

$$Moca = NbE_{AC}$$

Mathieu Boespflug

9 Juin 2008

EUROPA

EUROPA

- Vérificateur de preuves en $\lambda\Pi$ -modulo.
- Présence de type dépendents.
- En déduction modulo, les axiomes d'une théorie \mathcal{T} sont exprimés par des règles de réécriture.

Encodage du Calcul des Constructions dans $\lambda\Pi$ – *modulo*

$$\begin{aligned}
\epsilon_{Kind} (\dot{T}_{Type}) &\longrightarrow U_{Type} \\
\epsilon_{Kind} (\dot{\Pi}_{\langle Type, Type, Type \rangle} X Y) &\longrightarrow \Pi x : (\epsilon_{Type} X) (\epsilon_{Type} (Y x)) \\
\epsilon_{Kind} (\dot{\Pi}_{\langle Type, Kind, Kind \rangle} X Y) &\longrightarrow \Pi x : (\epsilon_{Type} X) (\epsilon_{Kind} (Y x)) \\
\epsilon_{Kind} (\dot{\Pi}_{\langle Kind, Type, Kind \rangle} X Y) &\longrightarrow \Pi x : (\epsilon_{Kind} X) (\epsilon_{Type} (Y x)) \\
\epsilon_{Kind} (\dot{\Pi}_{\langle Kind, Kind, Kind \rangle} X Y) &\longrightarrow \Pi x : (\epsilon_{Kind} X) (\epsilon_{Kind} (Y x))
\end{aligned}$$

Règle conversion dans CIC

$$\frac{\Gamma \vdash a : \tau \quad \tau \leftrightarrow_{\beta} \tau'}{\Gamma \vdash a : \tau'}$$

Règles de conversion dans $\lambda\Pi$ -modulo

$$\frac{\Gamma \vdash A : \textit{Type} \quad \Gamma \vdash B : \textit{Type} \quad \Gamma \vdash t : A}{\Gamma \vdash t : B} A \equiv_{\beta\mathcal{R}} B$$

$$\frac{\Gamma \vdash A : \textit{Kind} \quad \Gamma \vdash B : \textit{Kind} \quad \Gamma \vdash t : A}{\Gamma \vdash t : B} A \equiv_{\beta\mathcal{R}} B$$

Règles de conversion dans $\lambda\Pi$ -modulo

$$\frac{\Gamma \vdash A : \text{Type} \quad \Gamma \vdash B : \text{Type} \quad \Gamma \vdash t : A}{\Gamma \vdash t : B} \boxed{A \equiv_{\beta\mathcal{R}} B}$$

$$\frac{\Gamma \vdash A : \text{Kind} \quad \Gamma \vdash B : \text{Kind} \quad \Gamma \vdash t : A}{\Gamma \vdash t : B} \boxed{A \equiv_{\beta\mathcal{R}} B}$$

- 1 Introduction
- 2 Normalisation par évaluation
- 3 Conclusion

- 1 Introduction
- 2 Normalisation par évaluation
- 3 Conclusion

- 1 Introduction
- 2 Normalisation par évaluation
- 3 Conclusion

- termes : $t ::= x \mid c \mid \lambda x. t \mid t_1 t_2$
- patterns : $p ::= _ \mid x \mid c(p_1, \dots, p_n)$
- valeurs : $v ::= c(v_1, \dots, v_n)$
- règles : $r ::= k p_1 \cdots p_2 \longrightarrow t$

Exemple :

$$\begin{aligned} &+ 0 x \longrightarrow x \\ &+ x 0 \longrightarrow x \\ &+ (Sx) y \longrightarrow S (+ x y) \end{aligned}$$

L'histoire d'une convergence

- Calcul symbolique de Grégoire et Leroy (2002),
- Compilation vers la ZAM - Olivier Hermant (2005),
- Normalisation par evaluation - Berger et Schwichtenberg (1991), Danvy (1996), Aehlig et Haftmann et Nipkow (2008),
...
- Moca - Blanqui et Hardin et Weis (2007).

Objectifs

- simplifier la compilation des règles de réécriture
- porter la vitesse d'exécution de ces machines de β -réduction à notre problème.
 - Reprise du raisonnement de Grégoire et Leroy (2002) pour la réécriture.
 - Réduire la réécriture à de la β -réduction + X.

→ *C'est ce que fait Moca.*

→ **Problème** : *pas de normalisation forte*

Objectifs

- simplifier la compilation des règles de réécriture
- porter la vitesse d'exécution de ces machines de β -réduction à notre problème.
 - Reprise du raisonnement de Grégoire et Leroy (2002) pour la réécriture.
 - Réduire la réécriture à de la β -réduction + X.

→ *C'est ce que fait Moca.*

→ **Problème** : *pas de normalisation forte*

Objectifs

- simplifier la compilation des règles de réécriture
- porter la vitesse d'exécution de ces machines de β -réduction à notre problème.
 - Reprise du raisonnement de Grégoire et Leroy (2002) pour la réécriture.
 - Réduire la réécriture à de la β -réduction + X.

→ *C'est ce que fait Moca.*

→ *Problème : pas de normalisation forte*

Objectifs

- simplifier la compilation des règles de réécriture
- porter la vitesse d'exécution de ces machines de β -réduction à notre problème.
 - Reprise du raisonnement de Grégoire et Leroy (2002) pour la réécriture.
 - Réduire la réécriture à de la β -réduction + X.

→ *C'est ce que fait Moca.*

→ **Problème** : *pas de normalisation forte*

Normalisation par évaluation

Vue d'ensemble

Idée : Exprimer la normalisation forte d'un terme t du λ -calcul en terme d'évaluation d'un terme t' .

Première étape : Traduction vers le langage de l'évaluateur.
→ *modèle du λ -calcul.*

Deuxième étape : *Réification* de la dénotation dans le modèle pour retrouver un terme β -équivalent en forme η -longue.

Conditions : $t \leftrightarrow_{\beta} t' \Rightarrow \llbracket t \rrbracket = \llbracket t' \rrbracket$.

Normalisation d'un terme clos :

$$nbe_{\tau} t = \downarrow^{\tau} (\llbracket t \rrbracket \emptyset)$$

Problème : fonctions de réification/reflexion dirigées par les types.

Réponse : Reformulons le modèle — chaque élément du modèle est étiqueté pour guider la réification/reflexion.

Problème : fonctions de réification/reflexion dirigées par les types.

Réponse : Reformulons le modèle — chaque élément du modèle est étiqueté pour guider la réification/reflexion.

- Avons définis NbE pour λ -calcul non-typé avec règles de réécriture.
- Tombés sur Moca ! (presque)
- Pour que Moca fasse de la NbE : ajouter Var et Closure aux types Moca.
- Extension compatible avec l'associatif-commutatif (AC).
Remarque : l'AC est particulièrement important en présence de termes neutres.

- Avons définis NbE pour λ -calcul non-typé avec règles de réécriture.
- Tombés sur Moca ! (presque)
- Pour que Moca fasse de la NbE : ajouter Var et Closure aux types Moca.
- Extension compatible avec l'associatif-commutatif (AC).
Remarque : l'AC est particulièrement important en présence de termes neutres.

- Avons définis NbE pour λ -calcul non-typé avec règles de réécriture.
- Tombés sur Moca ! (presque)
- Pour que Moca fasse de la NbE : ajouter Var et Closure aux types Moca.
- Extension compatible avec l'associatif-commutatif (AC).
Remarque : l'AC est particulièrement important en présence de termes neutres.

NbE est une manière naturelle d'étendre
Moca à la normalisation forte.

Futurs travaux

- Revenir à de la NbE typée quand c'est possible pour éviter l'étiquetage.
- Étendre la NbE aux types dépendents. (cf travaux de Abel, Coquand, Dybjer (2007))
- Explorer l'application de ces techniques aux test de conversion.

→ *exploiter le partage maximal ?*