Réécriture : principales définitions et théorèmes

Laure Danthony

D'après le cours de Pierre Lescanne, janvier - mai 2001

Table des matières

1 Introduction

La réécriture est la donnée d'un ensemble de règles *orientées* qui permettent de façon plus ou moins déterministe de déduire une écriture d'une autre.
Les questions qui se posent sont les suivantes :

- les formes dites "simplifiées" sont-elles uniques ?
- comment orienter les équations ?
- le processus de simplification se termine-t-il ?
- ...

2 Rappels et notations sur les ordres

2.1 Caractéristiques d'une relation

DÉFINITION 1
Une relation est dite :

- transitive si xRy et $yRz \Rightarrow xRz$
- antisymétrique si xRy et $yRx \Rightarrow x = y$
- réflexive si $\forall x, xRx$

REMARQUE 1 Cette notion de l'antisymétrie est celle francophone et aussi celle du "All that", la notion anglosaxonne (qui interdit xRx) est la suivante : $\nexists x, y, xRy$ et yRx.

2.2 Définition d'un ordre

DÉFINITION 2
Un **préordre** est une relation transitive et réflexive.

DÉFINITION 3
Un **ordre** est une relation transitive, réflexive et symétrique.

DÉFINITION 4
Un ordre est dit **partiel** si il existe des couples qui ne sont pas en relation. Dans le cas contraire, il est dit **total**

DÉFINITION 5
Un ordre est **strict** si il a la relation d'irréflexivité, c'est à dire :

$$\forall x, \neg(xRx).$$

2.3 Eléments maximaux, maximum

On considère $(A, \leq)$ un poset (ensemble partiellement ordonné), et $M \subseteq A$, alors :

DÉFINITION 6
- Un **élément maximal** de M est un élément m qui vérifie : $\forall n \in M, n \geq m \Rightarrow n = m$.
- Le **maximum** de M est m si $\forall n \in M, m \geq n$.
- Un **majorant** de M est un élément $u \in A$ qui vérifie : $\forall n \in M, u \geq n$.

2.4 Caractère noethérien, bons ordres

DÉFINITION 7
Une relation R **termine** ou **est noethérienne** ssi il n'existe pas de suite infinie "décroissante" au sens suivant : $x_0 \xrightarrow[R]{} x_1 \xrightarrow[R]{} \dots x_n \xrightarrow[R]{} x_{n+1} \dots$

PROPOSITION 1 *Le produit lexicographique d'ordres qui termine termine.*

DÉFINITION 8
Un **bon ordre** est un ordre noethérien total.

2.5 Ordre incrémental, antichaîne, bel ordre

DÉFINITION 9
Un ordre est dit **incrémental** si

- il est noethérien ;
- tout ordre qui le contient est noethérien .

DÉFINITION 10
Une **antichaîne** est une suite d'éléments incomparables, c'est-à-dire un ensemble tel que $x \leq y \Rightarrow x = y$.

DÉFINITION 11
Un **bel ordre** est un ordre noethérien sans antichaîne infinie.

2.6 Suites et sous-suites bonnes, mauvaises

DÉFINITION 12
Une suite $(x_i)_{i\in\mathbb{N}}$ est dite **bonne** si $\exists i < j, x_i \leq x_j$. Dans le cas contraire, elle est dite **mauvaise**.

DÉFINITION 13
Une sous suite de $(x_i)_{i\in\mathbb{N}}$ est $(x_{\varphi(i)})_{i\in\mathbb{N}}$ où φ est une application $\mathbb{N} \to \mathbb{N}$ croissante.

2.7 Résultat important

THÉORÈME 1 *Les phrases suivantes sont équivalentes :*

- *l'ordre est incrémental*
- *l'ordre est noethérien et sans antichaîne infinie*
- *toute suite est bonne*
- *de toute suite on peut extraire une sous-suite croissante.*

3 Confluence

3.1 Définition

DÉFINITION 14
Une relation est **localement confluente** si elle vérifie : $y_1 \longleftarrow x \longrightarrow y_2 \Rightarrow (y_1, y_2)$ joignables.

DÉFINITION 15
Une relation est **semi confluente** si elle vérifie : $y_1 \xleftarrow{*} x \longrightarrow y_2 \Rightarrow (y_1, y_2)$ joignables.

DÉFINITION 16
Une relation est **confluente** si elle vérifie : $y_1 \xleftarrow{*} x \xrightarrow{*} y_2 \Rightarrow (y_1, y_2)$ joignables.

DÉFINITION 17
Une relation est **convergente** si elle termine et est confluente.

3.2 Propriété de Church-Rosser

DÉFINITION 18
La relation R a la **propriété de Church Rosser** si elle vérifie l'assertion suivante :

$$\text{si } N_1 \xleftrightarrow[R]{*} N_2, \text{ alors } \exists P, N_1 \xrightarrow[R]{*} P \text{ et } N_2 \xrightarrow[R]{*} P$$

PROPOSITION 2 *semi-confluence* $\Leftrightarrow$ *confluence* $\Leftrightarrow$ *Church-Rosser.*

3.3 Réductibilité

DÉFINITION 19
On dit que x et y sont **joignables** si y et x peuvent se réécrire en un même z.

DÉFINITION 20
x est dit **réductible** si il existe y tel que $x \xrightarrow[R]{} y$. Sinon, il est **irréductible**.

DÉFINITION 21
y est une **forme normale** de x si $x \xrightarrow[R]{*} y$ avec y irréductible.

DÉFINITION 22
Une relation est **normalisante** si chaque élément a une forme normale.

3.4 Quelques résultats

PROPOSITION 3 *Si R est confluente, alors chaque élément a au plus une forme normale.*

PROPOSITION 4 *Si R est normalisante et confluente, alors chaque élément a une unique une forme normale.*

PROPOSITION 5 *Si R est normalisante et confluente, alors $x \xleftrightarrow[R]{*} y$ est équivalent à $x$et y ont même forme normale.*

3.5 Induction bien fondée

$$\text{WFI} \quad \frac{\forall x \in A, \left(\forall y, x \xrightarrow{+} y, P(y)\right) \Rightarrow P(x)}{\forall x \in A, P(x)}.$$

REMARQUE 2 On n'a pas besoin de vérifier pour les éléments mininaux,

PROPOSITION 6 *Si $\longrightarrow$ termine, alors WFI est vérifiée.*

REMARQUE 3 La réciproque est aussi vraie.

3.6 Branchements

DÉFINITION 23

- Une relation ra est à **branchement fini** si chaque élément a un nombre fini de sucesseurs directs.
- Une relation est **globalement finie** si chaque élément a un nombre fini de sucesseurs (directs ou non).
- Une relation est **acyclique** s'il n'existe pas d'élément a tel que $a \xrightarrow{+} a$.

PROPOSITION 7 *Une relation à branchement fini qui termine est globalement finie.*

PROPOSITION 8 *Une relation globalement finie et acyclique termine.*

LEMME 1 (DE KÖNIG) *Un arbre à branchement fini est infini ssi il contient un chemin infini.*

4 Ordre Multiensemble

4.1 Notion de multiensemble

DÉFINITION 24

Un **multiensemble** M sur A est une fonction $M : A \to \mathbb{N}$; intuitivement c'est un ensemble où la répétition d'éléments est autorisée. $M(x)$ est alors le nombre de répétitions de x dans M. La notation standard est $\{a, a, b\}$ pour $\{a \to 2, b \to 1, c \to 0\}$.

DÉFINITION 25

Un multiensemble est **fini** si son support est fini, i.e. l'ensemble des x d'image non nulle est fini. On note $\mathcal{M}(A)$ l'ensemble des multiensembles finis sur A.

4.2 Opérations sur les multiensembles

- $x \in M$ si $M(x) > 0$.
- $N \subseteq M$ ssi $\forall x, N(x) \geq M(x)$.
- $(M \cup N)(x) = M(x) + N(x)$.
- $(M - N)(x) = M(x) \dot{-} N(x)$.

Dans la suite, on considère des multiensembles finis

4.3 Ordre multiset

DÉFINITION 26

Soit $>$ un ordre strict sur A. L'extension multiensemble $>_{mult}$ sur $M(A)$ est définie par : $M >_{mult} N$ si il existe des multiensembles X et Y vérifiant :

- $\emptyset \neq X \subseteq M$
- $N = (M - X) \cup Y$
- $\forall y \in Y, \exists x \in X, x > y$

REMARQUE 4 C'est l'ordre "de l'hydre". A chaque fois qu'on coupe une tête, il en repousse plusieurs, mais plus petites...

4.4 Principales propriétés

PROPOSITION 9 *Si $>$ est strict sur A, alors $>_{mult}$ est strict sur $M(A)$.*

PROPOSITION 10 *$>$ termine ssi $>_{mult}$ termine.*

5 Algèbres de termes

5.1 Domaine d'un arbre enraciné étiqueté

DÉFINITION 27
Un **domaine d'arbre** est un sous-ensemble fini non vide D de mots sur $\mathbb{N}^*$ tel que:

- tout préfixe d'un mot qui est dans D est dans D
- si $p.i \in D$ et $0 < j < i$, alors $p.j \in D$

FAIT 1 $\forall D, \varepsilon \in D$

5.2 Arbre enraciné étiqueté

DÉFINITION 28
Un **arbre étiqueté** par A est une application $T : D \to A$ telle que:

- $T(\epsilon)$ est la **racine** de l'arbre.
- D est le **domaine** (ou l'ensemble des positions) de T noté $Pos(T)$.
- $T(p)$ s'appelle l'**étiquette** à la position p.
- Si $p \in Pos(T)$ et $(\forall i \in \mathsf{IN}_\star) pi \notin Pos(T)$, alors p est une feuille de T.

5.3 Sous-arbre

DÉFINITION 29
$T_{|p}$ est le **sous-arbre à la position** p défini par $T_{|p}(q) = T(pq)$ où $Pos(T_{|p}) = \{q \mid pq \in Pos(T)\}$.

Autrement dit, c'est le sous-arbre de T de racine p.

FAIT 2 $(T_{|p})_{|q} = T_{|pq}$.

5.4 Remplacement

DÉFINITION 30
$T[U]_p$ est le **remplacement** du sous-arbre de T à la position p par l'arbre U, on le définit par:

$$Pos(T[U]_p) = \{q \in Pos(T) \mid \neg(p \,\textbf{prefix}\, q)\} \cup \{pp' \mid p' \in Pos(U)\}$$

et

$$T[U]_p(q) = \begin{cases} T(q) \text{ si } \neg(p \,\textbf{prefix}\, q) \\ U(p') \text{ si } q = pp' \text{ avec } p' \in Pos(U) \end{cases}$$

Autrement dit, on a remplacé par U le sous-arbre de T de racine p en renumérotant de façon adéquate.

PROPOSITION 11
- $T[U[V]_q]_p = T[U]_p[V]_{pq}$.

- $T[U]_p[V]_q = T[V]_q[U]_p$ *si* p *et* q *sont étrangers*[1].
- *Si* p *et* q *sont étrangers, alors* $(T[U]_p)_{|q} = T_{|q}$.
- $(T[U]_p)_{|pp'} = U_{|p'}$.
- $(T[U]_{pp'})_{|p} = (T_{|p})[U]_{p'}$.

5.5 Signature

DÉFINITION 31
Une **signature** est une famille d'ensembles de fonctions $(\Sigma_n)_{n\in\mathbb{N}}$.

$$\Sigma = \bigcup_{n\in\mathbb{N}} \Sigma_n$$

Si $f \in \Sigma_n$, f est dit d'**arité** n.
Σ_0 est l'ensemble des **constantes**.

5.6 Termes

Soit X un ensemble dit ensemble des *variables*.

DÉFINITION 32
Un **Σ-terme** sur X est un arbre t étiqueté par $\Sigma \cup X$, tel que

- Si $t(p) \in X$, alors p est une feuille.
- Si $f \in \Sigma$ d'arité n, alors c'est un nœud et il a exactement n feuilles.

DÉFINITION 33
$T(\Sigma, X)$ est l'ensemble des Σ-termes sur X.

DÉFINITION 34
$Var(t) = \{x \in X / \exists p \in Pos(t)\ t(p) = x\}$

FAIT 3 *Si* $t(p) \in \Sigma_0$, *alors* p *est une feuille.*

5.7 Σ-algèbres et morphismes

DÉFINITION 35
Une **Σ-algèbre** est un couple $\mathcal{A} = (A, \Sigma^{\mathbb{A}})$ où A est un ensemble dit *support* de l'algèbre,

$$\Sigma^{\mathbb{A}} = \bigcup_{n\in\mathbb{N}} \Sigma_n^{\mathbb{A}}$$

et $\Sigma_n^{\mathbb{A}}$ est l'ensemble des fonctions de type $A^n \to A$.

DÉFINITION 36
Un **morphisme** $\varphi : T(\Sigma, X) \to A$ est une application telle que

- $\varphi(x_i) = a_i$
- $\varphi(f(t_1, ..., t_n)) = f^{\mathbb{A}}(\varphi(t_1), ..., \varphi(t_n))$.

1. p et q sont dits **étrangers**, si p n'est pas préfixe de q et si q n'est pas préfixe de p

5.8 Substitutions, domaine, codomaine

Soit V un ensemble dénombrable de variables.

DÉFINITION 37
Une **substitution** est une application $\sigma : V \to T(\Sigma, V)$ qui est l'identité presque partout (c-à-d sauf sur un ensemble fini).

REMARQUES 5
- On note: $Dom(\sigma) = \{x \in V\}\, \sigma(x) \neq x$ le **domaine**.
- On note: $Range(\sigma) = \{\sigma(x)\}\, x \in Dom(\sigma)$ le **codomaine**.

DÉFINITION 38
(extension) On étend σ à $T(\Sigma, V)$ en une application $\hat{\sigma}$ (souvent notée σ) :
- $\hat{\sigma}(x) = \sigma(x)$,
- $\hat{\sigma}(f(t_1), ..., f(t_n)) = f(\hat{\sigma}(t_1), ..., \hat{\sigma}(t_n))$.

DÉFINITION 39
$Sub(T(\Sigma, V))$ est l'ensemble des substitutions.

5.9 Identités et réduction

DÉFINITION 40
Une **identité** $s \approx t$ est un couple[2] de termes.

DÉFINITION 41
Soit E un ensemble d'identités, la **réduction** $\xrightarrow[E]{}$ est définie par:
$s \xrightarrow[E]{} t$ ssi t est obtenu par remplacement dans s de la partie $\sigma(g)$ par $\sigma(d)$ où $(g \approx d) \in E$.

6 Prouver la terminaison d'un système de réécriture

6.1 Systèmes de réécriture

DÉFINITION 42
Une **règle de réécriture** est une identité $g \approx d$ telle que $Var(g) \supseteq Var(d)$. "en réécrivant on diminue le nombre de variables".

DÉFINITION 43
Un **système de réécriture** est un ensemble de règles de réécriture.

DÉFINITION 44
Un **redex** est une instance d'un membre gauche de règle de réécriture.

DÉFINITION 45
Contracter le redex $s_{|p}$ à la position p, c'est passer de s à $t = s[\sigma(d)]_p$.

2. c'est-à-dire un élément de $T(\Sigma, V) \times T(\Sigma, V)$

6.2 Ordres de réécriture, de réduction, de simplification

DÉFINITION 46
Un **ordre de réécriture** est un ordre $<$ sur $T(\Sigma, V)$ qui est :

- *compatible* cad si $s > s'$ alors $f(t_1, \ldots, t_{i-1}, s, t_{I+1}, \ldots, t_n) < f(t_1, \ldots, t_{i-1}, s', t_{I+1}, \ldots, t_n)$.
- *clos par substitution* cad $\forall \sigma \in Subst(T(\Sigma, V))$, $s_1 > s - 2 \Rightarrow \sigma(s_1) > \sigma(s_2)$.

DÉFINITION 47
Un **ordre de réduction** est un ordre de réécriture noethérien.

THÉORÈME 2 *Un système de réécriture R termine ssi il existe un ordre de réduction $>$ tel que $l > r$ pour tout $l \longrightarrow r \in R$.*

DÉFINITION 48
Un **ordre de simplification** est un ordre de réécriture qui satisfait la propriété de sous-terme, cad $\forall p \in Pos(t), t > t_{|p}$.

PROPOSITION 12 *On démontre les faits suivants :*

- *les ordres de simplification sont des beaux ordres ;*
- *les ordres de simplification terminent ;*
- *les ordres de simplification sont des ordres de réduction.*

6.3 Méthode directe

Pour prouver directement la terminaison d'un système de réécriture, on peut utiliser le théorème :

THÉORÈME 3 (LEMME DE KÖNIG) *Un arbre à branchement fini est infini ssi il contient un chemin infini*

REMARQUE 6 En fait on utilise ce théorème par la contraposée.

6.4 Interprétation polynômiale

DÉFINITION 49
Une interprétation polynômiale est la donnée d'une Σ-algèbre $\mathcal{A} = (A, \Sigma^A)$ où les fonctions $f \in \Sigma_n^A$ sont des polynômes à n variables à coefficients entiers.

PROPOSITION 13 *L'ordre induit par une interprétation polynomiale est un ordre de réduction, appelé* **ordre polynômial**.

REMARQUE 7 On peut ainsi prouver la terminaison d'un système de réécriture: si on trouve une interprétation polynômiale pour les symboles fonctionnels telle que $f^{\mathcal{A}}(s) > f^{\mathcal{A}}(t)$ pour tout $s \longrightarrow t \in R$ et qui est monotone (c'est à dire que les polynômes dépendent de toutes leurs indéterminées), alors on a prouvé que R termine.

REMARQUE 8 Le "All that" précise qu'il faut vérifier la clôture, c'est-à-dire

$$\forall x_1, x_2, \ldots x_n \in A, \forall f \in \Sigma_n^A, f(x_1, x_2, \ldots, x_n) \in A.$$

6.5 Utilisation de l'ordre lpo

DÉFINITION 50 (LPO)
Etant donné un ordre $\leq$ sur Σ appelé **précédence**: $s <_{lpo} t$ ssi

(A) s est une variable x, t n'est pas une variable et $x \in Var(t)$.

ou

(B) s s'écrit $f(s_1, \ldots, s_m)$ et t s'écrit $g(t_1, \ldots t_n)$ et

(B1) $\forall i \in [\![1, m]\!], s_i <_{lpo} t$

et

(B2) :

(B21) $f < g$

ou

(B22) $f \equiv g$ et $(s_1, \ldots, s_m) <_{lpo}^{lex} (t_1, \ldots t_n)$.

ou

(B23) $\exists j \in [\![1, n]\!], s \leq_{lpo} t_j$.

THÉORÈME 4 $<_{lpo}$ *"lexicographic path order" est un ordre de simplification.*

REMARQUE 9 On peut donc ainsi prouver la terminaison d'un système de réécriture en donnant une précédence sur les symboles fonctionnels de telle sorte à ce que $s <_{lpo} t$ pour tout $s \longrightarrow t \in R$.

REMARQUE 10 Le "All that" donne une autre définition pour $<_{lpo}$.

7 Unification

7.1 But

Le but est le suivant : étant donné un ensemble de paires (l'ordre ne compte pas) de termes de la forme $s_i \stackrel{?}{=} t_i$, trouver une solution à ce **problème équationnel**, c'est-à-dire une substitution σ telle que $\forall i, \sigma(s_i) = \sigma(t_i)$. Une telle substitution est appelée **unificateur** au problème équationnel. Si elle existe, on dit que le problème est **unifiable**.

DÉFINITION 51
$x \stackrel{?}{=} s$ est en **forme résolue** si x n'apparaît nulle part ailleurs dans le système équationnel (x n'est pas une variable de s en particulier). Si toutes les équations sont sous forme résolue, le système équationnel est sous **forme résolue**.

7.2 Méthode

On utilise des règles d'inférences qui transforment un problème en un problème équivalent à chaque application :

- Supprime $\dfrac{\{u \stackrel{?}{=} u\} \cup E}{E}$
- Décompose $\dfrac{\{f(u_1, \ldots u_n) \stackrel{?}{=} f(v_1, \ldots, v_n)\} \cup E}{\{u_1 \stackrel{?}{=} v_1, \ldots, u_n \stackrel{?}{=} v_n\} \cup E}$ si $f \in \Sigma_n$
- Elimine $\dfrac{\{x \stackrel{?}{=} v\} \cup E}{\{x \stackrel{?}{=} v\} \cup \sigma(E)}$ où $x \stackrel{?}{=} v$ n'est pas en forme résolue et $x \notin Var(v)$ et $\sigma = \{x \mapsto v\}$.

7.3 Résultats

PROPOSITION 14 (CORRECTION)

- *Ce système de règles conserve les unificateurs, c'est à dire que si* $E \Rrightarrow E'$ *avec ce système, les unificateurs de* E *sont les unificateurs de* E*'.*
- *Si* $E \overset{*}{\Rrightarrow} E'$ *aec* E' *sous forme résolue, alors le plus petit unificateur de* E *est l'unificateur trivial de* E'*, c'est à dire :* $\forall i, \sigma(x_i) = t_i$*.*

PROPOSITION 15 (COMPLÉTUDE) *Si* E *est unifiable, alors il existe* E' *obtenu à partir de* E *par le jeu de règles précédents, avec* E' *sous forme résolue.*

REMARQUE 11 L'unification peut être exponentielle en la taille du problème originel.

8 Complétion

8.1 Qu'est-ce que la complétion ?

Une **procédure de complétion** transforme un ensemble d'identités (et de règles confluentes) en un ensemble équivalent de règles confluentes et qui se terminent.

Cela nécessite un ordre pour orienter les identités, ce qui est l'objet de la procédure de complétion.

8.2 Paires critiques

DÉFINITION 52
Si il existe :

- deux règles $l \longrightarrow r$ et $g \longrightarrow d$,
- une position $p \in Pos(l)$,
- un mgu σ de $l_{|p}$ et de g,

le terme $\sigma(l)$ est applé une **superposition**, et la paire $\langle \sigma(r) = \sigma(\sigma(l[d]_p)) \rangle$ est appelée **une paire critique.**

DÉFINITION 53
Si les deux termes de la paire se réécrivent en un même terme, la paire est dite **convergente**. Sinon, elle est dite **divergente**.

THÉORÈME 5 (DE KNUTH ET BENDIX) *Soit un système de réécriture* R*. Alors* R *est confluent si et seulement si toutes ses paires critiques sont confluentes.*

8.3 Algorithme de complétion

On se donne un ordre $>$ sur les termes : Les règles suivante définissent la procédure : (E est un ensemble d'égalités, R un ensemble de règles). Au début, R est vide et E contient les égalités à compléter.

- Delete $\dfrac{E \cup \{s = s\}; R}{E; R}$
- Compose $\dfrac{E; R \cup \{s \longrightarrow t\}}{E; R \cup \{s \longrightarrow u\}}$ si $t \xrightarrow[R]{} u$.
- Simplify $\dfrac{E \cup \{s = t\}; R}{E \cup \{s = u\}; R}$ si $t \xrightarrow[R]{} u$.

- Collapse $\dfrac{E; R \cup \{s \longrightarrow t\}}{E \cup \{u = t\}; R}$ si $s \xrightarrow[R]{} u$ par une règle $l \longrightarrow r \in R$ avec $l \neq s$ instance d'un sous-terme de s ou $(s = l)$ et $t > r$.

- Orient $\dfrac{E \cup \{s = t\}; R}{E; R \cup \{s \longrightarrow t\}}$ si $s > t$.

- Deduce $\dfrac{E; R}{E \cup \{s = t\}; R}$, si $s \xleftarrow[R]{} u \xrightarrow[R]{} t$ pour un u.

REMARQUE 12 Pour appliquer cette procédure (règle Deduce), il faut savoir calculer les paires critiques à tout moment.

Justement, c'est le calcul des paires critiques qui fait toute la difficulté d'une procédure de complétion. Les différentes procédures utilisent des astuces qui leur permettent de calculer le moins de paires critiques possibles. L'ANS-complétion est la procédure qui calcule le moins de paires (vérifié expérimentalement).

9 Ordre sur les preuves

Si on dispose d'une preuve d'une égalité $a = b$, complétement par égalité, lors de la procédure de complétion on n'a plus d'égalité mais simplement des règles de réécriture. Ce que l'on souhaite c'est obtenir une preuve complétement par réécriture (une preuve "en vallée"). La procédure de complétion garantit (théorème) en fait l'existance d'une réduction commune à a et à b dans l'ensemble des règles persistantes. Pour montrer ce théorème, on a besoin d'introduire un ordre sur les preuves.

9.1 Notations

On note :

- $E_\infty = \bigcup_{i \geq 0} E_i$,
- $R_\infty = \bigcup_{i \geq 0} R_i$,
- $E_\omega = \bigcup_{i \geq 0} \bigcap_{j \geq i} E_j$ et
- $R_\omega = \bigcup_{i \geq 0} \bigcap_{j \geq i} R_j$ (R_ω est l'ensemble des **règles persistantes**).

9.2 Rappel sur l'enchâssement

9.3 Le coût d'une preuve

- **Coût d'un couple** A un stade de la procédure de complétion, on dispose d'un ensemble d'égalités, et de règles du type $a \to b$. On attribue le coût des paires de la manière suivante :
  - $a_i = a_{i+1}$ a un coût $(\{\!\{a_i, a_{i+1}\}\!\}, \_, \_)$;
  - $a_i \longrightarrow a_{i+1}$ a un coût $(\{\!\{a_i\}\!\}, l, r)$, où l est le membre gauche de la réécriture employée et r est le membre droit de la règle de réécriture employée.
  - $a_i \leftarrow a_{i+1}$ a un coût $(\{\!\{a_{i+1}\}\!\}, l, r)$, où l est le membre gauche de la réécriture employée et r est le membre droit de la règle de réécriture employée.
- **Coût d'une preuve** C'est le multiensemble des coûts des couples.

Si une preuve $\mathcal{P}$ est supérieure (pour l'ordre multiensemble) à une preuve $\mathcal{P}'$, on note $\mathcal{P}' \prec_C \mathcal{P}$.

DÉFINITION 54
Deux preuves sont dites **équivalentes** si elles prouvent la même identité.

On montre que chaque règle de la procédure de complétion fait décroître les coûts des preuves.
On obtient finalement le théorème :

THÉORÈME 6 *Si il existe dans $E_\infty \cup R_\infty$ une preuve $\mathcal{P}$ qui n'est pas une preuve de réécriture, alors il existe dans $E_\infty \cup R_\infty$ une preuve $\mathcal{P}'$ équivalente telle que $\mathcal{P}' \prec_C \mathcal{P}$.*

et donc les corollaires :

COROLLAIRE 1 *Si il existe une preuve $\mathcal{P}$ dans $E_\infty \cup R_\infty$, alors il existe une preuve de réécriture équivalente dans R_ω (c'est ce qu'on voulait).*

COROLLAIRE 2 *R_ω est convergent, c'est-à-dire confluent et termine.*

COROLLAIRE 3 *Si $R\omega$ est fini, alors le problème du mot dans E_0 est décidable.*

10 Bases de Gröbner

Voir le cours annexe.