RAIRO. INFORMATIQUE THÉORIQUE

J. P. JOUANNAUD
H. KIRCHNER

Construction d'un plus petit ordre de simplification

RAIRO. Informatique théorique, tome 18, nº 3 (1984), p. 191-208
<http://www.numdam.org/item?id=ITA_1984__18_3_191_0>

R.A.I.R.O. Informatique théorique/Theoretical Informatics
(vol. 18, n° 3, 1984, p. 191 à 207)

CONSTRUCTION D'UN PLUS PETIT ODRE DE SIMPLIFICATION (*)

par J. P. JOUANNAUD [1], H. KIRCHNER [1]

Communiqué par J. E. PIN

Résumé. — *Nous construisons explicitement le plus petit ordre de simplification (au sens de l'inclusion des relations) contenant une paire (g, d) donnée telle que g ne soit pas plongé dans d. La construction est simple mais la preuve techniquement complexe.*

Nous en déduisons un critère simple permettant de décider la terminaison d'une règle $g \to d$ à l'aide d'un ordre de simplification : il suffit pour cela que g ne s'unifie pas avec un terme non réduit à une variable et plongé dans d.

La méthode devra être étendue au cas de plusieurs règles avant de pouvoir concurrencer les méthodes existantes.

Abstract. — *We give a construction of the smallest (for inclusion of sets) simplification ordering which contains a given pair (g, d) such that g is not embedded in d. The construction is simple but the proof is complex.*

As a consequence, we show that it is decidable to prove the termination of a rewrite rule $g \to d$ using a simplification ordering: it must be checked that g does not unify with any non variable term embedded in d.

The method has to be extended to many rules systems before to compete with existant techniques.

1. INTRODUCTION

L'étude des systèmes de règles de réécriture s'est intensivement développée ces dernières années [6], car ils se sont révélés être un outil fondamental dans de nombreux domaines essentiels à la théorie de la programmation : types abstraits algébriques [3], preuve et transformation de programmes [4], démonstration automatique [7], unification [8]...

Le problème étudié dans cet article est celui de la terminaison, dont on sait qu'il est en général indécidable [5], mais qui a connu une activité intense depuis l'article de Dershowitz [1] sur les ordres de simplification montrant

(*) Reçu en avril 1982, révisé en juillet 1983.

(1) Centre de Recherche en Informatique de Nancy, Campus scientifique, B.P. n° 239, 54506 Vandœuvre-les-Nancy Cedex, France.

qu'il s'agissait d'un outil puissant et pratique à la fois pour la preuve de terminaison d'un système de règles de réécriture. La plupart des recherches dans ce domaine ont consisté à trouver et étudier de nouveaux ordres de simplification, plus simples ou plus puissants [10, 13, 1, 11 et 9]. Ces ordres de simplification sont des ordres partiels sur les termes d'une algèbre libre, récursivement engendrés à partir d'un ordre initial sur les symboles de fonction de l'algèbre. L'inconvénient est que l'ordre initial sur les symboles de fonction doit être donné *a priori*, ce qui exclut les preuves automatiques de terminaison (cela n'est pas vrai pour l'ordre de décomposition [9], mais la complexité du *calcul* de l'ordre initial n'a pas été étudiée jusqu'ici).

Nous tentons une autre approche dans cet article dont le but est de trouver un critère suffisant de terminaison qui soit décidable. La méthode consiste à vérifier l'existence d'un ordre de simplification stable par instanciation et contenant une relation initiale. En pratique, la relation initiale est constituée des paires (g_i, d_i) d'un système de règles de réécriture. La construction décrite dans cet article concerne les systèmes de réécriture à *une seule règle* : nous montrons qu'il existe un ordre de simplification $>$ tel que $\sigma g > \sigma d$ ssi g n'est unifiable avec aucun terme non variable plongé dans d.

Par exemple, la règle $f(x, x) \to f(h(a), b)$ termine car $f(x, x)$ ne s'unifie avec aucun des termes a, b, $h(a)$, $f(a, b)$ et $f(h(a), b)$. Notons que la terminaison de cette règle ne peut s'obtenir par les ordres de simplification habituels.

Néanmoins, la construction devra être généralisée au cas de systèmes de réécriture à plusieurs règles avant de prétendre concurrencer les autres méthodes.

2. PRÉLIMINAIRES

Étant donné un ensemble F de symboles de fonctions munis d'une arité $a : F \to N$, et un ensemble dénombrable X de symboles de variables (d'arité o par convention), $T(F, X)$ dénote l'ensemble des *termes* sur F (ou F-magma libre ou F-algèbre libre selon les auteurs). Les termes de $T(F, X)$ peuvent être considérés comme des arbres étiquetés de la manière suivante [2] :

Un arbre t est une fonction partielle de N_+^* (le monoïde libre sur N_+, ε étant le mot vide) dans FUX, dont le domaine $D(t)$,

$D(t) = \{ u \in N_+^* \mid t(u) \text{ est défini} \}$ vérifie :

$$\varepsilon \in D(t); \tag{1}$$

$$uv \in D(t) \quad \Rightarrow \quad u \in D(t) \quad \text{(clôture par préfixe)}; \tag{2}$$

$$\forall u \in D(t), \quad ui \in D(t) \text{ ssi } 1 \leqq i \leqq a(t(u)). \tag{3}$$

Un élément u de $D(t)$ est un *nœud* de t. Le *sous-arbre* de t issu du nœud u est l'arbre noté t/u tel que $t/u(v)=t(uv)$, $\forall v \in N_+^*$. La *substitution* de t' à t/u dans t produit l'arbre noté $t[u \leftarrow t']$ tel que $t[u \leftarrow t'](v)=t(v)$ si u n'est pas préfixe de v et $t[u \leftarrow t'](v)=t'(v/u)$ si u est préfixe de v, v/u dénotant le *quotient à droite* de v par u. Le *chemin* d'accès au *nœud* $u \in D(t)$ est le mot de $(\mathrm{FUX})^*$ noté $\hat{u}$ défini par $\hat{\varepsilon}=t(\varepsilon)$ et $u.i=\hat{u}.t(ui)$.

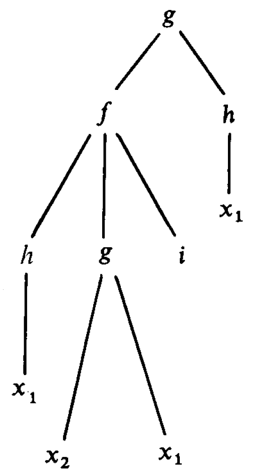
Deux nœuds u et v de t ou deux chemins $\hat{u}$ et $\hat{v}$ de t sont dits *compatibles* si l'un est préfixe de l'autre. Ils sont dits *incompatibles* dans le cas contraire et t/u et t/v sont alors dits *séparés* dans t. Il est utile de remarquer que si u et v sont incompatibles, alors :

$$t[u \leftarrow t']/v=t/v \qquad \text{et} \qquad t[u \leftarrow t'][v \leftarrow t'']=t[v \leftarrow t''][u \leftarrow t'],$$

et que si u et v sont compatibles, par exemple $u=vu'$ alors $t/u=t/v/u'$.

Exemple : $F=\{f, g, h, i, X=\{x_1, \ldots,\}$, $a(f)=3$, $a(g)=2$, $a(h)=1$, $a(i)=0$.

L'arbre



est défini par la fonction t : $t(\varepsilon)=g$; $t(1)=f$, $t(2)=h$; $t(11)=h$, $t(12)g$, $t(13)=i$, $t(21)=x_1$; $t(111)=x_1$, $t(121)=x_2$, $t(122)=x_1$.

La *taille* $|t|$ de l'arbre t est le nombre de ses symboles de fonction, c'est-à-dire le cardinal de l'ensemble $\{u \in D(t) \mid t(u) \in F\}$. L'ensemble $V(t)$ des variables de t est l'ensemble $\{x \in X \mid \exists u \in D(t),\ t(u)=x\}$. Le nombre $\theta(x, t)$ d'occurrences de x dans t est le cardinal de l'ensemble $\{u \in D(t) \mid t(u)=x\}$.

Nous allons maintenant donner quelques définitions et résultats dus à Kruskal [12] et Derschowitz [1].

DÉFINITION 1 : On appelle *plongement* la plus petite relation $t \trianglelefteq t'$ telle q :

(1) $$\exists i,\quad t \trianglelefteq t'_i \quad\Rightarrow\quad t \trianglelefteq f(\ldots t'_i \ldots),$$

(2) $$\forall 1 \leqq i \leqq a(f),\quad t_i \trianglelefteq t'_i \quad\Rightarrow\quad f(\ldots t_i \ldots) \trianglelefteq f(\ldots t'_i \ldots).$$

PROPOSITION 1 : *$\trianglelefteq$ est un ordre.* □

Preuve : Par induction sur $|t'|$.

Notons qu'en particulier $t \trianglelefteq f(\ldots t \ldots)$.

Il s'avère intéressant de traduire la définition 1 dans le formalisme des arbres étiquetés, afin d'étudier précisément certaines propriétés du plongement.

PROPOSITION 2 : $t \trianglelefteq t'$ ssi $\exists p : D(t) \to D(t')$ *vérifiant :*

(a) $\forall u \in D(t)$, $t'(p(u)) = t(u)$;

(*b*) $\forall u$, $v \in D(t)$, *u préfixe de* $v \Rightarrow p(u)$ *est préfixe de* $p(v)$;

(*c*) $\forall u \in D(t)$, $\forall 1 \leqq i \leqq a(t(u))$, $p(u).\, i$ *est préfixe de* $p(ui)$.

Preuve : La partie directe est prouvée par induction sur $|t'|$, la réciproque étant laissée au lecteur;

— $|t'| = 0$ évident;

— sinon, $t = f(\ldots t_i \ldots) \trianglelefteq t' = g(\ldots t'_i \ldots)$.

Cas (1) : $\exists i$, $t \trianglelefteq t'_i$. Par hypothèse de récurrence cela équivaut à l'existence $p' : D(t) \to D(t'_i)$ vérifiant les conditions (*a*), (*b*), (*c*) donc à l'existence de $p : D(t) \to D(t')$ telle que $p(u) = i.\, p'(u)$ vérifiant les conditions (*a*), (*b*) et (*c*).

Cas (2) : $f = g$ et $\forall 1 \leqq i \leqq a(f)$, $t_i \trianglelefteq t'_i$. Par hypothèse de récurrence, cela équivaut à l'existence $\forall i$ de $p'_i : D(t_i) \to D(t'_i)$ vérifiant les conditions (*a*), (*b*) et (*c*) donc à l'existence de $p : D(t) \to D(t')$ telle que $p(\varepsilon) = \varepsilon$, $p(iu) = ip'(u)$, $\forall 1 \leqq i \leqq a(f)$, vérifiant les conditions (*a*), (*b*) et (*c*). □

COROLLAIRE 1 : $\forall x \in V(t)$, $t \trianglelefteq t' \Rightarrow \theta(x, t) \leqq \theta(x, t')$ *et si* $t/u = x$ *alors* $\hat{u}$ *est sous-mot de* $\widehat{p(u)}$ *ce que nous notons* $\hat{u} \trianglelefteq \widehat{p(u)}$.

REMARQUE : La notion de sous-mot correspond en effet au plongement des termes monadiques complétés par un marqueur de fin de mot (d'arité 0), ce qui justifie la notation.

DÉFINITION 2 : On appelle ordre de simplification tout ordre strict $<$ clos par :

(1) sous-terme : $t<f(\ldots t\ldots)$;

(2) les opérations de l'algèbre :

$$t<t' \quad \Rightarrow \quad f(\ldots t\ldots)<f(\ldots t'\ldots).$$

LEMME 1 : *La monotonie peut également s'exprimer par :*

$$\left.\begin{array}{ll}\forall i, & t_i \leqq t'_i \\ \text{et} & \\ \exists i, & t_i<t'_i\end{array}\right\} \Rightarrow f(\ldots t_i \ldots)<f(\ldots t'_i \ldots). \qquad (2')$$

Preuve : $(2') \Rightarrow (2)$ est évident. On montre l'inverse en appliquant $a(f)$ fois la propriété (2). □

THÉORÈME 1 [1] : *L'ordre de plongement* $\triangle$ *est le plus petit (au sens de l'inclusion des relations) des ordres de simplification.*

Preuve : Découle directement des définitions. □

COROLLAIRE 2 : *L'ensemble des ordres de simplification muni de l'inclusion est un inf. demi-treillis dont le minimum est le plongement.*

Preuve : Les propriétés (1) et (2) sont stables par intersection. □

THÉORÈME 2 [1] : *Soit* $R=\{g_i \rightarrow d_i,\ i \in I\}$ *un système de règles de réécriture sur* $T(F)$. *Si il existe un ordre de simplification* $<$ *tel que :* $\forall i \in I$, $\forall \sigma$ *substitution,* $\sigma d_i<\sigma g_i$, *alors* R *termine.*

Ce théorème a une grande importance pratique car il signifie — grosso modo — que tout ordre de simplification est bien-fondé. De nombreux ordres de simplification ont été proposés dans la littérature [10, 13, 1, 11 et 9] dont le principe consiste à construire un ordre sur $T(F)$ à partir d'un ordre sur F. Ils permettent (ou ne permettent pas) de vérifier à la demande la terminaison d'un système de règles de réécriture. Tous ces ordres de simplification sont stables pour toute instanciation en particulier par instanciation des règles.

Nous allons maintenant décrire une tentative dans une autre direction, dont nous pensons qu'elle peut être fructueuse : la construction du plus petit ordre de simplification, noté $\underline{\triangle}$, tel que $d \underline{\triangle} g$. La construction, lorsqu'elle est possible, c'est-à-dire lorsque $g \not\underline{\triangle} d$, est très simple. La preuve de correction est toutefois assez délicate.

Nous généralisons ensuite la construction de telle sorte que l'ordre obtenu contienne toutes les paires $(\sigma g, \sigma d)$ pour σ instanciation quelconque fermée.

Nous obtenons ainsi un ordre de simplification stable par instanciation. La possibilité de réaliser cette construction fournit une condition suffisante, syntaxique, pour qu'une règle de réécriture termine. Il serait donc intéressant de généraliser cette construction au cas de systèmes de réécriture à plusieurs règles.

3. CONSTRUCTION D'UN PLUS PETIT ORDRE DE SIMPLIFICATION

Nous allons tout d'abord introduire la relation $\blacktriangle$ dont nous prouverons successivement qu'elle est ordre de simplification, puis qu'elle est le plus petit des ordres de simplification tel que d soit plus petit que g.

3.1. Définition de la relation $\blacktriangle$

DÉFINITION 3 : On dit que t est (g, d)*-plongé* dans t' et nous écrirons $t \underset{g,\,d}{\blacktriangle} t'$ (ou encore $t \blacktriangle t'$ s'il n'y a pas d'ambiguïté) ssi :

(1) $$t \triangle t';$$

(2) $\exists\, u \in D(t), \quad \exists\, u' \in D(t'), \quad \exists\, z \notin V(d) \cup V(t) \cup V(t')$ t. q.

$$(a)\ t/u \trianglelefteq d;$$

$$(b)\ g \trianglelefteq t'u';$$

$$(c)\ t[u \leftarrow z] \underline{\blacktriangle}\, t'[u' \leftarrow z].$$

REMARQUE 1 : Tout ordre de simplification contient le plongement et $\blacktriangle$ ne peut donc être un ordre de simplification si $g \trianglelefteq d$.

REMARQUE 2 : La variable z de la définition 3 est supposée prise en dehors de $V(d) \cup V(t) \cup V(t')$. Il s'ensuit qu'elle n'appartient pas à $V(g)$. En effet, d'après (b) $g \trianglelefteq t'/u'$ et comme $t'/u' \triangle t'$, on a $g \triangle t'$. Donc d'après le corollaire 1, $V(g) \subseteqq V(t')$.

REMARQUE 3 : On pourrait croire que le choix de la variable z a de l'importance. Il n'en est rien car z sera nécessairement absorbée au cours de la preuve $t \blacktriangle t'$ par cas (1). On peut donc en particulier renommer z si nécessaire, comme le prouvera le lemme 3.

3.2. $\blacktriangle$ est un ordre de simplification

Les lemmes qui suivent sont des lemmes techniques nécessaires aux preuves d'irréflexivité et de transitivité. Le corollaire 3 et le lemme 5 expriment toutefois des propriétés profondes de l'ordre $\blacktriangle$. De nombreuses preuves seront faites par induction sur $\blacktriangle$. On notera par $\overset{n}{\underline{\blacktriangle}}$ une preuve de longueur n pour $\underline{\blacktriangle}$.

LEMME 2 : *$y \notin V(d)$ et $t \underline{\blacktriangle} t' \Rightarrow \theta(y, t) \leqq \theta(y, t')$.*

Preuve : Par induction sur $\blacktriangle$:

— Cas (1) : c'est le corollaire 1.

— Cas (2) :

$$\exists u \in D(t), \quad u' \in D(t') \qquad \text{et} \qquad z \notin V(d) \cup V(t) \cup V(t') \quad \text{t. q. :}$$

$$t/u \underline{\triangle} d, \quad g \underline{\triangle} t'u' \qquad \text{et} \qquad t[u \leftarrow z] \underline{\blacktriangle} t'[u' \leftarrow z].$$

$$y \notin V(d) \quad \Rightarrow \quad y \notin t/u \quad \Rightarrow \quad \theta(y, t[u \leftarrow z]) = \theta(y, t) \quad \text{car } z \neq y.$$

Par hypothèse d'induction $\theta(y, t[u \leftarrow z]) \leqq \theta(y, t'[u' \leftarrow z])$.

D'où : $\theta(y, t) \leqq \theta(y, t'[u' \leftarrow z]) \leqq \theta(y, t')$. □

LEMME 3 (*lemme de renommage*): *soit $t \overset{n}{\underline{\blacktriangle}} t'$.*

$$\forall x \notin V(d) \cup V(t) \cup V(t'), \quad \theta(x, t) = \theta(x, t') = 1, \quad t/v = t'/v' = x$$

$$\Rightarrow \quad \forall y \in V(d) \cup V(t) \cup V(t'), \quad t[v \leftarrow y] \overset{n}{\underline{\blacktriangle}} t'[v' \leftarrow y].$$

Preuve : Par induction sur $\blacktriangle$.

— Cas (1) : évident;

— Cas (2) :

$$\exists u, \quad u' \quad \text{et} \quad z, \qquad t/u \underline{\triangle} d, \qquad g \underline{\triangle} t'u' \quad \text{et} \quad t[u \leftarrow z] \overset{n}{\underline{\blacktriangle}} t'[u' \leftarrow z].$$

Mais :

$$x \notin V(d) \quad \Rightarrow \quad x \notin V(t/u) \quad \Rightarrow \quad \theta(x, t[u \leftarrow z]) = 1.$$

Donc : $\theta(x, t'[u' \leftarrow z]) = 1$ d'après le lemme 2 et l'hypothèse. D'où par hypothèse d'induction :

$$t[u \leftarrow z][v \leftarrow y] \overset{n}{\underline{\blacktriangle}} t'[u' \leftarrow z][v' \leftarrow y].$$

Mais u et v ainsi que u' et v' étant des nœuds séparés ($x \notin V(d)$), on en déduit :

$$t[v \leftarrow y][u \leftarrow z] \overset{n}{\underline{\blacktriangle}} t'[v' \leftarrow y][u' \leftarrow z],$$

$$t/u = t[v \leftarrow y]/u \qquad \text{et} \qquad t'/u' = t'[v' \leftarrow y]/u'.$$

D'où par cas (2), $t[v \leftarrow y] \overset{n+1}{\underline{\blacktriangle}} t'[v' \leftarrow y]$. □

LEMME 4 : *Soit* $t \underline{\blacktriangle} t'$, $t/v = t'/v' = y \notin V(d)$ *et* $|\theta(y, t)| = |\theta(y, t'| = 1$. *Alors* $v \underline{\triangle} v'$ *et* $\hat{v} \underline{\triangle} \hat{v}$.

Preuve : Par induction sur $\blacktriangle$.

— Cas (1) : $t \underline{\triangle} t'$, par le corollaire 1.

— Cas (2) :

$$\exists u \in D(t), \quad u' \in D(t') \qquad \text{et} \qquad z \notin V(d) \cup V(t) \cup V(t') \text{ t. q. :}$$

$$t/u \underline{\triangle} d, \quad g \underline{\triangle} t'/u' \qquad \text{et} \qquad t[u \leftarrow z] \underline{\blacktriangle} t'[u' \leftarrow z].$$

$$y \notin V(d) \quad \Rightarrow \quad y \notin t/u \quad \Rightarrow \quad \theta(y, t[u \leftarrow z]) = 1$$

et donc $\theta(y, t'[u' \leftarrow z]) = y$, d'après le lemme 2 et l'hypothèse. Donc $t[u \leftarrow z]/v = t'[u' \leftarrow z]/v' = y$ et le résultat s'obtient par hypothèse d'induction. □

COROLLAIRE 3 : *Si* $t \underline{\blacktriangle} t'$ *par cas* (2) *alors* $u \underline{\blacktriangle} u'$ *et* $\hat{u} \underline{\triangle} \hat{u}'$. *Ce corollaire autorise une définition plus efficace du g. d. plongement.*

PROPOSITION 3 : $g \not\underline{\triangle} d \Rightarrow \blacktriangle$ *est irréflexive.*

Preuve : Par induction sur $\blacktriangle$.

— Cas (1) : $t \triangle t' \Rightarrow t \neq t'$ car $\triangle$ est irréflexif.

— Cas (2) : $t \blacktriangle t'$ par cas (2) : $\exists u, u', z$ t. q. $t/u \underline{\triangle} d$, $g \underline{\triangle}$ $t'u'$ et $u \underline{\triangle} u'$ par le corollaire 3.

Supposons $t = t'$. Alors $u' = u . v$ et t'u' $\underline{\triangle}$ $t'u$. D'où :

$$g \underline{\triangle} t'/u' \underline{\triangle} t'/u = t/u \underline{\triangle} d.$$

Contradiction. □

Le lemme suivant est à la base de la preuve de transitivité.

LEMME 5 : *Soit* $t \overset{n}{\underline{\blacktriangle}} t'$. $\forall v \in D(\tilde{t})$, $t/v \underline{\not\triangle} d \Rightarrow \exists v' \in D(t')$ t. q. :

(*a*) $v \underline{\triangle} v'$;

(*b*) $t/v \underline{\blacktriangle} t'/v'$;

(*c*) $t[v \leftarrow y] \overset{k}{\underline{\blacktriangle}} t'[v' \leftarrow y]$ *avec* $y \in V(d) \cup V(t) \cup V(t')$ *et* $k \leqq n$.

Preuve : Par induction sur $\blacktriangle$.

— Cas (1) : $t \underline{\triangle} t'$ évident.

— Cas (2) :

$$\exists u, \quad u', z, t/u \underline{\triangle} d,$$

$$g \underline{\triangle} t'/u' \qquad \text{et} \qquad t_1 = t[u \leftarrow z] \overset{n}{\underline{\blacktriangle}} t'_1 = t'[u' \leftarrow z] \qquad \text{et } u \underline{\triangle} u'$$

par le corollaire 3. $t/v \underline{\not\triangle} d \Rightarrow u$ n'est pas préfixe de v. Deux cas sont donc possibles : t/u et t/v sont deux sous-termes séparés de t (u et v sont incompatibles) ou bien t/u est un sous-terme strict de t/v (v est préfixe de u).

— Étudions d'abord le cas où u et v sont incompatibles. t/v est alors un sous-terme de t_1 vérifiant $t/v \underline{\not\triangle} d$. Par hypothèse d'induction :

$$\exists v' \in D(t'_1) \text{ t. q. :}$$

(*a*) $v' \underline{\triangle} v$; (*b*) $t_1/v \underline{\blacktriangle} t'_1/v$; (*c*) $t_1[v \leftarrow y] \overset{k}{\underline{\blacktriangle}} t'_1[v' \leftarrow y]$ et $k \leqq n$.

Mais $t_1/v = t/v$ puisque u et v sont incompatibles et de même $t'_1/v' = t'v'$. (*b*) est donc vraie pour t et t'. Utilisant maintenant l'hypothèse $t/u \underline{\triangle} d$ et $g \underline{\triangle} t'u'$ qui peut s'écrire $t[u \leftarrow y]/u \underline{\triangle} d$ et $g \underline{\triangle} t'[v' \leftarrow y]/u'$ puisque u et v d'une part, u' et v' d'autre part par le corollaire 3 sont incompatibles, ainsi que l'hypothèse d'induction (*c*) que nous réécrivons de la même manière en $t[v \leftarrow y]\,[u \leftarrow z] \overset{k}{\underline{\blacktriangle}} t'_1[v' \leftarrow y]\,[u' \leftarrow z]$, nous en déduisons par cas (2) de la définition 3 que $t[v \leftarrow y] \overset{k+1}{\underline{\blacktriangle}} t'[v' \leftarrow y]$ avec $k+1 \leqq n+1$.

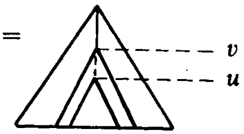
— Étudions maintenant le cas où t/u est un sous-terme strict de t/v (v est préfixe de u et $v \neq u$).

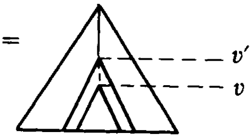
v préfixe propre de $u \Rightarrow z \in t_1/v$. Comme $z \notin V(d)$, $t_1/v \underline{\not\triangle} d$. On peut donc appliquer l'hypothèse d'induction à t_1 et t'_1 :

$$\exists v' \in D(t'_1) \text{ t. q. :}$$

(*a*) $v \underline{\triangle} v'$; (*b*) $t_1/v \underline{\blacktriangle} t'_1/v'$; (*c*) $t_1[v \leftarrow y] \overset{k}{\underline{\blacktriangle}} t'_1[v' \leftarrow y]$ avec $k \leqq n$.

Comme $z \in t_1/v \,\underline{\blacktriangle}\, t'_2/v'$, d'après le lemme 2, $z \in t'_1/v'$, ce qui implique que v' est un préfixe propre de u' (t'_1/v' ne peut être réduit à z, sinon $t_1/v \,\underline{\blacktriangle}\, t'_1/v'$ serait impossible).

$t=$ 

$t'=$ 

Donc $t[v \leftarrow y] = t_1[v \leftarrow y]$ et $t'[v' \leftarrow y] = t'_1[v' \leftarrow y]$ et la condition (c) est donc vraie pour t et t' avec $k \leqq n < n+1$.

Prouvons maintenant la condition (b) : comme v et v' sont des préfixes propres respectivement de u et u', nous avons :

$$t/v/u = t/u \,\underline{\triangle}\, d \qquad \text{et} \qquad g \,\underline{\triangle}\, t'/u' = t'/v'/u'.$$

D'autre part :

$$t/v[u \leftarrow z] = t[u \leftarrow z]/v = t_1/v \,\underline{\blacktriangle}\, t'_1/v' \qquad [\text{hypothèse } (b)],$$

et :

$$t'_1/v' = t'[u' \leftarrow z]/v' = t'v'[u' \leftarrow z].$$

Donc $t/v \,\underline{\blacktriangle}\, t'/v'$ par cas (2) de la définition 3, ce qui achève la preuve. □

LEMME 6 : $g \,\underline{\triangle}\, t \,\underline{\blacktriangle}\, t' \Rightarrow g \,\underline{\triangle}\, t'$.

Preuve : Par cas.

— Cas (1) : $t \,\underline{\triangle}\, t'$. Le lemme est vrai par transitivité de $\underline{\triangle}$.

— Cas (2) : $\exists\, u \in D(t)$, $u' \in D(t')$ t. q. $g \,\underline{\triangle}\, t'/u'$ ce qui implique $g \,\underline{\triangle}\, t'$.

PROPOSITION 4 : $g \not\underline{\triangle}\, d \Rightarrow \underline{\blacktriangle}$ *est transitif.*

Preuve : Soit $t \overset{n}{\underline{\blacktriangle}} t' \overset{m}{\underline{\blacktriangle}} t''$. La preuve que $t \,\underline{\blacktriangle}\, t''$ a lieu par récurrence sur $n+m$.

— $n+m=2$. Alors $t \,\underline{\triangle}\, t' \,\underline{\triangle}\, t'' \Rightarrow t \,\underline{\triangle}\, t'' \Rightarrow t \,\underline{\triangle}\, t''$.

— Supposons la proposition vraie pour $n+m$ et montrons la pour $n+m+1$. Trois cas sont possibles :

Cas 1 :

$$t \,\underline{\triangle}\, t' \overset{m+1}{\underline{\blacktriangle}} t''\,(n=1),$$

$$t \overset{m+1}{\blacktriangle} t'' \quad \Rightarrow \quad \exists v \in D(t'), \qquad w \in D(t'') \text{ t. q. } t'/v \,\underline{\triangle}\, d, \qquad g \,\underline{\triangle}\, t''/w$$

et :

$$t'[v \leftarrow z] \overset{m}{\underline{\blacktriangle}} t''[w \leftarrow z] \qquad \text{avec} \quad z \notin V(d) \cup V(t') \cup V(t''),$$

et $z \notin V(t)$ ce qui est toujours possible d'après le lemme de renommage :

$$t \,\underline{\triangle}\, t' \quad \Rightarrow \quad \exists u \in D(t) \text{ t. q. } t/u \,\underline{\triangle}\, t'/v \qquad \text{et} \qquad t[u \leftarrow z] \,\underline{\triangle}\, t'[v \leftarrow z].$$

Par hypothèse d'induction, on obtient $t[u \leftarrow z] \,\underline{\blacktriangle}\, t''[w \leftarrow z]$ et par transitivité de $\underline{\triangle}$, $t/u \,\underline{\triangle}\, d$ d'où $t \blacktriangle t''$ par cas (2) de la définition 3.

Cas 2 : $t \overset{n+1}{\underline{\blacktriangle}} t' \,\underline{\triangle}\, t''$ $(m=1)$. La preuve est similaire.

Cas 3 : $t \overset{n'}{\underline{\blacktriangle}} t' \overset{m'}{\underline{\blacktriangle}} t''$ avec $n'+m'=n+m+1$, $n' \neq 1$, $m \neq 1$.

Par définition,

$$\exists u \in D(t), \quad v \in D(t') \qquad \text{et} \qquad z \notin V(d) \cup V(t) \cup V(t') \cup V(t'') \text{ t. q. :}$$

$$t/u \,\underline{\triangle}\, d, \quad g \,\underline{\triangle}\, t'/v \qquad \text{et} \qquad t[u \leftarrow z] \overset{n'-1}{\underline{\blacktriangle}} t'[v \leftarrow z].$$

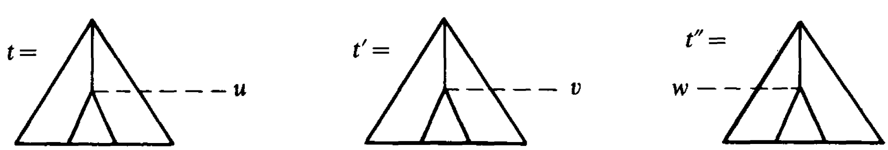
$g \not\underline{\triangle}\, d \Rightarrow t'/v \not\underline{\triangle}\, d$. Donc par le lemme 5 appliqué à t' et t'', $\exists\, w \in D(t'')$ t. q :

$$v \,\underline{\triangle}\, w, \quad t'/v \,\underline{\blacktriangle}\, t''/w \qquad \text{et} \qquad t'[v \leftarrow z] \overset{k'}{\underline{\blacktriangle}} t''[w \leftarrow z] \qquad \text{et} \qquad k' \leqq m'.$$

Par le lemme 5, $g \,\underline{\triangle}\, t'/v \,\underline{\blacktriangle}\, t''/w \Rightarrow g \,\underline{\triangle}\, t''/w$.

Par hypothèse d'induction avec $n'-1+k' \leqq n'-1+m'=n+m$, on obtient :

$$t[u \leftarrow z] \,\underline{\blacktriangle}\, t''[w \leftarrow z].$$



D'où par cas (2) de la définition (3), $t \,\underline{\blacktriangle}\, t''$. □

Si $g \not\underline{\triangle}\, d$, $\blacktriangle$ est donc un ordre strict. Nous allons maintenant prouver qu'il s'agit d'un ordre de simplification.

LEMME 6 DE MONOTONIE : $t \,\underline{\blacktriangle}\, t' \Rightarrow f(\ldots t \ldots) \,\underline{\blacktriangle}\, f(\ldots t' \ldots)$.

Preuve : Par induction sur $\blacktriangle$.

Cas (1) : $t \underline{\triangle} t'$, évident.

Cas (2) :

$$\exists u \in D(t), \quad v \in D(t'), \qquad z \notin V(d) \cup V(t) \cup V(t') \text{ t. q. :}$$
$$t/u \underline{\triangle} d, \qquad g \underline{\triangle} t'/v \quad \text{et} \quad t[u \leftarrow z] \underline{\blacktriangle} t'[v \leftarrow z].$$

Par hypothèse d'induction,

$$f(\ldots, t[u \leftarrow z], \ldots) \underline{\blacktriangle} f(\ldots, t' \ [v \leftarrow z], \ldots).$$

D'où le résultat par cas (2) de la définition 3. □

LEMME 7 (lemme du sous-terme) : $t \underline{\blacktriangle} f(\ldots t \ldots)$.

Preuve : Évident par cas (1) de la définition 3.

PROPOSITION 5 : $\blacktriangle$ *est un ordre de simplification.*

Preuve : Par les propositions 3 et 4, ainsi que les lemmes 6 et 7.

3.3. Minimalité de $\blacktriangle$

Afin de raisonner sur l'ordre de simplification minimal contenant la paire (d, g) (tel que $d \triangle g$), nous en donnons tout d'abord une définition inductive qui exprime qu'il est la fermeture par transitivité et monotonie de la relation $<$ telle que $d < g$ et $t < f(\ldots t \ldots)$.

NOTATION : Soit $\mathscr{S}$ la relation « sous-terme » c'est-à-dire $t \mathscr{S} f(\ldots t \ldots)$.

DÉFINITION 4 : Soit $\triangle$ le plus petit point fixe de l'équation :

$$\triangle = \{d, g\} \cup \mathscr{S} \cup \triangle \cdot \triangle \bigcup_{F} f(\triangle, \ldots, \triangle),$$

où $\triangle \cdot \triangle$ est le carré de la relation $\triangle$: $x \triangle^2 y$ ssi $\exists z$, $x \triangle z$ et $z \triangle y$, et : $(f(x_1, y_1), \ldots, (x_n, y_n)) = (f(x_1, \ldots, x_n), f(y_1, \ldots, y_n))$.

Il est clair que le plus petit point fixe de l'équation précédente existe et que la relation ainsi définie est la plus petite relation transitive, monotone et stable par sous-terme telle que $d \triangle g$. Afin de prouver que c'est le plus petit ordre de simplification tel que $d \triangle g$, il faut donc encore prouver l'irréflexivité de $\triangle$. On se contentera en fait de prouver l'équivalence de $\triangle$ et de $\blacktriangle$. Il s'ensuivra alors que $\triangle$ est irréflexif et que $\triangle$ et $\blacktriangle$ sont deux définitions équivalentes du plus petit ordre de simplification contenant la paire (d, g).

Montrons tout d'abord deux propriétés de $\dot{\triangle}$.

LEMME 8 : *Soit* $t \mathbin{\dot{\triangle}} t'$, $u \in D(t)$, $v \in D(t')$ t. q. :

$$t/u = t'/v = z \notin V(g) \cup V(d) \qquad \text{et} \qquad \theta(z, t) = \theta(z, t') = 1.$$

Alors :

$$\forall t'', \quad t[u \leftarrow t''] \mathbin{\dot{\triangle}} t'[v \leftarrow t''].$$

Preuve : Par induction sur $\dot{\triangle}$.

Cas (1) : $t = d$ et $t' = g$. Impossible car $z \notin V(d) \cup V(g)$.

Cas (2) : $t \mathbin{\dot{\triangle}} f(\ldots t \ldots) = t'$. $t[u \leftarrow z] \mathbin{\dot{\triangle}} f(\ldots t[u \leftarrow z] \ldots)$, car $\dot{\triangle}$ est stable par sous-terme (remarquons que l'on a implicitement utilisé l'hypothèse $\theta(z, t) = \theta(z, t') = 1$).

Cas (3) : $t \mathbin{\dot{\triangle}} t' \Rightarrow t[u \leftarrow t''] \mathbin{\dot{\triangle}} t'[v \leftarrow t'']$ par l'hypothèse de récurrence d'où $f(\ldots t[u \leftarrow t''] \ldots) \mathbin{\dot{\triangle}} f(\ldots t'[v \leftarrow t''] \ldots)$ par monotonie.

Cas (4) : $t \mathbin{\dot{\triangle}} t_1 \mathbin{\dot{\triangle}} t'$.

$$t[u \leftarrow t''] \mathbin{\dot{\triangle}} t_1[w \leftarrow t''] \qquad \text{et} \qquad t_1[w \leftarrow t''] \mathbin{\dot{\triangle}} t'[v \leftarrow t'']$$

par hypothèse de récurrence.

D'où $t[u \leftarrow t''] \mathbin{\dot{\triangle}} t'[v \leftarrow t'']$ par transitivité. □

LEMME 9 : $t' \mathbin{\dot{\triangle}} t''$ et $\theta(x, t) = 1$ et $t/u = x \Rightarrow t[u \leftarrow t'] \mathbin{\dot{\triangle}} t[u \leftarrow t'']$.

Preuve : C'est une propriété uniquement liée à la monotonie. On la montre par récurrence sur $|t|$:

— $|t| = 0 \Rightarrow t = x$ et la propriété est vraie;

— $t = f(\ldots t_i \ldots)$ et $x \in V(t_i)$. Alors $u = i.u'$, $v(x, t_i) = 1$. Donc par hypothèse d'induction, $t_i[u' \leftarrow t'] \mathbin{\dot{\triangle}} t_i[u' \leftarrow t'']$. Mais $\dot{\triangle}$ étant monotone, $f(\ldots t_i[u' \leftarrow t'] \ldots) \mathbin{\dot{\triangle}} f(\ldots t_i[u' \leftarrow t''] \ldots)$.

On concluera en remarquant que :

$$f(\ldots t_i[u' \leftarrow t] \ldots) = f(\ldots t_i \ldots)[u \leftarrow t].$$

LEMME 10 : $\triangle \subseteqq \dot{\triangle}$.

Preuve : C'est une conséquence de la transitivité, de la monotonie et de la stabilité par sous-terme. On la prouve par récurrence sur $|t'|$ si $t \mathbin{\dot{\triangle}} t'$.

— $|t'| = 0$. Évident.

— $t = f(\ldots t_i \ldots) \mathbin{\underline{\triangle}} t' = g(\ldots t'_i \ldots)$. Distinguons deux cas :

Cas (1) : $t \mathbin{\underline{\triangle}} t'_i$. Par hypothèse de récurrence, $t \mathbin{\underline{\dot{\triangle}}} t'_i$.

Par sous-terme $t'_i \mathbin{\dot{\triangle}} t'$ et par transitivité $t \mathbin{\dot{\triangle}} t'$.

Cas (2) : $f = g$ et $\forall\, 1 \leqq i \leqq a(f)$, $t_i \mathbin{\triangle} t'_i$. Par hypothèse de récurrence, $t_i \mathbin{\underline{\dot{\triangle}}} t'_i$ et par le lemme 1, $t \mathbin{\dot{\triangle}} t'$. □

On est maintenant en mesure de prouver l'identité de $\triangle$ et de $\blacktriangle$. Introduisons pour cela une troisième relation, notée $\lessdot$ ainsi définie, $t \lessdot t'$ *ssi :*

$$(1) \qquad t \triangle t',$$

$$(2) \quad \exists u \in D(t), \quad v \in D(t'), \qquad z \notin V(d) \cup V(t) \cup V(t') \text{ t. q. :}$$
$$t/u \trianglelefteq d, \qquad g \trianglelefteq t'v, \qquad t[u \leftarrow z] \triangle t'[u \leftarrow z].$$

LEMME 11 : $\lessdot \nsubseteq \triangle$.

Preuve : Par cas sur la définition de $\lessdot$.

Cas (1) : $t \triangle t'$. Évident.

Cas (2) :

$$\exists u, \quad v, z \text{ t. q. } t/u \trianglelefteq d \qquad \text{et} \qquad g \trianglelefteq t'v \qquad \text{et} \qquad t[u \leftarrow z] \triangle t'[u \leftarrow z].$$

Par définition de $\triangle$ et par le lemme 10 : $t/u \trianglelefteq d \trianglelefteq g \trianglelefteq t'v$, d'où par transitivité de $\triangle$: $t/u \triangle t'v$:

$$z \notin V(t) \cup V(t') \quad \Rightarrow \quad \theta(z, t[u \leftarrow z]) = \theta(z, t'[v \leftarrow z]) = 1.$$

Donc par le lemme 8, $t[u \leftarrow t/u] \triangle t'[v \leftarrow t/u]$ puis par le lemme 9,

$t'[v \leftarrow t/u] \triangle t'[v \leftarrow t'v]$ puis par transitivité :

$t[u \leftarrow t/u] \triangle t'[v \leftarrow t'v]$ soit $t \triangle t'$. □

Remplaçons maintenant $\triangle$ par $\lessdot$ dans la définition de $\lessdot$: on obtient exactement l'ordre $\blacktriangle$. Pour montrer que $\blacktriangle$ vérifie également le lemme 11, on va donc classiquement construire une suite d'ordres convergeant vers $\blacktriangle$ et ayant tous les propriétés du lemme 11.

Posons $\lessdot = \triangle$ et soit $\lessdot_{i+1}$ l'ordre ainsi défini, $t \lessdot_{i+1} t'$ ssi :

$$(1) \qquad t \triangle t',$$

$$(2) \quad \exists u \in D(t), \quad v \in D(t') \quad \text{et} \quad z \notin V(d) \cup V(t) \cup V(t') \text{ t. q. :}$$
$$t/u \trianglelefteq d, \ g \trianglelefteq t'v \qquad \text{et} \qquad t[u \leftarrow z] \lessdot_i t'[v \leftarrow z].$$

LEMME 12 : $\forall i \geqq 0, \lessdot_{i+1} \subseteqq \lessdot_i$.

Preuve : Par récurrence sur i.

— $i = 0$ vrai par le lemme 11 car $\lessdot_i = \lessdot$.

— Cas général : par cas sur la définition $\lessdot_{i+2}$:

Cas (1) : évident.

Cas (2) : par hypothèse de récurrence, $t[u \leftarrow z] \lessdot_{i+1} t'[v \leftarrow z]$ implique $t[u \leftarrow z] \lessdot_i t'[v \leftarrow z]$ d'où $t \lessdot_{i+1} t'$ par définition. □

LEMME 13 : $\bigcap_{i \in N} <_i = \lim_{i \to \infty} <_i = \blacktriangle$.

Preuve : Par le lemme 12, $\bigcap_{i \in N} <_i = \lim_{i \to \infty} <_i$. La seconde partie, $\lim_{i \to \infty} <_i = \blacktriangle$, s'obtient par continuité de $\blacktriangle$ et par construction des $<_i$.

D'où, en utilisant les lemmes 11, 12 et 13 :

COROLLAIRE 4 : $\blacktriangle \subseteqq \triangle$.

Pour conclure, il suffit maintenant de remarquer que $\blacktriangle$ est un ordre de simplification contenant la paire (d, g) : il est donc fermé par monotonie, transitivité et il contient la relation $\mathscr{S}$ de sous-terme. Or $\triangle$ est la plus petite relation ayant ces propriétés. Donc $\triangle \subseteqq \blacktriangle$. D'où :

THÉORÈME 3 : $\triangle$ *est le plus petit ordre de simplification tel que :*

$$d \blacktriangle g \quad \text{ssi} \quad g \not\triangle d.$$

4. CONSTRUCTION D'UN PLUS PETIT ORDRE DE SIMPLIFICATION STABLE PAR INSTANCIATION

La généralisation ne pose guère de problème : il s'agit tout simplement d'introduire dans la définition 3, la possibilité d'une instanciation :

DÉFINITION 5 : On dit que t est $(\Sigma - g,\ d)$*-plongé* dans t' et nous écrirons $t \blacktriangle_\Sigma t'$ ssi $\exists\, \theta \in \Sigma$ t. q. t soit $(\theta g,\ \theta d)$-plongé dans t'.

LEMME 14 : $\blacktriangle_\Sigma$ *est fermé par instanciation.*

Preuve : $t \blacktriangle_\Sigma t' \Rightarrow \exists\, \theta \in \Sigma,\ t \underset{\theta g,\, \theta d}{\blacktriangle} t'$. Montrons que cela implique $\sigma t \underset{\sigma\theta g,\, \sigma\theta d}{\blacktriangle} \sigma t'$, pour toute substitution σ. La preuve se fait par induction sur la preuve que $t \underset{\theta g,\, \theta d}{\blacktriangle} t'$.

Cas (1) : car $\triangle$ est fermé par instanciation.

Cas (2) :

$$\exists\, u \in D(t),\quad \exists\, u' \in D(t'),\quad \exists\, z \notin V(\theta d) \cup V(t) \cup V(t') \text{ t. q. :}$$

$$t/u \,\underline{\triangle}\, \theta g \,\underline{\triangle}\, t'/u' \qquad \text{et} \qquad t[u \leftarrow z] \underset{\theta g,\, \theta d}{\blacktriangle} t'[u' \leftarrow z].$$

Par le lemme 3 de renommage, on peut choisir $z \notin V(\sigma)$. $\underline{\triangle}$ étant stable par instanciation, on obtient :

$$\sigma(t/u) = \sigma t/u \,\underline{\triangle}\, \sigma\theta d \quad \text{et} \quad \sigma\theta g \,\underline{\triangle} \quad \sigma(t'u') = \sigma t'u'.$$

Par hypothèse de récurrence, on obtient : $\sigma(t[u \leftarrow z]) \underset{\sigma\theta g,\ \sigma\theta d}{\underline{\blacktriangle}} \sigma(t'[u' \leftarrow z])$ d'où, comme $z \notin V(\sigma)$:

$$\sigma t[u \leftarrow z] \underset{\sigma\theta g,\ \sigma\theta d}{\underline{\blacktriangle}} \sigma t'[u' \leftarrow z].$$

Donc : $\sigma t \underset{\sigma\theta g,\ \sigma\theta d}{\blacktriangle} \sigma t'$. □

Il nous reste maintenant deux problèmes à résoudre : le $(\Sigma\text{-}g, d)$-plongement est-il minimal d'une part, existe-t-il d'autre part ?

La minimalité provient de la minimalité de $\underset{g,\ d}{\blacktriangle}$: soit $<$ un ordre de simplification contenant (g, d) et clos par instanciation. Il contient $\underset{\theta g,\ \theta d}{\blacktriangle}$ pour tout θ c'est-à-dire $\blacktriangle_\Sigma$.

La preuve d'existence de $\blacktriangle_\Sigma$ est un peu plus délicate : il s'agit de savoir à quelle condition il n'existe aucun σ tel que $\sigma g \underline{\blacktriangle} \sigma d$, de sorte que $\underset{\sigma g,\ \sigma d}{\blacktriangle}$ existe pour tout σ.

LEMME 15 : $\sigma g \underline{\triangle} \sigma d \Rightarrow \exists d' \underline{\triangle} d$ t. q. $\sigma g = \sigma d'$.

Preuve : Par récurrence sur $|\sigma g| + |\sigma d|$.

— $|\sigma g| + |\sigma d| = 0$ alors $g = x$, $d = x$ et $d' = x$;

— $|\sigma g| + |\sigma d| = 1$ impossible car $x \not\underline{\triangle} a$ si $x \neq a$;

— cas général : $g = f(\ldots g_i \ldots)$ et $d = h(\ldots d_i \ldots)$.

(*a*) $f = h$ et $\sigma g \underline{\triangle} \sigma d \Rightarrow \sigma g_i \underline{\triangle} \sigma d_i$ pour chaque i. Par hypothèse de récurrence :

$$\exists d_i' \underline{\triangle} d_i \text{ t. q. } \sigma g_i = \sigma d_i'$$

d'où :

$$\sigma f(\ldots g_i \ldots) = \sigma f(\ldots d_i' \ldots) \quad \text{et} \quad f(\ldots d_i' \ldots) \underline{\triangle} f(\ldots d_i \ldots).$$

(*b*) $\exists i$, $\sigma g \underline{\triangle} \sigma d_i$. Par hypothèse de récurrence, $\exists d' \triangle d_i$ t. q. $\sigma g = \sigma d'$. Mais $d' \underline{\triangle} d_i \Rightarrow d' \triangle d$. □

Cette condition est évidemment décidable, car le nombre de termes d' plongés dans d est évidemment fini. Il suffira donc pour chacun d'eux de tester s'il s'unifie avec g. Cette méthode n'est certainement pas la meilleure pour trouver des σ s'ils existent tels que $\sigma g \underline{\triangle} \sigma d$. Une autre solution pourrait consister à adapter à ce problème les algorithmes d'unification connus. Nous obtenons finalement :

THÉORÈME 4 : $\blacktriangle_\Sigma$ *est le plus petit ordre de simplification stable par instanciation tel que* $d \blacktriangle_\Sigma g$ ssi $\nexists d' \trianglelefteq d$ *tel que* d' *s'unifie avec* g, *d'où le :*

COROLLAIRE : *La règle* $g \to d$ *termine si* g *ne s'unifie avec aucun terme plongé dans* d.

5. CONCLUSION

Le résultat obtenu est intéressant non pas tant par son importance pratique (quoique la décidabilité de la terminaison d'un système à une règle soit à notre connaissance un problème ouvert) que par les perspectives qu'il ouvre en cas de généralisation. Le problème qui se pose est tout simplement celui de la compatibilité de la relation initiale et du plongement.

Dans le cas d'une règle, il faut vérifier que $\forall \sigma \in \Sigma$, $(\sigma g, \sigma d) \cap \triangle = \emptyset$.

Dans le cas général, il faudra bien vérifier que $\forall \sigma \in \Sigma$, $\sigma R \cap \triangle = \emptyset$, si R est la relation initiale. Mais d'une part, il n'est pas sûr que cette condition suffise à assurer l'existence d'un plus petit ordre de simplification contenant R et stable par instanciation. D'autre part, il faut encore s'assurer que cette condition est décidable. Enfin, si elle l'est, il sera nécessaire d'étudier sa complexité avant d'affirmer son intérêt pratique.

BIBLIOGRAPHIE

1. N. DERSHOWITZ, *Orderings for Term Rewriting Systems*, Proc. 20th Symposium on Foundations of Computer Science, 1979, p. 123-131, also in TCS 17-3, 1982.
2. DONER, *Trees Acceptors and Some of Their Applications*, J. Computer System Science, vol. 4, 1970, p. 406-451.
3. J. V. GUTTAG, *The Specification and Application to Programming of Abstract Data Types*, Ph. D. Thesis, University of Toronto, 1975.
4. G. HUET et J. M. HULLOT, *Proofs by Induction in Equational Theories with Constructors*, Proc. 21th Symposium on Foundation of Computer Science, 1980.
5. G. HUET et D. S. LANKFORD, *On the Uniform Halting Problem for Term Rewriting Systems*, Rapport Laboria, n° 283, I.N.R.I.A., 1978.
6. G. HUET et D. C. OPPEN, *Equations and Rewrite Rules: a Survey*, in *Formal Languages: Perspectives and Open Problems*, R. BOOK, éd., Academic Press, 1980.
7. J. HSIANG et N. DERSHOWITZ, *Rewrite Methods for Clausal and non Clausal Theorem Proving*, Proc. 10th I.C.A.L.P., 1983.
8. J. P. JOUANNAUD, C. KIRCHNER et H. KIRCHNER, *Incremental Construction of Unification Algorithms in Equational Theories*, Proc. 10th I.C.A.L.P., 1983.

9. J. P. JOUANNAUD, P. LESCANNE et F. REINIG, *Recursive Decomposition Ordering*, in *Formal Description of Programming Concepts* 2, D. BJORNER, éd., North Holland, 1982.

10. D. KNUTH et P. BENDIX, *Simple word Problems in Universal Algebras*, in *Computational Problems in Abstract Algebra*, J. LEECH, éd., Pergamon Press, 1970.

11. KAMIN et J. J. LEVY, *Attempts for Generalizing the Recursive Path Ordering*, to be published.

12. J. B. KRUSKAL, *Well Quasi Ordering, the Tree Theorem and Vazsonyi's conjecture*, Trans. Amer. Math. Soc., vol. 95, 1960, p. 210-225.

13. D. PLAISTED, *A Recursively Defined Ordering for Proving Terminaison of Term Rewriting Systems*, Report 78-943, University of Illinois, 1978.