



**HAL**  
open science

# STORAGE OPERATORS AND DIRECTED LAMBDA-CALCULUS

René David, Karim Nour

► **To cite this version:**

René David, Karim Nour. STORAGE OPERATORS AND DIRECTED LAMBDA-CALCULUS. The Journal of Symbolic Logic, 1995, 64 (4), p 1054-1086. hal-00385174

**HAL Id: hal-00385174**

**<https://hal.science/hal-00385174v1>**

Submitted on 18 May 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**STORAGE OPERATORS  
AND  
DIRECTED LAMBDA-CALCULUS**

**René DAVID & Karim NOUR**

LAMA - Equipe de Logique  
Université de Savoie - Campus Scientifique  
73376 Le Bourget du Lac cédex  
email [david,nour]@univ-savoie.fr

## Abstract

Storage operators have been introduced by J.L. Krivine in [5] ; they are closed  $\lambda$ -terms which, for a data type, allow to simulate a "call by value" while using the "call by name" strategy. In this paper, we introduce the directed  $\lambda$ -calculus and show that it has the usual properties of the ordinary  $\lambda$ -calculus. With this calculus we get an equivalent - and simple - definition of the storage operators that allows to show some of their properties :

- the stability of the set of storage operators under the  $b$ -equivalence (theorem 5.1.1) ;
- the undecidability ( and its semi-decidability ) of the problem "is a closed  $\lambda$ -term  $t$  a storage operator for a finite set of closed normal  $\lambda$ -terms ? " (theorems 5.2.2 and 5.2.3) ;
- the existence of storage operators for every finite set of closed normal  $\lambda$ -terms (theorem 5.4.3) ;
- the computation time of the "storage operation" (theorem 5.5.2).

## Résumé

Les opérateurs de mise en mémoire ont été introduits par J.L. Krivine dans [5] ; il s'agit de  $\lambda$ -termes clos qui, pour un type de données, permettent de simuler "l'appel par nom" dans le cadre de "l'appel par valeur". Dans cet article, nous introduisons le  $\lambda$ -calcul dirigé et nous démontrons qu'il garde les propriétés usuelles du  $\lambda$ -calcul ordinaire. Avec ce calcul nous obtenons une définition équivalente - et simple - pour les opérateurs de mise en mémoire qui permet de prouver plusieurs de leurs propriétés :

- la stabilité de l'ensemble des opérateurs de mise en mémoire par la  $b$ -équivalence (théorème 5.1.1) ;
- l'indécidabilité (et sa semi-décidabilité ) du problème "un terme clos  $t$  est il un opérateur de mise en mémoire pour un ensemble fini de termes normaux clos ? " (théorèmes 5.2.2 et 5.2.3) ;
- l'existence d'opérateurs de mise en mémoire pour chaque ensemble fini de termes normaux clos (théorème 5.4.3) ;
- une inégalité controlant le temps calcul d'un opérateur de mise en mémoire (théorème

5.5.2).

## Summary

<b>§ 0. Introduction</b>	<b>4</b>
<b>§ 1. Basic notions of pure <math>\lambda</math>-calculus</b>	<b>8</b>
1.1. Notations	8
1.2. Properties of head reduction	
10	
<b>§ 2. Storage operators</b>	<b>11</b>
2.1. Definition of storage operators	11
2.2. General forms of the head normal form of a storage operator	12
2.3. Examples of storage operators	14
2.3.1. The projections	14
2.3.2. The Church integers	15
2.3.3. The recursive integers	16
2.3.4. The finite lists	17
<b>§ 3. The directed <math>\lambda</math>-calculus</b>	<b>19</b>
3.1. The $\lambda$ -terms	19
3.2. The $\beta$ -reduction	20

3.3. The  $b[\ ]$ -left reduction

24

3.4. The  $b[\ ]$ -head reduction

25

**§ 4. An equivalent definition for storage operators**  
27

**§ 5. Properties of storage operators**  
33

5.1. Storage operators and  $b$ -equivalence

33

5.2. Decidability

34

5.3. Storage operators and  $y$ -equivalence

35

5.4. Storage operators for a set of  $b$ -normal  $l$ -terms

37

5.5. Computation time of storage operators

38

**References**

40

## **§ 0. Introduction**

**0.1** Lambda-calculus as such is not a computational model. A reduction strategy is needed. In this paper, we consider  $l$ -calculus with the left reduction (iteration of the head reduction denoted by  $\Sigma$ ). This strategy has some advantages : it always terminates when applied to a normalizable  $l$ -term, and it seems more economic since we compute a  $l$ -term only when we need it. But the major drawback of this strategy is that a function must compute its argument every time it uses it. This is the reason why this strategy is not really used. We would like a solution to this problem.

Let  $F$  be a  $l$ -term,  $D$  a set of closed normal  $l$ -terms, and  $t \in D$ . During the computation, by left reduction, of  $(F)h_t$  (where  $h_t \rightarrow t$ ),  $h_t$  may be computed several times (as many

times as  $F$  uses it). We would like to transform  $(F)h_t$  to  $(F)t$ . We also want that this transformation depends only on  $h_t$  (and not  $F$ ). In other words we look for some closed l-term  $T$  which satisfies the following properties :

- For every  $F$ , and for every  $tAD$ ,  $(T)h_tF \sum (F)t$  ;
- The computation time of  $(T)h_tF$  depends only on  $h_t$ .

**Definition (temporary) :**

A closed l-term  $T$  is called a **storage operator** for  $D$  if and only if for every  $tAD$ , and for every  $h_t : b_t$ ,  $(T)h_t f \sum (f)t$  (where  $f$  is a new variable).

It is clear that a storage operator satisfies the required properties. Indeed,

- Since we have  $(T)h_t f \sum (f)t$ , then the variable  $f$  never comes in head position during the reduction, we may then replace  $f$  by any l-term.
- The computation time  $(T)h_tF$  depends only on  $h_t$ .

K. Nour has shown (see [9]) that it is not always possible to get a normal form (it is the case for the set of Church integers). We then change the definition.

**Definition (temporary) :**

A closed l-term  $T$  is called **storage operator** for  $D$  if and only if for every  $tAD$ , there is a closed l-term  $t_t : b_t$ , such that for every  $h_t : b_t$ ,  $(T)h_t f \sum (f)t_t$  (where  $f$  is a new variable).

J.L. Krivine has shown that, by using Gödel translation from classical to intuitionistic logic, we can find, for every data type, a very simple type for the storage operators. But the l-term  $t_t$  obtained may contain variables substituted by l-terms depending on  $h_t$ . Since the l-term  $t_t$  is by-equivalent to a closed l-term, the left reduction of  $t_t[u_1/x_1, \dots, u_n/x_n]$  is equivalent to the left reduction of  $t_t$ , the l-terms  $u_1, \dots, u_n$  will therefore never be evaluated during the reduction. We then modify again the definition.

**Definition (final) :**

A closed l-term  $T$  is called a **storage operator** for  $D$  if and only if for every  $tAD$ , there is a l-term  $t_t : b_t$ , such that for every  $h_t : b_t$ , there is a substitution  $s$ , such that  $(T)h_t f \sum (f)s(t_t)$  (where  $f$  is a new variable).

In the case where  $t_t = t$ , we say that  $T$  is a **strong storage operator**, and in the case where  $t_t$  is closed, we say that  $T$  is a **proper storage operator**. These special operators are studied in [9] and [12].

The previous definition is not well adapted to study these operators. Indeed, it is, a

priori, a Pstatement  $(\forall t E_t \forall h_t E_s A(T, t, t_t, h_t, s))$ . We will show that it is in fact equivalent to Pstatement  $(t_t \text{ can be computed from } t, \text{ and } s \text{ from } h_t)$ .

We now describe the intuitive meaning of the directed lambda calculus.

**0.2** Consider the particular case of the set  $\underline{N}$  of Church integers.

A closed l-term  $T$  is a storage operator for  $\underline{N}$  if and only if for every  $n \geq 0$ , there is a l-term  $t_n; b_n$ , such that for every  $h_n; b_n$ , there is a substitution  $s$ , such that  $(T)h_n f \sum (f)s(t_n)$ .

Let's analyse the head reduction  $(T)h_n f \sum (f)s(t_n)$ , by replacing each l-term which comes from  $h_n$  by a new variable.

If  $h_n; b_n$ , then  $h_n \sum l g l x (g) t_{n-1}$ ,  $t_{n-k} \sum (g) t_{n-k-1}$   $1 \leq k \leq n-1$ ,  $t_0 \sum x$ , and  $t_k; b_k (g)^k x$   $0 \leq k \leq n-1$ .

Let  $u_n$  be a new variable ( $u_n$  represents  $h_n$ ).  $(T)u_n f$  is solvable, and its head normal form does not begin by l, therefore it is a variable applied to some arguments. The free variables of  $(T)u_n f$  are  $u_n$  and  $f$ , we then have two possibilities for its head normal form :

:

(f)d (in this case we stop) or  $(u_n)a_1 \dots a_m$ .

Assume we obtain  $(u_n)a_1 \dots a_m$ . The variable  $u_n$  represents  $h_n$ , and  $h_n \sum l g l x (g) t_{n-1}$ , therefore  $(h_n)a_1 \dots a_m$  and  $((a_1)t_{n-1}[a_1/g, a_2/x])a_3 \dots a_m$  have the same head normal form. The l-term  $t_{n-1}[a_1/g, a_2/x]$  comes from  $h_n$ . Let  $u_{n-1, a_1, a_2}$  be a new variable ( $u_{n-1, a_1, a_2}$  represents  $t_{n-1}[a_1/g, a_2/x]$ ). The l-term  $((a_1)u_{n-1, a_1, a_2})a_3 \dots a_m$  is solvable, and its head normal form does not begin by l, therefore it is a variable applied to some arguments. The free variables of  $((a_1)u_{n-1, a_1, a_2})a_3 \dots a_m$  are among  $u_{n-1, a_1, a_2}$ ,  $u_n$ , and  $f$ , we then have three possibilities for its head normal form :

(f)d (in this case we stop) or  $(u_n)b_1 \dots b_r$  or  $(u_{n-1, a_1, a_2})b_1 \dots b_r$ .

Assume we obtain  $(u_{n-1, a_1, a_2})b_1 \dots b_r$ . The variable  $u_{n-1, a_1, a_2}$  represents  $t_{n-1}[a_1/g, a_2/x]$ , and  $t_{n-1} \sum (g) t_{n-2}$ , therefore  $(t_{n-1}[a_1/g, a_2/x])b_1 \dots b_r$  and  $((a_1)t_{n-2}[a_1/g, a_2/x])b_1 \dots b_r$  have the same head normal form. The l-term  $t_{n-1}[a_1/g, a_2/x]$  comes from  $h_n$ . Let  $u_{n-2, a_1, a_2}$  be a new variable ( $u_{n-2, a_1, a_2}$  represents  $t_{n-2}[a_1/g, a_2/x]$ ). The l-term  $((a_1)u_{n-2, a_1, a_2})b_1 \dots b_r$  is solvable, and its head normal form does not begin by l, therefore it is a variable applied to arguments. The free variables of  $((a_1)u_{n-2, a_1, a_2})b_1 \dots b_r$  are among  $u_{n-2, a_1, a_2}$ ,  $u_{n-1, a_1, a_2}$ ,  $u_n$ , and  $f$ , therefore we have four possibilities for its head normal form :

(f)d (in this case we stop) or  $(u_n)c_1 \dots c_s$  or  $(u_{n-1, a_1, a_2})c_1 \dots c_s$  or  $(u_{n-2, a_1, a_2})c_1 \dots c_s$

...and so on...

Assume we obtain  $(u_{0, d_1, d_2})e_1 \dots e_k$  during the construction. The variable  $u_{0, d_1, d_2}$

represents  $t_0[d_1/g, d_2/x]$ , and  $t_0 \sum x$ , therefore  $(t_0[d_1/g, d_2/x])e_1 \dots e_k$  and  $(d_2)e_1 \dots e_k$  have the same head normal form ; we then follow the construction with the l-term  $(d_2)e_1 \dots e_k$ .

The l-term  $(T)h_n f$  is solvable, and has  $(f)s(t)$  as head normal form, so this construction always stops on  $(f)d$ . We will prove later by a simple argument that  $d;_{by} \underline{n}$ .

According to the previous construction, the reduction  $(T)h_n f \sum (f)s(t_n)$  can be divided into two parts :

- A reduction that does not depend on  $\underline{n}$  :

$(T)u_n f \sum (u_n)a_1 \dots a_m,$

$((a_1)u_{n-1, a_1, a_2})a_3 \dots a_m \sum (u_{n-1, a_1, a_2})b_1 \dots b_r,$

$((a_1)u_{n-2, a_1, a_2})b_1 \dots b_r \sum (u_{n-2, a_1, a_2})b_1 \dots b_r,$

...

- A reduction that depends on  $\underline{n}$  (and not on  $h_n$ ) :

the reduction from  $(u_n)a_1 \dots a_m$  to  $((a_1)u_{n-1, a_1, a_2})a_3 \dots a_m,$

the reduction from  $(u_{n-1, a_1, a_2})b_1 \dots b_r$  to  $((a_1)u_{n-2, a_1, a_2})c_1 \dots c_s,$

....,

the reduction from  $(u_{0, d_1, d_2})e_1 \dots e_k$  to  $(d_2)e_1 \dots e_k,$

...

If we allow some new reduction rules to get the later reductions, (something as :  $(u_n)a_1 a_2 \sum (a_1)u_{n-1, a_1, a_2}$  ;  $u_{i+1, a_1, a_2} \sum (a_1)u_{i, a_1, a_2}$  (for  $i > 0$ ) ;  $u_{0, a_1, a_2} \sum a_2$ )

we obtain an equivalent -and easily expressed - definition for the storage operators for  $\underline{N}$  :

A closed l-term T is a storage operator for  $\underline{N}$  if and only if for every  $n \geq 0$ ,  $((T)u_n f \sum (f)d_n,$  and  $d_n;_{by} \underline{n}$ .

**0.3** The **directed l-calculus** is an extension of the ordinary l-calculus built for tracing a normal l-term t during some head reduction. Assume u is some, non normal, l-term having t as a subterm. We wish to trace the places where we really have to know what t is, during the reduction of u. Assume we have for every normal l-term t with free variables  $x_1, \dots, x_n$ , and any l-terms  $a_1, \dots, a_n$  a "new" variable  $u_{t, a_1, \dots, a_n}$ .

We want the following rules :

if  $t = lxv$ , then  $(u_{t, a_1, \dots, a_n})a \sum u_{v, a_1, \dots, a_n, a}$  or  $u_{t, a_1, \dots, a_n} \sum lx u_{v, a_1, \dots, a_n, x}$  ;

if  $t = (v)w$ , then  $u_{t, a_1, \dots, a_n} \sum (u_{v, a_1, \dots, a_n})u_{w, a_1, \dots, a_n}$  ;

if  $t = x_i$   $1 \leq i \leq n$ , then  $u_{t, a_1, \dots, a_n} \sum a_i$ .

We will prove later the following result (theorem 4-1) :

A closed l-term T is a storage operator for a set of closed normal l-terms D if and only



if for every  $t \in \text{AD}$ ,  $(T)u_t f \Sigma(f)d_t$ , and  $d_t :_{\text{b}} t$ .

**0.4** By interpreting the variable  $u_{t,a_1,\dots,a_n}$  (that will be denoted by  $[t]_{\langle a_1/x_1,\dots,a_n/x_n \rangle}$  and called a box) by  $t[a_1/x_1,\dots,a_n/x_n]$  (the l-term  $t$  with an explicit substitution), the new reduction rules are those that allow to really do the substitution. This kind of l-calculus (l-calculus with explicit substitution) has been studied by P.L.Curien (see [1] and [4]); his ls-calculus contains terms and substitutions and is intended to better control the substitution process created by b-reduction, and then the implementation of the l-calculus. The main difference between the ls-calculus and the directed l-calculus is :

- The first one produces an explicit substitution after each b-reduction ;
- The second only "executes" the substitutions given in advance.

We can therefore consider the directed l-calculus as a restriction (the interdiction of producing explicit substitutions) of ls-calculus ; a well adapted way to the study of the head reduction.

**0.5** This paper studies some properties of storage operators. It is organized as follows :

- The section 1 is devoted to preliminaries.
- In section 2, we define the storage operators, and we give the general form of their head normal forms.
- In section 3, we introduce the directed l-calculus, and we prove that it has the main properties of the ordinary l-calculus : the Church-Rosser theorem, the normalisation theorem, the resolution theorem. We focus on the head reduction, and we will prove that the reduction with the boxes represents correctly the reduction of terms where boxes are replaced by b-equivalent l-terms.
- In section 4, we present an equivalent definition for the storage operators.
- In section 5, we give some properties of storage operators :
  - If  $T$  is a storage operator for a set of closed normal l-terms, and if  $T :_{\text{b}} T'$ , then  $T'$  also is a storage operator for this set.
  - The problem " Let  $t$  be a closed l-term. Is it a storage operator for a set of closed normal l-terms ?" is undecidable. It is semi-decidable in case of a finite set.
  - Each finite set of closed normal l-terms has a storage operators.
  - the number of b-reductions to go from  $(T)h_t f$  to  $(f)s(t_t)$  is linear in the number of reductions to normalize  $h_t$ .

**Note** : The presentation made below hides some technical uninteresting difficulties. Since we work with name for the variables, and modulo  $\alpha$ -equivalence, there is a

problem to define precisely the notion of subterms.

- We suppose, for example, that the l-terms  $(x)x$ ,  $(y)y$ ,  $(z)z, \dots$  are subterms of the l-term  $lx(x)x$ .
- A l-term may have equal subterms ; we assume that we can distinguish these subterms.

These problems could be solved by indexing subterms with the paths from the root of the l-term and using de Bruijn notation. We will do not do that here.

**Acknowledgements.** We thank J.L. Krivine, S. Ronchi, and H. Barendregt for helpful discussions.

## § 1. Basic notions of pure l-calculus

### 1.1. Notations

They are standard (see [2] and [6]).

- We shall denote by  $L$  the set of terms of pure l-calculus, also called **l-terms**.
- Let  $t, u, u_1, \dots, u_n \in L$ , the application of  $t$  to  $u$  is denoted by  $(t)u$  or simply  $tu$ . In the same way we write  $(t)u_1 \dots u_n$  or  $tu_1 \dots u_n$  instead of  $(\dots((t)u_1)\dots)u_n$ .
- The  $b$  (resp.  $y$ , resp.  $by$ ) -reduction is denoted by  $t\mathbf{5}_b u$  (resp.  $t\mathbf{5}_y u$ , resp.  $t\mathbf{5}_{by} u$ ).
- One step of  $b$  (resp.  $y$ ) -reduction is denoted by  $t\mathbf{5}_{b0} u$  (resp.  $t\mathbf{5}_{y0} u$ ).
- The  $b$  (resp.  $y$ , resp.  $by$ ) -equivalence is denoted by  $t\mathbf{:}_b u$  (resp.  $t\mathbf{:}_y u$ , resp.  $t\mathbf{:}_{by} u$ ).
- The set of free variables of a l-term  $t$  is denoted by  $\mathbf{Fv}(t)$ .
- The notation  $t[a_1/x_1, \dots, a_n/x_n]$  represents the result of the simultaneous substitution of l-terms  $a_1, \dots, a_n$  to the free variables  $x_1, \dots, x_n$  of  $t$  (after a suitable renaming of the bounded variables of  $t$ ).

The notation  $\mathbf{s}(t)$  represents the result of the simultaneous substitution  $\mathbf{s}$  to the free variables of  $t$ .

- The length of a l-term  $t$  (number of symbols used to write  $t$ ) is denoted by  $\mathbf{lg}(t)$ .
- We denote by  $\mathbf{ST}(t)$ , the set of subterms of  $t$ .
- If  $t$  is  $b$ -normalizable, we denote by  $t^b$  its  $b$ -normal form.
- If  $t$  is  $by$ -normalizable, we denote by  $t^{by}$  its  $by$ -normal form.
- The notation  $t\mathbf{\Sigma\Sigma}0t'$  (resp.  $t\mathbf{\Sigma\Sigma}t'$ ) means that  $t'$  is obtained from  $t$  by one step of left reduction (resp. by some left reductions).

**Theorem 1.1.1 (normalization theorem).**  *$u$  is normalizable if and only if  $u$  is left normalizable.*

**Proof.** See [2] and [6]. ■

- If  $t$  is a normalizable  $l$ -term, then  $t \xrightarrow{\Sigma} t^b$ . We denote by  $\mathbf{Tps}(t)$ , the number of steps used to go from  $t$  to  $t^b$ .
- The notation  $t \xrightarrow{\Sigma_0} t'$  (resp.  $t \xrightarrow{\Sigma} t'$ ) means that  $t'$  is obtained from  $t$  by one step of head reduction (resp. by some head reductions).
- A  $l$ -term  $t$  is said **solvable** if and only if for every  $l$ -term  $u$ , there are variables  $x_1, \dots, x_k$ , and a  $l$ -terms  $u_1, \dots, u_k, v_1, \dots, v_l$ ,  $k, l \geq 0$ , such that  $(t[u_1/x_1, \dots, u_k/x_k])v_1 \dots v_l :_b u$ .

**Theorem 1.1.2 (resolution theorem).** *The following conditions are equivalent :*

- 1)  $t$  is solvable ;
- 2) the head reduction of  $t$  terminates ;
- 3)  $t$  is  $b$ -equivalent to a head normal form.

**Proof.** See [6]. ■

- If  $t$  is a solvable  $l$ -term, then there is a term  $t'$  in head normal form, such that  $t \xrightarrow{\Sigma} t'$ . We denote by  $\mathbf{tps}(t)$ , the number of step used to go from  $t$  to  $t'$ .
- For each  $l$ -term, we associate a set of  $l$ -terms denoted by  $\mathbf{STE}(t)$ , and called the set of essential subterms of  $t$ , by induction :
  - If  $t$  is unsolvable, then  $\mathbf{STE}(t) = \{U\}$  where  $U$  is a new symbol ;
  - If  $t$  is solvable, and  $ly_1 \dots ly_m(y)t_1 \dots t_r$  is its head normal form, then  $\mathbf{STE}(t) = \{t\} \cup \mathbf{STE}(t_i)$ .

**Theorem 1.1.3.** *If  $t$  is a normalizable  $l$ -term, then  $\mathbf{Tps}(t) = \mathbf{tps}(u)$ .*

**Proof.** Trivial. ■

## 1.2. Properties of head reduction

**Definitions.**

- We define an **equivalence relation** : on  $L$  by  $u : v$  if and only if there is a  $t$ , such that  $u \xrightarrow{\Sigma} t$ , and  $v \xrightarrow{\Sigma} t$ . In particular, if  $t$  is solvable, then  $u : t$  if and only if  $u$  is solvable, and has the same head normal form of  $t$ . If  $u$  is in head normal form, then  $t : u$  means  $u$  is the head normal form of  $t$ .
- If  $t \xrightarrow{\Sigma} t'$ , we denote by  $\mathbf{n}(t, t')$ , the number of steps to go from  $t$  to  $t'$ .

**Theorem 1.2.1.** *If  $t \xrightarrow{\Sigma} t'$ , then for every  $u_1, \dots, u_r \in L$  :*

- 1) *There is  $v \in L$ , such that  $(t)u_1 \dots u_r \xrightarrow{\Sigma} v$ ,  $(t')u_1 \dots u_r \xrightarrow{\Sigma} v$ , and  $\mathbf{n}((t)u_1 \dots u_r, v) = \mathbf{n}((t')u_1 \dots u_r, v)$*

$+n(t,t')$ .

2)  $t[u_1/x_1, \dots, u_r/x_r] \Sigma t'[u_1/x_1, \dots, u_r/x_r]$ , and  $n(t[u_1/x_1, \dots, u_r/x_r], t'[u_1/x_1, \dots, u_r/x_r]) = n(t,t')$ .

**Proof.** See [7]. ■

**Remarks.**

- 1) shows that to make the head reduction of  $(t)u_1 \dots u_n$ , it is equivalent (same result, and same number of steps) to make some steps in the head reduction of  $t$ , and then make the head reduction of  $(t')u_1 \dots u_n$ .

- 2) shows that to make the head reduction of  $t[u_1/x_1, \dots, u_n/x_n]$ , it is equivalent (same result, and same number of steps) to make some steps in the head reduction of  $t$ , and then make the head reduction of  $t'[u_1/x_1, \dots, u_n/x_n]$ .

This will be used everywhere without mention in the following.

**Corollary 1.2.2.** *Let  $t, u_1, \dots, u_n, v_1, \dots, v_m \in \mathcal{AL}$ . If  $(t[u_1/x_1, \dots, u_n/x_n])v_1 \dots v_m$  is solvable, then  $t$  is solvable.*

**Proof.** Easy. ■

**Corollary 1.2.3.** *If  $t:t'$ , then for every  $u_1, \dots, u_r \in \mathcal{AL}$  :*

1)  $(t)u_1 \dots u_r : (t')u_1 \dots u_r$ .

2)  $t[u_1/x_1, \dots, u_r/x_r] : t'[u_1/x_1, \dots, u_r/x_r]$ .

**Proof.** See [7]. ■

**Corollary 1.2.4.** *Let  $t :_b u$ , and  $u$  does not contain the variables  $x_1, \dots, x_n$ , then the left reduction of  $t[u_1/x_1, \dots, u_n/x_n]$  is equivalent to the left reduction of  $t$ . This reduction is independent of the  $l$ -terms  $u_1, \dots, u_n$  which will never be evaluated.*

**Proof.** See [7]. ■

## § 2. Storage operators

### 2.1 Definition of storage operators

**Definitions.**

- A  $l$ -term  $t$  is said **essential** if and only if it is  $b$ -equivalent to a  $b$ -normal closed  $l$ -term.

- Let  $T$  be a closed l-term, and  $t$  an essential l-term. We say that  $T$  is a **storage operator** (shortened to **o.m.m.** for *opérateur de mise en mémoire*) **for  $t$**  if and only if there is  $t_t:_{b}t$ , such that for every  $h_t:_{b}t$ ,  $(T)h_t \sum_l f(h_1/x_1, \dots, h_n/x_n)$ , where  $Fv(t_t) = \{x_1, \dots, x_n, f\}$ , and  $h_1, \dots, h_n$  are l-terms which depend on  $h_t$ .
- Let  $T$  be a closed l-term,  $D$  a set of essential l-terms. We say that  $T$  is an **o.m.m for  $D$**  if and only if it is an o.m.m. for every  $t$  in  $D$ .

**Lemma 2.1.1.**  *$T$  is an o.m.m. for  $t$  if and only if there is a l-term  $t_t:_{b}t$ , such that for every  $h_t:_{b}t$ ,  $(T)h_t f: (f)t_t[h_1/x_1, \dots, h_n/x_n]$ , where  $Fv(t_t) = \{x_1, \dots, x_n, f\}$ , and  $h_1, \dots, h_n$  are l-terms which depend on  $h_t$ .*

**Proof.**

**1** Clear.

**0** By corollary 1.2.2,  $(T)h_t$  is solvable. Let  $T'$  be its head normal form.

- If  $T' = lfw$ , then  $w$  is the head normal form of  $(T)h_t f$ , therefore  $w = (f)t_t[h_1/x_1, \dots, h_n/x_n]$ , therefore  $(T)h_t \sum_l f(h_1/x_1, \dots, h_n/x_n)$ .

- If  $T' = (v)T_1 \dots T_r$ ; we can choose  $h_t$ , such that  $f \notin Fv(h_t)$ ,  $v \neq f$ , therefore the head normal form of  $(T)h_t f$  is  $(v)T_1 \dots T_r f = (f)t_t[h_1/x_1, \dots, h_n/x_n]$ . A contradiction. ■

**Remark.** Let  $F$  be any l-term, and  $h_t$  a l-term b-equivalent to  $tAD$ . During the computation of  $(F)h_t$ ,  $h_t$  may be computed many times (for example, each time it comes in head position). Insead of computing  $(F)h_t$ , let us look at the head reduction of  $(T)h_t F$ . Since it is  $(T)h_t f[F/f]$ , by theorem 1.2.1, we shall first reduce  $(T)h_t f$  to its head normal form, which is  $(f)t_t[h_1/x_1, \dots, h_n/x_n]$ , and then compute  $(F)t_t[c_1/x_1, \dots, c_n/x_n, F/f]$  where  $c_i = h_i[F/f]$ . By corollary 1.2.4, the computation has been decomposed into two parts, the first being independent of  $F$ . This first part is essentially a computation of  $h_t$ , the result being  $t_t$ , which is a kind of normal form of  $h_t$ , because it only depends on the b-equivalent class of  $h_t$ : the substitutions made in  $t_t$  have no computational importance, since  $t$  is essential. So, in the computation of  $(T)h_t F$ ,  $h_t$  is computed first, and the result is given to  $F$  as an argument,  $T$  has stored the result, before giving it, as many times as needed, to any function.

## 2.2 General forms of head normal form of a storage operator

**Proposition 2.2.1.** *If  $T$  is an o.m.m. for  $t$ , then  $T$  is solvable, and its head normal form  $T'$  has one of the following form :  $T' = l^n(n)T_1 \dots T_r$   $r \geq 1$ ,  $T' = l^n l f(n)T_1 \dots T_r$   $r \geq 1$ , or*

$T' = \text{lnlf}(f)T_1$  where  $T_1 :_{\text{by}t}$ .

**Corollary 2.2.3.** *If  $t$  is unsolvable, and  $T$  is an o.m.m. for  $t$ , then  $T \sum \text{lnlf}(f)T_1$ , and  $T_1 :_{\text{by}t}$ .*

**Proof.** If  $T \sum \text{ln}(n)T_1 \dots T_r \ r \geq 1$  or  $T \sum \text{lnlf}(n)T_1 \dots T_r \ r \geq 1$ , then  $(T)t$  is unsolvable. Therefore, by proposition 2.2.1,  $T \sum \text{lnlf}(f)T_1$ , and  $T_1 :_{\text{by}t}$ . ■

**Proof of proposition 2.2.1.** If  $T$  is an o.m.m. for  $t$ , then there is a l-term  $t_t :_{\text{by}t}$ , such that for every  $h_t :_{\text{bt}}$ ,  $(T)h_t \sum \text{lf}(f)t_t[u_1/y_1, \dots, u_n/y_n]$ , with  $Fv(t_t) = \{y_1, \dots, y_n, f\}$ , and  $u_1, \dots, u_n$  are l-terms which depend on  $h_t$ . Therefore, by corollary 1.2.2,  $T$  is solvable. Let  $T'$  its head normal form. Since  $T$  is closed,  $T'$  also is closed, and  $T' = \text{lx}_1 \dots \text{lx}_m(x_i)T_1 \dots T_r \ r \geq 1$ .

By theorem 1.2.1,  $(T')h_t \sum \text{lf}(f)t_t[u_1/y_1, \dots, u_n/y_n]$ , therefore  $m = 1$  or  $2$ .

- If  $m=1$ , then  $T' = \text{ln}(n)T_1 \dots T_r \ r \geq 1$ .

- If  $m=2$  :

- If  $i=1$ , then  $T' = \text{lnlf}(n)T_1 \dots T_r \ r \geq 1$ .

- If  $i=2$ , then  $T' = \text{lnlf}(f)T_1 \dots T_r \ r \geq 1$ . Therefore  $\text{lf}(f)T_1[h_t/n] \dots T_r[h_t/n] = \text{lf}(f)t'[u_1/y_1, \dots, u_n/y_n]$ , therefore  $r=1$ , and  $T_1[h_t/n] = t_t[u_1/y_1, \dots, u_n/y_n]$ .

It remains to show that  $T_1 :_{\text{by}t}$ .

**Lemma 2.2.4.** *Let  $x, y$  be two variables of the l-calculus.*

1) *If  $t[(x)y/z] \ 5_{b_0} u$ , then  $u = v[(x)y/z]$ , and  $t \ 5_{b_0} v$ .*

2) *If  $t$  is a closed l-term, and  $t[(x)y/z] \ 5_{b_0} t$ , then  $t \ 5_{b_0} t$ .*

**Proof.**

1) By induction on  $t$ .

- If  $t$  is a variable, it is impossible.

- If  $t = \text{lrw}$ , then  $u = \text{lra}$ , and  $w[(x)y/z] \ 5_{b_0} a$ . By induction hypothesis, we have  $a = b[(x)y/z]$ , and  $w \ 5_{b_0} b$ . Therefore if we take  $v = \text{lrb}$ , we get  $u = v[(x)y/z]$ , and  $t \ 5_{b_0} v$ .

- If  $t = (a)b$ , and  $u = (c)b$  where  $a[(x)y/z] \ 5_{b_0} c$ . By induction hypothesis, we have  $c = d[(x)y/z]$ , and  $a \ 5_{b_0} d$ . Therefore if we take  $v = (d)b$ , we get  $u = v[(x)y/z]$ , and  $t \ 5_{b_0} v$ .

- If  $t = (a)b$ , and  $u = (a)c$  where  $b[(x)y/z] \ 5_{b_0} c$ . By induction hypothesis, we have  $c = d[(x)y/z]$ , and  $b \ 5_{b_0} d$ . Therefore if we take  $v = (a)d$ , we get  $u = v[(x)y/z]$ , and  $t \ 5_{b_0} v$ .

- If  $t = (\text{lra})b$ , and  $u = a[(x)y/z][b[(x)y/z]/r] = a[b/r][(x)y/z]$ , then, if we take  $v = a[b/r]$ , we get  $u = v[(x)y/z]$ , and  $t \ 5_{b_0} v$ .

2) By induction on the number of  $b_0$ -reductions. We use 1) to prove  $t = u[(x)y/z]$ , and  $t \ 5_{b_0} u$ . Since  $t$  is closed, then  $t = u$  and  $t \ 5_{b_0} t$ . ■

By lemma 2.2.4, we may assume that  $t_i$  does not contain a  $(y_i)y_j$   $1 \leq i, j \leq n$  as subterm.

**Lemma 2.2.5.** *Let  $d, t, t_1, \dots, t_n$  be l-terms, and  $s_1, \dots, s_n$  substitutions, such that :  
 $Fv(d) = \{x_1, \dots, x_n\} \{a_1, \dots, a_r\}$ ,  $Fv(t) = \{y_1, \dots, y_m\} \{b_1, \dots, b_k\}$ , and for all  $1 \leq i, j \leq m$   $(y_i)y_j$  is not a subterm of  $t$ . If for all  $1 \leq i \leq n$  and for every  $h_i: b_i t_i$ , there are  $h_1, \dots, h_{i-1}, h_{i+1}, \dots, h_n, u_1, \dots, u_m$ , such that  $d[s_1(h_1)/x_1, \dots, s_n(h_n)/x_n] = t[u_1/y_1, \dots, u_m/y_m]$ , then there are  $w_1, \dots, w_m$ , such that  $d = t[w_1/y_1, \dots, w_m/y_m]$ .*

**Proof.** By induction on  $d$  and  $t$ .

It is clear that we may assume that any variable  $x_1, \dots, x_n$  (resp.  $y_1, \dots, y_m$ ) appears at most once in  $d$  (resp.  $t$ ).

- If  $d = a_1$ , then  $a_1 = t[u_1/y_1, \dots, u_m/y_m]$ , therefore  $t = y_1$ , and  $u_1 = a_1$  or  $t = b_1 = a_1$ .
  - If  $d = x_1$ , then  $s_1(h_1) = t[u_1/y_1, \dots, u_m/y_m]$ .
    - If  $t = b_1$ , then  $s_1(h_1)$  is a variable, that is impossible if we take  $h_1 = (lxt_1)x$ .
    - If  $t = y_1$ , then  $d = t[x_1/y_1]$ .
    - If  $t = lxt'$ , then  $s_1(h_1)$  begins by  $l$ , that is impossible if we take  $h_1 = (lxt_1)x$ .
    - If  $t = (u)v$  :
      - If  $t = (\dots(((lxa)b)v_1)\dots)v_r$ , then  $s_1(h_1)$  begins with  $r+1$  (, that is impossible if we take  $h_1 = (\dots(((lx_1lx_2\dots lx_{n+2}t_1)x_1)x_2)\dots)v_{n+2}$ .
      - If  $t = (\dots((b_1)v_1)\dots)v_r$ , then that is impossible if we take  $h_1 = (lxt_1)x$ .
      - If  $t = (\dots((y_1)v_1)\dots)v_r$  and  $r \geq 2$ , then  $s_1(h_1)$  begins by at least  $r$  (, that is impossible if we take  $h_1 = (lxt_1)x$ . Therefore  $r = 1$  and  $t = (y_1)v_1$ .
- The l-term  $v_1$  can not begin by  $l$ . (it suffices to take  $h_1 = (lxt_1)(lxx)x$ )  
The l-term  $v_1$  can not begin by  $($ . (it suffices to take  $h_1 = (lxt_1)lxx$ )  
Therefore  $v_1$  is a variable.  
If  $v_1 = b_1$ , then that is impossible if we take  $h_1 = (lxt_1)(lxx)x$ .  
If  $v_1 = y_2$ , then that is impossible because in this case we have  $t = (y_1)y_2$ .

- If  $d = lxu$ , then :
  - If  $t = b_1$ , then  $lg(d) = 1$ , that is impossible.
  - If  $t = y_1$ , then  $d = t[lxu/y_1]$ .
    - If  $t = lxt'$ , then  $u[s_1(h_1)/x_1, \dots, s_n(h_n)/x_n] = t'[u_1/y_1, \dots, u_m/y_m]$ , and we use the induction hypothesis
    - If  $t = (u)v$ , then  $d$  begins by  $($ , that is impossible.
- If  $d = (u)v$ , then :
  - If  $t = b_1$ , then  $lg(d) = 1$ , that is impossible.
  - If  $t = y_1$ , then  $d = t[(u)v/y_1]$ .
  - If  $t = lxt'$ , then  $d$  begins by  $l$ , that is impossible.

- If  $t=(a)b$ , then  $u[s_1(h_1)/x_1, \dots, s_n(h_n)/x_n]=a[u_1/y_1, \dots, u_m/y_m]$ , and  $v[s_1(h_1)/x_1, \dots, s_n(h_n)/x_n]=b[u_1/y_1, \dots, u_m/y_m]$ , and we use the induction hypothesis. ■

By lemma 2.2.5, there are  $w_1, \dots, w_m$ , such that  $T_1=t_t[w_1/x_1, \dots, w_n/x_n]$ , we have  $T_1$ :byt. ■ (of proposition 2.2.1)

## 2.3 Examples of storage operators

### 2.3.1 The projections

For all  $0 \leq i \leq n$ , let  $P_i = \lambda x_1 \dots \lambda x_n x_i$  (the  $i^{\text{th}}$  projection among  $n$ ). Let  $P_n$  be the set of projections.

Define  $T = \lambda n (\lambda f) P_i \lambda f (f) P \dots \lambda f (f) P$ , and  $T = \lambda n \lambda f (n) (f) P P \dots (f) P$ .

$T$  and  $T$  are two o.m.m. for  $P_n$ .

Let  $h_i$ :  $P_1 \leq i \leq n$ , then  $h \sum P$ .

#### Behaviour of T:

$\text{Thf}:(h) \lambda f (f) P_i \lambda f (f) P \dots \lambda f (f) P f:(P) \lambda f (f) P_i \lambda f (f) P \dots \lambda f (f) P f:(\lambda f (f) P) f:(f) P$ .

It is easy to check that  $\text{tps}(\text{Thf}) = \text{Tps}(h) + n + 2$ . ■

#### Behaviour of T:

$\text{Thf}:(h) (f) P P \dots (f) P f:(P) (f) P P \dots (f) P f:(f) P$ .

It is easy to check that  $\text{tps}(\text{Thf}) = \text{Tps}(h) + n + 2$ . ■

### 2.3.2 The Church integers

For  $n \geq 0$ , we define the Church integer  $\underline{n} = \lambda f \lambda x (f)^n x$ . Let  $\underline{\mathbb{N}}$  be the set of Church integers.

Let  $\underline{s} = \lambda n \lambda f \lambda x (f)((n)f)x$ . It is easy to check that  $\underline{s}$  is a 1-term for the successor. Define

$T = \lambda n (n) G d$  where  $G = \lambda x \lambda y (x) \lambda z (y) (\underline{s}) z$ , and  $d = \lambda f (f) \underline{0}$  ;

$T = \lambda n \lambda f (n) F f \underline{0}$  where  $F = \lambda x \lambda y (x) (\underline{s}) y$ .

$T$  and  $T$  are o.m.m. for  $\underline{\mathbb{N}}$ .

Let  $h_n$ :  $\underline{n}$ , then  $h_n \sum \lambda g \lambda x (g) t_{n-1}, t_{n-k} \sum (g) t_{n-k-1} \ 1 \leq k \leq n-1, t_0 \sum x$ .

#### Behaviour of T:

$(T) h_n f:(h_n) G d f:(G) t_{n-1} [\underline{G}/g, d/x] f:(t_{n-1} [\underline{G}/g, d/x]) \lambda z (f) (\underline{s}) z$ .

We define a sequence of 1-terms  $(t_i)_{1 \leq i \leq n}$ :



$t_1 = lz(f)(s)z$ , and for all  $1 \leq k \leq n-1$  let  $t_{k+1} = lz(t_k)(s)z$ .

We prove (by induction on  $k$ ) that for all  $1 \leq k \leq n$  we have  $(T)h_n f: (t_{n-k}[G/g, d/x])t_k$ .

For  $k=1$  it is true.

Assume that is true for  $k$ , and prove it for  $k+1$ .

$$(T)h_n f: (t_{n-k}[G/g, d/x])t_k : (G)t_{n-k-1}[G/g, d/x]t_k : t_{n-k-1}[G/g, d/x]lz(t_k)(s)z = (t_{n-k-1}[G/g, d/x])t_{k+1}.$$

Therefore, in particular, for  $k=n$  we have  $(T)h_n f: (t_0[G/g, d/x])t_n = (d)t_n : (t_n)0$ .

We prove (by induction on  $k$ ) that for all  $1 \leq k \leq n$  we have  $t_k : lz(f)(s)^k z$ .

For  $k=1$  it is true.

Assume that is true for  $k$ , and prove it for  $k+1$ .

$$t_{k+1} = lz(t_k)(s)z : lz(lz(f)(s)^k z)(s)z : lz(f)(s)^{k+1} z.$$

Therefore, in particular, for  $k=n$  we have  $t_n : lz(f)(s)^n z$  and  $(T)h_n f: (lz(f)(s)^n z)0 : (f)(s)^n 0$ .

It is easy to check that  $tps((T)h_n f) = Tps(h_n) + 3n + 4$ . ■

### Behaviour of T:

$$(T)h_n f: (h_n)Ff0 : (F)t_{n-1}[F/g, f/x]0 : (t_{n-1}[F/g, f/x])(s)0.$$

We prove (by induction on  $k$ ) that for all  $1 \leq k \leq n$  we have

$$(T)h_n f: (t_{n-k}[F/g, f/x])(s)^k 0.$$

For  $k=1$  it is true.

Assume that is true for  $k$ , and prove it for  $k+1$ .

$$(T)h_n f: (t_{n-k}[F/g, f/x])(s)^k 0 : (F)t_{n-k-1}[F/g, f/x](s)^k 0 : t_{n-k-1}[F/g, f/x](s)^{k+1} 0.$$

Therefore, in particular, for  $k=n$  we have  $(T)h_n f: (t_0[F/g, f/x])(s)^n 0 = (f)(s)^n 0$ .

It is easy to check that  $tps((T)h_n f) = Tps(h_n) + 2n + 4$ . ■

## 2.3.3 The recursive integers

For  $n \geq 0$ , we define the recursive integer by  $=lflxx$  and  $=lflx(f)$ . Let be the set of recursive integers. Let  $=lnlflx(f)n$ . It is easy to check that is a l-term for the successor.

Define  $T = (Y)H$  where  $Y = (lxlf(f)(x)xf)lxlf(f)(x)xf$ ,  $H = lxly((y)lz(G)(x)z)d$ ,  $G = lxly(x)lz(y)()z$ , and  $d = lf(f)$  ;

$T = ln(n)rtr$  where  $t = ldlf(f)$ , and  $r = lylyz(G)(y)ztz$ .

$T$  and  $Tare$  o.m.m. for .

Let  $h_n : b$ , then :

if  $n=0$ ,  $h_n \sum lglxx$ , and if  $n \neq 0$ ,  $h_n \sum lglx(g)h_{n-1}$  where  $h_{n-1} : b$ .

### Behaviour of T:

We prove (by induction on  $n$ ) that  $((Y)H)h_n : lf(f)()^n$ .

If  $n=0$ , then  $((Y)H)h_0:((H)(Y)H)h_0:((h_0)lz(G)((Y)H)z)d:d=lf(f)$  .

If  $n \neq 0$ , then  $((Y)H)h_n:((H)(Y)H)h_n:((h_n)lz(G)((Y)H)z)d:$

$(lz(G)((Y)H)z)h_{n-1}[lz(G)((Y)H)z/g,d/x]:(G)((Y)H)h_{n-1}[lz(G)((Y)H)z/g,d/x]:$

$lf(((Y)H)h_{n-1}[lz(G)((Y)H)z/g,d/x])lz(f)( )z.$

Since  $h_{n-1}:_b$ , then  $h_{n-1}[lz(G)((Y)H)z/g,d/x]:_b$ , and, by induction hypothesis,  $((Y)H)h_{n-1}:lf(f)^{n-1}$ .

Therefore  $((Y)H)h_n:lf(lf(f)^{n-1})lz(f)( )z:lf(f)^n$ .

It is easy to check that  $tps((T)h_n f)=Tps(h_n)+10n+7$ . ■

### Behaviour of T:

We prove (by induction on  $n$ ) that  $(h_n)rtr:lf(f)^n$ .

If  $n=0$ , then  $(h_0)rtr:(t)r:lf(f)$ .

If  $n \neq 0$ , then  $(h_n)rtr:(r)h_{n-1}[r/g,t/x]:r:(G)(h_{n-1}[r/g,t/x])rtr:$

$lf((h_{n-1}[r/g,t/x])rtr)lz(f)( )z.$

Since  $h_{n-1}:_b$ , then  $h_{n-1}[r/g,t/x]:_b$ , and, by induction hypothesis,

$h_{n-1}[r/g,t/x]rtr:lf(f)^{n-1}$ .

Therefore  $(h_n)rtr:lf(lf(f)^{n-1})lz(f)( )z:lf(f)^n$ .

It is easy to check that  $tps((T)h_n f)=Tps(h_n)+7n+5$ . ■

## 2.3.4 The finite lists

Let  $U$  be a set of essential l-terms. We define the set of the finite lists of objects of  $U$ ,  $L_U = \{lf(x)((f)u_1)((f)u_2)\dots((f)u_n)x \text{ where } n \in \mathbf{N}, u_i \in U\}$ .

Let  $nil = lxlyy$ ,  $\underline{cons} = lxlylfla((f)x)((y)f)a$  and  $\underline{cons}' = lxlylfla((y)f)((f)x)a$ . It is easy to check that  $\underline{cons}$  and  $\underline{cons}'$  are two l-terms for the concatenation.

Let  $T_U$  be an o.m.m. for  $U$ .

Define  $T = ln(n)Hd$  where  $H = lxlylz((T_U)x)lu(y)lv(z)((\underline{cons})u)v$ , and  $d = lf(f)nil$  ;

$T = lnlf(n)K f nil$  where  $K = lxlylu((T_U)x)lv(y)(\underline{cons}')v)u$ .

$T$  and  $Tare$  o.m.m. for  $L_U$ .

Let  $h_n:lf(lf(x)((f)u_1)((f)u_2)\dots((f)u_n)x)$ , then :

$h_n \sum_{1 \leq i \leq n-1} glx(g)v_{i+1}t_i, v_{i+1}:_b u_{i+1}, t_i \sum_{1 \leq i \leq n-1} (g)v_{i+1}t_{i+1}, v_{i+1}:_b u_{i+1}, t_n \sum x.$

$T_U$  is an o.m.m. for  $U$ , therefore for all  $1 \leq i \leq n$ , there is  $t_i:u_i$ , such that

$(T_U)v_i[H/g,d/x] \sum lf(f)s_i(t_i).$

### Behaviour of T:

$(T)h_n f:(h_n)Hdf:(H)v_1[H/g,d/x]t_1[H/g,d/x]f:$

$((T_U)v_1[H/g,d/x])lu(t_1[H/g,d/x])lv(f)((\underline{cons})u)v:(lf(f)s_1(t_1))lu(t_1[H/g,d/x])lv(f)$

$((\underline{cons})u)v:(t_1[H/g,d/x])lv(f)((\underline{cons})s_1(t_1))v.$

We define a sequence of l-terms  $(d_i)_{1 \leq i \leq n} : d_1 = lv(f)((\underline{\text{cons}})s_1(t_1))v$ , and for  $1 \leq k \leq n-1$

Let  $d_{k+1} = lv(d_k)((\underline{\text{cons}})s_{k+1}(t_{k+1}))v$ .

We prove (by induction on k) that for all  $1 \leq k \leq n$  we have  $(T)h_n f : (t_k[H/g, d/x])d_k$ .

For  $k=1$  it is true.

Assume that is true for k, and prove it for  $k+1$ .

$(T)h_n f : (t_k[H/g, d/x])d_k : (H)v_{k+1}[H/g, d/x]t_{k+1}[H/g, d/x]d_k$

$((T_U)v_{k+1}[H/g, d/x])lu(t_{k+1}[H/g, d/x])lv(d_k)((\underline{\text{cons}})u)v :$

$(lf(f)s_{k+1}(t_{k+1}))lu(t_{k+1}[H/g, d/x])lv(d_k)((\underline{\text{cons}})u)v :$

$(t_{k+1}[H/g, d/x])lv(d_k)((\underline{\text{cons}})s_{k+1}(t_{k+1}))v = (t_{k+1}[H/g, d/x])d_{k+1}$ .

Therefore, in particular, for  $k=n$  we have  $(T)h_n f : (t_n[H/g, d/x])d_n = (d)d_n : (d_n)nil$ .

We prove (by induction on k) that for all  $1 \leq k \leq n$  we have

$d_k : lv(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_k(t_k))v$ .

For  $k=1$  it is true.

Assume that is true for k, and prove it for  $k+1$ .

$d_{k+1} = lv(d_k)((\underline{\text{cons}})s_{k+1}(t_{k+1}))v :$

$lz(lv(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_k(t_k))v)((\underline{\text{cons}})s_{k+1}(t_{k+1}))v :$

$lv(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_k(t_k))v)((\underline{\text{cons}})s_{k+1}(t_{k+1}))v$ .

Therefore, in particular, for  $k=n$  we have

$d_n : lv(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_n(t_n))v$

and  $(T)h_n f : (lv(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_n(t_n))nil :$

$(f)((\underline{\text{cons}})s_1(t_1))((\underline{\text{cons}})s_2(t_2)) \dots ((\underline{\text{cons}})s_n(t_n))nil = (f)s(\{((\underline{\text{cons}})t_1)((\underline{\text{cons}})t_2) \dots$

$((\underline{\text{cons}})t_n)nil\}$ ).

It is easy to check that if  $tps(T_U v_i) = Tps(v_i) + D_i$ , then  $tps((T)h_n f) = Tps(h_n) + 6n + 4$  +

### Behaviour of T:

$(T)h_n f : (h_n)K \quad f \quad nil : (K)v_1[K/g, f/x]t_1[K/g, f/x]nil : ((T_U)v_1[K/g, f/x])lv(t_1[K/g, f/x])$

$((\underline{\text{cons}}')v)nil : (lf(f)s_1(t_1))lv(t_1[K/g, f/x])((\underline{\text{cons}}')v)nil :$

$(t_1[K/g, f/x])((\underline{\text{cons}}')s_1(t_1))nil$ .

We prove (by induction on k) that for all  $1 \leq k \leq n$  we have

$(T)h_n f : (t_k[F/g, f/x])((\underline{\text{cons}}')s_k(t_k))((\underline{\text{cons}}')s_{k-1}(t_{k-1})) \dots ((\underline{\text{cons}}')s_1(t_1))nil$ .

For  $k=1$  it is true.

Assume that is true for k, and prove it for  $k+1$ .

$(T)h_n f : (t_k[K/g, f/x])((\underline{\text{cons}}')s_k(t_k)) \dots ((\underline{\text{cons}}')s_1(t_1))nil :$

$(K)v_{k+1}[K/g, f/x]t_{k+1}[K/g, f/x]((\underline{\text{cons}}')s_k(t_k)) \dots ((\underline{\text{cons}}')s_1(t_1))nil :$

$((T_U)v_{k+1}[K/g, f/x])lv(t_{k+1}[K/g, f/x])((\underline{\text{cons}}')v)((\underline{\text{cons}}')s_k(t_k)) \dots ((\underline{\text{cons}}')s_1(t_1))nil :$

$(lf(f)s_{k+1}(t_{k+1}))lv(t_{k+1}[K/g, f/x])((\underline{\text{cons}}')v)((\underline{\text{cons}}')v)((\underline{\text{cons}}')s_k(t_k)) \dots$

$((\underline{\text{cons}}')s_1(t_1))nil : (t_{k+1}[F/g, f/x])((\underline{\text{cons}}')s_{k+1}(t_{k+1})) \dots ((\underline{\text{cons}}')s_1(t_1))nil$ .

Therefore, in particular, for  $k=n$  we have

$(T)h_n f:(t_n[K/g, f/x])((\text{cons}')s_n(t_n))\dots((\text{cons}')s_1(t_1))\text{nil} =$

$(f)((\text{cons}')s_n(t_n))\dots((\text{cons}')s_1(t_1))\text{nil} = (f)s(\{((\text{cons}')t_n)\dots((\text{cons}')t_1)\text{nil}\})$ .

It is easy to check that if  $\text{tps}(T_{UV_i}) = \text{Tps}(v_i) + D_i$ , then  $\text{tps}((T)h_n f) = \text{Tps}(h_n) + 5n + 4$  +

## § 3. The directed l-calculus

### 3.1 The l[]-terms

#### Definitions.

• If  $L$  is the set of simple l-terms ( $L$  without a-equivalence), having  $V$  as set of variables, then the set of terms of **simple directed l-calculus**, denoted by  $L[]$ , is defined in the following way :

- If  $x \in V$ , then  $xAL[]$  ;
- If  $x \in V$ , and  $u \in AL[]$ , then  $lxuAL[]$  ;
- If  $u \in AL[]$ , and  $v \in AL[]$ , then  $(u)vAL[]$  ;
- If  $t \in AL[]$  is a *b-normal* l-term, such that  $Fv(t) = \{x_1, \dots, x_n\}$ , and  $a_1, \dots, a_n \in AL[]$ , then  $[t] \langle a_1/x_1, \dots, a_n/x_n \rangle \in AL[]$ .

A l[]-term of the form  $[t] \langle a_1/x_1, \dots, a_n/x_n \rangle$  is said a **box directed by t** (we also say that  $t$  is the **director** of the box).

This notation represents, intuitively, the l-term  $t$  where the free variables  $x_1, \dots, x_n$  will be replaced by  $a_1, \dots, a_n$ .

We extend the definition of the a-equivalence by :

$[u] \langle a_1/x_1, \dots, a_n/x_n \rangle :_a [v] \langle b_1/y_1, \dots, b_m/y_m \rangle$  if and only if there are permutations  $P_n$  and  $P_m$ ,  $0 \leq r \leq \inf(n, m)$ , and new variables  $z_1, \dots, z_r$ , such that :

- $Fv(u) = \{x_1, \dots, x_n\}$  and  $Fv(v) = \{y_1, \dots, y_m\}$ ,
- $u[z_1/x_1, \dots, z_r/x_r] :_a v[z_1/y_1, \dots, z_r/y_r]$ .
- $a :_a b \text{ for } 1 \leq i \leq r$ .

• The set of terms of the **directed l-calculus**, denoted by  $L[]$ , and also called **l[]-terms**, is defined by  $L[] = L[] / :_a$ .

• We will note  $\langle \mathbf{a}/\mathbf{x} \rangle$  the substitution  $\langle a_1/x_1, \dots, a_n/x_n \rangle$ . The substitution  $\langle a_1/x_1, \dots, a_n/x_n, b_1/y_1, \dots, b_m/y_m \rangle$  is denoted by  $\langle \mathbf{a}/\mathbf{x}, \mathbf{b}/\mathbf{y} \rangle$ , and the substitution  $\langle a_1[u_1/y_1, \dots, u_m/y_m]/x_1, \dots, a_n[u_1/y_1, \dots, u_m/y_m]/x_n \rangle$  is denoted by  $\langle \mathbf{a}[u_1/y_1, \dots, u_m/y_m]/\mathbf{x} \rangle$ .

• For every  $u, u_1, \dots, u_m \in AL[]$ , we extend the definitions of  $Fv(u)$  and  $u[u_1/y_1, \dots, u_m/y_m]$  by :

- $Fv([t] \langle \mathbf{a}/\mathbf{x} \rangle) = Fv(\mathbf{a}) =$
- $[t] \langle \mathbf{a}/\mathbf{x} \rangle [u_1/y_1, \dots, u_m/y_m] = [t] \langle \mathbf{a}[u_1/y_1, \dots, u_m/y_m]/\mathbf{x} \rangle$ , after a suitable renaming of the bounded variables of  $a_1, \dots, a_n$  that are free in  $u_1, \dots, u_m$ .

## 3.2 The $b[]$ -reduction

### Definitions.

• A  $l[]$ -term of the form  $(lxu)v$  is called **b-redex** ;  $u[v/x]$  is called its **contractum**.  
 A  $l[]$ -term of the form  $[t]\langle a/x \rangle$  is called **[]-redex** ; its **contractum**  $R$  is defined by induction on  $t$  :

- If  $t=x_i$   $1 \leq i \leq n$ , then  $R=a_i$  ;
- If  $t=lxu$ , then  $R=ly[u]\langle a/x, y/x \rangle$  where  $yFv(a)$  ;
- If  $t=(u)v$ , then  $R=( [u]\langle a/x \rangle ) [v]\langle a/x \rangle$ .

• We define a binary relation  $5_{b0}$  by :

$t5_{b0}t'$  if and only if  $t'$  is obtained by contracting a b-redex of  $t$ .

More precisely :

- If  $t$  is a variable,  $t5_{b0}t'$  is false for all  $t'$  ;
- If  $t=lxu$ , then  $t5_{b0}t'$  if and only if  $t'=lxu'$ , and  $u5_{b0}u'$  ;
- If  $t=(v)u$ , then  $t5_{b0}t'$  if and only if
  - $t'=(v)u'$  with  $u5_{b0}u'$  or
  - $t'=(v)u$  with  $v5_{b0}v'$  or
  - $v=lxw$ , and  $t'=w[u/x]$  ;
- If  $t=[u]\langle a/x \rangle$ , then  $t5_{b0}t'$  if and only if
  - $a_i5_{b0}a'_i$ ,  $x_iAFv(u)$   $1 \leq i \leq n$ , and  $t'=[u]\langle a_1/x_1, \dots, a_{i-1}/x_{i-1}, a'_i/x_i, a_{i+1}/x_{i+1}, \dots, a_n/x_n \rangle$ .

• We define a binary relation  $5_{[]0}$  by :

$t5_{[]0}t'$  if and only if  $t'$  is obtained by contracting a  $l[]$ -redex of  $t$ .

More precisely :

- If  $t$  is a variable,  $t5_{[]0}t'$  is false for all  $t'$  ;
- If  $t=lxu$ , then  $t5_{[]0}t'$  if and only if  $t'=lxu'$ , and  $u5_{[]0}u'$  ;
- If  $t=(v)u$ , then  $t5_{[]0}t'$  if and only if
  - $t'=(v)u'$  with  $u5_{[]0}u'$  or
  - $t'=(v)u$  with  $v5_{[]0}v'$  ;
- If  $t=[u]\langle a/x \rangle$ , then  $t5_{[]0}t'$  if and only if
  - $t'$  is the contractum of  $t$  or
  - $a_i5_{[]0}a'_i$ ,  $x_iAFv(u)$   $1 \leq i \leq n$ , and  $t'=[u]\langle a_1/x_1, \dots, a_{i-1}/x_{i-1}, a'_i/x_i, a_{i+1}/x_{i+1}, \dots, a_n/x_n \rangle$

• We define a binary relation  $5_{b[]0}$  on  $L[]$  by  $t5_{b[]0}t'$  or  $t5_{[]0}t'$ .

Therefore  $t5_{b[]0}t'$  if and only if  $t'$  is obtained by contracting a  $b[]$ -redex of  $t$ .

• We define the **b-conversion** (resp. the **[]-conversion**, resp. the  **$b[]$ -conversion**) as the reflexive and transitive closure of  $5_{b0}$  (resp.  $5_{[]0}$ , resp.  $5_{b[]0}$ ).

We have therefore  $t \mathbf{5}_b t'$  (resp.  $t \mathbf{5}_{\square} t'$ , resp.  $t \mathbf{5}_{b \square} t'$ ) if and only if there is a sequence  $t_0=t, t_1, \dots, t_{n-1}, t_n=t'$ , such that  $t_i \mathbf{5}_{b_0} t_{i+1}$  (resp.  $t_i \mathbf{5}_{\square} t_{i+1}$ , resp.  $t_i \mathbf{5}_{b \square} t_{i+1}$ ) for  $1 \leq i \leq n-1$ .

It is clear that if  $t \mathbf{5}_{b \square} t'$ , then  $Fv(t') \sqsubseteq Fv(t)$ .

• A  $\square$ -term  $t$  is said  **$\square$ -normal**, if it does not contain any redex.

A  $\square$ -term  $t$  is said  **$\square$ -normalizable**, if there is a  $\square$ -normal  $\square$ -term  $t'$ , such that  $t \mathbf{5}_{b \square} t'$ .

A  $\square$ -term  $t$  is said  **$\square$ -strongly normalizable**, if there is a no infinite sequence  $t_0=t, t_1, \dots, t_n, \dots$ , such that  $t_i \mathbf{5}_{b \square} t_{i+1}$  for  $i \geq 0$ .

**Lemma 3.2.1.**  *$t$  is  $\square$ -normal if and only if  $t \mathbf{A}$ , and  $t$  is  $b$ -normal.*

**Proof.** Clear. ■

**Lemme 3.2.2.** *A  $\square$ -reduction always terminates.*

**Proof.** Otherwise, there is an infinite sequence  $t_0, t_1, \dots, t_n, \dots$ , such that  $t_i \mathbf{5}_{\square} t_{i+1}$  for  $i \geq 0$ .

For each  $\square$ -term  $t$ , we associate an integer  **$b(t)$**  by induction on  $t$  :

- If  $t=x$ , then  $b(t)=0$  ;
- If  $t=lxu$ , then  $b(t)=b(u)$  ;
- If  $t=(u)v$ , then  $b(t)=b(u)+b(v)$  ;
- If  $t=[u] \langle \mathbf{a/x} \rangle$ , then :
  - If  $u=x_i$   $1 \leq i \leq n$ , then  $b(t)=b(a_i)+1$  ;
  - If  $u=lxv$ , then  $b(t)=b([v] \langle \mathbf{a/x}, y/x \rangle)+1 + Fv(\mathbf{a})$  ;
  - If  $u=(v)w$ , then  $b(t)=b([v] \langle \mathbf{a/x} \rangle)+b([w] \langle \mathbf{a/x} \rangle)+1$ .

**Lemma 3.2.3.**

1)  $b(t)=0$  if and only if  $t \mathbf{A}$ .

2) If  $b(a_i)=b(a'_i)$   $1 \leq i \leq n$ , then

$$b([u] \langle \mathbf{a/x} \rangle) = b([u] \langle a_1/x_1, \dots, a_{i-1}/x_{i-1}, a'_i/x_i, a_{i+1}/x_{i+1}, \dots, a_n/x_n \rangle).$$

3) If  $b(a_i) > b(a'_i)$ , and  $x_i \mathbf{A} Fv(u)$   $1 \leq i \leq n$ , then

$$b([u] \langle \mathbf{a/x} \rangle) > b([u] \langle a_1/x_1, \dots, a_{i-1}/x_{i-1}, a'_i/x_i, a_{i+1}/x_{i+1}, \dots, a_n/x_n \rangle).$$

**Proof.** By induction on  $t$ . (resp.  $u$ ) for 1) (resp 2), 3)). ■

**Lemma 3.2.4.** *If  $t \mathbf{5}_{\square} t'$ , then  $b(t) > b(t')$ .*

**Proof.** By induction on  $t$ . The only interesting case is  $t=[u] \langle \mathbf{a/x} \rangle$ . Then :

- If  $u=x_i$   $1 \leq i \leq n$ , then  $t'=a_i$ , and  $b(t)=b(a_i)+1 > b(t')$ .
- If  $u=lxv$ , then  $t'=[u] \langle \mathbf{a/x}, y/x \rangle + Fv(\mathbf{a})$ , therefore, by lemma 3.2.3,

$b(t)=b([u]\langle \mathbf{a}/\mathbf{x}, y/x \rangle)+1 > b(t')$ .

- If  $u=(v)w$ , then  $t'=( [v]\langle \mathbf{a}/\mathbf{x} \rangle)[w]\langle \mathbf{a}/\mathbf{x} \rangle$ , and  $b(t)=b([v]\langle \mathbf{a}/\mathbf{x} \rangle)+b([w]\langle \mathbf{a}/\mathbf{x} \rangle)+1 > b(t')$ .

- If  $a_i \mathfrak{F}_0 a'_i$ ,  $x_i \mathfrak{F}_v(u)$   $1 \leq i \leq n$ , and  $t'=[u]\langle a_1/x_1, \dots, a_{i-1}/x_{i-1}, a'_i/x_i, a_{i+1}/x_{i+1}, \dots, a_n/x_n \rangle$ . By induction hypothesis, we have  $b(a_i) > b(a'_i)$ , therefore, by lemma 3.2.3,  $b(t) > b(t')$ . ■

Therefore, by lemma 3.2.4, there is an infinite sequence  $b(t_0), b(t_1), \dots, b(t_n), \dots$ , such that  $b(t_i) > b(t_{i+1})$  for  $i \geq 0$ . A contradiction. ■ (of lemma 3.2.2)

**Definition.** For each  $l[]$ -term  $t$ , we associate a  $l$ -term  $l(t)$  by induction on  $t$  :

- If  $t=x$ , then  $l(t)=x$  ;

- If  $t=lxu$ , then  $l(t)=lxl(u)$  ;

- If  $t=(u)v$ , then  $l(t)=(l(u))l(v)$  ;

- If  $t=[u]\langle \mathbf{a}/\mathbf{x} \rangle$ , then  $l(t)=u[l(a_1)/x_1, \dots, l(a_n)/x_n]$ .

It is clear that for  $t \in AL[]$ ,  $Fv(t)=Fv(l(t))$ .

**Theorem 3.2.5.**  $t$  is  $b[]$ -strongly normalizable if and only if  $l(t)$  is strongly normalizable.

**Theorem 3.2.6 (Church-Rosser theorem).** Assume  $t_0 \mathfrak{F}_{b[]} t_1$ , and  $t_0 \mathfrak{F}_{b[]} t_2$ , then there is a  $t_3$ , such that  $t_1 \mathfrak{F}_{b[]} t_3$  and  $t_2 \mathfrak{F}_{b[]} t_3$ .

**Proof of theorem 3.2.5.**

1 If  $l(t)$  is not strongly normalizable, then there is an infinite sequence  $t_0=l(t), t_1, \dots, t_n, \dots$ , such that  $t_i \mathfrak{F}_{b_0} t_{i+1}$  for all  $i \geq 0$ .

**Lemma 3.2.7.** If  $t \mathfrak{F}_{[]} t'$ , then  $l(t)=l(t')$ .

**Proof.** By induction on  $t$ . ■

**Lemma 3.2.8.**

1)  $[u]\langle \mathbf{a}/\mathbf{x} \rangle \mathfrak{F}_{[]} u[a_1/x_1, \dots, a_n/x_n]$ .

2) If  $u_i \mathfrak{F}_{[]} v_i$   $1 \leq i \leq n$ , then  $u[u_1/x_1, \dots, u_n/x_n] \mathfrak{F}_{[]} u[v_1/x_1, \dots, v_n/x_n]$ .

**Proof.** By induction on  $u$ . ■

**Lemma 3.2.9.** If  $t$  is a  $l[]$ -term, then  $t \mathfrak{F}_{[]} l(t)$ .

**Proof.** By induction on  $t$ . The only interesting case is  $t=[u]\langle \mathbf{a}/\mathbf{x} \rangle$ .

By lemma 3.2.8,  $[u] \langle \mathbf{a}/\mathbf{x} \rangle \mathcal{S}_{\square} u[a_1/x_1, \dots, a_n/x_n]$ . By induction hypothesis, we have  $a_i \mathcal{S}_{\square} l(a_i) \ 1 \leq i \leq n$ , therefore, by lemma 3.2.8,  $t \mathcal{S}_{\square} u[l(a_1)/x_1, \dots, l(a_n)/x_n] = l(t)$ . ■

By lemma 3.2.9,  $t \mathcal{S}_{\square} l(t)$ , therefore  $t$  is not  $b[\ ]$ -strongly normalizable. A contradiction. ■ (of 1 theorem 3.2.5).

0 (theorem 3.2.5) If  $t$  is not  $b[\ ]$ -strongly normalizable then there is an infinite sequence  $t_0 = t, t_1, \dots, t_n, \dots$ , such that  $t_i \mathcal{S}_{b_0} t_{i+1}$  or  $t_i \mathcal{S}_{\square} t_{i+1}$  for  $i \geq 0$ .

**Lemma 3.2.10.**  $l(u[v/x]) = l(u)[l(v)/x]$ .

**Proof.** By induction on  $u$ . ■

**Lemma 3.2.11.** If  $u \mathcal{S}_{b_0} v$ , then  $l(u) \mathcal{S}_{b_0} l(v)$ .

**Proof.** By induction on  $u$ . The only non-trivial case is  $u = (lxt)w$  : we then have  $v = t[w/x]$ , therefore, by lemma 3.2.10,  $l(u) = (lxl(t))l(w) \mathcal{S}_{b_0} l(t)[l(w)/x] = l(v)$ . ■

**Corollary 3.2.12.** If  $u \mathcal{S}_{b[\ ]} v$ , then  $l(u) \mathcal{S}_{b[\ ]} l(v)$ .

**Proof.** Use lemmas 3.2.7 and 3.2.11. ■

By lemma 3.2.2, and lemma 3.2.11, there is an infinite sequence  $t'_0 = l(t), t'_1, \dots, t'_n, \dots$ , such that  $t'_i \mathcal{S}_{b_0} t'_{i+1}$  for all  $i \geq 0$ , therefore  $l(t)$  is not strongly normalizable. A contradiction. ■ (of 0 theorem 3.2.5)

**Proof of theorem 3.2.6.** If  $t_0 \mathcal{S}_{b[\ ]} t_1$ , and  $t_0 \mathcal{S}_{b[\ ]} t_2$ , then, by corollary 3.2.12,  $l(t_0) \mathcal{S}_{b[\ ]} l(t_1)$ , and  $l(t_0) \mathcal{S}_{b[\ ]} l(t_2)$ . Therefore, by the Church-Rosser theorem of  $l$ -calculus, there is a  $t_3$ , such that  $l(t_1) \mathcal{S}_{b[\ ]} t_3$ , and  $l(t_2) \mathcal{S}_{b[\ ]} t_3$ , therefore, by lemma 3.2.9,  $t_1 \mathcal{S}_{b[\ ]} t_3$ , and  $t_2 \mathcal{S}_{b[\ ]} t_3$ . ■

### Remarks.

- By the Church-Rosser theorem, the  $b[\ ]$ -normal form is unique.
- We define the  **$b[\ ]$ -equivalence** (denoted by  $:\mathbf{b}[\ ]$ ), as the symmetric closure of  $\mathcal{S}_{b[\ ]}$  ; In other words :  $t:\mathbf{b}[\ ]t'$  if there are  $t_0 = t, t_1, \dots, t_n = t'$  with  $t_i \mathcal{S}_{b[\ ]} t_{i+1}$  or  $t_{i+1} \mathcal{S}_{b[\ ]} t_i \ 0 \leq i \leq n-1$ . By the Church-Rosser theorem :  $t:\mathbf{b}[\ ]t'$  if and only if there is a  $l[\ ]$ -term  $u$ , such that  $t \mathcal{S}_{b[\ ]} u$  and  $t' \mathcal{S}_{b[\ ]} u$ , and a  $l[\ ]$ -term  $t$  is  $b[\ ]$ -normalizable if and only if there is a  $b[\ ]$ -normal  $l[\ ]$ -term  $u$  such that  $t:\mathbf{b}[\ ]u$ .



### 3.3 The $b[\ ]$ -left reduction

#### Definitions.

- A sequence of symbols of the form  $l$  or  $[\ ]$  corresponds to a redex. We may then define the **leftmost  $b$ -redex** and the **leftmost  $[\ ]$ -redex** of  $t$ . If  $t'$  is the  $l[\ ]$ -term obtained by contracting this redex, we say that :

$t$  gives  $t'$  by  **$b_0$ -left reduction** (resp. by  **$[\ ]_0$ -left reduction**, resp. by  **$b[\ ]_0$ -left reduction**), and write by  **$t \sum \sum_{b_0} t'$**  (resp.  **$t \sum \sum_{[\ ]_0} t'$** , resp.  **$t \sum \sum_{b[\ ]_0} t'$** ), if it is a  $b$ -redex (resp. a  $[\ ]$ -redex, resp. a  $b$ -redex or a  $[\ ]$ -redex ).

- We say that  $t$  reduces to  $t'$  by  **$b$ -left reduction** (resp.  **$[\ ]$ -left reduction**, resp.  **$b[\ ]$ -left reduction**), and we write  **$t \sum \sum_b t'$**  (resp.  **$t \sum \sum_{[\ ]} t'$** , resp.  **$t \sum \sum_{b[\ ]} t'$** ) if and only if  $t'$  is obtained from  $t$  by a sequence of  $b_0$ -left reductions (resp. of  $[\ ]_0$ -left reductions, resp. of  $b[\ ]_0$ -left reductions).

- A  $l[\ ]$ -term  $t$  is said  **$b[\ ]$ -left normalizable** if and only if there is a  $b[\ ]$ -normal  $l[\ ]$ -term  $t'$ , such that  $t \sum \sum_{b[\ ]} t'$ .

**Theorem 3.3.1.**  *$u$  is  $b[\ ]$ -left normalizable if and only if  $l(u)$  is left normalizable.*

**Theorem 3.3.2 (normalization theorem).**  *$u$  is  $b[\ ]$ -normalizable if and only if  $u$  is  $b[\ ]$ -left normalizable.*

#### Proof of theorem 3.3.1.

**1** Use lemmas 3.2.7 and 3.3.3.

#### Lemma 3.3.3.

1) *If  $R$  is the leftmost  $b$ -redex of  $u$ , then  $l(R)$  is the leftmost redex of  $l(u)$ .*

2) *If  $u \sum \sum_{b_0} v$ , then  $l(u) \sum \sum_{l_0} l(v)$ .*

#### Proof.

1) Clear.

2) By induction on  $u$ . The only non-trivial case is  $u=(lxt)w$  : then we have  $v=t[w/x]$ , then, by lemma 3.2.10,  $l(u)=(lxl(t))l(w) \sum \sum_{l_0} l(t)[l(w)/x]=l(t[w/x])$ . ■

**0** If not, there is an infinite sequence of  $l[\ ]$ -terms  $u_0=u, u_1, \dots, u_n, \dots$ , such that  $u_i \sum \sum_{b_0} u_{i+1}$  or  $u_i \sum \sum_{[\ ]_0} u_{i+1}$  for  $i \geq 0$ . Therefore, by lemmas 3.2.2, 3.2.7, and 3.3.3, there is an infinite sequence of  $l[\ ]$ -terms  $v_0=l(u), v_1, \dots, v_n, \dots$ , such that  $v_i \sum \sum_{l_0} v_{i+1}$  for  $i \geq 0$ , therefore  $l(u)$  is not left normalizable. A contradiction. ■

### Proof of theorem 3.3.2.

0 Clear.

1 If  $u$  is  $b[\ ]$ -normalizable, then  $l(u)$  is normalizable (same proof as theorem 3.3.1 1). By the normalization theorem of  $l$ -calculus,  $l(u)$  is left normalizable, therefore, by theorem 3.3.1,  $u$  is  $b[\ ]$ -left normalizable. ■

## 3.4 The $b[\ ]$ -head reduction

**Proposition 3.4.1.** *Every  $l[\ ]$ -term  $t$  can be - uniquely - written as  $lx_1 \dots lx_n(R)t_1 \dots t_m$   $n, m \geq 0$ ,  $R$  being a variable or a redex*

**Proof.** By induction on  $t$ . ■

### Definitions.

- Let  $t$  be a  $l[\ ]$ -term, then, by proposition 3.4.1,  $t = lx_1 \dots lx_n(R)t_1 \dots t_m$ .

If  $R$  is a variable, we say that  $t$  is a  **$b[\ ]$ -head normal form**.

If  $R$  is a redex, we say that  $R$  is the **head redex** of  $t$ .

If  $t'$  is the  $l[\ ]$ -term obtained from  $t$  by contracting its head redex, we say that :

$t$  gives  $t'$  by  **$b_0$ -head reduction** (resp. by  **$[\ ]_0$ -head reduction**, resp. by  **$b[\ ]_0$ -head reduction**), and we write  $t \sum_{b_0} t'$  (resp.  $t \sum_{[\ ]_0} t'$ , resp.  $t \sum_{b[\ ]_0} t'$ ), if the head redex is a  $b$ -redex (resp. a  $[\ ]$ -redex, resp. a  $b$ -redex or a  $[\ ]$ -redex).

- We say that  $t$  reduces to  $t'$  by  **$b$ -head reduction** (resp.  **$[\ ]$ -head reduction**, resp.  **$b[\ ]$ -head reduction**), and we write  $t \sum_b t'$  (resp.  $t \sum_{[\ ]} t'$ , resp.  $t \sum_{b[\ ]} t'$ ) if and only if  $t'$  is obtained from  $t$  by a sequence of  $b_0$ -head reduction (resp.  $[\ ]_0$ -head reduction, resp.  $b[\ ]_0$ -head reduction).

A  $b[\ ]$ -head reduction is, in particular, a  $b[\ ]$ -left reduction.

- If  $t \sum_{b[\ ]} t'$ , we denote by  $n(t, t')$ , the number of steps to go from  $t$  to  $t'$ .

- A  $l[\ ]$ -term  $t$  is said  **$b[\ ]$ -solvable** if and only if for every  $l[\ ]$ -term  $u$ , there are variables  $x_1, \dots, x_k$ , and  $l[\ ]$ -terms  $u_1, \dots, u_k, v_1, \dots, v_l$ ,  $k, l \geq 0$ , such that  $(t[u_1/x_1, \dots, u_k/x_k])v_1 \dots v_l ;_{b[\ ]} u$ .

**Theorem 3.4.2.** *If  $t \sum_{b[\ ]} t'$ , then for every  $u_1, \dots, u_r \in L[\ ]$  :*

1) *There is  $v \in L$ , such that  $(t)u_1 \dots u_r \sum_{b[\ ]} v$ ,  $(t')u_1 \dots u_r \sum_{b[\ ]} v$ , and  $n((t)u_1 \dots u_r, v) = n((t')u_1 \dots u_r, v) + n(t, t')$ .*

2)  *$t[u_1/x_1, \dots, u_r/x_r] \sum_{b[\ ]} t'[u_1/x_1, \dots, u_r/x_r]$ , and  $n(t[u_1/x_1, \dots, u_r/x_r], t'[u_1/x_1, \dots, u_r/x_r]) = n(t, t')$ .*

### Remarks.

- 1) shows that to make the  $b[]$ -head reduction of  $(t)u_1 \dots u_n$ , it is equivalent (same result, and same number of steps) to make some steps in the  $b[]$ -head reduction of  $t$ , and then make the  $b[]$ -head reduction of  $(t')u_1 \dots u_n$ .
- 2) shows that to make the  $b[]$ -head reduction of  $t[u_1/x_1, \dots, u_n/x_n]$ , it is equivalent (same result, and same number of steps) to make some steps in the  $b[]$ -head reduction of  $t$ , and then make the  $b[]$ -head reduction of  $t'[u_1/x_1, \dots, u_n/x_n]$ .

**Corollary 3.4.3.** *Let  $t, u_1, \dots, u_n, v_1, \dots, v_m \in AL[]$ . If the  $b[]$ -head reduction of  $(t[u_1/x_1, \dots, u_n/x_n])v_1 \dots v_m$  terminates, then the  $b[]$ -head reduction of  $t$  terminates.*

**Proof.** Use theorem 3.4.2. ■

**Theorem 3.4.4 (resolution theorem).** *The following conditions are equivalent :*

- 1)  $t$  is  $b[]$ -solvable ;
- 2) the  $b[]$ -head reduction of  $t$  terminates ;
- 3)  $t$  is  $b[]$ -equivalent to a  $b[]$ -head normal form.

**Proof of theorem 3.4.2.** It is enough to do the proof for one step of reduction.

1) By induction on  $r$  ; it is enough to do the proof for  $r=1$ . Then  $t = lx_1 \dots lx_n(R)t_1 \dots t_m$ , and  $t' = lx_1 \dots lx_n(R')t_1 \dots t_m$  where  $R'$  is the contractum of  $R$ .

If  $n=0$ , then  $(t)u = (R)t_1 \dots t_m u$ , and  $(t')u = (R')t_1 \dots t_m u$ , therefore  $(t)u \sum_{b[]} v$ , where  $v = (t')u$ .

If  $n \geq 1$ , then one step of  $b[]$ -head reduction of  $(t)u$  gives  $lx_2 \dots lx_n(\underline{R})t_1 \dots t_m$  (where  $\underline{w} = w[u/x_1]$  for every  $w \in AL[]$ ). One step of  $b[]$ -head reduction of  $(t')u$  gives  $lx_2 \dots lx_n(\underline{R}')t_1 \dots t_m$ .

**Lemma 3.4.5.** *If  $R$  a redex,  $R'$  its contractum, and  $u_1, \dots, u_m \in AL[]$ , then  $R[u_1/y_1, \dots, u_m/y_m]$  is a redex, and  $R'[u_1/y_1, \dots, u_m/y_m]$  is its contractum.*

**Proof.** If  $R$  is a  $b$ -redex, then  $R = (lxu)v$ , and  $R' = u[v/x]$ .

$R[u_1/y_1, \dots, u_m/y_m] = (lxu[u_1/y_1, \dots, u_m/y_m])v[u_1/y_1, \dots, u_m/y_m]$  is a  $b$ -redex, and its contractum is  $u[u_1/y_1, \dots, u_m/y_m][v[u_1/y_1, \dots, u_m/y_m]/x] = R'[u_1/y_1, \dots, u_m/y_m]$ .

If  $R$  is a  $[\ ]$ -redex, then  $R = [t] \langle \mathbf{a}/\mathbf{x} \rangle :$

- If  $t = x_i$   $1 \leq i \leq n$ , then  $R' = a_i$ .  $R[u_1/y_1, \dots, u_m/y_m] = [t] \langle \mathbf{a}[u_1/y_1, \dots, u_m/y_m]/\mathbf{x} \rangle$  is a  $[\ ]$ -redex, and its contractum is  $a_i[u_1/y_1, \dots, u_m/y_m] = R'[u_1/y_1, \dots, u_m/y_m]$ .

- If  $t = lxu$ , then  $R' = ly[u] \langle \mathbf{a}/\mathbf{x}, y/x \rangle$  where  $y Fv(\mathbf{a})$ .

$R[u_1/y_1, \dots, u_m/y_m] = [lxu] \langle \mathbf{a}[u_1/y_1, \dots, u_m/y_m]/\mathbf{x} \rangle$  is a  $[\ ]$ -redex, and its contractum is  $ly[u] \langle \mathbf{a}/\mathbf{x}, y/x \rangle [u_1/y_1, \dots, u_m/y_m] = R'[u_1/y_1, \dots, u_m/y_m]$  where  $y Fv(\mathbf{a})$ "

- If  $t = (u)v$ , then  $R' = ([u] \langle \mathbf{a}/\mathbf{x} \rangle)[v] \langle \mathbf{a}/\mathbf{x} \rangle$ .

$R[u_1/y_1, \dots, u_m/y_m] = ([u] \langle \mathbf{a}/\mathbf{x} \rangle) [v] \langle \mathbf{a}[u_1/y_1, \dots, u_m/y_m]/\mathbf{x} \rangle$  is a  $[\ ]$ -redex, and its contractum is  $([u] \langle \mathbf{a}/\mathbf{x} \rangle) [v] \langle \mathbf{a}/\mathbf{x} \rangle [u_1/y_1, \dots, u_m/y_m] = R'[u_1/y_1, \dots, u_m/y_m]$ . ■

By lemma 3.4.5,  $(t)u_{\sum b[\ ]}v$ , and  $(t')u_{\sum b[\ ]}v$  where  $v = l x_2 \dots l x_n (R') t_1 \dots t_m$ .

2) Same proof as 1). ■

### Proof of theorem 3.4.4.

**1)12)** If  $t$  is  $b[\ ]$ -solvable, then there are variables  $x_1, \dots, x_k$ , and terms  $u_1, \dots, u_k, v_1, \dots, v_l$  ( $k, l \geq 0$ ), such that  $(t[u_1/x_1, \dots, u_k/x_k])v_1 \dots v_l; b[\ ]lxx$ , therefore, by the Church-Rosser theorem,  $(t[u_1/x_1, \dots, u_k/x_k])v_1 \dots v_l \sum b[\ ]lxx$ , therefore, by corollary 3.4.3, the  $b[\ ]$ -head reduction of  $t$  terminates.

**2)13)** Clear.

**3)11)** Assume  $t; b[\ ]l x_1 \dots l x_n (y) t_1 \dots t_m$ , and let  $u$  be a  $l[\ ]$ -term :

- If  $y = x_i$   $1 \leq i \leq n$ , then  $((t)x_1 \dots x_{i-1})l y_1 \dots l y_m u x_{i+1} \dots x_n; b[\ ]u$  where  $y_j Fv(u)$   $1 \leq j \leq m$ .

- If  $y \neq x_i$   $1 \leq i \leq n$ , then  $(t[l y_1 \dots l y_m u / y])x_1 \dots x_n; b[\ ]u$  where  $y_j Fv(u)$   $1 \leq j \leq m$ .

Therefore  $t$  is  $b[\ ]$ -solvable. ■

**Lemma 3.4.6.** *If  $u \sum b_0 v$ , then  $l(u) \sum_0 l(v)$ .*

**Proof.** Same proof as lemma 3.4.5. ■

**Theoreme 3.4.7.**  *$u$  is  $b[\ ]$ -solvable if and only if  $l(u)$  is solvable.*

**Proof.**

**1** Use lemmas 3.2.7 and 3.4.6.

**0** Otherwise there is an infinite sequence of  $l[\ ]$ -terms  $u_0 = u, u_1, \dots, u_n, \dots$ , such that  $u_i \sum b_0 u_{i+1}$  or  $u_i \sum_{[\ ]} u_{i+1}$  for  $i \geq 0$ . Therefore, by lemmas 3.2.7, 3.4.6, and 3.2.2, there is an infinite sequence of  $l[\ ]$ -terms  $v_0 = l(u), v_1, \dots, v_n, \dots$ , such that  $v_i \sum_0 v_{i+1}$  for  $i \geq 0$ , therefore  $l(u)$  is unsolvable. A contradiction. ■

## § 4. An equivalent definition for storage operators

**Theorem 4.1.** *Let  $t$  be a closed  $b$ -normal  $l$ -term, and  $T$  a closed  $l$ -term.  $T$  is an o.m.m. for  $t$  if and only if there is a  $l$ -term  $t_i; b y t$ , such that*

$$T[t] f \sum b[\ ](f) t_i [ [t_1] \langle \mathbf{a}_1/\mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m/\mathbf{x}_m \rangle / y_m ].$$

To prove this theorem we need some definitions

**Definition.** Let  $t$  be a  $b$ -normal  $l$ -term, and  $u$  a  $l[]$ -term. We say that  $u$  is **directed by  $t$**  if and only if the directors of boxes of  $u$  are subterms of  $t$ .

More precisely  $u$  is directed by  $t$  if and only if :

- If  $u=x$ , then  $u$  is directed by  $t$  ;
- If  $u=lxv$ , then  $u$  is directed by  $t$  if and only if  $v$  is directed by  $t$  ;
- If  $u=(v)w$ , then  $u$  is directed by  $t$  if and only if  $v$  and  $w$  are directed by  $t$  ;
- If  $u=[v]<a/x>$ , then  $u$  is directed by  $t$  if and only if  $v$  is a subterm de  $t$ , and for all  $1 \leq i \leq n$ ,  $a_i$  is directed by  $t$ .

**Lemma 4.2.**

- 1) If  $u$  and  $v$  are directed by  $t$ , then  $u[v/x]$  is directed by  $t$ .
- 2) If  $u$  is directed by  $t$ , and  $u \mathcal{S}_{b[]} v$ , then  $v$  is directed by  $t$ .

**Proof.** By induction on  $u$ . ■

**Definition.** Let  $t$  be a  $b$ -normal  $l$ -term. A  **$t$ -special application**  $h$  is a function from  $ST(t)$  to  $L$  which satisfies the following properties :

- $h(x) \sum x$  ;
- $h(lxu) \sum lxh(u)$  ;
- $h((u)v) \sum (h(u))h(v)$ .

**Lemma 4.3.** If  $h$  is a  $t$ -special application, then, for every  $u \in ST(t)$ ,  $h(u) :_b u$ .

**Proof.** by induction on  $u$ . ■

**Lemma 4.4.** Let  $t$  be a  $b$ -normal  $l$ -term, and  $u \in ST(t)$ . For every  $h_u :_b u$ , there is a  $t$ -special application  $h$ , such that  $h(u) = h_u$ .

**Proof.** Let  $v \in ST(t)$  ; we define  $h(v)$  as follows :

- If  $v \in ST(u)$ ,  $h(v)$  is defined by induction on  $li(v) = lg(u) - lg(v)$ , and we check that  $h(v) :_b v$ .
  - If  $li(v) = 0$ , then  $v = u$ . Take  $h(v) = h_u$ , we have  $h(v) :_b v$ .
  - If  $li(v) \geq 1$ , then  $v$  is a proper subterm of  $u$  :
    - If there is an  $x$ , such that  $lxv \in ST(u)$  then by induction hypothesis, we have  $h(lxv) :_b lxv$ , therefore  $h(lxv) \sum lxh_v$  where  $h_v :_b v$ . Take  $h(v) = h_v$ , we have  $h(v) :_b v$ .
    - If there is  $w \in ST(t)$ , such that  $(v)w \in ST(t)$  then by induction hypothesis, we have  $h((v)w) :_b (v)w$ . Since  $t$  is  $b$ -normal, then  $h((v)w) \sum (h_v)h_w$  where  $h_v :_b v$  and  $h_w :_b w$ .

Take  $h(v)=h_v$ , we have  $h(v):_b v$ .

- If there is  $wAST(t)$ , such that  $(w)vAST(t)$  then by induction hypothesis, we have  $h((w)v):_b(w)v$ . Since  $t$  is  $b$ -normal, then  $h((w)v)\Sigma(h_w)h_v$  where  $h_v:_b v$ , and  $h_w:_b w$ .

Take  $h(v)=h_v$ , we have  $h(v):_b v$ .

- If  $uAST(v)\setminus\{v\}$ , take  $h(v)$  the  $l$ -term  $v$  where  $u$  is replaced by  $h(u)$ , we have  $h(v):_b v$ .

- Otherwise, we put  $h(v)=v$ .

By construction,  $h$  is a  $t$ -special application. ■

**Definition.** Let  $t$  be a  $b$ -normal  $l$ -term, and  $h$  a  $t$ -special application. The  **$t$ -special substitution**  $S_h$  is the function from the set of  $l[\ ]$ -terms directed by  $t$  into  $L$  defined by induction :

- If  $u=x$ , then  $S_h(u)=x$  ;

- If  $u=lxv$ , then  $S_h(u)=lyS_h(v[y/x])$  where  $yFv(h(t))$  ;

- If  $u=(v)w$ , then  $S_h(u)=(S_h(v))S_h(w)$  ;

- If  $u=[v]\langle a/x \rangle$ , then  $S_h(u)=h(v)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n]$ .

A  $t$ -special substitution is the function  $S_h$  associated to a  $b$ -normal  $l$ -term  $t$ , and some  $t$ -special application  $h$ .

It is easy to see that if  $u$  does not contain boxes, then  $S_h(u)=u$ .

**Lemma 4.5.** *If  $y_1, \dots, y_m Fv(h(t))$ , then*

$$S_h(u[v_1/y_1, \dots, v_m/y_m]) = S_h(u)[S_h(v_1)/y_1, \dots, S_h(v_m)/y_m].$$

**Proof.** By induction on  $u$ . ■

**Lemma 4.6.**  $S_h(u):_b l(u)$ .

**Proof.** By induction on  $u$ . ■

**Lemma 4.7.** *If  $u \Sigma_{b[\ ]} v$ , then  $S_h(u):S_h(v)$ .*

**Proof.** It is enough to do the proof for one step of reduction.

Let  $u=lx_1 \dots lx_n(R)u_1 \dots u_m$ , and  $v=lx_1 \dots lx_n(R')u_1 \dots u_m$  where  $R'$  is the contractum of redex  $R$  :

If  $R=(lxa)b$ , then  $R'=a[b/x]$ .

$$\begin{aligned} S_h((lxa)b) &= (lyS_h(a[y/x]))S_h(b)\Sigma S_h(a[y/x])[S_h(b)/y] = S_h(a)[y/x][S_h(b)/y] \\ &= S_h(a)[S_h(b)/x], \text{ therefore, by lemma 4.5, } S_h((lxa)b)\Sigma S_h(a[b/x]). \end{aligned}$$

If  $R=[u]\langle a/x \rangle$  :

- If  $u=x_i$   $1 \leq i \leq n$ , then  $R'=a_i$ , and  $S_h(R)=S_h(R')$ .

- If  $u=lxv$ , then  $R'=ly[v]<a/x, y/x>$  where  $yFv(a)$ .  
 $S_h(R)=h(u)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n] \sum lxh(v)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n]=$   
 $lzh(v)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n, z/x]$  where  $zFv(h(t))Fv(a)$ , therefore  
 $S_h(R) \sum S_h(R')$ .

- If  $u=(c)d$ , then  $R'=(c)<a/x>[d]<a/x>$   
 $S_h(R)=h(u)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n] \sum (h(c))h(d)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n]=$   
 $(h(c)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n])h(d)[S_h(a_1)/x_1, \dots, S_h(a_n)/x_n]$ ,  
therefore  $S_h(R) \sum S_h(R')$ . ■

**Corollary 4.8.**  $u$  is  $b[]$ -solvable if and only if  $S_h(u)$  is solvable.

**Proof.**

1 Use lemma 4.7.

0  $S_h(u)$  is solvable, therefore, by lemma 4.6,  $l(u)$  is solvable, therefore, by theorem 3.4.7,  $u$  is  $b[]$ -solvable. ■

**Definition.** We say that a  $l[]$ -term  $t$  is **good** if and only if there is a  $l$ -term  $u$ , such that  $t=u[[t_1]<a_1/x_1>/y_1, \dots, [t_m]<a_m/x_m>/y_m]$ , and for all  $1 \leq i \leq m$  if  $a_i = a_{1,i}, \dots, a_{n_i,i}$ , then  $a_{j,i}$  is good  $1 \leq j \leq n_i$ .

It is clear that we have :

- $x$  is good ;
- If  $lxt$  is good, then  $t$  is good ;
- $(u)v$  is good if and only if  $u$  and  $v$  are good ;
- $[w]<a/x>$  is good if and only if  $a_i$  is good  $1 \leq i \leq n$ .

**Example.** The  $l[]$ -term  $[x_1]<x/x_1>$  is good, but the  $l[]$ -term  $lx[x_1]<x/x_1>$  is not. Indeed, the variable  $x$  becomes bounded, and so we can not find a  $l$ -term  $u$ , such that  $lx[x_1]<x/x_1> = u[[t]<a/x>/y]$ .

**Lemma 4.9.** If  $t, v_1, \dots, v_r$  are good, then  $t[v_1/y_1, \dots, v_r/y_r]$  is good.

**Proof.** By induction on  $t$ . ■

**Definitions.**

- A **['-redex** is a  $l[]$ -term of the form  $([ly_1 \dots ly_m(y)u_1 \dots u_r]<a/x>)v_1 \dots v_m$ . Its contractum  $R$  is defined by :  $R=(b)[u_1]<a/x, v/y> \dots [u_r]<a/x, v/y>$  where  $b=v_i$  if  $y=y_i$   $1 \leq i \leq m$ , and  $b=a_i$  if  $y=x_i$   $1 \leq i \leq n$ .

It is easy to see that if  $R$  is a  $['$ -redex, and  $R'$  its contractum, then  $R \sum_{b[]} R'$ .

Let  $t = l x_1 \dots l x_n (R) t_1 \dots t_m$  where  $R$  is a  $l[\ ]$ -redex. If  $t'$  is the  $l[\ ]$ -term obtained from  $t$  by contracting the  $l[\ ]$ -redex  $R$ , we say that  $t$  gives  $t'$  by a  $l[\ ]_0$ -**head reduction**, and we write  $t \sum_{l[\ ]_0} t'$ .

We say that  $t$  reduces to  $t'$  by  $l[\ ]_0$ -**head reduction**, and we write  $t \sum_{l[\ ]_0} t'$  if and only if  $t'$  is obtained from  $t$  by a sequence of  $l[\ ]_0$ -head reductions.

- If  $t'$  is the  $l[\ ]$ -term obtained from  $t$  by contracting its head redex (b-redex or  $l[\ ]$ -redex), we say that  $t$  gives  $t'$  by **b** $l[\ ]_0$ -**head reduction**, and we write  $t \sum_{b l[\ ]_0} t'$ .

We say that  $t$  reduces to  $t'$  by **b** $l[\ ]_0$ -**head reduction**, and we write  $t \sum_{b l[\ ]_0} t'$  if and only if  $t'$  is obtained from  $t$  by a sequence of **b** $l[\ ]_0$ -head reductions.

- A head reduction  $t \sum_{b l[\ ]_0} t'$  is said **complete** if and only if for every  $l[\ ]$ -term  $u$ , if  $t' \sum_{b l[\ ]_0} u$ , then  $t' = u$ .

**Lemma 4.10.**

1) If  $f$  is a variable, and  $t \sum_{b l[\ ]_0} (f) u_1, \dots, u_r$ , then there is a sequence  $t_0 = t, t_1, \dots, t_n = (f) u_1, \dots, u_r$ , such that  $t_i \sum_{b l[\ ]_0} t_{i+1}$  is complete or  $t_i \sum_{l[\ ]_0} t_{i+1}$   $0 \leq i \leq n-1$ .

2) If moreover  $t$  is directed by  $u$ , then every director of  $t_i$   $0 \leq i \leq n$  is an element of  $STE(u)$ .

**Proof.**

1) If  $t \sum_{b l[\ ]_0} (f) u_1, \dots, u_r$ , then there is a sequence  $t = (v_0) w_0, (v_1) w_1, \dots, (v_m) w_m = (f) u_1, \dots, u_r$ , such that  $(v_i) w_i \sum_{b l[\ ]_0} (v_{i+1}) w_{i+1}$  or  $(v_i) w_i \sum_{l[\ ]_0} (v_{i+1}) w_{i+1}$   $0 \leq i \leq m-1$ . If  $(v_i) w_i \sum_{l[\ ]_0} (v_{i+1}) w_{i+1}$   $0 \leq i \leq m-1$ , then  $(v_i) w_i = (l y_1 \dots l y_p (y) d_1 \dots d_q) \langle \mathbf{a} / \mathbf{x} \rangle b_1 \dots b_p c_1 \dots c_s$ . Therefore there is  $j > i$ , such that  $(v_i) w_i \sum_{l[\ ]_0} (v_j) w_j$ , therefore there is a sequence  $t = (v'_0) w'_0, (v'_1) w'_1, \dots, (v'_k) w'_k = (f) u_1, \dots, u_r$ , such that  $(v'_i) w'_i \sum_{b l[\ ]_0} (v'_{i+1}) w'_{i+1}$  or  $(v'_i) w'_i \sum_{l[\ ]_0} (v'_{i+1}) w'_{i+1}$   $0 \leq i \leq k-1$ . Gathering consecutive b-reductions, it is clear that we can suppose that the b-reductions  $(v'_i) w'_i \sum_{b l[\ ]_0} (v'_{i+1}) w'_{i+1}$  are complete.

2) Easy. ■

**Lemma 4.11.** Let  $t$  be a good  $l[\ ]$ -term.

1) If  $t \sum_{b l[\ ]_0} t'$  then,  $t'$  is good.

2) If  $t \sum_{l[\ ]_0} (a) b$ , then  $(a) b$  is good.

3) If  $t \sum_{b l[\ ]_0} (f) u_1 \dots u_r$ , then  $u_1, \dots, u_r$  are good.

**Proof.** If  $t$  is good, then there is a  $l$ -term  $u$ , such that

$t = u[[t_1] \langle \mathbf{a}_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m / \mathbf{x}_m \rangle / y_m]$ , and for all  $1 \leq i \leq m$  if  $\mathbf{a}_i = a_{1,i}, \dots, a_{n_i,i}$ , then  $a_{j,i}$  is good  $1 \leq j \leq n_i$ .

1) It is enough to do the proof for one step of reduction. If  $t \sum_{b l[\ ]_0} t'$ , then  $t' = u'[[t_1] \langle \mathbf{a}_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m / \mathbf{x}_m \rangle / y_m]$  where  $u' \sum_{l[\ ]_0} u'$ , therefore  $t'$  is good.

2) It is enough do the proof for one step of reduction. For every  $l[\ ]$ -term  $u$ , denote by  $u''$



the  $l[\ ]$ -term  $u[[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$ .

It is clear that we may suppose that  $y_i Fv(\mathbf{a}_j) \ 1 \leq i, j \leq m$ .

If  $t \sum_{\square} (a) b$ , then  $u = (y_i)v_1 \dots v_q w_1 \dots w_s \ 1 \leq i \leq m, t_i = lf_1 \dots lf_q(y)u_1 \dots u_r$ , and

(a)b =  $\{(c)z_1 \dots z_r w_1 \dots w_s\} [[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m, [u_1]\langle \mathbf{a}_i/\mathbf{x}_i, \mathbf{v}''/\mathbf{f} \rangle/y_1, \dots, [u_r]\langle \mathbf{a}_i/\mathbf{x}_i, \mathbf{v}''/\mathbf{f} \rangle/z_r]$  where  $c = v_i$  if  $y = f_i \ 1 \leq i \leq q$ , and  $c = a_{j,i}$  if  $y = x_{j,i} \ 1 \leq i \leq m, 1 \leq j \leq \eta$ , and  $z_1, \dots, z_r$  are a new variables, therefore, by lemma 4.9, (a)b is good.

3) Use lemma 4.10, and 1) and 2). ■

### Proof of theorem 4.1.

**1** If  $T$  is an o.m.m. for  $t$ , then there is  $d_t:_{by}t$ , such that for every  $h_t:_{bt}$ , there is a substitution  $s$ , such that  $Th_t f \sum (f) s(d_t)$ .  $l(T[t]f) = Tt f$  is solvable, therefore, by theorem 3.4.7,  $T[t]f$  is  $b[\ ]$ -solvable, and  $T[t]f \sum_{b[\ ]} (f) t'$ . By lemma 4.4, let  $h$  be a  $t$ -special application, such that  $h(t) = h_t$ . Then  $S_h(t') = s(d_t)$ .  $T[t]f$  is a good  $l[\ ]$ -term, therefore, by lemma 4.11,  $t'$  is good, therefore  $t' = t_t [[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$  where  $t_t$  is a  $l$ -term. Therefore  $S_h(t') = t_t [h_1/y_1, \dots, h_m/y_m]$  where  $h_i = h(t_i) [S_h(a_{i,1})/x_{i,1}, \dots, S_h(a_{i,m_i})/x_{i,m_i}] \ 1 \leq i \leq m$ , therefore, by lemmas 2.2.4, 2-6, and 4.3,  $t_t = s'(d_t)$ , therefore  $t_t:_{by}t$ .

**0** Assume that  $T[t]f \sum_{b[\ ]} (f) t_t [[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$ , and  $t_t:_{by}t$ . Let  $h_t:_{bt}$ . By lemma 4.4, let  $h$  be a  $t$ -special application, such that  $h(t) = h_t$ . By lemma 4.7, we have  $S_h(T[t]f) = Th_t f \sum (f) t_t [h_1/y_1, \dots, h_m/y_m]$ . Therefore  $T$  is an o.m.m. for  $t$ . ■

### Examples.

-  $T$  is an o.m.m. for  $P_n$ . Indeed,

$$T[P] f \sum_{b0} ([P]) l f (f) P \dots l f (f) P f \sum_{\square} (l f (f) P) f \sum_{b0} (f) P \ 1 \leq i \leq n.$$

-  $T$  is an o.m.m. for  $\underline{N}$ . Indeed,

$$T[\underline{n}] f \sum_{b0} ([\underline{n}]) F f 0 \sum_{\square} (F) [(x_1)^{n-1} x_2] \langle F/x_1, f/x_2 \rangle 0 \sum_b ([x_1]^{n-1} x_2] \langle F/x_1, f/x_2 \rangle) (s) 0 \sum_{\square} (F) [(x_1)^{n-2} x_2] \langle F/x_1, f/x_2 \rangle (s) 0 \sum_b ([x_1]^{n-2} x_2] \langle F/x_1, f/x_2 \rangle) (s)^2 0 \sum_{\square} \dots \sum_b ([x_2] \langle F/x_1, f/x_2 \rangle) (s)^n 0 \sum_{\square} (f) (s)^n 0.$$

## § 5. Properties of storage operators

### 5.1 Storage operators and b-equivalence

**Theorem 5.1.1.** *Let  $t$  be a closed  $b$ -normal  $l$ -term,  $T$  and  $T'$  be closed  $l$ -terms. If  $T$  is an o.m.m. for  $t$ , and  $T':_bT$ , then  $T'$  also is an o.m.m. for  $t$ .*

**Proof.** On the set of good  $l[]$ -terms, we define an **equivalence relation  $g$**  by :

If  $t=u[[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$  where  $u$  is a  $l$ -term, then  $tg't'$  if and only if  $t'=u'[[t_1]\langle \mathbf{a}'_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}'_m/\mathbf{x}_m \rangle/y_m]$  where  $u:{}_b u'$ , and for all  $1 \leq i \leq m$  if  $\mathbf{a}_i = a_{1,i}, \dots, a_{n_i,i}$ , then  $\mathbf{a}'_i = a'_{1,i}, \dots, a'_{n_i,i}$ , and  $a_{j,i} g a'_{j,i}$   $1 \leq j \leq n_i$ .

It is clear that if  $lx_1 \dots lx_n(f)u_1 \dots u_m g t$  ( $f$  is a variable), then  $t = lx_1 \dots lx_n(f)u'_1 \dots u'_m$  where  $u_i g u'_i$   $0 \leq i \leq m$ .

**Lemma 5.1.2.** *If  $tg't'$ , and  $v_i g v'_i$   $1 \leq i \leq r$ , then  $t[v_1/y_1, \dots, v_r/y_r] g t'[v'_1/y_1, \dots, v'_r/y_r]$ .*

**Proof.** By induction on  $t$ . ■

**Lemma 5.1.3.** *Let  $t$  be a good  $l[]$ -term.*

1) *If  $t \sum_b t'$  is complete, and  $tgT$ , then for some  $T' : t'gT'$ , and  $T \sum_b T'$  is complete.*

2) *If  $t \sum_{[]} o(c)d$ , and  $tgT$ , then for some  $T'$  with the same  $b[]$ -head normal form as  $T : (c)dgT'$ .*

3) *If  $t \sum_b [f]u_1 \dots u_r$ , and  $tgT$ , then for some  $T' : (f)u_1 \dots u_r g T'$ , and  $T \sum_b [f]T'$ .*

**Proof.** If  $t$  is good, then there is a  $l$ -term  $u$ , such that

$t = u[[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$  where  $\mathbf{a}_i = a_{1,i}, \dots, a_{n_i,i}$   $1 \leq i \leq m$ .

1) If  $t \sum_b t'$  is complete, then  $t' = u'[[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$  where  $u'$  is the head normal form of  $u$ . If  $tgT$ , then  $T = U[[t_1]\langle \mathbf{a}'_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}'_m/\mathbf{x}_m \rangle/y_m]$  where  $u:{}_b U$ ,  $\mathbf{a}'_i = a'_{1,i}, \dots, a'_{n_i,i}$ , and  $a_{j,i} g a'_{j,i}$   $1 \leq j \leq n_i$ . Let  $U'$  be the head normal form of  $U$ .

Let  $T' = U'[[t_1]\langle \mathbf{a}'_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}'_m/\mathbf{x}_m \rangle/y_m]$ . It is clear that we have  $t'gT'$ , and  $T \sum_b T'$  is complete.

2) For every  $l[]$ -term  $u$ , we denote by  $u''$  the  $l[]$ -term  $u[[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m]$ .

It is clear that we can suppose that  $y_i \in Fv(\mathbf{a}_j)$   $1 \leq i, j \leq m$ .

If  $t \sum_{[]} o(c)d$ , then  $u = (y_i)v_1 \dots v_q w_1 \dots w_s$   $1 \leq i \leq m, t_i = lf_1 \dots lf_q(y)u_1 \dots u_r$ , and

$(c)d = \{(b)z_1 \dots z_r w_1 \dots w_s\} [[t_1]\langle \mathbf{a}_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}_m/\mathbf{x}_m \rangle/y_m, [u_1]\langle \mathbf{a}_i/\mathbf{x}_i, \mathbf{v}''/\mathbf{f} \rangle/z_1, \dots, [u_r]\langle \mathbf{a}_i/\mathbf{x}_i, \mathbf{v}''/\mathbf{f} \rangle/z_r]$  where  $b = v_i$  if  $y = f_i$   $1 \leq i \leq q$ , and  $b = a_{j,i}$  if  $y = x_{j,i}$   $1 \leq i \leq m$ , and  $1 \leq j \leq n_i$ , and  $z_1, \dots, z_r$  are a new variables.

If  $tgT$ , then  $T = U[[t_1]\langle \mathbf{a}'_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}'_m/\mathbf{x}_m \rangle/y_m]$  where  $u:{}_b U$ ,  $\mathbf{a}'_i = a'_{1,i}, \dots, a'_{n_i,i}$  and  $a_{j,i} g a'_{j,i}$   $1 \leq j \leq n_i$ . Since  $u:{}_b U$ , then  $U \sum (y_i)c_1 \dots c_q d_1 \dots d_s$  where  $v_i:{}_b c_i$   $1 \leq i \leq q$ , and  $w_j:{}_b d_j$   $1 \leq j \leq s$ .

For every  $l[]$ -term  $u$ , we denote by  $u'''$  the  $l[]$ -term  $[[t_1]\langle \mathbf{a}'_1/\mathbf{x}_1 \rangle/y_1, \dots, [t_m]\langle \mathbf{a}'_m/\mathbf{x}_m \rangle/y_m]$

$y_m]$ .

Let  $T' = \{(b)z_1 \dots z_r d_1 \dots d_s\} [[t_1] \langle \mathbf{a}'_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}'_m / \mathbf{x}_m \rangle / y_m, [u_1] \langle \mathbf{a}'_i / \mathbf{x}_i, \mathbf{c}''' / \mathbf{f} \rangle / z_1, \dots, [u_r] \langle \mathbf{a}'_i / \mathbf{x}_i, \mathbf{c}''' / \mathbf{f} \rangle / z_r]$  where  $b' = c_i$  if  $y = f_i$   $1 \leq i \leq q$ , and  $b' = a'_{j,i}$  if  $y = x_{j,i}$   $1 \leq i \leq m$ , and  $1 \leq j \leq \eta$ .

It is clear that  $T$  and  $T'$  have the same  $b[\ ]$ -head normal form, and, by lemma 5.1.2,  $t'gT'$ .

3) Use 1), 2), and lemma 4.10. ■

If  $T$  is an o.m.m. for  $t$ , then, by theorem 4.1,

$T[t]f \sum_{b[\ ]} (f)t_t [[t_1] \langle \mathbf{a}'_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}'_m / \mathbf{x}_m \rangle / y_m]$ , and  $t_t :_{b} y t$ . If  $T' :_b T$ , then  $T'[t]fgT[t]f$ , therefore, by 3) of lemma 5.1.3,

$T'[t]f \sum_{b[\ ]} (f)t'_t [[t_1] \langle \mathbf{a}'_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}'_m / \mathbf{x}_m \rangle / y_m]$ , and  $t'_t :_{b} t'_t :_{b} y t$ . Therefore, by theorem 4.1,  $T'$  is an o.m.m. for  $t$ . ■ (of theorem 5.1.1)

## 5.2 Decidability

**Theorem 5.2.1.** *If  $X$  is a non trivial set of closed  $l$ -terms stable by  $b$ -equivalence, then  $X$  is not recursive.*

**Proof.** See [2], [5], and [14]. ■

**Theorem 5.2.2.** *The set of o.m.m. for a set of closed  $b$ -normal  $l$ -terms is not recursive.*

**Proof.** Use theorems 5.1.1 and 5.2.1. ■

**Theorem 5.2.3.** *The set of o.m.m. for a finite set of closed  $b$ -normal  $l$ -terms is recursively enumerable.*

**Proof.** Use theorem 4.1. ■

## 5.3 Storage operators and $y$ -equivalence

**Theorem 5.3.1.** *Let  $t$  be a closed  $b$ -normal  $l$ -term, and  $T$  be closed  $l$ -term.*

*If  $T$  is an o.m.m. for  $t$ , and  $t5_y t'$ , then  $T$  also is an o.m.m. for  $t'$ .*

**Remark.** The theorem 5.3.1 is no more true if we replace  $t5_y t'$  by  $t :_y t'$ . Indeed,

if we take  $t = lxx$ ,  $t' = lxlz((x)y)z$ , and  $T = l(n)lf(f)lxx$ , then :

- $t \leq_y t'$ , therefore  $t;_y t'$ .
- For every l-term  $u$  such that  $u;_b t$ ,  $(T)u \sum lf(f)lxx$ , therefore  $T$  is an o.m.m. for  $t$ .
- $(T)t'f \sum lz((f)lxx)z$ , therefore  $T$  is not an o.m.m. for  $t'$ .

**Proof of theorem 5.3.1.** On the set  $L[]$ , we define the **binary relation c** as the least relation satisfying :

- $tct$  ;
- If  $tct'$ , then  $lxtclxt'$ ;
- If  $ucu'$ , and  $vcv'$ , then  $(u)vc(u')v'$ ;
- If  $t \leq_y x_i$ , and  $a_i ca'_i$   $1 \leq i \leq n$ , then  $[t] \langle \mathbf{a}/\mathbf{x} \rangle ca'_i$  ;
- If  $t \leq_y t'$ , and  $a_i ca'_i$   $1 \leq i \leq n$ , then  $[t] \langle \mathbf{a}/\mathbf{x} \rangle c[t'] \langle \mathbf{a}'/\mathbf{x} \rangle$ .

It is clear that :

- If  $lx_1 \dots lx_n(u_0)u_1 \dots u_m ct$ , then  $t = lx_1 \dots lx_n(u'_0)u'_1 \dots u'_m$  where  $u_i cu'_i$   $0 \leq i \leq m$ .
- Let  $t$  be a good  $l[]$ -term, therefore there is a l-term  $u$ , such that  $t = u[[t_1] \langle \mathbf{a}_1/\mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m/\mathbf{x}_m \rangle / y_m]$ , and for all  $1 \leq i \leq m$  if  $\mathbf{a}_i = a_{1,i}, \dots, a_{n_i,i}$ , then  $a_{j,i}$  is good  $1 \leq j \leq n_i$ . If  $tct'$ , then it is easy to check that  $t' = u[c_1/y_1, \dots, c_n/y_m]$  where  $c_i = [t'_i] \langle \mathbf{a}'_i/\mathbf{x}_i \rangle$  with  $t_i \leq_y t'_i$ ,  $a_{j,i} ca'_{j,i}$   $1 \leq i \leq m$ ,  $1 \leq j \leq n_i$ , or  $c_i = a'_{j,i}$   $1 \leq j \leq n$  with  $t_i \leq_y x_{j,i}$  and  $a_{j,i} ca'_{j,i}$   $1 \leq i \leq m$ ,  $1 \leq j \leq n_i$ .

**Lemma 5.3.2.** *If  $ucu'$ , and  $v_i cv'_i$   $1 \leq i \leq n$ , then :*

$$u[v_1/x_1, \dots, v_n/x_n]c u'[v'_1/x_1, \dots, v'_n/x_n].$$

**Proof.** By induction on  $u$ . ■

**Lemma 5.3.3.** *If  $u = lx_1 \dots lx_n(y)u_1 \dots u_m \leq_y v$ , then  $v = lx_1 \dots lx_{n-r}(y)u'_1 \dots u'_{m-r}$  where  $u_j \leq_y u'_j$   $1 \leq j \leq m-r$ ,  $u_{m-s} \leq_y x_{n-s}$   $0 \leq s \leq r-1$ , and  $x_{n-s} \neq y$  does not appear in  $u_1, \dots, u_{m-r}$ .*

**Proof.** By induction on the number  $n$  of  $y_0$ -reductions to go from  $u$  to  $v$ .

$n=0$  : clear.

If  $n \geq 1$ , then  $u \leq_y w \leq_y v$ , therefore  $w = lx_1 \dots lx_{n-r}(y)u'_1 \dots u'_{m-r}$  where  $u_j \leq_y u'_j$   $1 \leq j \leq m-r$ ,  $u_{m-s} \leq_y x_{n-s}$   $0 \leq s \leq r-1$ , and  $x_{n-s} \neq y$  does not appear in the  $u_1, \dots, u_{m-r}$ . Since  $w \leq_y v$ , then  $v = lx_1 \dots lx_{n-r}(y)u'_1 \dots u'_{m-r}$  where  $u'_k \leq_y u''_k$ , or  $u'_{m-r} = x_{n-r}$ ,  $x_{n-r} \neq y$  does not appear in the  $u_1, \dots, u_{m-r-1}$ , and  $v = lx_1 \dots lx_{n-r-1}(y)u'_1 \dots u'_{m-r-1}$  as required. ■

**Lemma 5.3.4.**

- 1) *If  $t \sum_{b0} t'$ , and  $tcT$ , then for some  $t' : t'cT'$ , and  $T \sum_{b0} T'$ .*
- 2) *If  $t \sum_{[]0} t'$ , and  $tcT$ , then for some  $t' : t'cT'$ , and  $T \sum_{[]0} T'$ .*

3) If  $t \sum_{b[]} t'$ , and  $tcT$ , then for some  $t' : t'cT'$ , and  $T \sum_{b[]} T'$ .

**Proof.**

1) If  $t \sum_{b0} t'$ , then  $t = lx_1 \dots lx_n (lxu) vt_1 \dots t_m$ , and  $t' = lx_1 \dots lx_n (u[v/x]) t_1 \dots t_m$ . If  $tcT$ , then  $T = lx_1 \dots lx_n (lxu') v' t'_1 \dots t'_m$  where  $ucu'$ ,  $vcv'$ , and  $t_i ct'_i$   $1 \leq i \leq m$ .

Let  $T' = lx_1 \dots lx_n (u'[v'/x]) t'_1 \dots t'_m$ . It is clear that  $T \sum_{b0} T'$ , and, by lemma 5.3.2,  $t'cT'$ .

2) If  $t \sum_{[]0} t'$ , then  $t = ly_1 \dots ly_m ([lz_1 \dots lz_k(y) u_1 \dots u_r] \langle \mathbf{a}/\mathbf{x} \rangle v_1 \dots v_k w_1 \dots w_s)$ , and  $t' = ly_1 \dots ly_m (b) [u_1] \langle \mathbf{a}/\mathbf{x}, \mathbf{v}/\mathbf{z} \rangle \dots [u_r] \langle \mathbf{a}/\mathbf{x}, \mathbf{v}/\mathbf{z} \rangle w_1 \dots w_s$  where  $b = v_i$  if  $y = y_i$   $1 \leq i \leq m$ , and  $b = a_i$  if  $y = x_j$   $1 \leq j \leq n$ . Assume  $tcT$ .

- If  $lz_1 \dots lz_k(y) u_1 \dots u_r \sum_{y} y$ , then  $k=r$ ,  $u_j \sum_{y} z_i$   $1 \leq i \leq m$ , and  $z_i \neq y = x_j$   $1 \leq j \leq n$ , then  $T = ly_1 \dots ly_m (a^j) v'_1 \dots v'_k w'_1 \dots w'_s$  where  $a_j c a^j$ ,  $v_i c v'_i$   $1 \leq i \leq k$ , and  $w_i c w'_i$   $1 \leq i \leq s$ . Let  $T' = T$ . It is clear that  $t'cT'$ , and  $T \sum_{[]0} T'$ .

- If  $lz_1 \dots lz_k(y) u_1 \dots u_r \sum_{y} lz_1 \dots lz_{k-1}(y) u'_1 \dots u'_{r-1}$  where  $u_j \sum_{y} u'_j$   $1 \leq j \leq r-1$ ,  $u_{r-s} \sum_{y} z_{k-s}$   $0 \leq s \leq l-1$ , and  $z_{k-s} \neq y$  does not appear in the  $u_1, \dots, u_{r-1}$ , then  $T = ly_1 \dots ly_m ([lz_1 \dots lz_{k-1}(y) u'_1 \dots u'_{r-1}] \langle \mathbf{a}'/\mathbf{x} \rangle v'_1 \dots v'_k w'_1 \dots w'_s)$  where  $v_i c v'_i$   $1 \leq i \leq k$ , and  $w_i c w'_i$   $1 \leq i \leq s$ . Let  $T' = ly_1 \dots ly_m (b_i) [u_1] \langle \mathbf{a}'/\mathbf{x}, v'_1/z_1, \dots, v'_{k-1}/z_{k-1} \rangle \dots [u_r] \langle \mathbf{a}'/\mathbf{x}, v'_1/z_1, \dots, v'_{k-1}/z_{k-1} \rangle v'_{m-l+1} \dots v'_k w'_1 \dots w'_s$  where  $b = v'_i$  if  $y = y_i$   $1 \leq i \leq m$ , and  $b = a_i$  if  $y = x_j$   $1 \leq j \leq n$ . It is clear that  $t'cT'$ , and  $T \sum_{[]0} T'$ .

3) Use 1) and 2). ■

If  $T$  is an o.m.m. for  $t$ , then, by theorem 4.1, there is a  $l$ -term  $t_t : b_y t$ , such that  $T[t] f \sum_{b[]} (f) t_t [[t_1] \langle \mathbf{a}_1/\mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m/\mathbf{x}_m \rangle / y_m]$ .

If  $t \sum_{y} t'$ , then  $T[t] f c T[t'] f$ , therefore, by lemma 5.3.4,  $T[t'] f \sum_{b[]} (f) t'$ , and  $t_t [[t_1] \langle \mathbf{a}_1/\mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m/\mathbf{x}_m \rangle / y_m] c t'$ . Therefore there is a  $l$ -term  $t''_t$ , such that  $t''_t : b_t t : b_y t$ , and  $t' = t''_t [[u_1] \langle \mathbf{b}_1/\mathbf{z}_1 \rangle / y_1, \dots, [u_m] \langle \mathbf{b}_r/\mathbf{z}_r \rangle / y_r]$ . Therefore, by theorem 4.1,  $T$  is an o.m.m. for  $t'$ . ■ (of theorem 5.3.1)

## 5.4. Storage operators for a set of b-normal l-terms

**Theorem 5.4.1.** *Let  $u_1, \dots, u_n, v_1, \dots, v_m$  be closed l-terms. Assume  $u_i b_y u_j$  for  $i < j$ , there is a closed l-term  $T$ , such that  $(T) u_i : b v_i$   $1 \leq i \leq n$ .*

**Proof.** See [3]. ■

**Theorem 5.4.2.** *Every finite set of b-normal l-terms having all distinct by-normal forms has an o.m.m..*

**Proof.** Let  $D=\{t_1, \dots, t_n\}$  be such a set. By theorem 5.4.1, there is a closed l-term  $T'$ , such that  $T'_{t_i};_b P$   $1 \leq i \leq n$ , therefore for every  $h_i;_b t_i$ ,  $T'_{h_i};_b P$ , therefore  $T'_{h_i} \Sigma P$ . Let  $T = \text{ln}((T')^n) \text{lf}(f)t_1 \dots \text{lf}(f)t_n$ . It is easy to check that  $T$  is an o.m.m. for  $D$ . ■

**Theorem 5.4.3.** *Every finite set of b-normal l-terms has an o.m.m..*

**Remarks.**

- The theorem 5.4.3 is no more true if we remove the hypothesis "the l-terms of  $D$  are b-normal". If we take  $t_1 = \text{lx}x$ , and  $t_2 = (\text{lx}(x)x)\text{lx}(x)x$ , then  $D = \{t_1, t_2\}$  have no o.m.m.. Indeed, if  $T$  is an o.m.m. for  $t_2$ , then, by corollary 2.2.3,  $T = \text{lnlf}(f)u_2$  where  $u_2;_{by} t_2$ , therefore  $T$  is not an o.m.m. for  $t_1$ .

- The theorem 5.4.3 is no more true if we remove the hypothesis " $D$  is finite". If we take  $D$  the set of all  $P_i \geq 1$ , then  $D$  have no o.m.m.. Indeed, if  $T$  is an o.m.m. for  $D$ , let  $T'$  its head normal form. By proposition 2.2.1,  $T' = \text{lx}_1 \dots \text{lx}_e(x_i)t_1 \dots t_n$  where  $e = 1$  or  $2$ , and, by theorem 5.1.1,  $T'$  also is an o.m.m. for  $D$ . It is easy to prove that  $T'$  is not an o.m.m. for the l-term  $P$ .

**Proof of theorem 5.4.3.** Let  $D = \{t_1, \dots, t_n\}$  be a finite set of b-normal l-terms. Gathering the l-terms having the same by-normal form, we can write  $D = \cup_{i=1}^m D_i$  where  $D_i = \{t_1, \dots, t_{j_i}\}$   $1 \leq i \leq m$ , for all  $1 \leq i \leq m$ , and  $1 \leq j, j' \leq m$ ,  $t^{by} = t'^{by}$ , and for all  $1 \leq i, i' \leq m$ ,  $t^{by} \neq t'^{by}$ .

**Lemma 5.4.4.** *Let  $t, t'$  be b-normal l-terms. If  $t;_y t'$ , then there is a b-normal l-term  $u$ , such that  $u;_y t$ , and  $u;_y t'$ .*

**Proof.** By induction on  $t$  and  $t'$ . If  $t;_y t'$ , then there is a b-normal l-term  $v$ , such that  $t;_y v$ , and  $t';_y v$ . If  $v = \text{lx}_1 \dots \text{lx}_n(y)v_1 \dots v_m$ , then, by lemma 5.3.3,  $t = \text{lx}_1 \dots \text{lx}_n \text{ly}_1 \dots \text{ly}_k(y)v'_1 \dots v'_m u_1 \dots u_k$ , and  $t' = \text{lx}_1 \dots \text{lx}_n \text{ly}_1 \dots \text{ly}_r(y)v''_1 \dots v''_m w_1 \dots w_r$  where  $v'_i;_y v_i$ ,  $v''_i;_y v_i$   $1 \leq i \leq m$ ,  $u_j;_y y_j$   $1 \leq j \leq k$ ,  $y_j \neq y$  does not appear in  $v_1, \dots, v_m$ ,  $w_j;_y y_j$   $1 \leq j \leq r$ , and  $y_j \neq y$  does not appear in  $v_1, \dots, v_m$ . Assume that  $k \leq r$ . By induction hypothesis, let  $a_i$  be a b-normal l-term, such that  $a_i;_y v'_i$ , and  $a_i;_y v''_i$   $1 \leq i \leq m$ , and  $b_j$  be a b-normal l-term, such that  $b_j;_y u_j$ , and  $b_j;_y w_j$   $1 \leq j \leq k$ . Let  $u = \text{lx}_1 \dots \text{lx}_n \text{ly}_1 \dots \text{ly}_r(y)a_1 \dots a_m b_1 \dots b_k w_{k+1} \dots w_r$ . It is clear that  $u$  is a b-normal l-term, and that  $u;_y t$ , and  $u;_y t'$ . ■

An **y-bound** for a set  $B=\{u_1,\dots,u_m\}$  is a b-normal l-term  $u$ , such that  $u \geq y u_i$   $1 \leq i \leq m$ .

**Corollary 5.4.5.** *Every finite set  $B$  of b-normal l-terms having all the same y-normal form has an y-bound.*

**Proof.** By induction on the number of l-terms of  $B$  using lemma 5.4.4. ■

By corollary 5.4.4, let  $u_i$  be a y-bound for  $D_i$   $1 \leq i \leq m$ . By theorem 5.4.2, the set  $\{u_1, \dots, u_m\}$  has an o.m.m., therefore, by theorem 5.3.1,  $D$  has an o.m.m.. ■ (of theorem 5.4.3)

## 5.5 Computation time of a storage operator

**Lemma 5.5.1.** *Let  $(t_i)_{1 \leq i \leq n}$  and  $(t'_i)_{1 \leq i \leq n}$  be sequences of l-terms, such that :*

- 1) *For all  $1 \leq i \leq n$ ,  $t_i \geq t'_i$ .*
- 2) *For all  $1 \leq i \leq n-1$ ,  $t_i = (u_i) v_{i,1} \dots v_{i,r_i}$ ,  $t'_i = (u'_i) v_{i,1} \dots v_{i,r_i}$  and  $u'_i \geq u_{i+1}$ .*
- 3)  *$t'_n = (f) v_{1,1} \dots v_{r,1}$  where  $f$  is a variable.*

*Then  $t_1 \geq t'_n$ , and  $tps(t_1) = n(t_1, t'_n) = +$ .*

**Proof.** By induction on  $n$ .

$n=1$ : trivial

for  $n > 2$  : Let  $n_i = n(t_i, t'_i)$  and  $m_i = n(u'_i, u_{i+1})$ . By induction hypothesis, we have  $t_2 \geq t'_n$ , and  $n(t_2, t'_n) = +$ .

$u'_1 \geq u_2$ , therefore, by theorem 1.2.1, for some  $w$ ,  $(u'_1) v_{1,1} \dots v_{1,r_1} \geq w$ ,  $(u_2) v_{1,1} \dots v_{1,r_1} \geq w$ , and  $n((u'_1) v_{1,1} \dots v_{1,r_1}, w) = n((u_2) v_{1,1} \dots v_{1,r_1}, w) + n(u'_1, u_2) = n((u_2) v_{1,1} \dots v_{1,r_1}, w) + m_1$ .

Therefore  $t'_1 \geq t'_n$ , and  $n(t'_1, t'_n) = n(t'_1, w) + n(w, t'_n)$ . Therefore  $t_1 \geq t'_n$ , and

$tps(t_1) = n(t_1, t'_n) = n(t_1, t'_1) + n(t'_1, t'_n) = n_1 + m_1 + + = +$ . ■

**Theorem 5.5.2.** *Let  $t$  be a closed b-normal l-term, and  $T$  a closed l-term.*

*If  $T$  is an o.m.m. for  $t$ , there are constants  $A_{T,t}$  and  $B_{T,t}$  such that for every  $h_t : b t$ ,  $tps(Th_t) \leq A_{T,t} Tps(h_t) + B_{T,t}$ .*

**Proof.**

If  $t \geq_b t'$ , denote by  $b(t, t')$ , the number of  $b_0$ -reductions used in this reduction.

For every  $v \in L[\ ]$ , we define  $D(v)$  by induction on  $v$  :

- If  $[u] < a/x >$  is the head redex of  $v$ , then  $D(v) = u$  ;

- If not,  $D(v)=o$  where  $o$  is a constant.

Let  $h$  be a  $t$ -special application. For every  $u \in \text{AST}(t) \setminus \{o\}$ , we define the integer  $n_h(u)$  by :

- $n_h(x)=n_h(o)=0$  ;
- $n_h(\lambda x u)=n(h(\lambda x u), \lambda x h(u))$  ;
- $n_h((u)v)=n(h((u)v), (h(u))h(v))$ .

If  $T$  is an o.m.m. for  $t$ , then

$T[t]f \sum_{b \in \Omega} (f) t_i [[t_1] \langle \mathbf{a}_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m / \mathbf{x}_m \rangle / y_m]$ , and  $t_i : b_i y_i$ . There is a sequence of  $\Omega$ -terms  $t_0 = T[t]f, t_1, \dots, t_n = (f) t_i [[t_1] \langle \mathbf{a}_1 / \mathbf{x}_1 \rangle / y_1, \dots, [t_m] \langle \mathbf{a}_m / \mathbf{x}_m \rangle / y_m]$ , such that  $t_{i-1} \sum_{b \in \Omega} t_i$  or  $t_{i-1} \sum_{b \in \Omega} t_i \ 1 \leq i \leq n$ .

Let  $A_{T,t} = \text{Max}\{\text{number of boxes directed by } u \text{ and appearing in head position of } t_i \mid 0 \leq i \leq n, u \in \text{AST}(t)\}$ , and  $B_{T,t} = b(t_0, t_n)$ .

Let  $h_t : b_t$ . By lemma 4.4, let  $h$  be a  $t$ -special application, such that  $h(t) = h_t$ .

By the proof of lemma 4.7, and by lemma 5.5.1, we have

$$\text{tps}(Th_t f) = b(t_0, t_n) + \dots$$

By theorem 1-3,  $\text{Tps}(h_t) = n_h(u)$ , and then  $\text{tps}(Th_t f) \leq A_{T,t} \text{Tps}(h_t) + B_{T,t}$ . ■

**Remark.** By the proof of theorem 5.5.2, we have  $\text{tps}(Th_t f) = A_{T,t} \text{Tps}(h_t) + B_{T,t}$  if and only if, for all  $u \in \text{AST}(t)$ ,  $A_{T,t}$  = the number of boxes directed by  $u$  and appearing in head position of  $t_i \ 0 \leq i \leq n$ .

## References

- [1] **M. Abadi, L. Cardelli, P.L. Curien, and J.L. Levy.**  
*Explicit Substitutions*. Technical report 1176, INRIA, 1990.  
To appear in Journal of Functional Programming.
- [2] **H. Barendregt.**  
*The lambda calculus : Its Syntax and Semantics*.  
North Holland, 1984.
- [3] **C. Böhm, M. Dezani Ciancaglini, A. Peretti & S. Ronchi Della Rocca.**



*A discriminator algorithm inside l-b-calculus.*  
Theoretical Computer Science 8, (271-291), 1979.

- [4] **P.L. Curien.**  
*The lr-calculi : an abstract framework for closures.*  
Technical report, LIENS - Ecole Normale Supérieure, 1988.
  
- [5] **J.L. Krivine.**  
*Lambda calcul, évaluation paresseuse et mise en mémoire.*  
Theoretical Informatics and Applications (R.A.I.R.O.), Vol. 25,1, p. 67-84,  
1991.
  
- [6] **J.L. Krivine.**  
*Lambda calcul, types et modèle.*  
Masson, Paris, 1990.
  
- [7] **J.L. Krivine.**  
*Opérateurs de mise en mémoire et traduction de Gödel.*  
Arch. Math. Logic 30. (241-267), 1990.
  
- [8] **K. Nour.**  
*Opérateurs de mise en mémoire en lambda-calcul pur et typé.*  
Thèse de doctorat, Université de Savoie, 1993.
  
- [9] **K. Nour.**  
*Strong storage operators and data types.*  
Preprint of LAMA, 1993.
  
- [10] **K. Nour.**  
*Opérateurs de mise en mémoire et types V-positifs.*  
Submitted to Theoretical Informatics and Applications, 1993.
  
- [11] **K. Nour.**  
*Storage operators and V-positive types in system TTR.*  
Preprint of LAMA, 1993.
  
- [12] **K. Nour.**  
*Opérateurs propres de mise en mémoire.*  
CRAS Paris, t. 317, Série I, p. 1.-.6, 1993.

[13] **K. Nour.**

*Preuve syntaxique d'un théorème de J.L. Krivine sur les opérateurs de mise en*

*mémoire.*

CRAS Paris, t. 318, Série I, p. 201-204, 1994.

[14] **D.S. Scott.**

*A system of functional abstraction.*

Unpublished, 1963.

\* \* \* \* \*