CPS Translation of Dependent Types

Amal Ahmed

Northeastern University

Work in progress, with Nick Rioux and William Bowman

Compiling Dependent Types

- Much recent focus on verified compilation of dependently typed languages: Coqonut, CertiCoq
- Our goal: *type-preserving*, compositional verified compilation of Coq/Agda
- Types at target level can be used to provide protection from target contexts/attackers (fully abstract compilation)

CPS Translation of Dependent Types

Prior work

- CPS Translations and Applications: The Cube and Beyond *[Barthe, Hatcliff, Sorenson HOSC'99]*
- *[Barthe & Uustalu PEPM'02]*

CPS Translation of Dependent Types

Prior work

- CPS Translations and Applications: The Cube and Beyond *[Barthe, Hatcliff, Sorenson HOSC'99]*
- *[Barthe & Uustalu PEPM'02]*
    - Good news: *"CPS translations… generalize for dependently typed calculi"*

CPS Translation of Dependent Types

Prior work

- CPS Translations and Applications: The Cube and Beyond *[Barthe, Hatcliff, Sorenson HOSC'99]*
- *[Barthe & Uustalu PEPM'02]*
  - Good news: *"CPS translations… generalize for dependently typed calculi"*
  - Bad news: *"No translation is possible along the same lines for small Σ-types and sum types with dependent case"*

This Talk: CPS-ing CoC with Σ

Kinds	κ	$::=$	$* \mid \Pi\, x : X.\, \kappa \mid \Pi\, \alpha : \kappa_1.\, \kappa_2$
Types	A, X	$::=$	$\alpha \mid \Pi\, x : X.\, Y \mid \Pi\, \alpha : \kappa.\, X \mid \Sigma\, x : X.\, Y \mid$ $\lambda\, x : X.\, A \mid A\ e \mid \lambda\, \alpha : \kappa.\, A \mid A\ B \mid e_1 =_X e_2$
Terms	e	$::=$	$x \mid \lambda\, x : X.\, e \mid \lambda\, \alpha : \kappa.\, e \mid e_1\ e_2 \mid e\ A \mid$ $\langle e_1, e_2 \rangle \mid \mathsf{fst}\ e \mid \mathsf{snd}\ e \mid \mathsf{refl}$

This Talk: CPS-ing CoC with Σ

Kinds	κ	$::=$	$* \mid \Pi\, x : X.\, \kappa \mid \Pi\, \alpha : \kappa_1.\, \kappa_2$
Types	A, X	$::=$	$\alpha \mid \Pi\, x : X.\, Y \mid \Pi\, \alpha : \kappa.\, X \mid \Sigma\, x : X.\, Y \mid$ $\lambda\, x : X.\, A \mid A\ e \mid \lambda\, \alpha : \kappa.\, A \mid A\ B \mid e_1 =_X e_2$
Terms	e	$::=$	$x \mid \lambda\, x : X.\, e \mid \lambda\, \alpha : \kappa.\, e \mid e_1\ e_2 \mid e\ A \mid$ $\langle e_1, e_2 \rangle \mid \mathsf{fst}\ e \mid \mathsf{snd}\ e \mid \mathsf{refl}$

X, Y denote types of kind $*$

Typed CPS: STLC (call by name)

Computation translation $\tau^{\div}$

$$\tau^{\div} = (\tau^{+} \to \bot) \to \bot$$

Value translation τ^{+}

$$\begin{aligned} \mathsf{bool}^{+} &= \mathsf{bool} \\ (\tau_1 \to \tau_2)^{+} &= \tau_1^{\div} \to \tau_2^{\div} \end{aligned}$$

Typed CPS: STLC (call by name)

Computation translation $\tau^{\div}$

$$\tau^{\div} = (\tau^{+} \to \bot) \to \bot$$

Value translation τ^{+}

$$\mathsf{bool}^{+} = \mathsf{bool}$$

$$(\tau_1 \to \tau_2)^{+} = \tau_1^{\div} \to \tau_2^{\div}$$

$$\tau_1^{\div} \to (\tau_2^{+} \to \bot) \to \bot$$

Typed CPS: Dependent Types (cbn)

Computation translation $X^{\div}$

$$X^{\div} = (X^{+} \to \bot) \to \bot$$

Value translation X^{+}

$$\begin{array}{rcl}
\alpha^{+} & = & \alpha \\
(\Pi\, x : X.\, Y)^{+} & = & \Pi\, x : X^{\div}.\, Y^{\div} \\
(\Sigma\, x : X.\, Y)^{+} & = & \Sigma\, x : X^{\div}.\, Y^{\div} \\
& \vdots &
\end{array}$$

Typed CPS: Dependent Types (cbn)

Computation translation $X^{\div}$

$\Gamma \vdash X : * \overset{\div}{\rightsquigarrow} X'$

$$X^{\div} = (X^{+} \to \bot) \to \bot$$

Value translation X^{+}

$$\begin{array}{rcl} \alpha^{+} & = & \alpha \\ (\Pi\, x : X.\, Y)^{+} & = & \Pi\, x : X^{\div}.\, Y^{\div} \\ (\Sigma\, x : X.\, Y)^{+} & = & \Sigma\, x : X^{\div}.\, Y^{\div} \\ & \vdots & \end{array}$$

Typed CPS: Dependent Types (cbn)

Computation translation $X^{\div}$

$\Gamma \vdash X : * \overset{\div}{\rightsquigarrow} X'$

$$X^{\div} = (X^{+} \to \bot) \to \bot$$

Value translation X^{+}

$\Gamma \vdash A : \kappa \overset{+}{\rightsquigarrow} A'$

$$\begin{array}{rcl} \alpha^{+} & = & \alpha \\ (\Pi\, x : X.\, Y)^{+} & = & \Pi\, x : X^{\div}.\, Y^{\div} \\ (\Sigma\, x : X.\, Y)^{+} & = & \Sigma\, x : X^{\div}.\, Y^{\div} \\ & \vdots & \end{array}$$

Typed CPS: pair … warm up

$$\frac{\Gamma \vdash e_1 : X \qquad \Gamma \vdash e_2 : Y[e_1/x]}{\Gamma \vdash \langle e_1,\, e_2 \rangle : \Sigma\, x : X.\, Y}$$

Typed CPS: pair … warm up

$$\frac{\Gamma \vdash e_1 : X \qquad \Gamma \vdash e_2 : Y[e_1/x]}{\Gamma \vdash \langle e_1,\, e_2 \rangle : \Sigma\, x : X.\, Y} \;\stackrel{\div}{\rightsquigarrow}$$

Typed CPS: pair ... warm up

$$\frac{\Gamma \vdash e_1 : X \overset{\div}{\rightsquigarrow} e_1^{\div} : X^{\div} \quad \Gamma \vdash e_2 : Y[e_1/x]}{\Gamma \vdash \langle e_1, e_2 \rangle : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow}}$$

Typed CPS: pair ... warm up

$$\frac{\Gamma \vdash e_1 : X \overset{\div}{\rightsquigarrow} e_1^{\div} : X^{\div} \quad \Gamma \vdash e_2 : Y[e_1/x] \overset{\div}{\rightsquigarrow} e_2^{\div} : Y^{\div}[e_1^{\div}/x]}{\Gamma \vdash \langle e_1, e_2 \rangle : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow}}$$

Typed CPS: pair ... warm up

$$\frac{\Gamma \vdash e_1 : X \overset{\div}{\rightsquigarrow} e_1^{\div} : X^{\div} \quad \Gamma \vdash e_2 : Y[e_1/x] \overset{\div}{\rightsquigarrow} e_2^{\div} : Y^{\div}[e_1^{\div}/x]}{\Gamma \vdash \langle e_1, e_2 \rangle : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} \lambda\, k : (\Sigma\, x : X^{\div}.\, Y^{\div}) \to \bot.}$$

Typed CPS: pair ... warm up

$$\frac{\Gamma \vdash e_1 : X \overset{\div}{\rightsquigarrow} e_1^{\div} : X^{\div} \quad \Gamma \vdash e_2 : Y[e_1/x] \overset{\div}{\rightsquigarrow} e_2^{\div} : Y^{\div}[e_1^{\div}/x]}{\Gamma \vdash \langle e_1, e_2 \rangle : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} \lambda\, k : (\Sigma\, x : X^{\div}.\, Y^{\div}) \to \bot.\; k\ \langle e_1^{\div}, e_2^{\div} \rangle}$$

Typed CPS: fst ... all's well

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y}{\Gamma \vdash \mathsf{fst}\; e : X} \quad \stackrel{\div}{\rightsquigarrow}$$

Typed CPS: fst ... all's well

$$\frac{\Gamma \vdash e : \Sigma\, x : X. Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma\, x : X^{\div}. Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{fst}\; e : X \overset{\div}{\rightsquigarrow}}$$

Typed CPS: fst ... all's well

$$\frac{\Gamma \vdash e : \Sigma x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{fst}\, e : X \overset{\div}{\rightsquigarrow} \lambda k : X^{+} \to \bot.}$$

Typed CPS: fst ... all's well

$$\frac{\Gamma \vdash e : \Sigma x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{fst}\; e : X \overset{\div}{\rightsquigarrow} \lambda k : X^{+} \to \bot.\; e^{\div}\; (\lambda p : (\Sigma x : X^{\div}.\, Y^{\div}).\, \mathsf{let}\; z = \mathsf{fst}\; p\; \mathsf{in}\; z\; k)}$$

Typed CPS: snd ... the evil case!

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{snd}\ e : Y[\mathsf{fst}\ e/x] \overset{\div}{\rightsquigarrow}}$$

Typed CPS: snd ... the evil case!

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \stackrel{\div}{\rightsquigarrow} e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{snd}\ e : Y[\mathsf{fst}\ e/x] \stackrel{\div}{\rightsquigarrow} \lambda\, k : (Y[\mathsf{fst}\ e/x])^{+} \to \bot.}$$

Typed CPS: snd … the evil case!

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \;\overset{\div}{\rightsquigarrow}\; e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{snd}\; e : Y[\mathsf{fst}\; e/x] \;\overset{\div}{\rightsquigarrow}\; \begin{array}{l} \lambda\, k : (Y[\mathsf{fst}\; e/x])^{+} \to \bot. \\ \quad e^{\div}\; (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}). \\ \qquad \mathsf{let}\; y = \mathsf{snd}\; p\; \mathsf{in}\; y\; k) \end{array}}$$

Typed CPS: snd … the evil case!

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \;\overset{\div}{\rightsquigarrow}\; e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{snd}\; e : Y[\mathsf{fst}\; e/x] \;\overset{\div}{\rightsquigarrow}\; \lambda\, k : (Y[\mathsf{fst}\; e/x])^{+} \to \bot.\; e^{\div}\; (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).\; \mathsf{let}\; y = \mathsf{snd}\; p\; \mathsf{in}\; y\; k)}$$

$Y^{+}[(\mathsf{fst}\; e)^{\div}/x] \to \bot$

$Y^{\div}[\mathsf{fst}\; p/x]$

Typed CPS: snd … the evil case!

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot}{\Gamma \vdash \mathsf{snd}\ e : Y[\mathsf{fst}\ e/x] \overset{\div}{\rightsquigarrow} \lambda\, k : (Y[\mathsf{fst}\ e/x])^{+} \to \bot.\ e^{\div}\ (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).\ \mathsf{let}\ y = \mathsf{snd}\ p\ \mathsf{in}\ y\ k)}$$

$$Y^{+}[(\mathsf{fst}\ e)^{\div}/x] \to \bot$$

$$(Y^{+}[\mathsf{fst}\ p/x] \to \bot) \to \bot$$

Typed CPS: snd … the evil case!

$$\Gamma \vdash e : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div} : (\Sigma\, x : X^{\div}.\, Y^{\div} \to \bot) \to \bot$$

$$\Gamma \vdash \mathsf{snd}\ e : Y[\mathsf{fst}\ e/x] \overset{\div}{\rightsquigarrow} \lambda\, k : (Y[\mathsf{fst}\ e/x])^{+} \to \bot.\ e^{\div}\ (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).\ \mathsf{let}\ y = \mathsf{snd}\ p\ \mathsf{in}\ y\ k)$$

$$Y^{+}[(\mathsf{fst}\ e)^{\div}/x] \to \bot$$

$$(Y^{+}[\mathsf{fst}\ p/x] \to \bot) \to \bot$$

Reasoning about value passed to cont.

$$e : (X^+ \to \bot) \to \bot$$

Want to extract the content of type X^+ inside e

Reasoning about value passed to cont.

$$e : (X^+ \to \bot) \to \bot$$

Want to extract the content of type X^+ inside e

Idea: change the type translation

$$e : \Pi\, \alpha : * \,.\, (X^+ \to \alpha) \to \alpha$$

Reasoning about value passed to cont.

$$e : (X^+ \to \bot) \to \bot$$

Want to extract the content of type X^+ inside e

Idea: change the type translation

$$e : \Pi\, \alpha : * \,.\, (X^+ \to \alpha) \to \alpha$$

Now, we can extract via: $e\; X^+\; \mathsf{id}$

Reasoning about value passed to cont.

$$e : (X^{+} \to \bot) \to \bot$$

Want to extract the content of type X^{+} inside e

Idea: change the type translation

$$e : \Pi\, \alpha : * \,.\, (X^{+} \to \alpha) \to \alpha$$

Now, we can extract via: $e\; X^{+}\; \mathsf{id}$

$$X^{\div} = \Pi\, \alpha : * \,.\, (X^{+} \to \alpha) \to \alpha$$

Typed CPS: snd again

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div}}{\begin{array}{r}\Gamma \vdash \mathsf{snd}\; e : Y[\mathsf{fst}\; e/x] \overset{\div}{\rightsquigarrow} \lambda\, \alpha : * .\, \lambda\, k : (Y[\mathsf{fst}\; e/x])^{+} \to \alpha. \\ e^{\div}\; \alpha\; (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}). \\ \mathsf{let}\; y = \mathsf{snd}\; p\; \mathsf{in}\; y\; \alpha\; k)\end{array}}$$

Typed CPS: snd again

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \;\overset{\div}{\rightsquigarrow}\; e^{\div}}{\Gamma \vdash \mathsf{snd}\; e : Y[\mathsf{fst}\; e/x] \;\overset{\div}{\rightsquigarrow}\; \lambda\, \alpha : * \,.\, \lambda\, k : (Y[\mathsf{fst}\; e/x])^{+} \to \alpha.\; e^{\div}\; \alpha\; (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).\; \mathsf{let}\; y = \mathsf{snd}\; p\; \mathsf{in}\; y\; \alpha\; k)}$$

$$e^{\div}\; (\Sigma\, x : X^{\div}.\, Y^{\div})\; \mathsf{id}$$

Typed CPS: snd again

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} e^{\div}}{\Gamma \vdash \mathsf{snd}\ e : Y[\mathsf{fst}\ e/x] \overset{\div}{\rightsquigarrow} \lambda\, \alpha : *\, .\, \lambda k : (Y[\mathsf{fst}\ e/x])^{+} \rightarrow \alpha.}$$

$$e^{\div}\ \alpha\ (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).$$

$$\mathsf{let}\ y = \mathsf{snd}\ p\ \mathsf{in}\ y\ \alpha\ k)$$

$Y^{+}[(\mathsf{fst}\ e)^{\div}/x] \rightarrow \alpha$

$e^{\div}\ (\Sigma\, x : X^{\div}.\, Y^{\div})\ \mathsf{id}$

$(Y^{+}[\mathsf{fst}\ p/x] \rightarrow \alpha) \rightarrow \alpha$

Typed CPS: snd again

$$\frac{\Gamma \vdash e : \Sigma\, x : X.\, Y \;\stackrel{\div}{\rightsquigarrow}\; e^{\div}}{\Gamma \vdash \mathsf{snd}\; e : Y[\mathsf{fst}\; e/x] \;\stackrel{\div}{\rightsquigarrow}\; \lambda\, \alpha : *\, .\, \lambda k : (Y[\mathsf{fst}\; e/x])^{+} \to \alpha.\; e^{\div}\; \alpha\; (\lambda\, p : (\Sigma\, x : X^{\div}.\, Y^{\div}).\; \mathsf{let}\; y = \mathsf{snd}\; p\; \mathsf{in}\; y\; \alpha\; k)}$$

$Y^{+}[(\mathsf{fst}\; e)^{\div}/x] \to \alpha$

$e^{\div}\; (\Sigma\, x : X^{\div}.\, Y^{\div})\; \mathsf{id}$

$(Y^{+}[\mathsf{fst}\; p/x] \to \alpha) \to \alpha$

2 issues:

- how to typecheck above continuation
- how to prove $\mathsf{fst}\; p \equiv (\mathsf{fst}\; e)^{\div}$

For Issue 1: New Typing Rule

$$\frac{\Gamma \vdash e : \Pi\, \alpha : *\,.\,(X \to \alpha) \to \alpha \qquad \Gamma \vdash Y : * \qquad \Gamma, x : X, u : x =_X e\; X\; \mathsf{id} \vdash e_b : Y}{\Gamma \vdash e\; Y\; (\lambda\, x : X.\, e_b) : Y}$$

For Issue 1: New Typing Rule

$$\frac{\Gamma \vdash e : \Pi\,\alpha : *\,.\,(X \to \alpha) \to \alpha \qquad \Gamma \vdash Y : * \qquad \Gamma, x : X, u : x =_X e\,X\,\mathsf{id} \vdash e_b : Y}{\Gamma \vdash e\,Y\,(\lambda\,x : X.\,e_b) : Y}$$

Note: e may apply continuation many times…

For Issue 1: New Typing Rule

$$\frac{\Gamma \vdash e : \Pi\, \alpha : *\, .\, (X \to \alpha) \multimap \alpha \qquad \Gamma \vdash Y : * \qquad \Gamma, x : X, u : x =_X e\, X\, \mathsf{id} \vdash e_b : Y}{\Gamma \vdash e\, Y\, (\lambda\, x : X.\, e_b) : Y}$$

Need linearity to ensure safe execution of e_b

For Issue 1: New Typing Rule

$$\frac{\Gamma \vdash e : \Pi\, \alpha : *\,.\, (X \to \alpha) \multimap \alpha \qquad \Gamma \vdash Y : * \qquad \Gamma, x : X, u : x =_X e\, X\, \mathsf{id} \vdash e_b : Y}{\Gamma \vdash e\, Y\, (\lambda\, x : X.\, e_b) : Y}$$

Need linearity to ensure safe execution of e_b

Need to prove soundness of the above rule — requires internalizing parametricity:

- *[Krishnaswami-Dreyer, CSL'13]*
- *[Bernardy et al., ICFP'10, JFP'12]*

For Issue 2: Parametricity Condition

Given that $p \equiv e^{\div}\ (\Sigma\, x : X^{\div} .\, Y^{\div})\ \mathsf{id}$

we can prove $\mathsf{fst}\ p \equiv (\mathsf{fst}\ e)^{\div}$

if the parametricity condition holds.

Parametricity Condition: *[Wadler'89]*

If $e : \Pi\, \alpha : * .\, (X \to \alpha) \multimap \alpha$

and $k : X \to Y$

then $e\, Y\, k \ \equiv\ k\, (e\, X\ \mathsf{id})$

Concluding thoughts…

Type-preserving CPS for Coq/Agda

- do the same issues arise if we translate to ANF?

What is the "right" type for a CPS'd term?

- $\tau^{\div} = (\tau^{+} \to \bot) \to \bot$
- $\tau^{\div} = (\tau^{+} \to \bot) \multimap \bot$
- $\tau^{\div} = \forall \alpha.\, (\tau^{+} \to \alpha) \multimap \alpha$
- reversibility of CPS and linear use of continuation seem critical (e.g., for fully abstract compilation, interoperability with direct-style languages)

Questions?

$$\frac{\Gamma \vdash e_1 : X \overset{\div}{\rightsquigarrow} e_1^{\div} \qquad \Gamma \vdash e_2 : Y[e_1/x] \overset{\div}{\rightsquigarrow} e_2^{\div}}{\Gamma \vdash \langle e_1, e_2 \rangle : \Sigma\, x : X.\, Y \overset{\div}{\rightsquigarrow} \lambda\, \alpha : *\, .\, \lambda\, k : (\Pi\, \_ : (\Sigma\, x : X^{\div}.\, Y^{\div}).\, \alpha).\, k\ \langle e_1^{\div}}$$