Récriture d'Ordre Supérieur

Jean-Pierre Jouannaud
École Polytechnique
91400 Palaiseau, France

email: jouannaud@lix.polytechnique.fr
http: //w^3.lix.polytechnique.fr/Labo/Jean-Pierre.Jouannaud

Project LogiCal, Pôle Commun de Recherche en Informatique du Plateau de Saclay, CNRS, École Polytechnique, INRIA, Université Paris-Sud.

January 3, 2005

Outline

1. Confluence de la récriture d'ordre supérieur
   - Confluence de la récriture normale abstraite
   - Confluence de la récriture d'ordre supérieur
   - Filtrage et unification de patterns

Confluence de la récriture normale

La confluence d'une relation qui termine se ramène à un calcul de paires critiques, l'algorithme d'unification utilisé dépendant de l'algorithme de filtrage utilisé.

La confluence de la récriture d'ordre supérieur simple repose sur le calcul de paires critiques obtenues par unification simple.

Ce n'est pas le cas de la récriture normale, qui utilise le filtrage d'ordre supérieur. Nous nous proposons de définir la récriture normale abstraite, avant d'en déduire les propriétés de confluence, puis d'appliquer les résultats obtenus à la récriture d'ordre supérieur.

Récriture normale [Jouannaud-Rubio-VanRaamsdonk]

- Ensemble R de règles, et $S = S_1 \cup S_2$ de règles CR telles que $t{\downarrow_S} = (t{\downarrow_{S_1}}){\downarrow_{S_2}}$
- Récriture normale $s \xrightarrow[R_{S\downarrow}]{p} t$ si $s = s{\downarrow_S},\ s|_p =_S l\sigma$ and $t = s[r\sigma]_p{\downarrow_S}$
- Joignabilité : $u{\downarrow_S} \xrightarrow[R_{S\downarrow}]{*} w \xleftarrow[R_{S\downarrow}]{*} v{\downarrow_S}$
- CR : $s \xleftrightarrow[R \cup S]{*} t$ ssi (s, t) est joignable
- Terminaison de $\longrightarrow_{R_{S\downarrow}} \cup \longrightarrow_S$
- Paires critiques : S-paires critiques de R, paires critiques simples de S_1 avec R
- Extensions : S-extensions gauches S-normales, S_2-extensions droites S-normales

Church-Rosser

- Hypothèses :
  (a) S est Church-Rosser
  (b) $\longrightarrow_{R_S} \cup \longrightarrow_S$ termine
  (c) Règles de R sont S-normales
-

Theorem

R est CR ssi il est localement confluent et cohérent.

Church-Rosser

- Hypothèses :
  (a) S est Church-Rosser
  (b) $\longrightarrow_{R_S} \cup \longrightarrow_S$ termine
  (c) Règles de R sont S-normales
-

Theorem

R est CR ssi il est localement confluent et cohérent.

- Hypothèses :
  (a) S est Church-Rosser
  (b) $\longrightarrow_{R_S} \cup \longrightarrow_S$ termine
  (c) Règles de R sont S-normales
-

Theorem

R est CR ssi il est localement confluent et cohérent.

Confluence et coherence locales

- **Definition**

  La récriture normale est *localement confluente* si $\forall t, u, v$ avec $t = t{\downarrow}_S$, $t \longrightarrow_{R_S} u$ et $t \longrightarrow_{R_S} v$, alors (u, v) est joignable.

- **Definition**

  La récriture normale est *localement coherente* si $\forall s, t$ tel que $s \longrightarrow_{R_S} t$, alors (s, t) est joignable.

Confluence et coherence locales

- **Definition**

  La récriture normale est *localement confluente* si $\forall t, u, v$ avec $t = t{\downarrow_S}$, $t \longrightarrow_{R_S} u$ et $t \longrightarrow_{R_S} v$, alors (u, v) est joignable.

- **Definition**

  La récriture normale est *localement coherente* si $\forall s, t$ tel que $s \longrightarrow_{R_S} t$, alors (s, t) est joignable.

Confluence et coherence locales

- **Definition**

  La récriture normale est *localement confluente* si $\forall t, u, v$ avec $t = t{\downarrow}_S$, $t \longrightarrow_{R_S} u$ et $t \longrightarrow_{R_S} v$, alors (u, v) est joignable.

- **Definition**

  La récriture normale est *localement coherente* si $\forall s, t$ tel que $s \longrightarrow_{R_S} t$, alors (s, t) est joignable.

Confluence et coherence locales

- **Definition**

  La récriture normale est *localement confluente* si $\forall t, u, v$ avec $t = t{\downarrow}_S$, $t \longrightarrow_{R_S} u$ et $t \longrightarrow_{R_S} v$, alors (u, v) est joignable.

- **Definition**

  La récriture normale est *localement coherente* si $\forall s, t$ tel que $s \longrightarrow_{R_S} t$, alors (s, t) est joignable.

Coherence locale

- **Definition**

  Étant donnés $l \to r \in S$, $g \to d \in R$ et $p \in \mathcal{FP}os(l) \setminus \{\wedge\}$ tels que $l|_p = g$ est unifiable, alors $l[g]{\downarrow_S} \to l[d]{\downarrow_S}$ est une extension gauche normale.

- **Lemma**

  Soit $s \longrightarrow_{R_S} t$. Supposons que R est clos pour les extensions gauches normales et que CR est satisfaites pour les preuves plus petites que $\{s, t\}$. Alors (s, t) est joignable.

Coherence locale

- **Definition**

  Étant donnés $l \to r \in S$, $g \to d \in R$ et $p \in \mathcal{FP}os(l) \setminus \{\wedge\}$ tels que $l|_p = g$ est unifiable, alors $l[g]\downarrow_S \to l[d]\downarrow_S$ est une extension gauche normale.

- **Lemma**

  Soit $s \longrightarrow_{R_S} t$. Supposons que R est clos pour les extensions gauches normales et que CR est satisfaites pour les preuves plus petites que $\{s, t\}$. Alors (s, t) est joignable.

Coherence locale

- **Definition**

  Étant donnés $l \rightarrow r \in S$, $g \rightarrow d \in R$ et $p \in \mathcal{FPos}(l) \setminus \{\wedge\}$ tels que $l|_p = g$ est unifiable, alors $l[g]\downarrow_S \rightarrow l[d]\downarrow_S$ est une extension gauche normale.

- **Lemma**

  Soit $s \longrightarrow_{R_S} t$. Supposons que R est clos pour les extensions gauches normales et que CR est satisfaites pour les preuves plus petites que $\{s, t\}$. Alors (s, t) est joignable.

Coherence locale

- **Definition**

  Étant donnés $l \rightarrow r \in S$, $g \rightarrow d \in R$ et $p \in \mathcal{FP}os(l) \setminus \{\wedge\}$ tels que $l|_p = g$ est unifiable, alors $l[g]\downarrow_S \rightarrow l[d]\downarrow_S$ est une extension gauche normale.

- **Lemma**

  Soit $s \longrightarrow_{R_S} t$. Supposons que R est clos pour les extensions gauches normales et que CR est satisfaites pour les preuves plus petites que $\{s, t\}$. Alors (s, t) est joignable.

- **Lemma**

  La récriture normale est localement confluente ssi $\forall s, t, u, v$ tels que $s \longrightarrow^{p}_{R} u$, $s \longrightarrow^{(\geq p)^*}_{S} t$ et $t \longrightarrow^{q}_{R_S} v$ avec $q \# p$ ou $q \geq p$, alors (u, v) est joignable.

- **Definition**

  $S_1 \cup S_2$ est une dichotomie de S si pour tout terme t il existe une dérivation $t \longrightarrow^{*}_{S_1} u \longrightarrow^{*}_{S_2} t{\downarrow}_S$.

Confluence locale

- **Lemma**

  La récriture normale est localement confluente ssi $\forall s, t, u, v$ tels que $s \xrightarrow[R]{p} u$, $s \xrightarrow[S]{(\geq p)^*} t$ et $t \xrightarrow[R_S]{q} v$ avec $q \# p$ ou $q \geq p$, alors (u, v) est joignable.

- **Definition**

  $S_1 \cup S_2$ est une dichotomie de S si pour tout terme t il existe une dérivation $t \xrightarrow[S_1]{*} u \xrightarrow[S_2]{*} t{\downarrow}_S$.

Confluence locale

- **Lemma**

  La récriture normale est localement confluente ssi $\forall s, t, u, v$ *tels que* $s \xrightarrow[R]{p} u$, $s \xrightarrow[S]{(\geq p)^*} t$ *et* $t \xrightarrow[R_S]{q} v$ *avec* $q \# p$ *ou* $q \geq p$, *alors* (u, v) *est joignable.*

- **Definition**

  $S_1 \cup S_2$ est une dichotomie de S si pour tout terme t il existe une dérivation $t \xrightarrow[S_1]{*} u \xrightarrow[S_2]{*} t{\downarrow_S}$.

Confluence locale

- **Lemma**

  La récriture normale est localement confluente ssi $\forall s, t, u, v$ *tels que* $s \longrightarrow^{p}_{R} u$, $s \longrightarrow^{(\geq p)^*}_{S} t$ *et* $t \longrightarrow^{q}_{R_S} v$ *avec* $q \# p$ *ou* $q \geq p$, *alors* (u, v) *est joignable.*

- **Definition**

  $S_1 \cup S_2$ est une dichotomie de S si pour tout terme t il existe une dérivation $t \longrightarrow^{*}_{S_1} u \longrightarrow^{*}_{S_2} t{\downarrow_S}$.

Confluence locale, suite

- **Definition**

  Soit $g \to d \in S_2$, $l \to r \in R$, $p \in \mathcal{F}\mathcal{D}om(d) \setminus \{\wedge\}$ tels que $l = d|_p$ est unifiable. Alors $d[l]_p\downarrow \to d[r]_p\downarrow$ est une extension droite normale.

- **Definition**

  Soit $g \to d \in S_1$, $l \to r \in R$, $p \in \mathcal{F}\mathcal{P}os(l)$ t.q. $\sigma = mgu(l|_p = g)$. Alors $(r\sigma, l\sigma[d\sigma]_p)$ est une paire superficielle de $g \to d$ sur $l \to r$ à p. Cette paire est fortement joignable si $l\sigma[d\sigma]_p \xrightarrow[S]{*} \xrightarrow[R]{\Lambda} v$ et $(r\sigma, v)$ est joignable.

Confluence locale, suite

- **Definition**

  Soit $g \to d \in S_2$, $l \to r \in R$, $p \in \mathcal{FDom}(d) \setminus \{\wedge\}$ tels que $l = d|_p$ est unifiable. Alors $d[l]_p\!\downarrow \longrightarrow d[r]_p\!\downarrow$ est une extension droite normale.

- **Definition**

  Soit $g \to d \in S_1$, $l \to r \in R$, $p \in \mathcal{FPos}(l)$ t.q. $\sigma = mgu(l|_p = g)$. Alors $(r\sigma, l\sigma[d\sigma]_p)$ est une paire superficielle de $g \to d$ sur $l \to r$ à p. Cette paire est fortement joignable si $l\sigma[d\sigma]_p \xrightarrow[S]{*} \xrightarrow[R]{\Lambda} v$ et $(r\sigma, v)$ est joignable.

Confluence locale, suite

- **Definition**

  Soit $g \to d \in S_2$, $l \to r \in R$, $p \in \mathcal{FDom}(d) \setminus \{\wedge\}$ tels que $l = d|_p$ est unifiable. Alors $d[l]_p{\downarrow} \to d[r]_p{\downarrow}$ est une extension droite normale.

- **Definition**

  Soit $g \to d \in S_1$, $l \to r \in R$, $p \in \mathcal{FPos}(l)$ t.q. $\sigma = mgu(l|_p = g)$. Alors $(r\sigma, l\sigma[d\sigma]_p)$ est une paire superficielle de $g \to d$ sur $l \to r$ à p. Cette paire est fortement joignable si $l\sigma[d\sigma]_p \xrightarrow[S]{*} \xrightarrow[R]{\Lambda} v$ et $(r\sigma, v)$ est joignable.

Confluence locale, suite

- **Definition**

  Soit $g \to d \in S_2$, $l \to r \in R$, $p \in \mathcal{FD}om(d) \setminus \{\wedge\}$ tels que $l = d|_p$ est unifiable. Alors $d[l]_p{\downarrow} \to d[r]_p{\downarrow}$ est une extension droite normale.

- **Definition**

  Soit $g \to d \in S_1$, $l \to r \in R$, $p \in \mathcal{FP}os(l)$ t.q. $\sigma = mgu(l|_p = g)$. Alors $(r\sigma, l\sigma[d\sigma]_p)$ est une paire superficielle de $g \to d$ sur $l \to r$ à p.
  Cette paire est fortement joignable si $l\sigma[d\sigma]_p \xrightarrow[S]{*} \xrightarrow[R]{\Lambda} v$ et $(r\sigma, v)$ est joignable.

Confluence locale, conclusion

Lemma

Soient $t = t{\downarrow_S}$, $t \longrightarrow_{l \to r} u$ *et* $t \longrightarrow_{R_S} v$ *avec* $g \to d$ *t.q.* $t = t{\downarrow_S}$. *Alors la paire* (u, v) *est joignable si*

(i) Les paires S*-critiques de* R *qui sont* S_1*-irréductibles sont joignables.*

(ii) R *est clos pour les extensions normales.*

(iii) Les paires superficielles avec S_1 *sont joignables fortement.*

(iv) R *est clos pour les* S_2 *extensions droites normales.*

(v) Les preuves plus petites que $\{s\}$ *sont CR.*

Theorem

CR est vraie sous les conditions suivantes:
(a) $S_1 \cup S_2$ est une dichotomie de S CR.
(b) $\longrightarrow_{R_S} \cup \longrightarrow_S$ termine
(c) Les règles de R sont S-normales
(i) Les paires S-critiques de R qui sont S_1-irréductibles sont joignables.
(ii) R est clos pour les extensions normales.
(iii) Les paires superficielles avec S_1 sont joignables fortement.
(iv) R est clos pour les S_2-extensions droites normales.

Confluence de la récriture d'ordre supérieur

- Récriture d'ordre supérieur de Nipkow
- La variante Jouannaud-Rubio-VanRaamsdonk

Confluence de la récriture d'ordre supérieur

- Récriture d'ordre supérieur de Nipkow
- La variante Jouannaud-Rubio-VanRaamsdonk

Nipkow's HOR

- $R = \{\Gamma_i \vdash l_i \to r_i\}_i$ such that
  (i) $\Gamma_i \vdash l_i, r_i : \alpha_i,\ \alpha_i$ of base type
  (ii) l_i, r_i are in η-long β-normal form
  (iii) l_i is a pattern : each subterm headed by a free variable $X : \beta_1 \to \ldots \to \beta_n \to \beta$ is of the form $X(x_1, \ldots, x_n)$, $x_1 : \beta_1, \ldots, x_n : \beta_n$ being distinct bound variables.

Theorem

If $R \cup \longrightarrow_\beta \cup \longrightarrow_{\eta^1}$ is terminating, then Nipkow's HOR is Church-Rosser.

Nipkow's HOR

- $R = \{\Gamma_i \vdash l_i \rightarrow r_i\}_i$ such that
  (i) $\Gamma_i \vdash l_i, r_i : \alpha_i,\ \alpha_i$ of base type
  (ii) l_i, r_i are in η-long β-normal form
  (iii) l_i is a pattern : each subterm headed by a free variable $X : \beta_1 \rightarrow \ldots \rightarrow \beta_n \rightarrow \beta$ is of the form $X(x_1, \ldots, x_n)$, $x_1 : \beta_1, \ldots, x_n : \beta_n$ being distinct bound variables.

Theorem

If $R \cup \longrightarrow_\beta \cup \longrightarrow_{\eta^1}$ is terminating, then Nipkow's HOR is Church-Rosser.

- Proof : let $R_2 = \emptyset$.
- Since $\longrightarrow_{\eta^{-1}}$ has a variable as its lefthand side, there are no extensions associated with the η-rule and no shallow critical pairs.
- Since the rules in R are of base type, their lefthand side cannot unify with an abstraction : no β-extensions are needed.
- Since rules are in long-$\beta\overline{\eta}$-normal form, no subterm of a rule can unify with the lefthand side of β : no shallow pairs are needed.

- Proof : let $R_2 = \emptyset$.
- Since $\longrightarrow_{\eta^{-1}}$ has a variable as its lefthand side, there are no extensions associated with the η-rule and no shallow critical pairs.
- Since the rules in R are of base type, their lefthand side cannot unify with an abstraction : no β-extensions are needed.
- Since rules are in long-$\beta\overline{\eta}$-normal form, no subterm of a rule can unify with the lefthand side of β : no shallow pairs are needed.

- Proof : let $R_2 = \emptyset$.
- Since $\longrightarrow_{\eta^{-1}}$ has a variable as its lefthand side, there are no extensions associated with the η-rule and no shallow critical pairs.
- Since the rules in R are of base type, their lefthand side cannot unify with an abstraction : no β-extensions are needed.
- Since rules are in long-$\beta\overline{\eta}$-normal form, no subterm of a rule can unify with the lefthand side of β : no shallow pairs are needed.

- Proof : let $R_2 = \emptyset$.
- Since $\longrightarrow_{\eta^{-1}}$ has a variable as its lefthand side, there are no extensions associated with the η-rule and no shallow critical pairs.
- Since the rules in R are of base type, their lefthand side cannot unify with an abstraction : no β-extensions are needed.
- Since rules are in long-$\beta\overline{\eta}$-normal form, no subterm of a rule can unify with the lefthand side of β : no shallow pairs are needed.

Examples

- Modelling untyped lambda calculus:

$$\begin{array}{rcl} \mathrm{app} & : & T \to T \to T \\ \mathrm{abs} & : & (T \to T) \to T \end{array}$$

with T a sort representing the set of terms.

$$\begin{array}{rcl} \mathrm{app}(\mathrm{abs}(\lambda x.Z(x)), Z') & \to & Z(Z') \\ \mathrm{abs}(\lambda x.\mathrm{app}(Y(x))) & \to & Y \end{array}$$

- Modelling differentiation:

$$\begin{array}{rcl} \mathrm{diff} & : & (R \to R) \to (R \to R) \\ \sin, \cos & : & R \to R \end{array}$$

A rewrite rule for differentiation:

$$\mathrm{diff}(\lambda x.\sin(F(x)), y) \to$$
$$\cos(F(y)) \times \mathrm{diff}(\lambda x.F(x), y)$$

Examples

- Modelling untyped lambda calculus:

$$\begin{aligned} \mathrm{app} &: T \to T \to T \\ \mathrm{abs} &: (T \to T) \to T \end{aligned}$$

  with T a sort representing the set of terms.

$$\begin{aligned} \mathrm{app}(\mathrm{abs}(\lambda x.Z(x)), Z') &\to Z(Z') \\ \mathrm{abs}(\lambda x.\mathrm{app}(Y(x))) &\to Y \end{aligned}$$

- Modelling differentiation:

$$\begin{aligned} \mathrm{diff} &: (R \to R) \to (R \to R) \\ \mathrm{sin}, \mathrm{cos} &: R \to R \end{aligned}$$

  A rewrite rule for differentiation:

$$\mathrm{diff}(\lambda x.\sin(F(x)), y) \to \cos(F(y)) \times \mathrm{diff}(\lambda x.F(x), y)$$

Relaxed assumptions

- Consider the one rule rewriting system $R = \{\lambda x.a \to \lambda x.b\}$, where a and b are two constants of a given base type.
  $a \overset{\Lambda}{\underset{\beta}{\longleftrightarrow}} (\lambda x.a\ u) \overset{1}{\underset{R}{\longleftrightarrow}} (\lambda x.b\ u) \overset{\Lambda}{\underset{\beta}{\longleftrightarrow}} b.$
  But a and b are in normal form for Nipkow's rewriting, and therefore $a \underset{R \cup \beta\eta}{\not\longleftrightarrow} b$.
- It would be easy to change the rule into $a \to b$, therefore avoiding the problem, but this cannot be done in general.
  Consider $\lambda x.f(x, Z(x)) \to \lambda x.g(x, Z(x))$.
  Removing the abstraction yields $f(X, Z(X)) \to g(X, Z(X))$, which does not conform the format.

Relaxed assumptions

- Consider the one rule rewriting system $R = \{\lambda x.a \to \lambda x.b\}$, where a and b are two constants of a given base type.
  $a \overset{\Lambda}{\underset{\beta}{\longleftrightarrow}} (\lambda x.a\ u) \overset{1}{\underset{R}{\longleftrightarrow}} (\lambda x.b\ u) \overset{\Lambda}{\underset{\beta}{\longleftrightarrow}} b.$
  But a and b are in normal form for Nipkow's rewriting, and therefore $a \underset{R\cup\beta\eta}{\not\longleftrightarrow} b$.
- It would be easy to change the rule into $a \to b$, therefore avoiding the problem, but this cannot be done in general.
  Consider $\lambda x.f(x, Z(x)) \to \lambda x.g(x, Z(x))$.
  Removing the abstraction yields $f(X, Z(X)) \to g(X, Z(X))$, which does not conform the format.

Relaxed assumptions

- This problem can be solved by adding the normal β-extensions when appropriate.

Theorem

Assume R is a set of higher-order rules s.t.
(i) lefthand sides are patterns;
(ii) $R \cup \beta \cup \eta^{-1}$ is terminating;
(iii) irreducible HO critical pairs of R are joinable;
(iv) for each rule $\lambda x.l \rightarrow r \in R$, R contains its β-extension $l \rightarrow @(r, x) \downarrow_\beta^\eta$.
Then R is Church-Rosser.

- There are finitely many β-extensions.
- The union of $\longrightarrow_{R_{\beta\eta}}$ and $\longrightarrow_{\beta\cup\eta^{-1}}$ must be terminating.

Relaxed assumptions

- This problem can be solved by adding the normal β-extensions when appropriate.

Theorem

Assume R is a set of higher-order rules s.t.
(i) lefthand sides are patterns;
(ii) $R \cup \beta \cup \eta^{-1}$ is terminating;
(iii) irreducible HO critical pairs of R are joinable;
(iv) for each rule $\lambda x.l \to r \in R$, R contains its β-extension $l \to @(r, x) \updownarrow_{\beta}^{\eta}$.
Then R is Church-Rosser.

- There are finitely many β-extensions.
- The union of $\longrightarrow_{R_{\beta\eta}}$ and $\longrightarrow_{\beta\cup\eta^{-1}}$ must be terminating.

Relaxed assumptions

- This problem can be solved by adding the normal β-extensions when appropriate.

Theorem

Assume R is a set of higher-order rules s.t.
(i) lefthand sides are patterns;
(ii) $R \cup \beta \cup \eta^{-1}$ is terminating;
(iii) irreducible HO critical pairs of R are joinable;
(iv) for each rule $\lambda x.l \to r \in R$, R contains its β-extension $l \to @(r, x) \updownarrow^{\eta}_{\beta}$.
Then R is Church-Rosser.

- There are finitely many β-extensions.
- The union of $\longrightarrow_{R_{\beta\eta}}$ and $\longrightarrow_{\beta\cup\eta^{-1}}$ must be terminating.

Relaxed assumptions

- This problem can be solved by adding the normal β-extensions when appropriate.

Theorem

Assume R is a set of higher-order rules s.t.
(i) lefthand sides are patterns;
(ii) $R \cup \beta \cup \eta^{-1}$ is terminating;
(iii) irreducible HO critical pairs of R are joinable;
(iv) for each rule $\lambda x.l \to r \in R$, R contains its β-extension $l \to @(r, x) \updownarrow_{\beta}^{\eta}$.
Then R is Church-Rosser.

- There are finitely many β-extensions.
- The union of $\longrightarrow_{R_{\beta\eta}}$ and $\longrightarrow_{\beta\cup\eta^{-1}}$ must be terminating.

- Variables have arities. When unifying an equation in an environment Γ, a variable $X : \alpha_1 \times \ldots \alpha_n \rightarrow \alpha$ must be substituted by an abstraction of the form $\lambda x_1 : \alpha_1 \ldots x_n : \alpha_n.t$ such that $\Gamma \cup \{x_1 : \alpha_1 \ldots x_n : \alpha_n\} \vdash_{\mathcal{F}} t : \alpha$, where Γ is the current environment for unification.

Definition

A *rewrite rule* is an expression of the form $\Gamma \vdash l \rightarrow r : \sigma$ such that
(i) l and r are in $\beta\eta$-normal form,
(ii) $\Gamma \vdash l, r : \sigma$,
(iii) $l = f(l_1, \ldots, l_n)$ is a pattern,
(iv) $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$

- Variables have arities. When unifying an equation in an environment Γ, a variable $X : \alpha_1 \times \ldots \alpha_n \rightarrow \alpha$ must be substituted by an abstraction of the form $\lambda x_1 : \alpha_1 \ldots x_n : \alpha_n . t$ such that $\Gamma \cup \{x_1 : \alpha_1 \ldots x_n : \alpha_n\} \vdash_{\mathcal{F}} t : \alpha$, where Γ is the current environment for unification.

Definition

A *rewrite rule* is an expression of the form $\Gamma \vdash l \rightarrow r : \sigma$ such that
(i) l and r are in $\beta\eta$-normal form,
(ii) $\Gamma \vdash l, r : \sigma$,
(iii) $l = f(l_1, \ldots, l_n)$ is a pattern,
(iv) $\mathcal{V}ar(r) \subset \mathcal{V}ar(l)$

The variation

Theorem

Given a set R of higher-order rules such that
(i) $R \cup \beta\eta$ is terminating;
(ii) β-irreducible higher-order critical pairs of R are joinable;
(iii) For each rule $@(l, x) \to r \in R$ such that $x \notin \mathcal{V}ar(l)$, R contains its η-extension $l \to \lambda x.r$.
Then R is Church-Rosser.

The variation

- Proof: let $S_1 = \{\beta\}$ and $S_2 = \{\eta\}$.
- No shallow pairs are needed when lefthand sides are patterns.
- No β-extensions are needed since lefthand sides are now headed by a defined function symbol.
- No forward extension with η is needed, since the righthand side is a variable.
- For each rule of the form $@(l, x) \to r$ with $x \notin \mathcal{V}ar(l)$, we need the normal extension $l \to \lambda x.r$. A rule can have only finitely many extensions.

The variation

- Proof: let $S_1 = \{\beta\}$ and $S_2 = \{\eta\}$.
- No shallow pairs are needed when lefthand sides are patterns.
- No β-extensions are needed since lefthand sides are now headed by a defined function symbol.
- No forward extension with η is needed, since the righthand side is a variable.
- For each rule of the form $@(l, x) \to r$ with $x \notin \mathcal{V}ar(l)$, we need the normal extension $l \to \lambda x.r$. A rule can have only finitely many extensions.

The variation

- Proof: let $S_1 = \{\beta\}$ and $S_2 = \{\eta\}$.
- No shallow pairs are needed when lefthand sides are patterns.
- No β-extensions are needed since lefthand sides are now headed by a defined function symbol.
- No forward extension with η is needed, since the righthand side is a variable.
- For each rule of the form $@(l, x) \rightarrow r$ with $x \notin \mathcal{V}ar(l)$, we need the normal extension $l \rightarrow \lambda x.r$. A rule can have only finitely many extensions.

The variation

- Proof: let $S_1 = \{\beta\}$ and $S_2 = \{\eta\}$.
- No shallow pairs are needed when lefthand sides are patterns.
- No β-extensions are needed since lefthand sides are now headed by a defined function symbol.
- No forward extension with η is needed, since the righthand side is a variable.
- For each rule of the form $@(l, x) \rightarrow r$ with $x \notin \mathcal{V}ar(l)$, we need the normal extension $l \rightarrow \lambda x.r$. A rule can have only finitely many extensions.

The variation

- Proof: let $S_1 = \{\beta\}$ and $S_2 = \{\eta\}$.
- No shallow pairs are needed when lefthand sides are patterns.
- No β-extensions are needed since lefthand sides are now headed by a defined function symbol.
- No forward extension with η is needed, since the righthand side is a variable.
- For each rule of the form $@(l, x) \to r$ with $x \notin \mathcal{V}ar(l)$, we need the normal extension $l \to \lambda x.r$. A rule can have only finitely many extensions.

Examples

- Modelling simply typed lambda calculus:

$$\begin{aligned} \text{app} &: (\alpha \to \beta) \times \alpha \Rightarrow \beta \\ \text{abs} &: (\alpha \to \beta) \Rightarrow (\alpha \to \beta) \end{aligned}$$

The β- and η-rules are encoded as:

$$\begin{aligned} \text{app}(\text{abs}(\lambda x.Z(x)), Z') &\to Z(Z') \\ \text{abs}(\lambda x.\text{app}(X, x)) &\to X \end{aligned}$$

- Modelling differentiation:

$$\begin{aligned} \text{diff} &: (R \to R) \Rightarrow (R \to R) \\ \sin, \cos &: R \Rightarrow R \end{aligned}$$

A rewrite rule for differentiation:

$$\text{diff}(\lambda x.\sin(F(x))) \to \lambda x.\cos(F(x)) \times \text{diff}(\lambda x.F(x))$$

Examples

- Modelling simply typed lambda calculus:

$$\begin{aligned} \mathsf{app} &: (\alpha \to \beta) \times \alpha \Rightarrow \beta \\ \mathsf{abs} &: (\alpha \to \beta) \Rightarrow (\alpha \to \beta) \end{aligned}$$

The β- and η-rules are encoded as:

$$\begin{aligned} \mathsf{app}(\mathsf{abs}(\lambda x.Z(x)), Z') &\to Z(Z') \\ \mathsf{abs}(\lambda x.\mathsf{app}(X, x)) &\to X \end{aligned}$$

- Modelling differentiation:

$$\begin{aligned} \mathsf{diff} &: (R \to R) \Rightarrow (R \to R) \\ \mathsf{sin}, \mathsf{cos} &: R \Rightarrow R \end{aligned}$$

A rewrite rule for differentiation:

$$\mathsf{diff}(\lambda x.\mathsf{sin}(F(x))) \to \lambda x.\mathsf{cos}(F(x)) \times \mathsf{diff}(\lambda x.F(x))$$

More examples

- Modelling finite polymorphic lists:

$$
\begin{array}{lcl}
\text{list} & : & * \longrightarrow * \\
\text{nil} & : & \text{list}(\alpha) \\
\text{cons} & : & \alpha \times \text{list}(\alpha) \Rightarrow \text{list}(\alpha) \\
\text{map} & : & \text{list}(\alpha) \times (\alpha \rightarrow \alpha) \Rightarrow \text{list}(\alpha)
\end{array}
$$

$$
\begin{array}{rcl}
\text{map}(\text{nil}, F) & \longrightarrow & \text{nil} \\
\text{map}(\text{cons}(H, T), F) & \longrightarrow & \text{cons}(@(F, H), \text{map}(T, F))
\end{array}
$$

- Modelling differentiation again:

$$
\begin{array}{lcl}
\alpha & : & \text{an algebraic closed field} \\
\text{diff}, \sin, \cos & : & (\alpha \rightarrow \alpha) \Rightarrow (\alpha \rightarrow \alpha) \\
F & : & \alpha \rightarrow \alpha \\
\text{diff}(\sin(F)) & \longrightarrow & \cos(F) \times \text{diff}(F)
\end{array}
$$

More examples

- Modelling finite polymorphic lists:

$$\begin{array}{lcl} \text{list} & : & * \longrightarrow * \\ \text{nil} & : & \text{list}(\alpha) \\ \text{cons} & : & \alpha \times \text{list}(\alpha) \Rightarrow \text{list}(\alpha) \\ \text{map} & : & \text{list}(\alpha) \times (\alpha \longrightarrow \alpha) \Rightarrow \text{list}(\alpha) \end{array}$$

$$\begin{array}{rcl} \text{map}(\text{nil}, F) & \longrightarrow & \text{nil} \\ \text{map}(\text{cons}(H, T), F) & \longrightarrow & \text{cons}(@(F, H), \text{map}(T, F)) \end{array}$$

- Modelling differentiation again:

$$\begin{array}{lcl} \alpha & : & \text{an algebraic closed field} \\ \text{diff}, \text{sin}, \text{cos} & : & (\alpha \longrightarrow \alpha) \Rightarrow (\alpha \longrightarrow \alpha) \\ F & : & \alpha \longrightarrow \alpha \\ \text{diff}(\text{sin}(F)) & \longrightarrow & \text{cos}(F) \times \text{diff}(F) \end{array}$$

Our framework is slightly different from the usual one by using free variables with arities. The impact of this change on pattern matching and unification is simply that a free variable X of arity n always comes together with the (all different) bound variables to which it is applied. For example, a solved equation $X = t$ is replaced by an equations of the form $X(x_1, \ldots, x_n) = t$ in which the variables $x_1, \ldots, x_n$ must be all different because of the pattern condition, and the variable X does not occur in t. And therefore, the most general higher-order unifier must substitute the variable X by the term $\lambda x_1 \cdots x_n.t$.