inductive cases are equal up until the base case, in which they are either equal (*MType* and *M?*) or the former is strictly greater than the latter (*MCurr*).

□

5.12 Lemma: Subject type reveals an arrow in $\vdash^{\cdot}$

Lemma 18.

If $\Gamma \vdash^{\cdot} p{:}T \cdot t' : T' \rightsquigarrow (p', \sigma')$ *then* $T \ = \ \forall \overline{X}. T''$ *for some* T''

Proof. By a straightforward inductive argument on the assumed derivation. □

5.13 Lemma: Peek-ahead for $\vdash^{\cdot}$

Lemma 19.

If $\Gamma \vdash^{\cdot} (p{:}\forall \overline{X}. S \rightarrow T_{\overline{Y}}, \sigma) \cdot t' : T' \rightsquigarrow (p', \sigma')$ *then there exists some* $\sigma_{\overline{Y}}$ *with* $\overline{Y} \in MV(\Gamma, p) \cup \overline{X}$ *such that* $\sigma_{\overline{Y}}(T_{\overline{Y}}) = T'$

Proof. By a straightforward induction on the assumed derivation: in case *PChk* $\overline{Y} = \varnothing$, and in case *PSyn* $\sigma_{\overline{Y}}$ comes from synthesizing the type of t' and matching it against some expected type based on $T_{\overline{Y}}$ (and some guessing done in *PForall*). □

6 Qualified Completeness of $\vdash_{\delta}$ *wrt* $\vdash$

Definition 1. *(Annotation Requirements for Typing the External Language): Let e_P be a term of the internal language such that $\Gamma \vdash e_P : T_P$*[3] *Furthermore, let t_P be a term in the external language such that $t_P \in \lfloor e_P \rfloor$. We say that t_P meets our annotation requirements when the following conditions hold for each sub-expression e of e_P, corresponding sub-expression t of t_P, and corresponding sub-derivation $\Gamma' \vdash e : T$ of a derivation of $\Gamma \vdash e_P : T_P$:*

1. *If $e{=}\lambda\, x{:}S.\, e'$ for some S and e', then $t{=}\lambda\, x{:}S.\, t'$ for some t*
2. *If e occurs as a maximal term application in e_P and if $\Gamma' \vdash^{P} t : T' \rightsquigarrow (p, \sigma_{id})$ for some T' and p, then $MV(\Gamma, p){=}\varnothing$.*
3. *If e is a term application and $t = t_1\ t_2$ for some t_1 and t_2, and if $\Gamma' \vdash^{P} t_1 : T' \rightsquigarrow (p, \sigma_{id})$ for some T' and p, then $T'{=}\forall \overline{X}. S_1 \rightarrow S_2$ for some S_1 and S_2.*
4. *If e is a type application and $t = t'[S]$ for some t' and S, and $\Gamma' \vdash^{P} t' : T' \rightsquigarrow (p, \sigma_{id})$ for some T' and p, then $T'{=}\forall X. S'$ for some S'.*

If Definition 1 holds for some pair of internal and external terms (e_P, t_P) then we can show the following theorems.

[3]The subscript $_P$ indicates nothing more than an expression which we consider to be the whole **p**rogram we are typing.

- If e occurs somewhere in e_P as not a term applicand*, or if $\neg App(e)$, then $\Gamma' \vdash_{\Uparrow} t : T \rightsquigarrow e$
  $\Gamma' \vdash_{\Uparrow} t_P : T_P \rightsquigarrow e_P$ is a special case of this.
- If e occurs as an applicand in e_P and $e = e'\ \overline{[S]}$ for some e' and $\overline{S}$, $t = t'\ \overline{[S]}$ for some t', with $\neg TpApp(e')$, then $\Gamma \vdash^{\mathrm{P}} t : T' \rightsquigarrow (p, \sigma_{id})$ with some σ such that $dom(\sigma) = MV(\Gamma, p)$ and $\sigma\ (p, T') = (e, T)$
- If $TmApp(e)$ and $\Gamma \vdash e : T'$ then $\Gamma \vdash^{\mathrm{P}} t : T \rightsquigarrow (p', \sigma_{id})$
  with some σ such that $dom(\sigma) = MV(\Gamma, p')$ and $\sigma\ (p', T) = (e, T')$
- If
  - $\Gamma \vdash e\ \overline{[U_1]}\ \overline{[U_2]} : S' \to T'$ and $\Gamma \vdash_{\Uparrow} t' : S' \rightsquigarrow e'$
  - and some σ with $dom(\sigma) = MV(\Gamma, p\ \overline{[X_1]}\ \overline{[X_2]})$
    where $\sigma\ (p\ \overline{[X_1]}\ \overline{[X_2]}, S \to T) = (e\ \overline{[U_1]}\ \overline{[U_2]}, S' \to T')$
    (and $(|\overline{X_1}|, |\overline{X_2}|) = (|\overline{U_1}|, |\overline{U_2}|)$)

  then
  - $\Gamma \vdash^{\cdot} (p\overline{[X_1]} : \forall \overline{X_2}.\, S \to T, \sigma_{id}) \cdot t' : T'' \rightsquigarrow (p', \sigma_{id})$
  - with some σ' with $dom(\sigma') = MV(\Gamma, p')$ and where $\sigma\ (p', T'') = (e\ \overline{[U_1]}\ \overline{[U_2]}\ e', T')$

6.1 Qualified Completeness $\vdash_{\Uparrow}$ wrt $\vdash$

Theorem 11.
Under the qualifications of Definition 1, if e occurs as a non-applicand in e_P or if $\neg App(e)$ then $\Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e$

Proof. By induction on the assumed derivation

Case Var Our assumed derivation is

$$\frac{}{\Gamma \vdash x : \Gamma(x)}\ Var$$

There is only one partial erasure of x – x. We apply rule Var of $\vdash_{\Uparrow}$ to conclude

$$\frac{}{\Gamma \vdash_{\Uparrow} x : \Gamma(x) \rightsquigarrow x}\ Var$$

Case Abs Our assumed derivation is

$$\frac{\Gamma, x{:}T \vdash e : S}{\Gamma \vdash \lambda\, x{:}T.\, e : T \to S}\ AAbs$$

By our first assumed qualification, we have that our partial erasure t' of $\lambda\, x{:}T.\, e$ has the form $\lambda\, x{:}T.\, t$ for some partial erasure t of e. We invoke the IH (the body of our λ-abstraction, e, is not itself an applicand) and conclude

$$\frac{\dfrac{\Gamma, x{:}T \vdash e : S}{\Gamma, x{:}T \vdash_{\Uparrow} t : S \rightsquigarrow e}\ IH}{\Gamma \vdash_{\Uparrow} \lambda\, x{:}T.\, t : T \to S \rightsquigarrow \lambda\, x{:}T.\, e}\ AAbs$$

Case $TAbs$ Our assumed derivation is

$$\frac{\Gamma, X \vdash e : T}{\Gamma \vdash \Lambda\, X.\, e : \forall\, X.\, T}\ TAbs$$

We have a partial erasure $\Lambda\, X.\, t$ of $\Lambda\, X.\, e$, meaning that t is a partial erasure of e. We invoke the IH to conclude

$$\frac{\dfrac{\Gamma, X \vdash e : T}{\Gamma, X \vdash_{\Uparrow} t : T \rightsquigarrow e}\ IH}{\Gamma \vdash_{\Uparrow} \Lambda X.t : \forall\, X.\, T \rightsquigarrow \Lambda X.e}\ TAbs$$

Case $TApp$ Our assumed derivation is

$$\dfrac{\Gamma \vdash e : \forall X.T}{\Gamma \vdash e[S] : [S/X]T}\ TApp$$

By assumption, $e[S]$ occurs somewhere not as a term-applicand* in e_P. This means that its erasure t corresponding to the same position in t_P has form $t = t'[S]$ by the definition of erasure (we only erase type arguments between term to term applications).

Because $e[S]$ is not a term applicand*, neither is e[4]. Therefore, we can invoke the IH to conclude

$$\dfrac{\dfrac{\Gamma \vdash e : \forall X.T}{\Gamma \vdash_{\Uparrow} t : \forall X.T \rightsquigarrow e}\ IH}{\Gamma \vdash_{\Uparrow} t[S] : [S/X]T \rightsquigarrow e[S]}\ TApp$$

Case App Our assumed derivation is

$$\dfrac{\Gamma \vdash e : S' \to T' \quad \Gamma \vdash e' : S'}{\Gamma \vdash e\ e' : T'}\ App$$

Since the elaborated expression in question is $e\ e'$ we know that its erasure must be of the form $t\ t'$. We invoke mutual induction for the qualified completeness of $\vdash^{\mathrm{P}}$ for applications to get

$\Gamma \vdash^{\mathrm{P}} t\ t' : T \rightsquigarrow (p, \sigma_{id})$ with σ such that

- $dom(\sigma) = MV(\Gamma, p)$
- $\sigma\ (p, T) = (e\ e', T')$

Now, by assumption $e\ e'$ occurs as not a term applicand* in e_P. By qualification #2 we have $MV(\Gamma, p) = \varnothing$. We use this to rewrite the post-conditions of our invocation of mutual induction above:

- $dom(\sigma) = MV(\Gamma, p) = \varnothing \implies \sigma = \sigma_{id}$
- $\sigma\ (p, T) = \sigma_{id}(p, T) = (p, T) = (e\ e', T')$

We can now conclude

$$\dfrac{\Gamma \vdash^{\mathrm{P}} t\ t' : T' \rightsquigarrow (e\ e', \sigma_{id}) \quad MV(\Gamma, e\ e') = MV(\Gamma, T') = \varnothing}{\Gamma \vdash_{\Uparrow} t\ t' : T' \rightsquigarrow e\ e'}\ AppSyn$$

□

6.2 Qualified Completeness of $\vdash^{\mathrm{P}}$ wrt $\vdash$ (TApp)

Theorem 12. *Under the qualifications of Definition 1, if* e *occurs as a term applicand* in* e_P *and* $e = e'\overline{[S]}$ *for some* e' *and* $\overline{S}$*, and* $t = t'\overline{[S]}$ *for some* t'*, with* $\neg TpApp(e')$*, then* $\Gamma \vdash^{P} t : T' \rightsquigarrow (p, \sigma_{id})$ *with some* σ *such that* $dom(\sigma) = MV(\Gamma, p)$ *and* $\sigma\ (p, T) = (e, T')$

Proof. By induction on the length of $\overline{S}$

Case $\overline{[S]} = \varnothing$ Our assumed derivation is

$\Gamma' \vdash e : T'$

By assumption, $\neg TpApp(e)$. We therefore have that either $TmApp(e)$ or else $\neg App(e)$ In either case we can appeal to mutual induction on qualified completeness to conclude:

Subcase $TmApp(e)$**:** We appeal to qualified completeness of $\vdash^{\mathrm{P}}$ for applications (Theorem 13) to get

$\Gamma' \vdash^{\mathrm{P}} t : T \rightsquigarrow (p, \sigma_{id})$ with σ s.t. $dom(\sigma) = MV(\Gamma', p)$ and $\sigma\ (p, T) = (e, T')$, which is what we need to conclude.

[4]Recall the definitions of term applicand* earlier in the document

Subcase $\neg App(e)$**:** We appeal to qualified completeness of $\vdash_{\Uparrow}$ (Theorem 11) to get

$\Gamma' \vdash_{\Uparrow} t : T' \rightsquigarrow e$

and using rule $PHead$ we derive

$\Gamma' \vdash^{\mathrm{P}} t : T' \rightsquigarrow (e, \sigma_{id})$

noting that $MV(\Gamma', e) = \varnothing$ and $\sigma_{id}\ (e, T') = (e, T')$, which is what we need to conclude.

Case $\overline{[S]} = \overline{[S']}\ [S]$ Our assumed derivation is

$$\frac{\Gamma \vdash e\overline{[S']} : \forall X. T'}{\Gamma \vdash e\overline{[S']}\ [S] : [S/X]T'}\ TApp$$

By the IH we have

$$\frac{\Gamma \vdash e\ \overline{[S']} : \forall X. T'}{\Gamma \vdash^{\mathrm{P}} t\ \overline{[S']} : T \rightsquigarrow (p, \sigma_{id})}\ IH$$ with σ such that

- $dom(\sigma) = MV(\Gamma, p)$
- $\sigma\ (p, T) = (e\ \overline{[S']}, \forall X. T')$

By qualification #4 we have $T = \forall X. T''$. By combining this with the second post-conditions from the IH we get $\sigma\ T'' = T'$. We derive

$$\frac{\Gamma \vdash^{\mathrm{P}} t\ \overline{[S']} : \forall X. T'' \rightsquigarrow (p, \sigma_{id})}{\Gamma \vdash^{\mathrm{P}} t\ \overline{[S']}\ [S] : [S/X]T'' \rightsquigarrow (p[S], \sigma_{id})}\ PTApp$$

and note we can produce σ as the output substitutions, since

- $dom(\sigma) = MV(\Gamma, p[S]) = MV(\Gamma, p)$
- $\sigma\ (p[S], [S/X]T'') = (e\ \overline{[S']}\ [S], [S/X]T')$
  ($dom(\sigma) \cap X = \varnothing$ and $cod(\sigma)$ is only those types well-formed under Γ)

$\square$

6.3 Qualified Completeness of $\vdash^{\mathrm{P}}$ wrt $\vdash$ (App)

Theorem 13. *Under the qualifications of Definition 1, if* $TmApp(e)$ *and* $\Gamma' \vdash e : T'$ *then* $\Gamma \vdash^{P} t : T \rightsquigarrow (p, \sigma_{id})$ *with some* σ *such that* $dom(\sigma) = MV(\Gamma, p)$ *and* $\sigma\ (p, T) = (e, T')$

Proof. Directly. Our internal term is $e\ e'$ and external term is $t\ t'$, and our assumed derivation is

$$\frac{\Gamma \vdash e : S' \to T' \qquad \Gamma \vdash e' : S'}{\Gamma \vdash e\ e' : T'}\ App$$

We can rewrite $e = e''\ \overline{[U]}$, making visiable all of the outermost type applications in e (if any). Since e is an applicand, we know that its erasure t may have had some number of the right-most type applications erased – so $t = t''\ \overline{[U_1]}$ where $\overline{[U]} = \overline{[U_1]}\ \overline{[U_2]}$

We examine the first premise of our assumed derivation. We now know it must have the following form:

$$\frac{\dfrac{\Gamma \vdash e''\ \overline{[U_1]} : \forall \overline{X}. T'_{\overline{X}}}{\Gamma \vdash e''\ \overline{[U_1]}\ \overline{[U_2]} : S' \to T'}\ TApp... \qquad \Gamma \vdash e' : S'}{\Gamma \vdash e''\ \overline{[U_1]}\ \overline{[U_2]}\ e' : T'}\ App$$

where the (left-to-right ordered) substitution $\overline{[U_2/X]}T'_{\overline{X}} = S' \to T'$

We appeal to mutual induction on the completeness of $\vdash^{\mathrm{P}}$ $wrt \vdash$ (Theorem 12) to get:

$\Gamma \vdash^{\mathrm{P}} t''\ \overline{[U_1]} : T_{\overline{X}} \rightsquigarrow (p, \sigma_{id})$ with σ such that

- $dom(\sigma) = MV(\Gamma, p)$

- $\sigma\ (p, T_{\overline{X}}) = (e\ \overline{[U_1]}, \forall \overline{X}.T'_{\overline{X}})$

Now, by qualification #3, from our $\vdash^{\mathrm{P}}$ derivation of applicand t we have that $T_{\overline{X}} = \forall \overline{X}.S \to T$ for some S and T. The use of the same bound type variables $\overline{X}$ as used in $\forall \overline{X}.T'_{\overline{X}}$ is justified by rewriting the equality concerning $\sigma\ T_{\overline{X}}$ above with this new information:

$\sigma\ (p, \forall\, \overline{X}.\, S \to T = (e''\overline{[U_1]}, \forall\, \overline{X}.\, T'_{\overline{X}})$

We now appeal to completeness of $\vdash^{\cdot}$ *wrt* $\vdash$. We satisfy its preconditions:

- $\Gamma \vdash e''\ \overline{[U_1]}\ \overline{[U_2]} : S' \to T'$ and $\Gamma \vdash_{\Uparrow} t' : S' \rightsquigarrow e'$

  The second of these we get by mutual induction on the completeness of $\vdash_{\Uparrow}$ *wrt* $\vdash$, noting that e' occurs in a non-applicand position.

- some σ'' with $dom(\sigma'') = MV(\Gamma, p\ \overline{[X]})$
  where $\sigma''(p\ \overline{[X]}, S \to T) = ((e''\ \overline{[U_1]})\ \overline{[U_2]}, S' \to T')$
  (and $(|\varnothing|, |\overline{X}|) = (|\varnothing|, |\overline{U_2}|)$)

  Note that we parenthesize $(e''\ \overline{[U_1]})$ for clarification [5] We are not providing vectorized type arguments $\overline{U_1}$ and $\overline{U_2}$ to the theorem – we are providing type arguments $\varnothing$ and $\overline{U_2}$, and corresponding $\varnothing$ and $\overline{X}$ for the vectorized type meta-variables.

  The σ'' we provide is $\overline{[U_2/X]} \circ \sigma$

Having set this up, we get the following from mutual induction:

- $\Gamma \vdash^{\cdot} (p{:}\forall\, \overline{X}.\, S \to T, \sigma_{id}) \cdot t' : T'' \rightsquigarrow (p', \sigma_{id})$
- some σ' with $dom(\sigma') = MV(\Gamma, p')$
  where $\sigma'\ (p', T'') = ((e''\ \overline{[U_1]})\ \overline{[U_2]}\ e', T')$

which is what we need to conclude. ∎

6.4 Qualified Completeness of $\vdash^{\cdot}$ wrt $\vdash$

Theorem 14. *Under the qualifications of Definition 1, if*

- $\Gamma \vdash e\ \overline{[U_1]}\ \overline{[U_2]} : S' \to T'$ *and* $\Gamma \vdash_{\Uparrow} t' : S' \rightsquigarrow e'$
- *and some* σ *with* $dom(\sigma) = MV(\Gamma, p\ \overline{[X_1]}\ \overline{[X_2]})$
  where $\sigma\ (p\ \overline{[X_1]}\ \overline{[X_2]}, S \to T) = (e\ \overline{[U_1]}\ \overline{[U_2]}, S' \to T')$
  (and $(|\overline{X_1}|, |\overline{X_2}|) = (|\overline{U_1}|, |\overline{U_2}|)$*)*

then

- $\Gamma \vdash^{\cdot} (p\overline{[X_1]}{:}\forall\, \overline{X_2}.\, S \to T, \sigma_{id}) \cdot t' : T'' \rightsquigarrow (p', \sigma_{id})$
- *with some* σ' *with* $dom(\sigma') = MV(\Gamma, p')$ *and where* $\sigma\ (p', T'') = (e\ \overline{[U_1]}\ \overline{[U_2]}\ e', T')$

Proof. By induction on $\overline{X_2}$

Case $\overline{[X_2]} = X, \overline{X'_2}$ We have

- $\Gamma \vdash e\ \overline{[U_1]}\ [U]\ \overline{[U'_2]} : S' \to T'$
- $\sigma\ (p\overline{[X_1]}[X]\overline{[X'_2]}, S \to T) = (e\ \overline{[U_1]}\ [U]\ \overline{[U'_2]}, S' \to T')$

We appeal to the IH using variable groups $\overline{X'_1} = \overline{X_1}, X$ and $\overline{X'_2}$, noting that this regrouping does not keep us from providing the conditions we received on our assumed derivation to the inductive invocation. We get

- $\Gamma \vdash^{\cdot} (p\overline{[X_1]}[X]{:}\forall\, \overline{X'_2}.\, S \to T, \sigma_{id}) \cdot t' : T'' \rightsquigarrow (p', \sigma_{id})$
- with some σ' with $dom(\sigma') = MV(\Gamma, p')$ and where $\sigma\ (p', T'') = (e\ \overline{[U_1]}\ \overline{[U_2]}\ e', T')$

[5] or at least some attempt at it

From this we derive

$$\frac{\Gamma \vdash^{\cdot} (p\overline{[X_1]}[X]\!:\!\forall \overline{X_2'}.\, S \to T, \sigma_{id}) \cdot t' : T' \rightsquigarrow (p', \sigma_{id})}{\Gamma \vdash^{\cdot} (p\overline{[X_1]}\!:\!\forall X, \overline{X_2'}.\, S \to T, \sigma_{id}) \cdot t' : T' \rightsquigarrow (p', \sigma_{id})}\ PForall$$

And provide the σ' prime we received from our IH, noting that the conditions on it are precisely what we need to conclude.

Case $\overline{[X_2]} = \varnothing$ We have

- $\Gamma \vdash e\ \overline{[U_1]} : S' \to T'$
- $\Gamma \vdash_{\Uparrow} t' : S' \rightsquigarrow e'$
- and $\sigma\ (p\ \overline{[X_1]}, S \to T) = (e\ \overline{[U_1]}, S' \to T')$.

To proceed, we must do case analysis on whether $MV(\Gamma, S) = \varnothing$ or not.

Subcase $MV(\Gamma, S) = \varnothing$**:** Because $MV(\Gamma, S) \subseteq MV(\Gamma, p) = dom(\sigma)$, we have $\sigma\ S = (\sigma \cap MV(\Gamma, S))(S) = \sigma_{id}(S) = S'$. So we have by rewriting our second assumption that

$\Gamma \vdash_{\Uparrow} t' : S \rightsquigarrow e'$

By using the fact that checking mode extends synthesizing for the specificational rules (Lemma 20) we can derive $\Gamma \vdash_{\Downarrow} t' : S \rightsquigarrow e'$ to get

$$\frac{MV(\Gamma, S) = \varnothing \quad \Gamma \vdash_{\Downarrow} t' : S \rightsquigarrow e'}{\Gamma \vdash^{\cdot} (p\overline{[X_1]}\!:\!S \to T, \sigma_{id}) \cdot t' : T \rightsquigarrow (p\overline{[X_1]}\ e', \sigma_{id})}\ PChk$$

We must now provide a suitable σ' completing our partial type synthesis. Pick our assumed σ. Then we have

- $dom(\sigma) = MV(\Gamma, p\ \overline{[X_1]}\ e')$
- $\sigma\ (p\overline{[X_1]}\ e', T) = (e\ \overline{[U_1]}\ e', T')$

allowing us to conclude this sub-case.

Subcase $MV(\Gamma, S) = \overline{Y} \neq \varnothing$**:** We know that $MV(\Gamma, S) = \overline{Y} \subseteq MV(\Gamma, p) = dom(\sigma)$. Let $\sigma_{\overline{Y}} = \sigma \cap \overline{Y}$. Then we know $\sigma\ S = \sigma_{\overline{Y}}(S) = S'$. We have

$\Gamma \vdash_{\Uparrow} t' : \sigma_{\overline{Y}}\ S \rightsquigarrow e'$

We can derive

$$\frac{MV(\Gamma, S) = \overline{Y} \neq \varnothing \quad \Gamma \vdash_{\Uparrow} t' : \sigma_{\overline{Y}}\ S \rightsquigarrow e}{\Gamma \vdash^{\cdot} (p\overline{[X_1]}\!:\!S \to T, \sigma_{id}) \cdot t' : \sigma_{\overline{Y}}\ T \rightsquigarrow ((\sigma_{\overline{Y}}\ p\overline{[X_1]})\ e', \sigma_{id})}\ PSyn$$

We must now pick a suitable σ'. Pick $\sigma - \sigma_{\overline{Y}}$. We have

- $dom(\sigma - \sigma_{\overline{Y}}) = MV(\Gamma, p\overline{[X_1]}\ e') - \overline{Y} = MV(\Gamma, (\sigma_{\overline{Y}}\ p\overline{[X_1]})\ e')$
- $\sigma'\ \sigma_{\overline{Y}}\ (p\overline{[X_1]}\ e', \sigma_{\overline{Y}}\ T)$
  $= \sigma\ (p\ \overline{[X_1]}\ e', T)$
  $= (e\ \overline{[U_1]}\ e', T')$

which is what we need to conclude. □

6.5 Lemma: Checking extends Synthesizing (Specification)

Lemma 20. *If* $\Gamma \vdash_{\Uparrow} t : T \rightsquigarrow e$ *then* $\Gamma \vdash_{\Downarrow} t : T \rightsquigarrow e$

Proof. Directly. Take the assumed derivation of $\vdash_{\Uparrow}$, invoke completeness of $\Vdash_{\Uparrow}$ wrt $\vdash_{\Uparrow}$ (refthmcomplete-alg), use the fact that checking extends synthesizing for the algorithmic rules (Lemma 16), and then finish by invoking soundness of the $\Vdash_{\Downarrow}$ wrt $\vdash_{\Downarrow}$ (Theorem 2). □