Typed recursion in the rewriting calculus

Benjamin Wack

joint work with C. Kirchner, L. Liquori, H. Cirstea

ρWorkshop, March 10th 2004, Nancy

The type system

Typing fixpoints

Encoding typed objects and TRS

Logical failure

The Type System I

$$\frac{\mathcal{X} : \sigma \in \Gamma}{\Gamma \vdash \mathcal{X} : \sigma} \; (Var)$$

The Type System I

$$\frac{\mathcal{X} : \sigma \in \Gamma}{\Gamma \vdash \mathcal{X} : \sigma} \; (Var)$$

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \rightarrow \tau \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 \; \mathcal{T}_2 : \tau} \; (Appl)$$

The Type System I

$$\frac{\mathcal{X} : \sigma \in \Gamma}{\Gamma \vdash \mathcal{X} : \sigma} \ (Var)$$

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \rightarrowtriangle \tau \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 \ \mathcal{T}_2 : \tau} \ (Appl)$$

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 \longrightarrow_{\Delta} \mathcal{T}_2 : \sigma \rightarrowtriangle \tau} \ (Abs)$$

The Type System I

$$\frac{\mathcal{X} : \sigma \in \Gamma}{\Gamma \vdash \mathcal{X} : \sigma} \ (Var)$$

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \rightarrow \tau \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 \ \mathcal{T}_2 : \tau} \ (Appl)$$

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 \rightarrow_{\Delta} \mathcal{T}_2 : \sigma \rightarrow \tau} \ (Abs)$$

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma \vdash \mathcal{T}_2 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_3 : \tau}{\Gamma \vdash [\mathcal{T}_1 \ll_{\Delta} \mathcal{T}_2]\mathcal{T}_3 : \tau} \ (Match)$$

The Type System II

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 ; \mathcal{T}_2 : \sigma} \ (Struct)$$

The Type System II

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 ; \mathcal{T}_2 : \sigma} \ (Struct)$$

$$\frac{\Gamma \vdash \mathcal{T} : \sigma \quad \alpha \notin FV(\Gamma)}{\Gamma \vdash \mathcal{T} : \forall \alpha . \sigma} \ (Abs{-}\forall) \qquad \frac{\Gamma \vdash \mathcal{T} : \sigma \quad \alpha \notin FV(\Gamma)}{\Gamma \vdash \alpha \rightarrow \mathcal{T} : \forall \alpha . \sigma} \ (Abs{-}\forall)$$

The Type System II

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1; \mathcal{T}_2 : \sigma} \; (Struct)$$

$$\frac{\Gamma \vdash \mathcal{T} : \sigma \quad \alpha \notin FV(\Gamma)}{\Gamma \vdash \mathcal{T} : \forall \alpha . \sigma} \; (Abs{-}\forall) \qquad \frac{\Gamma \vdash \mathcal{T} : \sigma \quad \alpha \notin FV(\Gamma)}{\Gamma \vdash \alpha \to \mathcal{T} : \forall \alpha . \sigma} \; (Abs{-}\forall)$$

$$\frac{\Gamma \vdash \mathcal{T} : \forall \alpha . \sigma}{\Gamma \vdash \mathcal{T} : \sigma\{\tau / \alpha\}} \; (Appl{-}\forall) \qquad \frac{\Gamma \vdash \mathcal{T} : \forall \alpha . \sigma}{\Gamma \vdash \mathcal{T} \tau : \sigma\{\tau / \alpha\}} \; (Appl{-}\forall)$$

Typing properties

Well-typed matching:

If $\mathcal{S}ol(\mathcal{P} \ll \mathcal{T}) = \theta$, then $\forall X \in \mathcal{P}, \quad \Gamma \vdash X : \sigma \;\Rightarrow \Gamma \vdash X\theta : \sigma$.

Subject Reduction:

If $\Gamma \vdash \mathcal{T}_1 : \sigma$ and $\mathcal{T}_1 \mapsto\!\!\!\!\rightarrow_{\rho\sigma\delta} \mathcal{T}_2$, then $\Gamma \vdash \mathcal{T}_2 : \sigma$.

Uniqueness:

If $\Gamma \vdash \mathcal{T} : \varphi$ and $\Gamma \vdash \mathcal{T} : \psi$, then $\varphi =_\alpha \psi$.

Decidability:

$$\left.\begin{array}{ll} \textit{(typechecking)} & \Gamma \vdash \mathcal{T} : \varphi\, ? \\ \textit{(type inference)} & \Gamma \vdash \mathcal{T} : ? \end{array}\right\} \text{are decidable.}$$

The type system

Typing fixpoints

Encoding typed objects and TRS

Logical failure

Normalization failure

$f : (\alpha \rightarrowtriangle \alpha) \rightarrowtriangle \alpha$ and $\Gamma = X : \alpha \rightarrowtriangle \alpha$, $\qquad \omega \triangleq f \bullet X \rightarrow X \bullet (f \bullet X)$

$$
\begin{array}{rcl}
\omega \bullet (f \bullet \omega) & \equiv & (f \bullet X \rightarrow X \bullet (f \bullet X)) \bullet (f \bullet \omega) \\
 & \mapsto_\rho & [f \bullet X \ll f \bullet \omega].(X \bullet (f \bullet X)) \\
 & \mapsto_\sigma & \omega \bullet (f \bullet \omega) \\
 & \mapsto_\rho & \ldots
\end{array}
$$

Normalization failure (cont'd)

$f : (\alpha \rightarrowtriangle \alpha) \rightarrowtriangle \alpha$ and $\Gamma = X : \alpha \rightarrowtriangle \alpha$, $\qquad \omega \triangleq f \bullet X \to X \bullet (f \bullet X)$

$$\dfrac{\Gamma \vdash f : (\alpha \rightarrowtriangle \alpha) \rightarrowtriangle \alpha \quad \Gamma \vdash X : \alpha \rightarrowtriangle \alpha}{(b) \quad \Gamma \vdash f \bullet X : \alpha} \qquad \dfrac{\Gamma \vdash X : \alpha \rightarrowtriangle \alpha \quad \dfrac{(b)}{\Gamma \vdash f \bullet X : \alpha}}{\Gamma \vdash X \bullet (f \bullet X) : \alpha}$$
$$\overline{(a) \vdash \omega \equiv f \bullet X \to X \bullet (f \bullet X) : \alpha \rightarrowtriangle \alpha}$$

$$\dfrac{\dfrac{(a)}{\vdash \omega : \alpha \rightarrowtriangle \alpha} \qquad \dfrac{\vdash f : (\alpha \rightarrowtriangle \alpha) \rightarrowtriangle \alpha \quad \dfrac{(a)}{\vdash \omega : \alpha \rightarrowtriangle \alpha}}{\vdash f \bullet \omega : \alpha}}{\vdash \omega \bullet (f \bullet \omega) : \alpha}$$

Inductive types with “positive” occurrences

- In CaML, try to type:

```
type t = F of (t -> t);;

let omega x = match x with (F y) -> y (F y);;
```

Inductive types with “positive” occurrences

- In CaML, try to type:

```
type t = F of (t -> t);;

let omega x = match x with (F y) -> y (F y);;
```

- In CIC, the constructor $F : (x_1 : A_1) \dots (x_n : A_n).R$ is accepted only if R is *positive* in each A_i:

1. R is positive in T if R does not occur in T;
2. R is positive in $(R\vec{t})$ if R does not occur in $\vec{t}$;
3. R is positive in $(x : A)C$ if R does not occur in A and R is positive in C.

The type system

Typing fixpoints

Encoding typed objects and TRS

Logical failure

Detecting matching failures: the symbol stk

1. The relation $P \not\sqsubseteq A$ detects (some) definitive matching failures:

$$
\begin{array}{llll}
f & \not\sqsubseteq & g & \\
f(\overline{A_n}) & \sqsubseteq & B & \text{if } \ (B \equiv f(\overline{B_n})\,) \ \wedge \ \exists i, A_i \not\sqsubseteq B_i \\
P & \not\sqsubseteq & A & \text{if } A \equiv ([Q \ll_\Delta A_1].A_2 \wedge Q \not\sqsubseteq A_1 \vee P \not\sqsubseteq A_2)
\end{array}
$$

Detecting matching failures: the symbol stk

1. The relation $P \not\sqsubseteq A$ detects (some) definitive matching failures:

$$
\begin{array}{lcll}
f & \not\sqsubseteq & g & \\
f(\overline{A_n}) & \not\sqsubseteq & B & \text{if } \ (B \equiv f(\overline{B_n})\,) \ \wedge \ \exists i, A_i \not\sqsubseteq B_i \\
P & \not\sqsubseteq & A & \text{if } A \equiv ([Q \ll_\Delta A_1].A_2 \wedge Q \not\sqsubseteq A_1 \vee P \not\sqsubseteq A_2)
\end{array}
$$

2. The relation $\longmapsto_{\mathsf{stk}}$ treats matching failures uniformly:

$$
\begin{array}{rcll}
[P \ll_\Delta A].B & \longmapsto_{\mathsf{stk}} & \mathsf{stk} & \quad \text{if } P \not\sqsubseteq A \\
\mathsf{stk}; A & \longmapsto_{\mathsf{stk}} & A & \\
A; \mathsf{stk} & \longmapsto_{\mathsf{stk}} & A & \\
\mathsf{stk}\bullet A & \longmapsto_{\mathsf{stk}} & \mathsf{stk} &
\end{array}
$$

Rho and objects

- Object = record with an explicit account of self, $i.e.$

$$[m_i = \varsigma(X_i)t_i]^{i \in I} \quad \triangleq \quad (m_i(X_i) \to t_i)^{i \in I}$$

Rho and objects

- Object = record with an explicit account of self, $i.e.$

$$[m_i = \varsigma(X_i)t_i]^{i \in I} \quad \triangleq \quad (m_i(X_i) \to t_i)^{i \in I}$$

- Self-application = the application of an object to the object itself, $i.e.$

$$t_1.t_2 \quad \triangleq \quad t_1 \bullet t_2(t_1)$$

Rho and objects

- Object = record with an explicit account of self, $i.e.$

$$[m_i = \varsigma(X_i)t_i]^{i\in I} \quad \triangleq \quad (m_i(X_i) \to t_i)^{i\in I}$$

- Self-application = the application of an object to the object itself, $i.e.$

$$t_1.t_2 \quad \triangleq \quad t_1 \bullet t_2(t_1)$$

- Ex: $t \triangleq a(S) \to b$. Then: $t.a \triangleq t \bullet a(t) \stackrel{\rho}{\mapsto} [a(S) \ll a(t)]b \stackrel{\sigma}{\mapsto} b$

Typed objects

- An object has type:

$$\dfrac{\dfrac{S : lab \to \phi \vdash meth : (lab \to \phi) \to lab}{\vdash meth(S) : lab}\ (Appl) \qquad \vdash \mathcal{T}_{meth} : \phi}{\vdash meth(S) \to \mathcal{T}_{meth} : lab \to \phi}\ (Abst)$$

Typed objects

- An object has type:

$$\dfrac{\dfrac{S : lab \to \phi \vdash meth : (lab \to \phi) \to lab}{\vdash meth(S) : lab}\,(Appl) \qquad \vdash \mathcal{T}_{meth} : \phi}{\vdash meth(S) \to \mathcal{T}_{meth} : lab \to \phi}\,(Abst)$$

- $obj.meth \triangleq obj \bullet meth(obj)$ can be typed as follows:

$$\dfrac{\vdash obj : lab \to \phi \qquad \dfrac{\vdash meth : (lab \to \phi) \to lab \quad \vdash obj : lab \to \phi}{\vdash meth(obj) : lab}}{\vdash obj \bullet meth(obj) : \phi}$$

Encoding rewriting in the ρ-calculus

1. The following operator selects the first applicable rule of a set:

$$\mathsf{first}(\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n) \quad \triangleq \quad X \to ((\mathsf{stk} \to A_n{}^\bullet X; I)^\bullet (\ldots{}^\bullet(\mathsf{stk} \to A_2{}^\bullet X; I)^\bullet(A_1 X)))$$

$$\mathsf{first}(\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n)^\bullet \mathsf{B} \quad \mapsto\!\!\!\!\to_{\rho\sigma\delta} \quad A_{j+1}{}^\bullet B \;\; \text{if } \forall i \leq j,\; A_i{}^\bullet B \mapsto\!\!\!\!\to_{\rho\sigma\delta} \mathsf{stk} \quad \text{and } A_{j+1}{}^\bullet B \not\mapsto\!\!\!\!\to_{\rho\sigma\delta} \mathsf{stk}$$

Encoding rewriting in the ρ-calculus

1. The following operator selects the first applicable rule of a set:

$$\mathsf{first}(\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n) \quad \triangleq \quad X \to ((\mathsf{stk} \to A_n{}^\bullet X; I)^\bullet (\ldots {}^\bullet(\mathsf{stk} \to A_2{}^\bullet X; I)^\bullet (A_1 X)))$$

$$\mathsf{first}(\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_n)^\bullet \mathsf{B} \quad \mapsto\!\!\!\!\rightarrow_{\rho\sigma\delta} \quad A_{j+1}{}^\bullet B \ \text{ if } \forall i \le j,\ A_i{}^\bullet B \mapsto\!\!\!\!\rightarrow_{\rho\sigma\delta} \mathsf{stk} \quad \text{and } A_{j+1}{}^\bullet B \not\mapsto\!\!\!\!\rightarrow_{\rho\sigma\delta} \mathsf{stk}$$

2. The Term Rewrite System $\mathcal{R} = \{t_i \mapsto s_i\}$ with signature $\{a_j\}$ is encoded by:

$$\mathcal{R} \ \rightsquigarrow \ (rec{}^\bullet S) \to \mathsf{first}\left(\begin{array}{l} t_1 \to S^\bullet(rec{}^\bullet S)^\bullet s_1, \\ \ldots\ , \\ a_1{}^\bullet \overline{X} \to S^\bullet(Rec{}^\bullet S)^\bullet(a_1{}^\bullet \overline{S(rec{}^\bullet S)^\bullet X}), \\ \ldots\ , \end{array}\right),$$

$$(Rec{}^\bullet S) \to \mathsf{first}\left(\begin{array}{l} t_1 \to S^\bullet(rec{}^\bullet S)^\bullet s_1, \\ \ldots\ , \\ I \end{array}\right)$$

Example

- Addition over Peano integers:

$$plus \triangleq \left[\begin{array}{l} S \to add(0, y) \to y; \\ S \to add(suc(x), y) \to suc\big((S \bullet S) \bullet add(x, y)\big) \end{array} \right]$$

Example

- Addition over Peano integers:

$$plus \triangleq \left[\begin{array}{l} S \to add(0, y) \to y; \\ S \to add(suc(x), y) \to suc\big((S \bullet S) \bullet add(x, y)\big) \end{array} \right]$$

$$\begin{array}{ll} & (plus \bullet plus) \bullet add(N, M) \\ \mapsto\!\!\!\twoheadrightarrow_{\rho\sigma\delta} & [0 \ll N].M;\ [0 \ll N-1].(M+1)\ \cdots\ \underline{[0 \ll 0].(M+N)};\ [suc \bullet x \ll 0]\ldots. \\ \longmapsto_{\mathsf{stk}} & M + N \end{array}$$

Example

- Addition over Peano integers:

$$plus \triangleq \left[\begin{array}{l} S \to add(0, y) \to y; \\ S \to add(suc(x), y) \to suc\big((S \bullet S) \bullet add(x, y)\big) \end{array} \right]$$

$$\begin{array}{ll} & (plus \bullet plus) \bullet add(N, M) \\ \mapsto\!\!\!\twoheadrightarrow_{\rho\sigma\delta} & [0 \ll N].M;\ [0 \ll N-1].(M+1)\ \cdots\ \underline{[0 \ll 0].(M+N)};\ [suc \bullet x \ll 0]. \ldots \\ \mapsto_{\mathsf{stk}} & M + N \end{array}$$

- Fill in the blanks with your favorite rewrite system...

$$func \triangleq \left[\begin{array}{l} S \to \qquad\qquad ; \\ S \to \qquad\qquad (S \bullet S) \bullet \end{array} \right]$$

Example

- Addition over Peano integers:

$$plus \triangleq \left[\begin{array}{l} S \to add(0,y) \to y; \\ S \to add(suc(x),y) \to suc\big((S \bullet S) \bullet add(x,y)\big) \end{array}\right]$$

$$\begin{array}{ll} (plus \bullet plus) \bullet add(N,M) & \\ \mapsto\!\!\!\twoheadrightarrow_{\rho\sigma\delta} & [0 \ll N].M;\ [0 \ll N-1].(M+1)\ \cdots\ \underline{[0 \ll 0].(M+N)};\ [suc \bullet x \ll 0].\ldots \\ \mapsto_{\mathsf{stk}} & M+N \end{array}$$

- Fill in the blanks with your favorite rewrite system...

$$func \triangleq \left[\begin{array}{l} S \to\ len([\,]) \to 0\ \ ; \\ S \to\ len(Cons(x,l)) \to suc\big((S \bullet S) \bullet len(l)\big) \end{array}\right]$$

Example

- Addition over Peano integers:

$$plus \triangleq \left[\begin{array}{l} S \to add(0,y) \to y; \\ S \to add(suc(x),y) \to suc\big((S{\bullet}S){\bullet}add(x,y)\big) \end{array}\right]$$

$$\begin{array}{ll} & (plus \bullet plus) \bullet add(N, M) \\ \mapsto\!\!\!\twoheadrightarrow_{\rho\sigma\delta} & [0 \ll N].M;\ [0 \ll N-1].(M{+}1)\ \cdots\ \underline{[0 \ll 0].(M{+}N)};\ [suc{\bullet}x \ll 0]\ldots. \\ \mapsto_{\mathsf{stk}} & M + N \end{array}$$

- Fill in the blanks with your favorite rewrite system... provided it is convergent and ground reducible if you want completeness.

$$func \triangleq \left[\begin{array}{l} S \to\ len([\,]) \to 0\ \ ; \\ S \to\ len(Cons(x,l)) \to suc\big((S{\bullet}S) \bullet len(l)\big) \end{array}\right]$$

The type system

Typing fixpoints

Encoding typed objects and TRS

Logical failure

Logical inconsistency

- As is, the Curry-Howard isomorphism is not valid:

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 \longrightarrow_{\Delta} \mathcal{T}_2 : \sigma \rightarrowtriangle \tau} \ (Abs) \qquad \frac{\Gamma, \Delta \vdash \sigma \quad \Gamma, \Delta \vdash \tau}{\Gamma \vdash \sigma \rightarrowtriangle \tau} \ (\rightarrow I)$$

Logical inconsistency

- As is, the Curry-Howard isomorphism is not valid:

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 \rightarrow_{\Delta} \mathcal{T}_2 : \sigma \rightarrow \tau} \ (Abs) \qquad \frac{\Gamma, \Delta \vdash \sigma \quad \Gamma, \Delta \vdash \tau}{\Gamma \vdash \sigma \rightarrow \tau} \ (\rightarrow I)$$

- Thus, for instance, $\dfrac{\vdash (h^{\bot \rightarrow \alpha \rightarrow \alpha}(X^{\bot}) \rightarrow X^{\bot}) : \alpha \rightarrow \alpha \rightarrow \bot}{\vdash (h^{\bot \rightarrow \alpha \rightarrow \alpha}(X^{\bot}) \rightarrow X^{\bot}) \bullet (Y^{\alpha} \rightarrow Y^{\alpha}) : \bot}$

Logical inconsistency

- As is, the Curry-Howard isomorphism is not valid:

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 \rightarrow_{\Delta} \mathcal{T}_2 : \sigma \twoheadrightarrow \tau} \ (Abs) \qquad \frac{\Gamma, \Delta \vdash \sigma \quad \Gamma, \Delta \vdash \tau}{\Gamma \vdash \sigma \twoheadrightarrow \tau} \ (\rightarrow I)$$

- Thus, for instance, $\dfrac{\vdash (h^{\perp \twoheadrightarrow \alpha \twoheadrightarrow \alpha}(X^{\perp}) \rightarrow X^{\perp}) : \alpha \twoheadrightarrow \alpha \twoheadrightarrow \perp}{\vdash (h^{\perp \twoheadrightarrow \alpha \twoheadrightarrow \alpha}(X^{\perp}) \rightarrow X^{\perp}) \bullet (Y^{\alpha} \rightarrow Y^{\alpha}) : \perp}$

- How to fix it ?

$$\frac{\Gamma, X_i : \varphi_i \vdash B : \psi}{\Gamma \vdash A \rightarrow B : (\bigwedge \varphi_i) \twoheadrightarrow \psi} \ (Abs) \quad , \quad FV(A) = \{X_i^{\varphi_i}\}$$

Dependent type discipline

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_1 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash \mathcal{T}_1 : \Delta \to \mathcal{T}_2 : \sigma \twoheadrightarrow \tau} \; (Abs)$$

$$\frac{\Gamma \vdash \mathcal{T}_1 : \sigma \twoheadrightarrow \tau \quad \Gamma \vdash \mathcal{T}_2 : \sigma}{\Gamma \vdash \mathcal{T}_1 \; \mathcal{T}_2 : \tau} \; (Appl)$$

$$\frac{\Gamma, \Delta \vdash \mathcal{T}_2 : \tau}{\Gamma \vdash (\mathcal{T}_1 : \Delta) \to \mathcal{T}_2 : (\mathcal{T}_1 : \Delta) \twoheadrightarrow \tau} \; (Abs)$$

$$\frac{\Gamma \vdash \mathcal{T}_1 : (\mathcal{T}_{11} : \Delta) \twoheadrightarrow \tau \quad \Gamma \vdash \mathcal{T}_2 : \sigma \quad \Gamma, \Delta \vdash \mathcal{T}_{11} : \sigma}{\Gamma \vdash \mathcal{T}_1 \; \mathcal{T}_2 : [\mathcal{T}_{11} \ll_\Delta \mathcal{T}_2].\tau} \; (Appl)$$