

# Algorithms for equivalence and reduction to minimal form for a class of simple recursive equations\*

B. Courcelle, G. Kahn, J. Vuillemin  
 IRIA-LABORIA, 78150 ROCQUENCOURT, France

**Résumé.** Nous décrivons ici un algorithme pour décider de l'équivalence dans un domaine dont les éléments sont définis par des systèmes d'équations au point fixe. Cet algorithme est utilisé pour trouver diverses représentations minimales des éléments du domaine.

## Introduction.

Plusieurs travaux récents, par exemple [4], [8], [9], [11] utilisent la notion d'équation au point fixe pour exprimer la sémantique des langages de programmation. Nous étudions ici un "pur langage de points fixes" dans lequel aucune fonction de base n'est interprétée, ce qui élimine en particulier l'opérateur conditionnel *si-alors-sinon*.

Au sujet des équations au point fixe, dont un exemple typique est l'équation  $X = f(X, g(X))$  nous nous posons un certain nombre de questions, par exemple:

- est-ce que  $X = f(X, g(X))$  est équivalente à  $Y = f(f(Y, g(Y)), g(Y))$ ?
- existe-t-il une équation plus simple équivalente à

$$Z = g(g(Z))$$

ou au système

$$\begin{cases} X = f(Y, g(X), Y) \\ Y = g(X) \end{cases}$$

- la variable  $X$  définie par le système

$$\begin{cases} X = f(X, Y) \\ Y = g(X, Y) \end{cases}$$

pourrait-elle l'être par une seule équation?

Dans la première partie, nous étudions les *équations récursives simples*. Nous montrons l'existence d'une *forme canonique* caractérisant une classe d'équivalence d'équations. La forme canonique minimise la taille de l'équation dans sa classe d'équivalence.

La deuxième partie, indépendante de la première (sauf en ce qui concerne les définitions de la syntaxe

**Abstract.** In this paper, we describe an algorithm for deciding equivalence in a domain whose objects are defined by uninterpreted fixed-point equations. The algorithm is then applied to finding a minimal representation of those objects or to removing mutual recursion whenever possible.

## Introduction.

Many recent works, for example [4],[8],[9],[11] use the notion of fixpoint equations to express semantics of programming languages. We study here a "pure language of fixpoints" with uninterpreted function symbols, which omits in particular the conditional operator *if-else-elseif*.

In the study of fixpoint equations, of which a typical example is the equation  $X = f(X, g(X))$ , we ask ourselves certain questions, for example:

- is  $X = f(X, g(X))$  equivalent to  $Y = f(f(Y, g(Y)), g(Y))$ ?
- does there exist a simpler equation equivalent to

$$Z = g(g(Z))$$

or to the system

$$\begin{cases} X = f(Y, g(X), Y) \\ Y = g(X) \end{cases}$$

- can the variable  $X$  defined by the system

$$\begin{cases} X = f(X, Y) \\ Y = g(X, Y) \end{cases}$$

be represented by a single equation?

In the first section, we study *simple recursive equations*. We show the existence of *canonical forms* characterizing a class of equivalent equations. The canonical form minimizes the size of the equation in its equivalence class.

The second section, independent of the first (except where definitions of syntax and semantics are con-

\*This document presents a translation for historical perspective of the paper: [ B. Courcelle, G. Kahn, and J. Vuillemin. Algorithmes d'équivalence et de réduction à des expressions minimales dans une classe d'équations récursives simples. In Jacques Loeckx, editor, *Automata, Languages and Programming*, volume 14 of *Lecture Notes in Computer Science*, pages 200–213. Springer Verlag, August 1974. ] The left column is the original French with minor typographical changes, see Appendix B; the right column is intended to be a faithful English translation. See Appendix A for a commentary by B. Courcelle.

et de la sémantique) étudie les mêmes problèmes pour les *systèmes d'équations récursives*, dégage une *notion de forme canonique* mais qui ne minimise par le *nombre d'équations du système*. Ce dernier problème est ensuite abordé et résolu.

Ce travail est motivé par des questions aussi différentes que l'étude des définitions récursives des modes en Algol 68 (Lewis-Rosen [6]), la formalisations des preuves d'équivalences entre programmes parallèles (Kahn [4]) ou la recherche de sous-théories décidables dans la théorie des schémas de programmes.

D'autres auteurs (J. Engelfriet [3], C. Pair [10], J. Kral [5]) ont obtenu indépendamment des résultats voisins dans un cadre syntaxique et sémantique différent.

## I. Equations Récursives Simples

Pour la clarté de l'exposé, nous commençons par définir les équations au point-fixe à une seule inconnue.

### 1.1 Syntaxe

Les termes sont construits à l'aide de symboles de fonction  $\{F, G, H, \dots\}$  ayant chacun une certaine arité, et du symbole de variable  $X$  par les règles:

- (i) Les symboles de fonction d'arité 0, ou *constants* et le symbole  $X$  sont des termes.
- (ii) Si  $T_1, T_2, \dots, T_n$  sont des termes et  $F$  est un symbole de fonction d'arité  $n$ , alors  $F(T_1, T_2, \dots, T_n)$  est un terme.

Si  $T$  est un terme,  $X = T$  est une équation au point fixe.

Pour la suite il sera utile de munir l'ensemble des termes d'une structure d'ordre partiel  $\leq$  définie par les règles suivantes:

- (i) Pour tout terme  $T$ , nous avons  $X \leq T$ .
- (ii) Si  $T_1 \leq T'_1, T_2 \leq T'_2, \dots, T_n \leq T'_n$  alors  $F(T_1, \dots, T_n) \leq F(T'_1, \dots, T'_n)$ .

En d'autres termes  $T_1 \leq T_2$  si et seulement si  $T_2$  est le résultat de la substitution dans  $T_1$  de certains termes pour certaines occurrences de  $X$ .

#### Exemple:

$$F(X, G(X, X)) \leq F(H(X), G(X, H(X)))$$

mais  $F(X, G(X, X)) \not\leq F(X, H(X))$ .

Etant donnés deux termes  $T$  et  $T'$  nous pouvons définir un minorant  $\Sigma(T, T')$  tel que  $\Sigma(T, T') \leq T$  et  $\Sigma(T, T') \leq T'$  de la façon suivante:

cerned) studies the same problems for *systems of recursive equations*, deriving a *notion of canonical form* that minimizes the *number of equations in the system*. This last problem is then confronted and resolved.

This work is motivated by a variety of questions such as the study of recursive datatype definitions in Algol 68 (Lewis-Rosen [6]), formalizing equivalence proofs of parallel programs (Kahn [4]) and research into decidable sub-theories of the theory of program schemas.

Other authors (J. Engelfriet [3], C. Pair [10], J. Kral [5]) have independently obtained related results in syntactically and semantically different frameworks.

## I. Simple Recursive Equations

For clarity of exposition, we start by defining fixpoint equations in a single unknown.

### 1.1 Syntax

Terms are constructed from function symbols  $\{F, G, H, \dots\}$  each having some arity, and from the variable symbol  $X$  by the rules:

- (i) Function symbols of arity 0 (or *constants*) and the symbol  $X$  are terms.
- (ii) If  $T_1, T_2, \dots, T_n$  are terms and  $F$  is a function symbol of arity  $n$ , then  $F(T_1, T_2, \dots, T_n)$  is a term.

If  $T$  is a term,  $X = T$  is a fixpoint equation.

For what follows, it is useful to define a partial order  $\leq$  on the set of terms by the following rules:

- (i) For all terms  $T$ , we have  $X \leq T$ .
- (ii) If  $T_1 \leq T'_1, T_2 \leq T'_2, \dots, T_n \leq T'_n$  then  $F(T_1, \dots, T_n) \leq F(T'_1, \dots, T'_n)$ .

For other terms  $T_1 \leq T_2$  if and only if  $T_2$  is the result of substituting in  $T_1$  some terms for some occurrences of  $X$ .

#### Exemple:

$$F(X, G(X, X)) \leq F(H(X), G(X, H(X)))$$

but  $F(X, G(X, X)) \not\leq F(X, H(X))$ .

When given two terms  $T$  and  $T'$  we can define a reduction  $\Sigma(T, T')$  such that  $\Sigma(T, T') \leq T$  and  $\Sigma(T, T') \leq T'$  in the following manner:

$$\begin{aligned} \Sigma(T, X) &= \Sigma(X, T) = X \\ \Sigma(F(T_1, T_2, \dots, T_n), G(T'_1, T'_2, \dots, T'_m)) &= X \quad \text{if } F \neq G \\ \Sigma(F(T_1, T_2, \dots, T_n), F(T'_1, T'_2, \dots, T'_n)) &= F(\Sigma(T_1, T'_1), \dots, \Sigma(T_n, T'_n)) \end{aligned}$$

#### Exemple:

$$\Sigma(F(X, G(X)), F(G(X), H(X))) = F(X, X)$$

#### Exemple:

$$\Sigma(F(X, G(X)), F(G(X), H(X))) = F(X, X)$$

**Notation:** Pour un terme  $T$  dans lequel figure  $m$  fois la lettre  $X$  nous désignons par  $T\{T_1, \dots, T_m\}$  le résultat de la substitution du terme  $T_i$  à la  $i^{\text{ème}}$  occurrence de  $X$  dans  $T$ , pour chaque  $i$  dans  $[1, m]$ .

**Lemme 1:** Pour tous termes  $T$  et  $T'$  il existe des termes  $T_1, \dots, T_m$  et  $T'_1, \dots, T'_m$  tels que:

$$\begin{aligned} T &= \Sigma(T, T')\{T_1, \dots, T_m\} \\ T' &= \Sigma(T, T')\{T'_1, \dots, T'_m\} \end{aligned}$$

## 1.2 Sémantique

(a) Nous interpréterons les équations au point fixe sur un *domaine*  $\mathcal{D}$  qui doit satisfaire les propriétés suivantes:

(i) L'ensemble  $\mathcal{D}$  est muni d'une relation d'ordre partiel  $\subseteq$ . Nous notons  $\equiv$  la relation d'équivalence induite.

(ii) Il existe dans  $\mathcal{D}$  un plus petit élément  $\perp$ .

(iii) Toute chaîne croissante dénombrable possède une borne supérieure.

Cette structure, qui est légèrement moins restrictive que celle de treillis complet dont se sert Scott [11], est aussi utilisée dans [7],[8] et [12].

(b) - A chaque symbole de constante  $C$  est associé un élément  $C$  dans  $\mathcal{D}$ .

- A chaque symbole de fonction  $F$  d'arité  $n$  est associée une application  $f$  de  $\mathcal{D}^n$  dans  $\mathcal{D}$  qui est *monotone* et *continue* en chacun de ses arguments (Voir [7] pour une définition de ces notions).

- A tout terme  $T$  se trouve alors associée de façon naturelle une application  $t$  de  $\mathcal{D}$  dans  $\mathcal{D}$  et à l'équation au point fixe  $X = T$  on associera le plus petit point fixe de  $t : \mathcal{D} \rightarrow \mathcal{D}$  que l'on appellera encore plus petit point fixe de l'équation. Il est défini comme la borne supérieure dans  $\mathcal{D}$  de l'ensemble  $\{t^n(\perp) \mid n \geq 0\}$ . (Voir encore Milner [7]).

**Convention:** Nous utiliserons systématiquement des lettres majuscules pour désigner les objets syntaxiques et les minuscules correspondantes pour désigner les objets sémantiques associés.

(c) Nous allons maintenant construire une *interprétation canonique* de notre langage, qui joue le rôle de l'univers de Herbrand pour les théories du premier ordre. Le domaine  $\mathcal{D}$  de l'interprétation canonique est constitué par l'ensemble des suites *infinies de termes*  $\{T_i \mid i \in \mathbb{N}\}$  que  $T_i \leq T_{i+1}$  pour tout  $i \geq 0$  construits sur la variable  $X$ .

Nous définissons maintenant une relation d'ordre  $\subseteq$  sur  $\mathcal{D}$ : si  $\tau = \{T_i \mid i \in \mathbb{N}\}$  et  $\tau' = \{T'_i \mid i \in \mathbb{N}\}$  sont deux éléments de  $\mathcal{D}$ , alors  $\tau \subseteq \tau'$  si et seulement si  $\forall i \exists j T_i \leq T'_j$ .

L'élément minimal de  $\mathcal{D}$  est  $\perp = \{T_i \mid \forall i \in \mathbb{N}, T_i = X\}$ . Toute chaîne  $\tau_1 \subseteq \tau_2 \subseteq \dots \subseteq \tau_i \subseteq \dots$  dans

**Notation:** For a term  $T$  in which the letter  $X$  occurs  $m$  times, we denote by  $T\{T_1, \dots, T_m\}$  the result of substituting the term  $T_i$  for the  $i^{\text{th}}$  occurrence of  $X$  in  $T$ , for each  $i$  in  $[1, m]$ .

**Lemma 1:** For all terms  $T$  and  $T'$  there exist terms  $T_1, \dots, T_m$  and  $T'_1, \dots, T'_m$  such that:

$$\begin{aligned} T &= \Sigma(T, T')\{T_1, \dots, T_m\} \\ T' &= \Sigma(T, T')\{T'_1, \dots, T'_m\} \end{aligned}$$

## 1.2 Semantics

(a) We interpret fixpoint equations in a *domain*  $\mathcal{D}$  which must satisfy the following requirements:

(i) The set  $\mathcal{D}$  is provided with a partial order relation  $\subseteq$ . We write  $\equiv$  for the induced equivalence relation.

(ii) There exists in  $\mathcal{D}$  a least element  $\perp$ .

(iii) All denumerable ascending chains have an upper bound.

This structure, which is slightly less restrictive than that of complete lattices, is after Scott [11], and was also used in [7],[8] and [12].

(b) - With each constant symbol  $C$  is associated with an element  $C$  in  $\mathcal{D}$ .

- With each function symbol  $F$  of arity  $n$  is associated a map  $f$  from  $\mathcal{D}^n$  to  $\mathcal{D}$  which is *monotone* and *continuous* in each of its arguments (See [7] for a definition of these notions).

- To all terms  $T$  there is associated in a natural way a map  $t$  from  $\mathcal{D}$  to  $\mathcal{D}$ , and to the fixpoint equation  $X = T$  we associate the least fixpoint of  $t : \mathcal{D} \rightarrow \mathcal{D}$  which we again call the least fixpoint of the equation. It is defined as the upper bound in  $\mathcal{D}$  of the set  $\{t^n(\perp) \mid n \geq 0\}$ . (See again Milner [7]).

**Convention:** We systematically use upper-case letters to designate syntactic objects, and lower-case letters to designate the associated semantic objects.

(c) We are now going to construct a *canonical interpretation* of our language, which plays the role of the Herbrand universe for first-order theories. The domain  $\mathcal{D}$  of the canonical interpretation is comprised of the set of infinite sequences of terms  $\{T_i \mid i \in \mathbb{N}\}$  constructed on the variable  $X$  such that  $T_i \leq T_{i+1}$  for all  $i \geq 0$ .

We now define an order relation  $\subseteq$  on  $\mathcal{D}$ : if  $\tau = \{T_i \mid i \in \mathbb{N}\}$  and  $\tau' = \{T'_i \mid i \in \mathbb{N}\}$  are two elements of  $\mathcal{D}$ , then  $\tau \subseteq \tau'$  if and only if  $\forall i \exists j T_i \leq T'_j$ .

The minimal element of  $\mathcal{D}$  is  $\perp = \{T_i \mid \forall i \in \mathbb{N}, T_i = X\}$ . All chains  $\tau_1 \subseteq \tau_2 \subseteq \dots \subseteq \tau_i \subseteq \dots$  in which

laquelle  $\tau_i = \{T_j^i \mid j \in \mathbb{N}\}$  admet une borne supérieure  $\tau = \bigcup_{i \in \mathbb{N}} \tau_i = \{T_i^i \mid i \in \mathbb{N}\}$  où pour tout  $i$   $T_i^i = T_{n_i}^i$ , la suite  $n_i$  choisie de façon que  $\forall i (n_i \geq i \text{ et } T_{n_i}^i \leq T_{n_i+1}^{i+1})$  (Il existe une telle suite puisque les  $\tau_i$  forment une chaîne croissante).

Une constante  $C$  est alors interprétée par la suite:

$$C = \{T_i \mid T_i = C, i \in \mathbb{N}\}$$

L'interprétation  $f$  du symbole  $n$ -aire  $F$  associe aux suites  $\{T_k^i \mid k \in \mathbb{N}\}$  pour  $i$  dans  $[1, n]$  la suite  $\{F(T_k^1, T_k^2, \dots, T_k^n) \mid k \in \mathbb{N}\}$ . Il est aisé de vérifier que nous avons bien là interprétation légitime. (Plus rigoureusement, le domaine d'interprétation que nous avons considéré est  $\mathcal{D}/\equiv$ ).

Notons  $P\{A/X\}$  le résultat de la substitution dans un terme  $P$  de  $A$  à toutes les occurrences de  $X$ . On vérifiera que dans notre interprétation canonique  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  avec

$$T^1 = T \text{ et } T^{i+1} = T\{T_i/X\}$$

L'intérêt de l'interprétation canonique vient du lemme suivant:

**Lemme 2:** *Deux équations au point fixe sont équivalentes si et seulement si elles sont équivalentes dans l'interprétation canonique.*

**Démonstration:**

Il nous suffit de démontrer que si  $X = T$  et  $X = T'$  sont équivalentes dans l'interprétation canonique  $C$ , elles seront équivalentes dans toute autre interprétation  $I$ .

Soient  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  et  $Y(t') = \{T'^j \mid j \in \mathbb{N}\}$  les points fixes de ces équations dans l'interprétation canonique. Il est aisé de vérifier que  $T_1 \leq T_2$  implique  $t_1(\perp) \subseteq_I t_2(\perp)$  pour toute interprétation  $I$ . Comme  $\forall i \exists j T^i \leq T'^j$  on en déduit

$$\forall i \exists j t^i(\perp) \subseteq_I t'^j(\perp) \text{ et par symétrie}$$

$$\forall k \exists l t'^k(\perp) \subseteq_I t^l(\perp).$$

Par suite, dans  $I$ ,  $\bigcup_{i \in \mathbb{N}} t^i(\perp) \equiv_I \bigcup_{i \in \mathbb{N}} t'^i(\perp)$ .  $\square$

Nous pouvons maintenant énoncer deux lemmes techniques qui seront utiles dans la suite.

**Lemme 3:** *Si deux termes  $F(T_1, \dots, T_n)$  et  $G(T'_1, \dots, T'_m)$  sont égaux dans l'interprétation canonique, alors  $F = G$ ,  $n = m$ , et pour tout  $i$   $T_i = T'_i$  dans l'interprétation canonique.*

**Démonstration:** Soient  $\{T_k^i \mid i \in \mathbb{N}\}$  et  $\{T'_k{}^i \mid i \in \mathbb{N}\}$  les interprétations de  $T_k$  et  $T'_k$ . Puisque:

$$\forall i \exists j F(T_1^i, \dots, T_n^i) \leq G(T_1'^j, \dots, T_m'^j) \text{ on en déduit } F = G, n = m \text{ et}$$

$$\forall i \exists j T_k^i \leq T_k'^j \quad (k \in [1, n])$$

$\tau_i = \{T_j^i \mid j \in \mathbb{N}\}$  admit an upper bound  $\tau = \bigcup_{i \in \mathbb{N}} \tau_i = \{T_i^i \mid i \in \mathbb{N}\}$  where for all  $i$ ,  $T_i^i = T_{n_i}^i$ , the sequence  $n_i$  chosen so that  $\forall i (n_i \geq i \text{ and } T_{n_i}^i \leq T_{n_i+1}^{i+1})$  (There exists such a sequence since the  $\tau_i$  form an ascending chain.)

A constant  $C$  is therefore interpreted as the sequence:

$$C = \{T_i \mid T_i = C, i \in \mathbb{N}\}$$

The interpretation  $f$  of  $n$ -ary symbols  $F$  maps from sequences  $\{T_k^i \mid k \in \mathbb{N}\}$  for  $i$  in  $[1, n]$  to the sequence  $\{F(T_k^1, T_k^2, \dots, T_k^n) \mid k \in \mathbb{N}\}$ . It is easy to verify that we have a legitimate interpretation. (More rigorously, the interpretation domain that we have considered is  $\mathcal{D}/\equiv$ ).

We write  $P\{A/X\}$  for the result of substituting  $A$  for all occurrences of  $X$  in a term  $P$ . We can verify that in our canonical interpretation  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  with

$$T^1 = T \text{ and } T^{i+1} = T\{T_i/X\}$$

The interest in canonical interpretation arises from the following lemma:

**Lemma 2:** *Two fixpoint equations are equivalent if and only if they are equivalent in the canonical interpretation.*

**Proof:**

It suffices to demonstrate that if  $X = T$  and  $X = T'$  are equivalent in the canonical interpretation  $C$ , they must be equivalent in all other interpretations  $I$ .

Let the fixpoints of these equations be  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  and  $Y(t') = \{T'^j \mid j \in \mathbb{N}\}$  in the canonical interpretation. It is easy to verify that  $T_1 \leq T_2$  implies  $t_1(\perp) \subseteq_I t_2(\perp)$  for all interpretations  $I$ . From  $\forall i \exists j T^i \leq T'^j$  we deduce

$$\forall i \exists j t^i(\perp) \subseteq_I t'^j(\perp) \text{ and by symmetry}$$

$$\forall k \exists l t'^k(\perp) \subseteq_I t^l(\perp).$$

Consequently, in  $I$ ,  $\bigcup_{i \in \mathbb{N}} t^i(\perp) \equiv_I \bigcup_{i \in \mathbb{N}} t'^i(\perp)$ .  $\square$

We can now state two technical lemmas that will be useful later.

**Lemma 3:** *If two terms  $F(T_1, \dots, T_n)$  and  $G(T'_1, \dots, T'_m)$  are equal in the canonical interpretation, then  $F = G$ ,  $n = m$ , and for all  $i$ ,  $T_i = T'_i$  in the canonical interpretation.*

**Proof:** Let  $\{T_k^i \mid i \in \mathbb{N}\}$  and  $\{T'_k{}^i \mid i \in \mathbb{N}\}$  be the interpretations of  $T_k$  and  $T'_k$ . Since:

$$\forall i \exists j F(T_1^i, \dots, T_n^i) \leq G(T_1'^j, \dots, T_m'^j) \text{ from which we deduce } F = G, n = m \text{ and}$$

$$\forall i \exists j T_k^i \leq T_k'^j \quad (k \in [1, n]) \text{ and antisymmetry. } \square$$

et l'inégalité symétrique.  $\square$

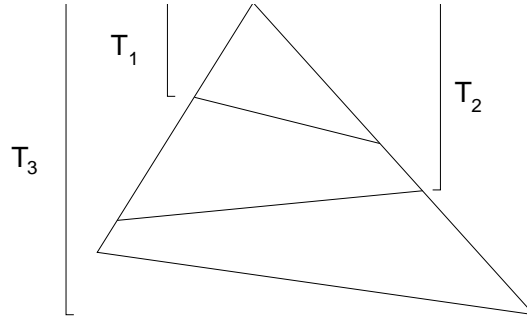
**Lemme 4:** Pour tous termes  $T$  et  $T'$ , si  $Y(t) \subseteq Y(t')$  dans l'interprétation canonique, alors, ou bien  $T = X$ , ou bien  $Y(t) \equiv Y(t')$ .

**Démonstration:**

(a) Définitions d'abord deux notions de "profondeur" d'un terme  $T$ ,  $\text{pmax}(T)$  et  $\text{pmin}(T)$ :

$$\begin{cases} \text{pmax}(X) = \text{pmax}(C) = 1 \\ \text{(si } C \text{ est une constante)} \\ \text{pmax}(F(T_1, \dots, T_n)) = 1 + \max_{1 \leq i \leq n} \{\text{pmax}(T_i)\} \\ \\ \text{pmin}(X) = 1 ; \text{pmin}(C) = +\infty \\ \text{(si } C \text{ est une constante)} \\ \text{pmin}(F(T_1, \dots, T_n)) = 1 + \min_{1 \leq i \leq n} \{\text{pmin}(T_i)\} \end{cases}$$

Ces notions nous permettent d'énoncer la propriété d'"alignement" suivante: si  $T_1, T_2, T_3$  vérifient  $T_1 \leq T_3$  et  $T_2 \leq T_3$  et  $\text{pmax}(T_1) \leq \text{pmin}(T_2)$  alors  $T_1 \leq T_2$ . La preuve se fait aisément par induction sur la structure de  $T_3$ . On peut représenter cette situation par le dessin suivant:



(b) Etant donnés maintenant deux points fixes  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  et  $Y(t') = \{T'^j \mid j \in \mathbb{N}\}$ , s'ils vérifient  $Y(t) \subseteq Y(t')$  alors  $\forall i \exists j T^i \leq T'^j$ .

Montrons que l'on a aussi  $\forall j \exists i T'^j \leq T^i$ , si  $T \neq X$ , étant donné  $T'^j$ , on peut toujours trouver un  $i$  tel que

$$\text{pmin}(T^i) \geq \text{pmax}(T'^j)$$

Il est alors possible de trouver un  $k$  tel que  $k \geq j$  et  $T^i \leq T'^k$ . Donc  $T'^j \leq T'^k$  et d'après la propriété d'alignement  $T'^j \leq T^i$ . Donc  $Y(t') \subseteq Y(t)$ .  $\square$

### 1.3 Forme normale et algorithme d'équivalence

Nous allons montrer que l'ensemble des termes conduisant à des équations au point fixe équivalentes est clos par rapport à l'opération  $\Sigma$ .

**Notation:** Si deux termes  $T$  et  $T'$  ont pour interprétation  $t$  et  $t'$  dans l'interprétation canonique, nous noterons  $\sigma(t, t')$  l'interprétation de  $\Sigma(T, T')$ .

**Lemme 5:** Si  $t(a) \equiv t'(a)$  pour un certain  $a$ , alors  $\sigma(t, t')(a) \equiv t(a) \equiv t'(a)$ .

**Lemma 4:** For all terms  $T$  and  $T'$ , if  $Y(t) \subseteq Y(t')$  in the canonical interpretation, then either  $T = X$  or  $Y(t) \equiv Y(t')$ .

**Proof:**

(a) First we define two notions of the "depth" of a term  $T$ ,  $\text{pmax}(T)$  and  $\text{pmin}(T)$ :

$$\begin{cases} \text{pmax}(X) = \text{pmax}(C) = 1 \\ \text{(if } C \text{ is a constant)} \\ \text{pmax}(F(T_1, \dots, T_n)) = 1 + \max_{1 \leq i \leq n} \{\text{pmax}(T_i)\} \\ \\ \text{pmin}(X) = 1 ; \text{pmin}(C) = +\infty \\ \text{(if } C \text{ is a constant)} \\ \text{pmin}(F(T_1, \dots, T_n)) = 1 + \min_{1 \leq i \leq n} \{\text{pmin}(T_i)\} \end{cases}$$

These notions allow us to state the following "alignment" property: if  $T_1, T_2, T_3$  satisfy  $T_1 \leq T_3$  and  $T_2 \leq T_3$  and  $\text{pmax}(T_1) \leq \text{pmin}(T_2)$  then  $T_1 \leq T_2$ . The proof is done easily by structural induction. We can represent this situation by the following figure:

(b) Now, given two fixpoints  $Y(t) = \{T^i \mid i \in \mathbb{N}\}$  and  $Y(t') = \{T'^j \mid j \in \mathbb{N}\}$ , if they satisfy  $Y(t) \subseteq Y(t')$  then  $\forall i \exists j T^i \leq T'^j$ .

Let us also show that if we have  $\forall j \exists i T'^j \leq T^i$ , if  $T \neq X$ , given  $T'^j$ , we can always find an  $i$  such that

$$\text{pmin}(T^i) \geq \text{pmax}(T'^j)$$

It is therefore possible to choose  $k$  such that  $k \geq j$  and  $T^i \leq T'^k$ . Then  $T'^j \leq T'^k$  and by the alignment property  $T'^j \leq T^i$ . Then  $Y(t') \subseteq Y(t)$ .  $\square$

### 1.3 Normal form and equivalence algorithm

We are going to show that the set of terms leading to equivalent fixpoint equations is closed under the operation  $\Sigma$ .

**Notation:** If two terms  $T$  and  $T'$  have interpretations  $t$  and  $t'$  in the canonical interpretation, we write  $\sigma(t, t')$  for the interpretation of  $\Sigma(T, T')$ .

**Lemma 5:** If  $t(a) \equiv t'(a)$  for some  $a$ , then  $\sigma(t, t')(a) \equiv t(a) \equiv t'(a)$ .

**Démonstration:** Par induction sur la structure de  $T$ :

(1) Si  $T = X$  ou  $T = C$  la propriété est évidente.

(2) Si  $T = F(T_1, \dots, T_n)$  deux cas se présentent:

-  $T' = X$ , donc  $\Sigma(T, T') = X$  et la propriété est vraie.

-  $T' = G(T'_1, \dots, T'_m)$  et alors nécessairement  $F = G$ ,  $m = n$  et  $t'_i(a) = t_i(a)$  pour tout  $i$  dans  $[1, n]$ . Par l'hypothèse d'induction,  $\sigma(t_i, t'_i)(a) = t_i(a)$  et donc  $t(a) = f(\sigma(t_1, t'_1)(a), \dots, \sigma(t_n, t'_n)(a)) = \sigma(t, t')(a)$ .  $\square$

**Lemme 6:** Si deux termes  $T$  et  $T'$  ont le même point fixe, alors  $\Sigma(T, T')$  a aussi le même point fixe. Autrement dit  $Y(t) \equiv Y(t')$  implique  $Y(t) = Y(\sigma(t, t'))$ .

**Démonstration:** Nous avons  $Y(t) \equiv t(Y(t)) \equiv Y(t') \equiv t'(Y(t')) \equiv t'(Y(t))$ . Par le lemme 4 nous obtenons  $\sigma(t, t')(Y(t)) = Y(t)$  et par minimalité  $Y(\sigma(t, t')) \subseteq Y(t)$ . Le lemme 3 implique alors:

- ou bien  $Y(\sigma(t, t')) \equiv Y(t)$  et la preuve est finie;

- ou bien  $\Sigma(T, T') \equiv X$ , mais ceci n'est compatible avec  $Y(t) = Y(t')$  que si  $T = T' = X$  auquel cas nous avons encore  $Y(\sigma(t, t')) = Y(t)$ .  $\square$

Nous sommes maintenant prêts à montrer (de manière non constructive) l'existence d'une *forme minimale* pour l'ensemble des termes conduisant à des équations au point fixe équivalentes.

Désignons par  $\|T\|$  la taille d'un terme  $T$  définie récursivement par:

1)  $\|X\| = 0$

2)  $\|F(T_1, \dots, T_n)\| = 1 + \sum_{i=1}^n \|T_i\|$ .

**Lemme 7:** Dans l'ensemble  $E(T) = \{T' \mid Y(t') \equiv Y(t)\}$  des termes ayant même plus petit point fixe que  $T$ , il existe un élément  $T^*$  de taille minimale.

**Démonstration:** Soient  $T_1$  et  $T_2$  deux termes de  $E(T)$  différents et de taille minimale. Alors  $T_1$  et  $T_2$  sont forcément incomparables, sinon, d'après le Lemme 1, l'un des deux serait de taille strictement plus petite que l'autre. Mais alors, toujours d'après le lemme,  $\|\Sigma(T_1, T_2)\| < \|T_1\|$ . Comme  $E(T)$  est fermé par  $\Sigma$  d'après le lemme 6,  $T_1$  et  $T_2$  ne pourraient être minimaux.  $\square$

Pour l'instant, le lemme 7 ne permet pas de construire  $T^*$ , mais nous allons exhiber une relation syntactique entre tous les termes ayant même point fixe que  $T^*$ . Notons  $T' \rightarrow T''$  la relation définie par les axiomes suivants:

i)  $\vdash T \rightarrow T$

ii)  $T \rightarrow U \vdash T \rightarrow U\{\{X/T\}\}$

où la notation  $U\{\{X/T\}\}$  signifie que certaines occurrences de  $X$  dans  $U$  ont été remplacées par  $T$ . Notons  $D(T) = \{U \mid \vdash T \rightarrow U\}$ .

**Théorème 1:** L'ensemble  $E(T)$  des termes ayant

**Proof:** By structural induction on  $T$ :

(1) If  $T = X$  or  $T = C$  the property is obvious.

(2) If  $T = F(T_1, \dots, T_n)$  two cases arise:

-  $T' = X$ , then  $\Sigma(T, T') = X$  and the property holds;

-  $T' = G(T'_1, \dots, T'_m)$  and so necessarily  $F = G$ ,  $m = n$  and  $t'_i(a) = t_i(a)$  for all  $i$  in  $[1, n]$ . By the induction hypothesis,  $\sigma(t_i, t'_i)(a) = t_i(a)$  and therefore  $t(a) = f(\sigma(t_1, t'_1)(a), \dots, \sigma(t_n, t'_n)(a)) = \sigma(t, t')(a)$ .  $\square$

**Lemma 6:** If two terms  $T$  and  $T'$  have the same fixpoint, then  $\Sigma(T, T')$  also has the same fixpoint. In other words  $Y(t) \equiv Y(t')$  implies  $Y(t) = Y(\sigma(t, t'))$ .

**Proof:** We have  $Y(t) \equiv t(Y(t)) \equiv Y(t') \equiv t'(Y(t')) \equiv t'(Y(t))$ . By lemma 4 we obtain  $\sigma(t, t')(Y(t)) = Y(t)$  and by minimality  $Y(\sigma(t, t')) \subseteq Y(t)$ . Lemma 3 therefore implies:

- either  $Y(\sigma(t, t')) \equiv Y(t)$  and the proof is finished;

- or  $\Sigma(T, T') \equiv X$ , but this is not compatible with  $Y(t) = Y(t')$  since if  $T = T' = X$  in which case we again have  $Y(\sigma(t, t')) = Y(t)$ .  $\square$

We are now ready to show (non-constructively) the existence of a minimal form for the set of terms leading to equivalent fixpoint equations.

We write  $\|T\|$  for the size of a term  $T$ , defined recursively by:

1)  $\|X\| = 0$

2)  $\|F(T_1, \dots, T_n)\| = 1 + \sum_{i=1}^n \|T_i\|$ .

**Lemma 7:** In the set  $E(T) = \{T' \mid Y(t') \equiv Y(t)\}$  of terms having the same least fixpoint as  $T$ , there exists an element  $T^*$  of minimal size.

Let  $T_1$  and  $T_2$  be two different terms of  $E(T)$  of minimal size. Then  $T_1$  and  $T_2$  are inevitably incomparable, otherwise, from Lemma 1, one of the two would be of size strictly less than the other. But then, always  $\|\Sigma(T_1, T_2)\| < \|T_1\|$  from the lemma. Since  $E(T)$  is closed under  $\Sigma$  by Lemma 6,  $T_1$  and  $T_2$  cannot be minimal.  $\square$

For the moment, lemma 7 does not allow the construction of  $T^*$ , but we will present a syntactic relation between all the terms having the same fixpoint as  $T^*$ . We write  $T' \rightarrow T''$  for the relation defined by these axioms:

i)  $\vdash T \rightarrow T$

ii)  $T \rightarrow U \vdash T \rightarrow U\{\{X/T\}\}$

where the notation  $U\{\{X/T\}\}$  indicates some occurrences of  $X$  in  $U$  have been replaced by  $T$ . We write  $D(T) = \{U \mid \vdash T \rightarrow U\}$ .

**Theorem 1:** The set  $E(T)$  of terms having the same

même point fixe que  $T$  est identique à l'ensemble  $D(T^*) = \{U \mid \vdash T^* \rightarrow U\}$  des termes dérivant de l'élément minimal  $T^*$  de  $E(T)$ .

### Démonstration:

Bien entendu  $E(T) = E(T^*)$ . Le fait que  $D(T^*) \subseteq E(T^*)$  est déjà connu (par exemple cf. [10]). Montrons maintenant que  $E(T^*) \subseteq D(T^*)$ : soit  $T'$  un terme de longueur minimale appartenant à  $E(T^*)$  et non à  $D(T^*)$ . On a nécessairement  $T^* \leq T'$  puisque  $T^*$  est une forme normale. Donc  $T' = T^*\{T_1, \dots, T_k\}$ . Comme  $T'$  est dans  $E(T^*)$ , nous avons:

$$Y(t^*) \equiv Y(t') \equiv t'(Y(t')) \equiv t'(Y(t^*)).$$

$$\text{Donc } t^*(Y(t^*)) = t'(Y(t^*)).$$

Par le lemme 3, on en déduit  $Y(t^*) \equiv t_i(Y(t^*))$  pour  $i \in [1, k]$ . Par minimalité puis le lemme 4, on obtient  $Y(t_i) = Y(t^*)$ . Mais comme les  $T_i$  sont plus petits que  $T'$  ils sont par hypothèse dans  $D(T^*)$ . Par suite  $T'$  est dans  $D(T^*)$ .  $\square$

Le théorème 1 implique que la forme normale  $T^*$  de  $T$  est un sous-terme de  $T$ , à savoir le plus petit sous-terme  $T^*$  tel que  $T^* \rightarrow T$ . Ceci nous donne donc un algorithme de calcul de cette forme normale et un processus de décision de l'équivalence de deux équations au point fixe. Plutôt qu'exposer ces deux algorithmes en plus de détail ici, nous passons maintenant au cas général, celui des systèmes d'équations au point fixe.

## II. Systèmes d'Equations Récursives

Les résultats obtenus dans la première partie se généralisent au cas des systèmes d'équations au point fixe: à tout système, on peut associer un système dit canonique de taille minimale (La "taille" d'un système étant la somme des tailles des équations qui le constituent). On obtient ainsi un algorithme pour décider de l'équivalence de deux systèmes et l'on peut aussi calculer un système équivalent ayant un nombre minimum d'équations. Cependant, ce dernier système n'est pas nécessairement unique.

### 2.1 Syntaxe

Nous utilisons un ensemble  $\Xi = \{X_1, \dots, X_n\}$  de variables.

Un *système* est la donnée:

i) d'une famille d'équations au point fixe  $X_i = T_i$ , pour  $i = 1, 2, \dots, n$ . Les termes  $T_i$  sont construits sur les variables  $X_i$ ,  $i \in [1, n]$ .

ii) d'une *variable principale*  $X_1$

Un système est dit *connexe* si toutes ses équations sont nécessaires au calcul de  $X_1$ , avec la définition suivante: l'équation  $X_i = T_i$  et toutes les équations nécessaires aux variables de  $T_i$  sont nécessaires au calcul de  $X_i$ . La *taille* du système  $S = \{X_i = T_i \mid i \in [1, n]\}$  est définie par  $\|S\| = \sum_1^n \|T_i\|$  et un système dit *uniforme* si pour tout  $i$ ,  $\|T_i\| = 1$ , autrement dit si les  $T_i$  ne comportent qu'un seul symbole de fonction.

fixpoint as  $T$  is identical to the set  $D(T^*) = \{U \mid \vdash T^* \rightarrow U\}$  of terms deriving from the minimal element  $T^*$  of  $E(T)$ .

### Proof:

Of course  $E(T) = E(T^*)$ . The fact that  $D(T^*) \subseteq E(T^*)$  is already known (for example cf. [10]). We now show that  $E(T^*) \subseteq D(T^*)$ : let  $T'$  be a term of minimal length belonging to  $E(T^*)$  and not to  $D(T^*)$ . We necessarily have  $T^* \leq T'$  because  $T^*$  is a normal form. So,  $T' = T^*\{T_1, \dots, T_k\}$ . Since  $T'$  is in  $E(T^*)$ , we have:

$$Y(t^*) \equiv Y(t') \equiv t'(Y(t')) \equiv t'(Y(t^*)).$$

$$\text{Therefore } t^*(Y(t^*)) = t'(Y(t^*)).$$

From lemma 3, we deduce  $Y(t^*) \equiv t_i(Y(t^*))$  for  $i \in [1, k]$ . By minimality and lemma 4, we obtain  $Y(t_i) = Y(t^*)$ . But since the  $T_i$  are smaller than  $T'$  they are by hypothesis in  $D(T^*)$ . Consequently  $T'$  is in  $D(T^*)$ .  $\square$

Theorem 1 implies that the normal form  $T^*$  of  $T$  is a subterm of  $T$ , namely the smallest subterm  $T^*$  such that  $T^* \rightarrow T$ . This gives us an algorithm for computing this normal form and a decision procedure for the equivalence of two fixpoint equations. Rather than presenting these two algorithms in great detail here, we now proceed to the general case, that of systems of fixpoint equations.

## II. Systems of Recursive Equations

The results obtained in the first section generalize to the case of systems of fixpoint equations: to each system, one can associate a canonical system of minimal size (the "size" of a system is the sum of the sizes of its constituent equations). Thus, one obtains an algorithm for deciding the equivalence of two systems and one can also compute an equivalent system with a minimal number of equations. However, this last system is not necessarily unique.

### 2.1 Syntax

We use a set  $\Xi = \{X_1, \dots, X_n\}$  of variables.

A *system* is:

i) a family of fixpoint equations  $X_i = T_i$ , for  $i = 1, 2, \dots, n$ . The terms  $T_i$  are constructed over the variables  $X_i$ ,  $i \in [1, n]$ .

ii) A *principle variable*  $X_1$

A system is said to be *connected* if all its equations are needed to compute  $X_1$ , with the following definition: the equation  $X_i = T_i$  and all the equations necessary for variables of  $T_i$  are needed to compute  $X_i$ . The *size* of a system  $S = \{X_i = T_i \mid i \in [1, n]\}$  is defined by  $\|S\| = \sum_1^n \|T_i\|$  and a system is said to be *uniform* if for all  $i$ , either  $\|T_i\| = 1$  or  $T_i$  contains only a single function symbol.

## 2.2 Sémantique.

Si l'on interprète les termes comme au §1.2. sur un domaine  $\mathcal{D}$ , à l'ensemble  $\{T_i \mid i \in [1, n]\}$  se trouve naturellement associé une application de  $\mathcal{D}^n$  dans  $\mathcal{D}^n$  et donc un point fixe dans  $\mathcal{D}^n$ . Nous noterons  $Y(x_i)$  la composante de ce vecteur correspondant à la variable  $X_i$ .

Deux systèmes  $S$  et  $S'$  sont *équivalents* si, pour toute interprétation  $Y(x_1) \equiv Y(x'_1)$ . La notion d'interprétation canonique se généralise trivialement et le lecteur pourra vérifier que le lemme 2 reste valide.

Un système  $\{X_i = T_i \mid i \in [1, n]\}$  sera dit *normal* s'il n'est équivalent à aucun système  $\{X_j = T_j \mid j \in [1, n]\}$  dans lequel les  $T_j$  sont des sous-termes des  $T_i$  et au moins l'un deux est un sous-terme propre.

Un sous-terme propre de  $T$  est un sous-terme de  $T$  différent de  $T$  et de  $X_i$  pour tout  $i \in [1, n]$ .

Dans le cas d'un système comportant une seule équation, ce système est normal si l'équation est en forme normale.

## 2.3 Systèmes canoniques.

Etant donné un système  $S = \{X_i = T_i \mid i \in [1, n]\}$  nous commençons par construire un système uniforme  $\bar{S}$  qui lui est équivalent. Par exemple:

$$S = \begin{cases} X_1 & = & F(X_1, G(X_1, X_2)) \\ X_2 & = & H(F(X_1, X_2)) \end{cases}$$

(variable principale  $X_1$ )

$$\bar{S} = \begin{cases} Y_1 & = & F(Y_1, Y_3) \\ Y_2 & = & H(Y_4) \\ Y_3 & = & G(Y_1, Y_2) \\ Y_4 & = & F(Y_1, Y_2) \end{cases}$$

(variable principale  $Y_1$ )

Formellement,  $\bar{S}$  est construit ainsi: appelons  $\mathcal{J} = \{\tau_i \mid i \in [1, m]\}$  l'ensemble des sous-termes propres de  $S$  (dans le cas de notre exemple:  $\mathcal{J} = \{X_1, X_2, F(X_1, X_2), G(X_1, X_2)\}$ .) Le système  $\bar{S}$  est construit sur les nouvelles variables  $Y_i, i \in [1, m]$  associées ainsi aux éléments de  $\mathcal{J}$ :

1) Si  $\tau_i = X_1$  et si  $X_1 = F(\tau_{1_1}, \dots, \tau_{1_k})$  est une équation de  $S$ , alors  $Y_i = F(Y_{1_1}, \dots, Y_{1_k})$  est une équation de  $\bar{S}$ .

2) Si  $\tau_i = F(\tau_{i_1}, \dots, \tau_{i_k})$  alors  $Y_i = F(Y_{i_1}, \dots, Y_{i_k})$  est une équation de  $\bar{S}$ .

3) La variable principale de  $\bar{S}$  est la nouvelle variable  $Y_1$  associée à  $X_1$ .

**Lemme 8:**  $S \equiv \bar{S}$

**Lemme 9:** Si  $S$  est connexe,  $\bar{S}$  est connexe. Si  $S$  est normal,  $\bar{S}$  est normal.

Pour un système uniforme  $S = \{X_i = F_i(X_{i_1}, \dots, X_{i_{k(i)}}) \mid i \in I\}$  nous calculons la relation d'équival-

## 2.2 Semantics.

If we interpret terms as in §1.2 in a domain  $\mathcal{D}$ , for the set  $\{T_i \mid i \in [1, n]\}$  one finds naturally a map from  $\mathcal{D}^n$  to  $\mathcal{D}^n$  and thus a fixpoint in  $\mathcal{D}^n$ . We write  $Y(x_i)$  for the component of this vector corresponding to the variable  $X_i$ .

Two systems  $S$  and  $S'$  are *equivalent* if, for all interpretations  $Y(x_1) \equiv Y(x'_1)$ . The notion of canonical interpretation generalizes trivially and the reader may verify that lemma 2 remains valid.

A system  $\{X_i = T_i \mid i \in [1, n]\}$  is said to be *normal* if it is not equivalent to a system  $\{X_j = T_j \mid j \in [1, n]\}$  in which the  $T_j$  are subterms of the  $T_i$  and at least one of the two is a proper subterm.

A proper subterm of  $T$  is a subterm of  $T$  different from  $T$  and from  $X_i$  for all  $i \in [1, n]$ .

The case of a system containing a single equation, the system is normal if the equation is in normal form.

## 2.3 Canonical systems.

Given a system  $S = \{X_i = T_i \mid i \in [1, n]\}$  we start by constructing a uniform system  $\bar{S}$  equivalent to it. For example:

$$S = \begin{cases} X_1 & = & F(X_1, G(X_1, X_2)) \\ X_2 & = & H(F(X_1, X_2)) \end{cases}$$

(principal variable  $X_1$ )

$$\bar{S} = \begin{cases} Y_1 & = & F(Y_1, Y_3) \\ Y_2 & = & H(Y_4) \\ Y_3 & = & G(Y_1, Y_2) \\ Y_4 & = & F(Y_1, Y_2) \end{cases}$$

(principal variable  $Y_1$ )

Formally,  $\bar{S}$  is constructed as follows: let us call  $\mathcal{J} = \{\tau_i \mid i \in [1, m]\}$  the set of proper subterms of  $S$  (in the case of our example:  $\mathcal{J} = \{X_1, X_2, F(X_1, X_2), G(X_1, X_2)\}$ .) The system  $\bar{S}$  is constructed over new variables  $Y_i, i \in [1, m]$  associated as follows with elements of  $\mathcal{J}$ :

1) If  $\tau_i = X_1$  and if  $X_1 = F(\tau_{1_1}, \dots, \tau_{1_k})$  is an equation of  $S$ , then  $Y_i = F(Y_{1_1}, \dots, Y_{1_k})$  is an equation of  $\bar{S}$ .

2) If  $\tau_i = F(\tau_{i_1}, \dots, \tau_{i_k})$  then  $Y_i = F(Y_{i_1}, \dots, Y_{i_k})$  is an equation of  $\bar{S}$ .

3) The principle variable of  $\bar{S}$  is the new variable  $Y_1$  associated with  $X_1$ .

**Lemma 8:**  $S \equiv \bar{S}$

**Lemma 9:** If  $S$  is connected,  $\bar{S}$  is connected. If  $S$  is normal,  $\bar{S}$  is normal.

For a uniform system  $S = \{X_i = F_i(X_{i_1}, \dots, X_{i_{k(i)}}) \mid i \in I\}$  we compute the equivalence over variables defined



ence entre ses variables définie par  $Y(x_i) \equiv Y(x_j)$  pour toute interprétation, que nous notons  $X_i \equiv X_j$ . Définissons inductivement une suite croissante de parties de  $\Xi \times \Xi$ , si  $\Xi = \{X_i \mid i \in I\}$ :

- 1)  $D_0 = \{(X_i, X_j) \in \Xi \times \Xi \mid F_i \neq F_j\}$
- 2)  $D_{n+1} = D_n \cup \{(X_i, X_j) \in \Xi \times \Xi \mid F_i = F_j \wedge \exists m \in [1, k(i)] \text{ tel que } (X_{i_m}, X_{j_m}) \in D_n\}$

**Lemme 10:**

- i) Il existe un certain  $l$  tel que  $D_l = \bigcup_{n=0}^{\infty} D_n$ .
- ii)  $X_i \equiv X_j$  si et seulement si la paire  $(X_i, X_j)$  n'appartient pas à  $D_l$ .

**Démonstration:** L'existence de  $l$  vient simplement du fait que  $\Xi$  est fini. Montrons ii) par induction de Scott (cf. [5]) sur la formule:

$$\begin{aligned} & \wedge \wedge X_i \equiv X_j \\ & (X_i, X_j) \notin D_l \end{aligned}$$

Tout d'abord  $\wedge \wedge \Omega \equiv \Omega$ . Par ailleurs,  $\Phi$  implique:

$$\Phi\{T_1/X_1, \dots, T_i/X_i, \dots, T_n/X_n\}$$

puisque  $(X_i, X_j) \notin D_l$  entraîne  $T_i \equiv T_j$ .  $\square$

**Théorème 2:**

*L'équivalence de deux systèmes d'équations au point fixe est décidable.*

**Démonstration:**

Le Lemme 9 permet de se ramener au cas des systèmes uniformes. Si  $S = \{X_i = T_i \mid i \in I\}$  et  $S' = \{X'_i = T'_i \mid i \in I'\}$ , il est toujours possible de s'assurer que  $\Xi = \{X_i \mid i \in I\}$  et  $\Xi' = \{X'_i \mid i \in I'\}$  sont *disjoints* et de considérer le système

$$S'' = \{X_i = T_i \cup X'_j = T'_j \mid i \in I, j \in I'\}$$

L'algorithme du lemme 10 nous dira si  $X_1 \equiv X'_1$ .  $\square$

**Corollaire 1:**

*L'équivalence  $T \equiv T'$ , où  $T$  et  $T'$  sont des termes quelconques sur les variables de deux systèmes  $S$  et  $S'$  est décidable.*

**Démonstration:**

On rajoute les équations  $Z = T$  et  $Z' = T'$  et on vérifie si  $Z \equiv Z'$ .  $\square$

**Théorème 3:**

*A tout système  $S$  on peut faire correspondre un système équivalent  $S'$  qui est normal, dont la taille est au plus la taille de  $S$  et qui comporte au plus le même nombre de variables.*

**Démonstration:**

Soit  $T$  un sous-terme propre de  $S$ ,  $T \neq X_i$ , tel que  $X_i \equiv T$ . Cette égalité permet de "réduire" le système  $S$ :

by  $Y(x_i) \equiv Y(x_j)$  for all interpretations, which we write  $X_i \equiv X_j$ . We inductively define an increasing sequence of subsets of  $\Xi \times \Xi$ , if  $\Xi = \{X_i \mid i \in I\}$ :

- 1)  $D_0 = \{(X_i, X_j) \in \Xi \times \Xi \mid F_i \neq F_j\}$
- 2)  $D_{n+1} = D_n \cup \{(X_i, X_j) \in \Xi \times \Xi \mid F_i = F_j \wedge \exists m \in [1, k(i)] \text{ such that } (X_{i_m}, X_{j_m}) \in D_n\}$

**Lemma 10:**

- i) There exists some  $l$  such that  $D_l = \bigcup_{n=0}^{\infty} D_n$ .
- ii)  $X_i \equiv X_j$  if and only if the pair  $(X_i, X_j)$  does not appear in  $D_l$ .

**Proof:** The existence of  $l$  comes simply from the fact that  $\Xi$  is finite. Let us show ii) by Scott induction (cf. [5]) on the formula:

$$\begin{aligned} & \bigwedge X_i \equiv X_j \\ & (X_i, X_j) \notin D_l \end{aligned}$$

First of all,  $\bigwedge \Omega \equiv \Omega$ . In addition,  $\Phi$  implies:

$$\Phi\{T_1/X_1, \dots, T_i/X_i, \dots, T_n/X_n\}$$

since  $(X_i, X_j) \notin D_l$  leads to  $T_i \equiv T_j$ .  $\square$

**Theorem 2:**

*The equivalence of two systems of fixpoint equations is decidable.*

**Proof:**

Lemma 9 makes it possible to reduce to the case of uniform systems. If  $S = \{X_i = T_i \mid i \in I\}$  and  $S' = \{X'_i = T'_i \mid i \in I'\}$ , it is always possible to ensure that  $\Xi = \{X_i \mid i \in I\}$  and  $\Xi' = \{X'_i \mid i \in I'\}$  are *disjoint* and to consider the system

$$S'' = \{X_i = T_i \cup X'_j = T'_j \mid i \in I, j \in I'\}$$

The algorithm of lemma 10 tells us whether  $X_1 \equiv X'_1$ .  $\square$

**Corollary 1:**

*The equivalence  $T \equiv T'$ , where  $T$  and  $T'$  are any terms on the variables of the two systems  $S$  and  $S'$  is decidable.*

**Proof:**

One adds the equations  $Z = T$  and  $Z' = T'$  and verifies whether  $Z \equiv Z'$ .  $\square$

**Theorem 3:**

*For any system  $S$  one can construct a corresponding equivalent system  $S'$  which is normal, whose size is at most the size of  $S$  and which consists of at most the same number of variables.*

**Proof:**

Let  $T$  be a proper subterm of  $S$ ,  $T \neq X_i$ , such that  $X_i \equiv T$ . This equality permits "reduction" of the system  $S$ :

- Si  $T$  est une variable  $X_j$ , on remplace partout  $X_j$  par  $X_i$  et on supprime de  $S$  l'équation de définition de  $X_j$ ,  $X_j = T_j$ . Cette transformation élimine une variable du système sans accroître sa taille. On aurait aussi bien pu éliminer  $X_i$  que  $X_j$ ; ce degré de liberté sera exploité plus loin.

- Si  $T$  est un sous-terme de taille supérieure ou égale à  $l$ , remplaçons toutes ses occurrences dans  $S$  par  $X_i$ . La taille du système ne peut que diminuer et le système obtenu est équivalent au système initial.

Cette construction ne peut être itérée qu'un nombre fini de fois puisqu'il ne peut y avoir qu'un nombre fini d'équivalences  $X_i \equiv T$ . Par définition, le système auquel on aboutit est alors normal.  $\square$

**Exemple:**

$$S_0 \begin{cases} X = F(F(X, Y), Z) \\ Y = G(X, Z) \\ Z = G(X, G(X, Y)) \end{cases}$$

On découvre que  $Y \equiv Z$ .  $S_0$  est équivalent à  $S_1$ :

$$S_1 \begin{cases} X = F(F(X, Y), Y) \\ Y = G(X, Y) \end{cases}$$

Mais  $X \equiv F(X, Y)$ .  $S_1$  est donc équivalent à  $S_2$ :

$$S_2 \begin{cases} X = F(X, Y) \\ Y = G(X, Y) \end{cases}$$

$S_2$  est normal.

La construction du théorème 3 peut être effectuée plus facilement sur des systèmes uniformes où les seuls sous-termes propres sont des variables.

Le lemme 10 définit une relation d'équivalence entre ces variables et le système normal associé à un système uniforme relie les classes d'équivalence.

**Notations:** Si  $S$  est uniforme, nous notons  $\tilde{S}$  le système normal qui lui correspond. A tout système  $S$ , on associe un système  $\bar{S}$  uniforme équivalent, puis  $\hat{S} = \bar{\tilde{S}}$ , système normal et uniforme.

**Lemme 11:** Si  $S$  et  $S'$  sont deux systèmes connexes équivalents, alors  $\hat{S} = \hat{S}'$ .

**Démonstration:** Bien entendu, l'égalité entre  $\hat{S}$  et  $\hat{S}'$  s'entend à un renommage près des variables. Montrons d'abord que toute variable  $X_i$  de  $\bar{S}$  il existe une variable  $X'_i$  de  $\bar{S}'$  telle que  $X_i \equiv X'_i$ , par récurrence. Comme  $S \equiv S'$  et  $\bar{S} \equiv \bar{S}'$ , nous avons bien  $X_1 \equiv X'_1$ .

Si  $X_i \equiv X'_j$  et  $X_i = F(X_{i_1}, \dots, X_{i_n})$ ,  $X'_i = F(X_{j_1}, \dots, X_{j_n})$ , alors  $X_{i_1} = X'_{j_1}, \dots, X_{i_n} = X'_{j_n}$ . Par suite, toutes les variables nécessaires à  $X_1$  ont un correspondant dans  $\bar{S}'$ . Comme  $\bar{S}$  est connexe, ce sont toutes les variables de  $\bar{S}$ . Bien entendu la propriété symétrique est vraie:

- If  $T$  is a variable  $X_j$ , one replaces everywhere  $X_j$  by  $X_i$  and removes from  $S$  the equation defining  $X_j$ ,  $X_j = T_j$ . This transformation eliminates one variable from the system without increasing its size. One could also have eliminated  $X_i$  instead of  $X_j$ ; this degree of freedom will be exploited later.

- If  $T$  is a subterm of size greater or equal to  $l$ , we replace all its occurrences in  $S$  with  $X_i$ . The size of the system can only decrease and the system obtained is equivalent to the initial system.

This construction can only be iterated a finite number of times because there may be only a finite number of equivalences  $X_i \equiv T$ . By definition, the system which one then reaches is normal.  $\square$

**Example:**

$$S_0 \begin{cases} X = F(F(X, Y), Z) \\ Y = G(X, Z) \\ Z = G(X, G(X, Y)) \end{cases}$$

One finds that  $Y \equiv Z$ .  $S_0$  is equivalent to  $S_1$ :

$$S_1 \begin{cases} X = F(F(X, Y), Y) \\ Y = G(X, Y) \end{cases}$$

But  $X \equiv F(X, Y)$ .  $S_1$  is therefore equivalent to  $S_2$ :

$$S_2 \begin{cases} X = F(X, Y) \\ Y = G(X, Y) \end{cases}$$

$S_2$  is normal.

The construction of theorem 3 can be more easily carried out on uniform systems where the only proper subterms are variables.

Lemma 10 defines an equivalence relation between these variables and the normal system associated with a uniform system bound to the equivalence classes.

**Notations:** If  $S$  is uniform, we write  $\tilde{S}$  for the normal system to which it corresponds. To all systems  $S$ , one associates an equivalent uniform system  $\bar{S}$ , so  $\hat{S} = \bar{\tilde{S}}$ , then  $\hat{S} = \bar{\tilde{S}}$ , a normal and uniform system.

**Lemma 11:** If  $S$  and  $S'$  are two equivalent, connected systems, then  $\hat{S} = \hat{S}'$ .

**Proof:** Of course, the equality between  $\hat{S}$  and  $\hat{S}'$  is understood to be up to variable renaming. We first show that for all variables  $X_i$  of  $\bar{S}$  there exists a variable  $X'_i$  of  $\bar{S}'$  such that  $X_i \equiv X'_i$ , by recurrence. Since  $S \equiv S'$  and  $\bar{S} \equiv \bar{S}'$ , we have  $X_1 \equiv X'_1$ .

If  $X_i \equiv X'_j$  and  $X_i = F(X_{i_1}, \dots, X_{i_n})$ ,  $X'_i = F(X_{j_1}, \dots, X_{j_n})$ , then  $X_{i_1} = X'_{j_1}, \dots, X_{i_n} = X'_{j_n}$ . Consequently, all the variables necessary to  $X_1$  have a corresponding variable in  $\bar{S}'$ . Since  $\bar{S}$  is connected, these are all the variables of  $\bar{S}$ . Of course, the symmetric property is true:

$$\forall j \exists i X'_j \equiv X_i$$

Donc il existe une bijection entre les classes d'équivalence des variables de  $S$  et  $S'$ . Comme par construction les variables de  $\hat{S}$  et  $\hat{S}'$  respectivement sont indépendantes, nous avons  $\hat{S} = \hat{S}'$  à un changement de nom près des variables.  $\square$

Le lemme 11 justifie le nom de système canonique donné au système  $\hat{S}$ .

**Théorème 4:** *Dans l'ensemble des systèmes équivalents à un système donné  $S$ , le système canonique  $\hat{S}$  est de taille minimale.*

**Démonstration:** De la construction de  $\bar{S}$ , il est clair que  $\|S\| \geq \|\bar{S}\|$ . Comme  $\|\bar{S}\| \geq \|\tilde{S}\| = \|\hat{S}\|$  et  $\hat{S} = \hat{S}'$  pour tout  $S'$  équivalent à  $S$ ,  $\hat{S}$  est de taille minimale.  $\square$

**Remarque:** Il peut exister plusieurs systèmes équivalents de taille minimale, bien qu'un seul soit canonique, comme le montre l'exemple suivant:

$$S_1 = \begin{cases} X = F(X, Y) \\ Y = G(Z) \\ Z = H(X, Y) \end{cases}$$

$$S_2 = \begin{cases} X = F(X, Y) \\ Y = G(H(X, Y)) \end{cases}$$

Ici,  $\|S_1\| = \|S_2\|$ ,  $S_1 \equiv S_2$  et  $S_1$  est canonique.

## 2.4 Minimisation du nombre d'équations

On peut être intéressé par la représentation d'un système qui en minimise non pas la taille mais le nombre d'équations au point fixe. L'exemple suivant montre qu'il n'y a pas toujours de système normal minimal (en ce sens) unique:

$$S_1 = \begin{cases} X = F(X, Y) \\ Y = G(H(X, Y)) \end{cases}$$

$$S_2 = \begin{cases} X' = F(X', G(Y')) \\ Y' = H(X', G(Y')) \end{cases}$$

$S_1$  et  $S_2$  sont équivalents ( $X \equiv X'$ ), normaux et comportent un nombre minimum d'équation. Mais  $S_1$  et  $S_2$  ne sont pas identiques à un changement près des noms des variables.

Nous montrons maintenant comment construire tous les systèmes normaux et tous les systèmes normaux minimaux.

### Définition:

Soit  $S = \{X_i = T_i \mid i \in I\}$  et  $C$  un sous-ensemble des variables de  $S$ ,  $C = \{X_k \mid k \in K\}$ , contenant  $X_1$  et ayant la propriété suivante:

$\forall l \in I \setminus K, \exists T(l, C)$ , terme ne faisant intervenir que les variables de  $C$  tel que  $X_1 \equiv T(1, C)$ . [FOOTNOTE: Et par  $T(l, C)$  nous désignons dans la suite le plus petit terme permettant de définir  $X_1$ .]

$$\forall j \exists i X'_j \equiv X_i$$

Therefore there exists a bijection between the equivalence classes of  $S$  and  $S'$ . Since by construction the variables of (respectively)  $\hat{S}$  and  $\hat{S}'$  are independent, we have  $\hat{S} = \hat{S}'$  modulo a renaming.  $\square$

Lemma 11 justifies calling the system  $\hat{S}$  a canonical system.

**Theorem 4:** *In the set of systems equivalent to a given system  $S$ , the canonical system  $\hat{S}$  is of minimal size.*

**Proof:** In the construction of  $\bar{S}$ , it is clear that  $\|S\| \geq \|\bar{S}\|$ . Since  $\|\bar{S}\| \geq \|\tilde{S}\| = \|\hat{S}\|$  and  $\hat{S} = \hat{S}'$  for all  $S'$  equivalent to  $S$ ,  $\hat{S}$  is of minimal size.  $\square$

**Remark:** There may exist multiple equivalent systems of minimal size, but only one may be canonical ?, as the following example illustrates:

$$S_1 = \begin{cases} X = F(X, Y) \\ Y = G(Z) \\ Z = H(X, Y) \end{cases}$$

$$S_2 = \begin{cases} X = F(X, Y) \\ Y = G(H(X, Y)) \end{cases}$$

Here,  $\|S_1\| = \|S_2\|$ ,  $S_1 \equiv S_2$  and  $S_1$  is canonical.

## 2.4 Minimizing the number of equations

One might be interested in a representation of a system that minimizes not the size but the number of fix-point equations. The following example illustrates that there is not always a unique system that is normal and minimal (in this sense):

$$S_1 = \begin{cases} X = F(X, Y) \\ Y = G(H(X, Y)) \end{cases}$$

$$S_2 = \begin{cases} X' = F(X', G(Y')) \\ Y' = H(X', G(Y')) \end{cases}$$

$S_1$  and  $S_2$  are equivalent ( $X \equiv X'$ ), normal and each have a minimum number of equations. But  $S_1$  and  $S_2$  are not identical up to renaming.

We now show how to construct all the normal systems and all the minimal normal systems.

### Definition:

Let  $S = \{X_i = T_i \mid i \in I\}$  and  $C$  be a subset of variables of  $S$ ,  $C = \{X_k \mid k \in K\}$ , containing  $X_1$  and having the following property:

$\forall l \in I \setminus K, \exists T(l, C)$ , a term using only variables of  $C$  such that  $X_1 \equiv T(1, C)$ . [FOOTNOTE: And by  $T(l, C)$  we designate in what follows the least term allowing the definition of  $X_1$ .]

On appelle *coupe* de  $S$ ,  $C(S)$  le système associé à un tel ensemble  $C$ :

$$C(S) = \{X_i = T'_i \mid X_i \in C\}$$

dans lequel  $T'_i$  est obtenu à partir de  $T_i$  en remplaçant, pour tout  $l$  dans  $I \setminus K X_1$  par  $T(l, C)$ . A un  $C$  comportant un nombre minimum de variables correspond une *coupe minimale*.

**Théorème 5:** *Les systèmes normaux équivalents à  $S$  sont les coupes de  $\hat{S}$ .*

**Démonstration:**

a) Si  $S'$  est une coupe de  $\hat{S}$ ,  $S' \equiv S$  et  $S'$  est normal sinon  $\hat{S}$  ne le serait pas.

b) Si  $S'$  est normal et  $S' \equiv S$ , alors  $\bar{S}'$  est normal donc  $\bar{S}' = \hat{S}' = \hat{S}$ .

Mais  $S'$  est une coupe de  $\hat{S}$  si et seulement si  $\bar{S}' = \hat{S}$  et  $S'$  est normal.  $\square$

**Corollaire 2:**

Les systèmes normaux minimaux équivalents à  $S$  sont les coupes minimales de  $\hat{S}$ . Comme il n'y a qu'un nombre fini de coupes de  $\hat{S}$ , on peut construire effectivement ces systèmes minimaux.

**Exemple:**

$$S = \begin{cases} X = F(F(X, Y), Y) \\ Y = G(H(X, G(Z))) \\ Z = H(X, G(Z)) \end{cases} \quad \hat{S} = \begin{cases} X = F(X, Y) \\ Y = G(Z) \\ Z = H(X, Y) \end{cases}$$

$$C_1 = \{X, Y, Z\} \quad C_1(\hat{S}) = \hat{S}$$

$$C_2 = \{X, Y\} \quad C_2(\hat{S}) = \begin{cases} X = F(X, Y) \\ Y = G(H(X, Y)) \end{cases}$$

$$C_3 = \{X, Z\} \quad C_3(\hat{S}) = \begin{cases} X = F(X, G(Z)) \\ Z = H(X, G(Z)) \end{cases}$$

On retrouve bien les deux formes normales minimales de  $S$ .

**Remarque:** Si l'on part d'une seule équation récursive comme dans la première partie, on doit trouver une seule forme normale minimale, la forme normale de cette équation.

**Exemple:**

$$\begin{aligned} S &= \{X = F(X, G(F(X, G(X, X)), X))\} \\ \hat{S} &= \begin{cases} X = F(X, X') \\ X' = G(X, X) \end{cases} \\ C &= \{X\} \quad C(\hat{S}) = \{X = F(X, G(X, X))\} \end{aligned}$$

**Conclusion:**

Dans l'esprit des auteurs, l'intérêt de ce travail ne réside pas tant dans les résultats de décidabilité

We call an *S-cut*,  $C(S)$  the system associated with such a set  $C$ :

$$C(S) = \{X_i = T'_i \mid X_i \in C\}$$

in which  $T'_i$  is obtained from  $T_i$  by replacing for all  $l$  in  $I \setminus K X_1$  by  $T(l, C)$ . To a  $C$  containing a minimal number of variables there is a corresponding *minimal cut*.

**Theorem 5:** *The normal systems equivalent to  $S$  are the cuts of  $\hat{S}$ .*

**Proof:** a) If  $S'$  is a cut of  $\hat{S}$ ,  $S' \equiv S$  and  $S'$  are normal otherwise  $\hat{S}$  would not be.

b) If  $S'$  is normal and  $S' \equiv S$ , then  $\bar{S}'$  is normal since  $\bar{S}' = \hat{S}' = \hat{S}$ .

But  $S'$  is a cut of  $\hat{S}$  if and only if  $\bar{S}' = \hat{S}$  and  $S'$  is normal.  $\square$

**Corollary 2:**

The minimal normal systems equivalent to  $S$  are the minimal cuts of  $\hat{S}$ . Since there are only a finite number of cuts of  $\hat{S}$ , one can effectively construct these minimal systems.

**Example:**

So, we find the two minimal normal forms of  $S$ .

**Remark:** If one starts from a *single* recursive equation as in the first section, one always finds a single minimal normal form, the normal form of this equation.

**Example:**

$$\begin{aligned} S &= \{X = F(X, G(F(X, G(X, X)), X))\} \\ \hat{S} &= \begin{cases} X = F(X, X') \\ X' = G(X, X) \end{cases} \\ C &= \{X\} \quad C(\hat{S}) = \{X = F(X, G(X, X))\} \end{aligned}$$

**Conclusion:**

In the opinion of the authors, the interest of this work does not lie in the decidability results obtained

obtenus (qui peuvent l'être à moindre frais) que dans les méthodes utilisées, en particulier la construction d'un domaine canonique.

Cette méthode a été utilisée par Courcelle et Vuillemin [1] pour des systèmes fonctionnels.

Enfin, il est possible de considérer les résultats obtenus comme des résultats de complétude de sous-théories de la logique LCF (Logic for Computable Functions) étudiée par Milner [7].

**Remerciement:**

Une partie des résultats obtenus ici étaient connus de R. Milner, qui a collaboré à ce travail dans sa phase initiale.

(which can be had with less effort) than in the methods used, in particular the construction of a canonical domain.

This method was used by Courcelle and Vuillemin [1] for functional systems.

Finally, it is possible to consider the results obtained as completeness results of subtheories of the logic LCF (Logic for Computable Functions) studied by Milner [7].

**Acknowledgement:**

A portion of the results obtained here were already known by R. Milner, who collaborated in the early phase of this work.

## References

- [1] Courcelle B. et J. Vuillemin. Completeness results for the equivalence of recursive schemes, *J. Comput. System Sci.* 12 (1976) 179-197.
- [2] De Roeper, W. P. "Operational and Mathematical Semantics for first-order Recursive Program Schemas". (Private Communication).
- [3] J. Engelfriet. A note on infinite trees. *Information Processing Letters*, vol 1, no. 6, December 1972.
- [4] Kahn G. "A preliminary Theory for Parallel Programs". (Rapport Laboria no. 6, January 1973).
- [5] J. Kral. "Equivalence of modes and the Equivalence of Finite Automata" *Algol Bulletin* No. 35, March 1973, (34-35).
- [6] Lewis C.H., Rosen B.K. "Recursively defined Data Types". *IEEE SWAT Symposium* (October 1973)
- [7] Milner R. "Models of LCF". *Stanford Computer Science Department Report*. CS-332 (1973).
- [8] Milner R., Weyrauch R. "Proving Compiler Correctness in a mechanized Logic" in *Machine Intelligence 7*. Edinburgh University Press (1972).
- [9] Nivat M. "Sur l'Interprétation des Schémas de Programmes Monadiques". *Rapport Laboria* No. 1 (Décembre 1972).
- [10] Pair C. "Concerning the syntax of Algol 68," *Algol Bulletin* No. 31 (March 1970) 16-27.
- [11] Scott D. "Outline of a Mathematical Theory of Computation". *Oxford Mono. PRG-2*. Oxford University (1970).
- [12] Vuillemin J. "Proof Techniques for Recursive Programs". *Stanford Computer Science Department*. Ph.D. Thesis (1973).

## A Commentary by B. Courcelle

It was quite a surprise for me to receive a message from Todd Veldhuizen saying that he had translated from French a 30 years old conference paper presented at the second ICALP held in Saarbrücken in 1974, of which I am coauthor with G. Kahn and J. Vuillemin. He did that work because he felt the paper was important.

First of all I would like to thank him for this work, and I am writing this postface as a testimony of my gratitude.

I will not comment on the importance of the paper, which is quite a subjective appreciation, but I will indicate a few research directions that can be traced back to it, and I will give a few references I have in mind. I have definitely no time for a thorough bibliographical review of related papers.

The research group around Maurice Nivat at INRIA (INRIA was called Laboria in 1974) and Paris-7 University, developed a theory of program schemes with uninterpreted conditional operators, handling them as basic functions rather than as control structures. One obtains in this way a smoother algebraic treatment and more decidability results (although most decidable problems are actually intractable). This theory is exposed in the book by I. Guessarian [4], and in a chapter I wrote for the Handbook of Theoretical Science [3].

This theory uses in a fundamental way infinite trees (i.e., infinite terms comparable to formal power series) as syntactic objects representing the behaviours of program schemes under all interpretations. The present paper deals with regular trees. More complex program schemes correspond to algebraic trees. These trees are related with deterministic context-free languages like regular trees are with regular languages. Regular and algebraic trees are surveyed in [1].

Another feature of the present paper developed later is the focus on systems of equations and their least fixed points. In this way, program schemes get closer to denotational semantics than to operational semantics. And this is coherent with the use of uninterpreted conditional operators. Term rewriting systems are useful as a tool for proving properties. The term "algebraic semantics" referring to this approach was currently used in the late 70's. See [4] and [3].

Program transformations by "folding and unfolding" recursive definitions have raised a lot of interest. But the hopes have not materialized. The characterization of equivalences of programs in terms of formal systems has certainly benefited from these developments. Recursive program schemes and grammars of various types are treated in a uniform way in my survey paper [2]. The existence of a formal system able to express the equivalence of recursive program schemes is the central fact proved by G. Sénizergues and from which follows the decidability of the equivalence problem for deterministic push-down automata [5,6]. I. Walukiewicz has obtained [7] a formal proof system for the Mu-calculus, a language based on least and greatest fixed-points of systems of equations.

## References

- [1] B. Courcelle, Fundamental properties of infinite trees, *Theoret. Comp. Sci.* 25 (1983) 95-169.
- [2] B. Courcelle, Equivalences and transformations of regular systems. Applications to recursive program schemes and grammars. *Theoret. Comp. Sci.* 42 (1986) 1-122.
- [3] B. Courcelle, Recursive applicative program schemes, Chapter 9 of *Handbook of Theoretical Computer Science*, Volume B, J. Van Leeuwen ed., Elsevier, 1990, pp. 459-492.
- [4] I. Guessarian, Algebraic semantics, *Lec. Notes Comput. Sci.*, volume 99, Springer Verlag, 1981.
- [5] G. Sénizergues, Complete formal systems for equivalence problems, *Theoret. Comp. Sci.* 231 (1999) 309-334.
- [6] G. Sénizergues,  $L(A) = L(B) ?$ , *Theoret. Comp. Sci.* 251 (2001) 1-166.
- [7] I. Walukiewicz, Completeness of Kozen's axiomatization of propositional Mu-calculus, *Information and Computation* 157 (2000) 142-182.

## B Translation notes and typographical changes

This translation is based on:

B. Courcelle, G. Kahn, and J. Vuillemin. Algorithmes d'équivalence et de réduction à des expressions minimales dans une classe d'équations récursives simples. In Jacques Loeckx, editor, *Automata, Languages and Programming*, volume 14 of *Lecture Notes in Computer Science*, pages 200–213. Springer Verlag, August 1974.

Typographical changes:

- Many uses of underlining in the original (typewritten) paper were replaced with bold or italic in this version. The english abstract is from the original paper; this accounts for the different reading of the french and english versions. In the original paper, equations were written as  $Z \leq g(g(Z))$ , where the  $\leq$  was intended to be read “defined by.” For clarity, in this version equations are written  $Z = g(g(Z))$ . (B. Courcelle)
- Bottom of page 201 in the equation  $\Sigma(F(T_1, T_2, \dots, T_n), F(T'_1, T'_2, \dots, T'_n)) = \dots$ , a missing right parenthesis added to the right-hand side.
- Top of page 202 changed “Il existe dans  $\mathcal{D}$  un élément minimal  $\perp$ ” to “Il existe dans  $\mathcal{D}$  un plus petit élément  $\perp$ ” (B. Courcelle).
- Top of page 203 changed “dans laquelle  $\tau_i = \{T_j^1 \mid j \in \mathbb{N}\}$ ” to “dans laquelle  $\tau_i = \{T_j^i \mid j \in \mathbb{N}\}$ ”
- Bottom of page 203 changed “Soient  $T_k^i \mid i \in \mathbb{N}$  et  $T_k'^i \mid i \in \mathbb{N}$  les interprétations de  $T_k$  et  $T_k'$ .” to “Soient  $\{T_k^i \mid i \in \mathbb{N}\}$  et  $\{T_k'^i \mid i \in \mathbb{N}\}$  les interprétations de  $T_k$  et  $T_k'$ .”
- On the top of page 204 there is a symbol missing in the equation  $\forall i \exists j T_k^i T_k'^j$  ( $k \in [1, n]$ ) between  $T_k^i$  and  $T_k'^j$ . An  $\leq$  symbol has been inserted.
- Near the top of page 206, “Donc  $t^*(Y(t^*)) = t'(Y(t^*))$ .” changed to “Donc  $t^*(Y(t^*)) = t'(Y(t^*))$ .”
- Near the bottom of page 206, “La *taille* du système  $S = X_i \leq T_i \mid i \in [1, n]$ ” changed to “La *taille* du système  $S = \{X_i \leq T_i \mid i \in [1, n]\}$ ”.
- Near the bottom of page 207, changed “ $\mathcal{J} = \{\tau_i \mid i \in [1, m]\}$ ” to “ $\mathcal{J} = \{\tau_i \mid i \in [1, m]\}$ ”.
- Middle of page 207, changed “construire un système uniforme  $S$ ” to “construire un système uniforme  $\bar{S}$ ”.
- Near top of p. 208, changed “2)  $D_{n+1} = D_n \cup \{(X_i, X_j) \in \Xi \times \Xi \mid F_j = F_j \wedge \dots\}$ ” to “2)  $D_{n+1} = D_n \cup \{(X_i, X_j) \in \Xi \times \Xi \mid F_i = F_j \wedge \dots\}$ ”.
- Middle of p. 208 in proof of Lemma 10, changed “ $(X_i X_j) \notin D_l$ ” to “ $(X_i, X_j) \notin D_l$ ”
- Middle of p. 208 in proof of Lemma 10, changed “entraîne  $T_i \equiv T_j$ ” to “entraîne  $T_i \equiv T_j$ ”
- Page 210, changed the statement of Theorem 4 from “Dans l'ensemble des systèmes équivalents à un système donné  $\hat{S}$ , le système canonique  $S$  est de taille minimale” to “Dans l'ensemble des systèmes équivalents à un système donné  $S$ , le système canonique  $\hat{S}$  est de taille minimale.”
- Top of p. 210 in proof of lemma 11, changed “nous avons bien  $X_1 \equiv X_1$ ” to “nous avons bien  $X_1 \equiv X_1'$ ”.
- In bibliography, changed reference [1] from Courcelle B. et J. Vuillemin. “Semantics and Axiomatics of a Simple Programming Language”. *Rapport Laboria no. 60, 1974.* to Courcelle B. et J. Vuillemin. *Completeness results for the equivalence of recursive schemes, J. Comput. System Sci. 12 (1976) 179-197.*

Translation by T. Veldhuizen; corrections to [tveldhui@acm.org](mailto:tveldhui@acm.org).