Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science 171 (2007) 3–19
Electronic Notes in Theoretical Computer Science
www.elsevier.com/locate/entcs

Term Collections in λ and ρ-calculi

Germain Faure[1]

Université Henri Poincaré & LORIA, BP 239
F-54506 Vandoeuvre-lès-Nancy, France

Abstract

The ρ-calculus generalises term rewriting and the λ-calculus by defining abstractions on arbitrary patterns and by using a pattern-matching algorithm which is a parameter of the calculus. In particular, equational theories that do not have unique principal solutions may be used. In the latter case, all the principal solutions of a matching problem are stored in a "structure" that can also be seen as a collection of terms.

Motivated by the fact that there are various approaches to the definition of structures in the ρ-calculus, we study in this paper a version of the λ-calculus with term collections.

The contributions of this work include a new syntax and operational semantics for a λ-calculus with term collections, which is related to the λ-calculi with strict parallel functions studied by Boudol and Dezani et al. and a proof of the confluence of the β-reduction relation defined for the calculus (which is a suitable extension of the standard rule of β-reduction in the λ-calculus).

Keywords: lambda-calculus, rho-calculus, parallel operator, canonical sets, term collections.

1 Introduction

The ρ-calculus, also called the rewriting calculus, originally emerged from different motivations—and from a different community—than the λ-calculus. It was introduced to make explicit all the ingredients of rewriting such as rule application and result [7]. *In fine*, the ρ-calculus provides an extension of the λ-calculus with additional concepts originating from rewriting and functional programing, namely built-in pattern-matching, represented using matching constraints, and term collections.

There are several aspects of the ρ-calculus that have been studied so far. The dynamics of the computations has been studied [13] by defining interaction nets for the ρ-calculus. We can mention also the study of type systems [3,24] and their applications to a proof theory that handles rich proof-terms in the generalized deduction modulo [25].

[1] Email: germain.faure@loria.fr

1571-0661 © 2007 Elsevier B.V. Open access under CC BY-NC-ND license.
doi:10.1016/j.entcs.2006.12.037

On a more practical side, the ρ-calculus has been used both to encode object calculi [8] and to give a semantics to rewrite based languages [7] such as ELAN [4]. More recently, there were undergoing works to specify in the ρ-calculus the Focal Project [17,18], a programming environment in which certified programs can be developed. Also, the ρ-calculus has been used to specify efficient decision procedures [21].

Term collections are fundamental in the context of the ρ-calculus but also in logic programming and in web query languages. Typically, matching constraints that are involved in the calculus may have more than one solution —this is also the case for example in programming language like TOM [23], Maude [16], ASF+SDF [1] or ELAN [11]—and thus generates a collection of results.

Different works on the ρ-calculus propose different approaches to deal with term collections. They were originally [7,6] represented using sets. In more recent works [9], they are represented via a *structure* construction whose operational semantics is parametrised by a theory (typically a combination of the axioms of associativity, commutativity and/or idempotence) that the user chooses depending on the way (s)he wants to deal with non-determinism in the calculus. For example, the original semantics of "sets of results" is recovered by considering the associative, commutative and idempotent (ACI) theory on structures.

The generality given by those recent works is broken when the matching constraints involved in the calculus may have more than one solution, that is when the solving of the matching problem gives several solutions (substitutions). In fact, to the knowledge of the author there is no satisfactory (total) order to compare substitutions. Thus we must represent collections of terms by sets (or at least by an associative and commutative structure).

If we look carefully at the different presentations of the ρ-calculus, we can remark that the different operational semantics always share a common structure: the set of evaluation rules can be divided into a first subset consisting of the ρ-rule, an extension of the β-rule of the λ-calculus to deal with application of pattern-abstractions and into a second one dealing with term collections (including the δ-rule that distributes term collections over the application operator). These two kinds of rules are both treated at the same level whereas the former captures in a nutshell the computational mechanism of the calculus when the latter only performs "administrative simplifications" concerning term collections.

The first attempt to a denotational semantics of the ρ-calculus proposed in [12] enlightened a relation between the ρ-calculus and the λ-calculi for (strict) parallel functions [5]. Syntactically, the latters are extensions of the λ-calculus with a parallel operator that distributes left *w.r.t.* applications and *w.r.t.* λ-abstractions (as the structure operator of the ρ-calculus). It has been extended to parallel and non-deterministic λ-calculi like in [10].

The Scott models of the ρ-calculus and of the λ-calculi for (strict) parallel functions are surprisingly close: the structure operator are adequately represented by the join operator.

This suggests a clear relation between these formalisms: the λ-calculi for (strict) parallel functions are extensions of the λ-calculus with term collections and the ρ-calculus is an extension of the λ-calculus with term collections and built-in pattern-matching.

As a first step in the study of the ρ-calculus with non-unitary matching theories, we propose in this paper the study of the λ-calculus with term collections, called in the following the $\lambda_{\shortparallel}$-calculus.

In the spirit of the normalized rewriting [15], the approach introduced in this paper manages term collections at the meta-level by considering only *canonical sets* that is, sets that are normalized for some rules (the ones previously refereed as "administrative simplifications"). The result is a confluent calculus where the computational mechanism becomes easier to understand since only the β-rule is an explicit evaluation step. While the work of [5] mainly insists on models (looking at the full-abstraction problem for programming languages), in this work we propose to look at the $\lambda_{\shortparallel}$-calculus from an operational point of view by giving a clear operational semantics to the equivalence relation introduced in [5]. The same approach can be used if one prefers to consider canonical multisets since neither the definitions nor the proofs are related to the idempotence.

Road-map The paper is organized as followed. The first section introduces the syntax of the $\lambda_{\shortparallel}$-calculus by defining simple terms, parallel terms and substitutions. The second section is a general study of the relations on the terms on the $\lambda_{\shortparallel}$-calculus. The third section introduces the operational semantics of the $\lambda_{\shortparallel}$-calculus. We finally study in the fourth section the Church-Rosser property. We conclude by some remarks on the operationality of the reduction.

2 Syntax of the $\lambda_{\shortparallel}$-calculus

2.1 Preliminaries

For better a readability, a relation τ is also denoted by $\rightarrow_\tau$. Its reflexive and transitive closure is denoted either by $(\tau)^*$ or by $\twoheadrightarrow_\tau$. The successors of A for the relation τ are the elements of the set $\{B \mid A \rightarrow_\tau B\}$. The composition of two relations is simply denoted by juxtaposition.

We denote by $\mathcal{P}^+_{<\infty}(X)$ the finite non-empty subsets of X. A non-empty set $\{S_1, \ldots, S_n\}$ is often denoted by $\{S_i\}_{i=1}^n$ or even simpler by $\{S_i\}_i$.

2.2 Terms

In this section, we introduce the syntax of the $\lambda_{\shortparallel}$-calculus which consists of simple terms and parallel terms. A parallel term is simply a set of simple terms. Simple terms differ from λ-terms since they cannot be applied to a simple term but only to *a set* of simple terms that is, to a parallel term.

Definition 2.1 (Simple terms and terms) *Given a denumerable set of variables $\mathcal{X}$, we define by induction on k an increasing family of sets $(\mathbb{S}_k)$. We set $\mathbb{S}_0 = \mathcal{X}$ and we define $\mathbb{S}_{k+1}$ as follows:*

- **Monotonicity:** $\mathbb{S}_k \subseteq \mathbb{S}_{k+1}$ *;*
- **Abstraction:** *if* $S \in \mathbb{S}_k$ *then* $\lambda x.S \in \mathbb{S}_{k+1}$ *;*
- **Application:** *if* $S \in \mathbb{S}_k$ *and* $M \in \mathcal{P}^+_{<\infty}(\mathbb{S}_k)$ *then* $S\,M \in \mathbb{S}_{k+1}$*.*

We denote by $\mathbb{S}$ *the union of all the sets* $\mathbb{S}_k$ *and by* $\mathbb{M}$ *the set* $\mathcal{P}^+_{<\infty}(\mathbb{S})$*. We call* simple terms *the elements of* $\mathbb{S}$ *and* parallel terms *or simply* terms *the elements of* $\mathbb{M}$*.*

In the application SM, the simple term S is said to be in functional position while the term M is in applicative position.

Simple terms are written $S, T, U \ldots$ and terms are denoted by $M, N, P, E \ldots$. We denote by η the canonical injection of simple terms into terms, that is the relation such that $S \to_\eta \{S\}$ for all simple terms S. We consider terms modulo the α-conversion and the *hygiène* convention of Barendregt. The α-conversion on sets is defined by:

$$M =_\alpha N \Leftrightarrow \begin{cases} \forall S \in M, \exists T \in N, S =_\alpha T \\ \forall T \in N, \exists S \in M, T =_\alpha S \end{cases}$$

Note that the α-conversion does not preserve cardinality. For example,we have $\{\lambda x.x, \lambda y.y\} =_\alpha \{\lambda z.z, \lambda z.z\} = \{\lambda z.z\}$.

We can remark that proving a property by induction on terms means proving this property for each term M by induction on the least k such that $M \in \mathcal{P}^+_{<\infty}(\mathbb{S}_k)$. This number denotes the height of M and is written $\mathfrak{h}\,(M)$.

Remark 2.2 *We can equivalently defined the simple terms and terms by*

$$\begin{array}{rrcl} \textit{simple terms} & S \in \mathbb{S} & ::= & x \quad | \quad \lambda x.S \quad | \quad S\,M \\ \textit{terms} & M \in \mathbb{M} & ::= & \{S, \ldots, S\} \end{array}$$

Then the height of a term can be (re)defined as follows:

$$\begin{array}{llll} \mathfrak{h}\,(x) & = 0 & \mathfrak{h}\,(\lambda x.S) & = 1 + \mathfrak{h}\,(S) \\ \mathfrak{h}\,(S\,M) & = 1 + max\ \{\mathfrak{h}\,(S), \mathfrak{h}\,(M)\} & \mathfrak{h}\,(\{S_i\}_i) & = max_i\ \{\mathfrak{h}\,(S_i)\} \end{array}$$

Syntactic sugar For better readability, the λ-abstraction and the application operator are often applied to terms. These are simply syntactic sugar defined as follows:

$$\begin{array}{rcl} \lambda x.\{S_i\}_i & = & \{\lambda x.S_i\}_i \\ \{S_i\}_i\,M & = & \{S_i\,M\}_i \end{array}$$

For example, we write either

$$\lambda x.\{x, y\} \text{ or } \{\lambda x.x, \lambda x.y\}.$$

Similarly,

$$\{z,t\}\ \{y\} \text{ and } \{z\ \{y\},\ t\ \{y\}\}$$

denote the same term.

2.3 Substitutions

We define the application of substitutions in three steps:

(i) First, we define the substitution of a variable by a simple term in a simple term. This operation has type $\mathbb{S} \times \mathcal{X} \times \mathbb{S} \to \mathbb{S}$.

(ii) Then, we define the substitution of a variable by a term in a simple term. This operation has type $\mathbb{S} \times \mathcal{X} \times \mathbb{M} \to \mathbb{M}$. Note that the substitution of variable by a term in a simple term gives a term.

(iii) Finally, we extend the previous operation to term, that is we define the substitution of a variable by a term in a term. This operation has type $\mathbb{M} \times \mathcal{X} \times \mathbb{M} \to \mathbb{M}$.

Definition 2.3 (Simple substitution) *Let x be a variable and T be a simple term. We define by induction on the simple term S the substitution $[\ :=\ ] : \mathbb{S} \times \mathcal{X} \times \mathbb{S} \to \mathbb{S}$ as follows:*

$$\begin{array}{lcl} y[x := T] & = & \begin{cases} T & \textit{if } x = y \\ y & \textit{otherwise} \end{cases} \\ (\lambda y.S_0)[x := T] & = & \lambda y.(S_0[x := T]) \\ (S_0\ \{S_i\}_i)[x := T] & = & (S[x := T])\ \{S_i[x := T]\}_i \end{array}$$

In the abstraction case, we take the usual precautions assuming without loss of generality and thanks to the α-conversion that $y \neq x$ and y does not occur free in T.

Definition 2.4 (Substitution in simple terms) *Let x be a variable and M be a term. We define by induction on the simple term S the substitution $[\ :=\ ] : \mathbb{S} \times \mathcal{X} \times \mathbb{M} \to \mathbb{M}$ as follows:*

$$\begin{array}{lcl} y[x := M] & = & \begin{cases} M & \textit{if } x = y \\ \{y\} & \textit{otherwise} \end{cases} \\ (\lambda y.S)[x := M] & = & \lambda y.(S[x := M]) \\ (S\ \{T_i\}_i)[x := M] & = & (S[x := M])\ \bigcup_i T_i[x := M] \end{array}$$

Definition 2.5 (Substitution in terms) *The definition of substitutions can be* *extended to terms by setting:*

$$\{S_i\}_{i=1}^n[x := M] = \Big\{S_i[x := M]\Big\}_{i=1}^n$$

Example 2.6 *We consider the simple term $y\,\{x\}$. The substitution of x by the term $\{z,t\}$ gives:*

$$\begin{aligned}(y\,\{x\})[x := \{z,t\}] &= (y[x := \{z,t\}]) \cup x[x := \{z,t\}]\\ &= \{y\} \cup x[x := \{z,t\}]\\ &= \{y\}\{z,t\}\\ &= \{y\{z,t\}\}\end{aligned}$$

The substitution of y by $\{z,t\}$ in $y\,\{x\}$ gives:

$$\begin{aligned}(y\{x\})[y := \{z,t\}] &= (y[y := \{z,t\}]) \cup x[y := \{z,t\}]\\ &= (y[y := \{z,t\}])\{x\}\\ &= (\{z,t\})\{x\}\\ &= \{z\{x\},\ t\{x\}\}\end{aligned}$$

The substitution lemma is valid.

Lemma 2.7 (Substitution lemma) *Let M, N and P be terms and x be a variable. If x is not free in P then we have*

$$M[x := N][y := P] = M[y := P][x := N[y := P]]$$

3 Relations on the terms of the $\lambda_{\shortparallel}$-calculus

We shall consider two kinds of relations: relation from simple terms to terms which are subsets of $\mathbb{S} \times \mathbb{M}$ and relations from terms to terms which are subsets of $\mathbb{M} \times \mathbb{M}$. To simplify the reading, we use the following definitions:

Definition 3.1 (Relations on (simple) terms) *A relation of simple terms is a relation which is a subset of $\mathbb{S} \times \mathbb{M}$. A relation on terms is a relation which is a subset of $\mathbb{M} \times \mathbb{M}$.*

To define a relation on the terms of the $\lambda_{\shortparallel}$-calculus we will always proceed in the two following steps: first, we define a relation on simple terms and then we extend it to terms. The second point is studied in details in Sect. 3.1. We focus on the first one.

The specificity of this work is that the syntax consists in two different sets (the set of simple terms and the set of terms) and that a simple term is going to be reduced to a term (thus moving from a syntactical category to another one). The usual approach (see for example [2]) to define the operational semantics of higher-order languages (usually called the reduction relation) first defines the reduction at the head position (usually called the notion of reduction) and then considers its compatible closure (that gives the reduction relation).

In the framework of the $\lambda_{||}$-calculus, the same approach must be used with care. Since a simple term reduces to a term, we have to make clear what a compatible relation is or in other words, we have to make clear how the compatible closure of a notion of reduction is computed.

For relations on simple terms, we adapt the usual notion of "contextual relations" that are relations that verify

$$\text{If } S \rightarrow_\tau M \text{ then } C[S] \rightarrow_\tau C[M]$$

for any context $C[\,]$. In the $\lambda_{||}$-calculus, the context $C[\,]$ may put the simple term S under an abstraction, or in functional position of an application or in a set that is located in the applicative position of an application. The notion of contextual relation (Def. 3.7) formalises this idea.

For relation of terms, we have moreover to specify how the relation behaves with respect to sets. This gives the notion of additive relations (Def. 3.4).

Nevertheless, in the context of relations that simultaneously reduce several redexes, the notion of contextual and additive relations should be adapted. This leads to the notion of multiplicative and parallel relations (Def. 3.5 and 3.8).

In the following sections, when defining a new relation we will always state its related properties (additive, multiplicative, contextual or parallel). This helps us not only in proofs but also to increase intuitions.

3.1 Extending a relation on simple terms to a relation on terms.

Let τ be a relation on simple terms. We want to determine the different ways to extend it to a relation on terms.

The first way is to use the canonical injection of simple terms into terms (previously denoted η). Then, if $S \rightarrow_\tau M$ then $\{S\} \rightarrow_{\widehat{\tau}} M$. This extension is called *the singleton* extension and it will be used only for technical reasons.

Let us examine other ways. Suppose given a term M that we consider as the union of (singletons of) simple terms. We want to determine the successors of M by an extended relation. They can be either the terms obtained by replacing *one* simple term of M by one of its successors by τ or the terms obtained by replacing *all* the simple terms of M by one of its successors by τ. By analogy with the terminology used in linear logics we call the former the *additive* extension of τ and the latter the *multiplicative* extension of τ.

Definition 3.2 (Extending a relation) *Given a relation τ on simple terms, we* *define three relations $\widehat{\tau}$, $\overline{\tau}$ and $\widetilde{\tau}$ on terms as follows:*

$$\textsc{Singleton}\ \frac{S \to_\tau M}{\{S\} \to_{\widehat{\tau}} M}$$

$$\textsc{Additive}\ \frac{S \to_\tau M}{\{S\} \cup E \to_{\widetilde{\tau}} M \cup E}$$

$$\textsc{Multiplicative}\ \frac{\forall i, S_i \to_\tau M_i}{\cup_i \{S_i\} \to_{\overline{\tau}} \cup_i M_i}$$

We will often refer to the following remark that express that the three extensions (the singleton, the additive and the multiplicative) are continuous.

Lemma 3.3 *Let $(\tau_i)_i$ be an increasing sequence of relation on simple terms and let τ be the relation defined as $\tau = \cup_i \tau_i$. Then we have $\widehat{\tau} = \cup_i\ \widehat{\tau_i}$, $\widetilde{\tau} = \cup_i\ \widetilde{\tau_i}$ and $\overline{\tau} = \cup_i\ \overline{\tau_i}$.*

Proof We only prove the first equality. The two other ones are similar.

$$\begin{aligned}
M \to_{\cup_i \widehat{\tau_i}} N &\Leftrightarrow \exists i_0, M \to_{\widehat{\tau_{i_0}}} N \\
&\Leftrightarrow \exists i_0, M = \{S\} \text{ and } S \to_{\tau_{i_0}} N \\
&\Leftrightarrow M = \{S\} \text{ and } S \to_{\cup \tau_i} N \\
&\Leftrightarrow M \to_{\widehat{\cup_i \tau_i}} N
\end{aligned}$$

□

3.2 Relations

We first define two different behaviors of a relation with respect to sets.

Definition 3.4 (Additive relations) *A relation τ on terms is* additive *if for all terms M_1, M_1' such that $M_1 \to_\tau M_1'$ then for all terms M_2 we have $M_1 \cup M_2 \to_\tau M_1' \cup M_2$.*

We can remark that if τ is a relation on simple terms then the relation $\widetilde{\tau}$ is an additive relation.

Definition 3.5 (Multiplicative relations) *A relation τ on terms is* multiplicative *if for all terms M_1, M_1', M_2 and M_2' such that $M_1 \to_\tau M_1'$ and $M_2 \to_\tau M_2'$ we have $M_1 \cup M_2 \to_\tau M_1' \cup M_2'$.*

We can remark that if τ is a relation on simple terms then the relation $\overline{\tau}$ is a multiplicative relation.

Additive and multiplicative relations are dual in the following sense:

Proposition 3.6 *The reflexive and transitive closure of an additive relation is a multiplicative relation.*

Proof Let τ be an additive relation. We want to show that $(\tau)^*$ is multiplicative that is

$$\forall M_1, M_1', M_2, M_2' \qquad M_1 \mapsto\!\!\!\twoheadrightarrow_\tau M_1',\ M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_2' \Rightarrow M_1 \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2' \tag{1}$$

It is sufficient to prove that

$$\forall M_1, M_1' \qquad M_1 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \Rightarrow \forall M_2\ \ M_1 \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2. \tag{2}$$

Suppose that (2) is true and let us show (1). Let M_1, M_1', M_2, M_2' be terms such that $M_1 \mapsto\!\!\!\twoheadrightarrow_\tau M_1'$ and $M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_2'$. Applying (2) twice we get $M_1 \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2$ and then $M_1' \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2'$. Then by transitivity of $(\tau)^*$, we obtain $M_1 \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2'$. To prove (2), it is sufficient to prove that

$$\forall M_1, M_1' \qquad M_1 \rightarrow_\tau M_1' \Rightarrow \forall M_2\ \ M_1 \cup M_2 \rightarrow_\tau M_1' \cup M_2. \tag{3}$$

Suppose that (3) is true and let us show (2). Let M_1 and M_1' be terms such that $M_1 \mapsto\!\!\!\twoheadrightarrow_\tau M_1'$. By definition of $(\tau)^*$ there exist n terms $M_1^1, \dots M_1^n$ such that $M_1^1 \rightarrow_\tau M_1^2 \rightarrow_\tau \dots \rightarrow_\tau M_1^n$ with $M_1 = M_1^1$ and $M_1^n = M_1'$. Applying (3) n times we get $M_1 \cup M_2 \rightarrow_\tau \dots \rightarrow_\tau M_1' \cup M_2$ that is $M_1 \cup M_2 \mapsto\!\!\!\twoheadrightarrow_\tau M_1' \cup M_2$. We finally remark that (3) is true since τ is additive by hypothesis. □

We introduce in the two following definitions two different behaviors of a relation on terms with respect to the application and the abstraction operator.

Definition 3.7 (Contextual relations) *A relation τ on simple terms is contextual if it satisfies the two following conditions:*

(i) Contextuality w.r.t. to the abstraction operator: *For each variable x of $\mathcal{X}$, for each simple term S and for each term M such that $S \rightarrow_\tau M$, we have $\lambda x.S \rightarrow_\tau \lambda x.M$.*

(ii) Contextuality w.r.t. to the application operator:
   (a) *For all terms M, N and for each simple term S such that $S \rightarrow_\tau M$, we have $SN \rightarrow_\tau MN$.*
   (b) *For all terms M, N and for each simple term S such that $M \rightarrow_{\tilde{\tau}} N$, we have $SM \rightarrow_\tau SN$.*

Definition 3.8 (Parallel relations) *A relation τ on terms is* parallel *if it is reflexive, multiplicative and satisfies the two following conditions:*

(i) Parallelism w.r.t. to the abstraction operator: *For all variable $x \in \mathcal{X}$, for all terms M and M' such that $M \rightarrow_\tau M'$, we have $\lambda x.M \rightarrow_\tau \lambda x.M'$.*

(ii) Parallelism w.r.t. to the application operator: *For all terms M_1, M_2, M_1' and M_2' such that $M_1 \rightarrow_\tau M_1'$ and $M_2 \rightarrow_\tau M_2'$, $M_1 M_2 \rightarrow_\tau M_1' M_2'$.*

We often say that a relation has the parallelism property to means that it is a parallel relation.

The duality between additive and multiplicative relations can be extended to contextual and parallel relations:

Proposition 3.9 *Let τ be a contextual relation on simple terms. Then the reflexive and transitive closure of the relation $\widetilde{\tau}$, that is the relation $(\widetilde{\tau})^*$, is parallel.*

Proof The proof is similar to Prop. 3.6. □

Every parallel relation is compatible with the substitution application in the following sense:

Lemma 3.10 *Let τ be a parallel relation on terms. If M, N, M' are terms such that $M \to_\tau M'$ then $N[x := M] \to_\tau N[x := M']$.*

Proof The proof is by induction on N. If $N \in \mathcal{P}^+_{<\infty}(\mathbb{S}_0)$, then the result is obvious. Suppose that the result is true for all terms in $\mathcal{P}^+_{<\infty}(\mathbb{S}_k)$. Let N be a term belonging to $\mathcal{P}^+_{<\infty}(\mathbb{S}_{k+1})$. Then $N = \{S_i\}_i$ with $S_i \in \mathbb{S}_{k+1}$. We prove that for all i, $S_i[x := M] \to_\tau S_i[x := M']$ and then we conclude by multiplicativity of τ. There are three cases:

(i) If S_{i_0} belongs to $\mathbb{S}_{k+1}$ and it belongs also to $\mathbb{S}_k$ then the result holds by induction.

(ii) If S_{i_0} belongs to $\mathbb{S}_{k+1}$ with $S_{i_0} = \lambda x.S'_{i_0}$ and $S'_{i_0} \in \mathbb{S}_k$. Then by induction hypothesis, we have $S'_{i_0}[x := M] \to_\tau S'_{i_0}[x := M']$. By parallelism of the relation $\to_\tau$, we obtain $\lambda x.S'_{i_0}[x := M] \to_\tau \lambda x.S'_{i_0}[x := M']$ which means $S_{i_0}[x := M] \to_\tau S_{i_0}[x := M']$

(iii) If S_{i_0} belongs to $\mathbb{S}_{k+1}$ with $S_{i_0} = S'_{i_0}N'_{i_0}$, $S'_{i_0} \in \mathbb{S}_k$ and $N'_{i_0} \in \mathcal{P}^+_{<\infty}(\mathbb{S}_k)$. This case is similar to the previous one: we apply the induction hypothesis on S'_{i_0} and N'_{i_0} and we conclude by the parallelism of τ.

□

The following definition and the following lemma will be crucial in the proof of the Church-Rosser property of the Sect. 5. They formalize the idea that if a multiplicative relation on simple terms satisfies the multi-diamond property (formally if its singleton extension satisfies the diamond property) then its multiplicative extension also verifies this property.

Definition 3.11 (Multi-diamond) *A pair of binary relations (τ, φ) on terms satisfies the multi-diamond property if for any term M, for any $m > 0$ and for any terms $M_1, \ldots, M_m$ such that $M \to_\tau M_i$ for all i, then there exists a term M' such that for all i we have $M_i \to_\varphi M'$.*

$$
\begin{tikzcd}
 & M \arrow[ld, "\tau"'] \arrow[d, "\tau"] \arrow[rrd, "\tau"] & & \\
M_1 \arrow[rd, dashed, "\varphi"'] & M_2 \arrow[d, dashed, "\varphi"] & \cdots & M_n \arrow[lld, dashed, "\varphi"] \\
 & M' & &
\end{tikzcd}
$$

Lemma 3.12 *Let $\tau \subseteq \mathbb{S} \times \mathbb{M}$ be a relation on simple terms and let $\varphi \subseteq \mathbb{M} \times \mathbb{M}$ be a multiplicative relation on terms. If the pair $(\widehat{\tau}, \varphi)$ satisfies the multi-diamond property then the pair $(\overline{\tau}, \varphi)$ also satisfies the multi-diamond property.*

Proof Let M and $M_1, \ldots, M_n$ be terms such that for all i we have $M \rightarrow_{\overline{\tau}} M_i$. Then $M = \{S_j\}_j$ and $M_i = \{N_j^i\}_j$ with $S_j \rightarrow_\tau N_j^i$ for all i and all j. Since $(\widehat{\tau}, \varphi)$ satisfies the multi-diamond property there exists a term $N_j^i \rightarrow_\varphi N_j'$. Since φ is multiplicative we have $\{N_j^i\}_j \rightarrow_\varphi \{N_j'\}_j$ for all i, that is to say, $M_i \rightarrow_\varphi M'$ with $M' = \{N_j'\}_j$.

□

4 Operational semantics of the $\lambda_{||}$-calculus

To define the operational semantics of the $\lambda_{||}$-calculus, we first define the one-step reduction β^1 on simple terms. Then we extend it on terms using the additive extension: this provides the one-step reduction on terms. Next, we consider the reflexive and transitive closure of the latter that defines the relation β. In the following section, we will show that this relation is confluent.

4.1 Definition of β^1-reduction

Definition 4.1 *We define an increasing sequence $(\beta_k^1)_k$ of relations on simple terms by induction. The relation β_0^1 is the empty relation and the relation β_{k+1}^1 is defined by induction as follows:*

$$\frac{S \rightarrow_{\beta_k^1} M}{S \rightarrow_{\beta_{k+1}^1} M} \qquad \frac{}{(\lambda x.S)\ M \rightarrow_{\beta_{k+1}^1} S[x := M]}$$

$$\frac{S \rightarrow_{\beta_k^1} M}{\lambda x.S \rightarrow_{\beta_{k+1}^1} \lambda x.M} \qquad \frac{M \rightarrow_{\widetilde{\beta_k^1}} M'}{S\ M \rightarrow_{\beta_{k+1}^1} \{S\ M'\}} \qquad \frac{S \rightarrow_{\beta_k^1} M'}{S\ M \rightarrow_{\beta_{k+1}^1} M'\ M}$$

We define

$$\beta^1 \triangleq \bigcup_{k=0}^{\infty} \beta_k^1$$

We have defined the one-step reduction β^1 that reduces a simple term into a term. The one-step reduction on terms is obtained by considering the additive extension of β^1 namely $\widetilde{\beta^1}$. Its reflexive and transitive closure is the relation denoted β that is:

$$\beta \triangleq (\widetilde{\beta^1})^*$$

We can consider the λ-calculus as a sub-calculus of the $\lambda_{||}$-calculus where all terms are singletons. This is illustrated in the following example.

Example 4.2 The λ-term $(\lambda z.\lambda t.zt)(\lambda xy.x)$ can be encoded in the $\lambda_{\shortparallel}$-calculus and we can simulate the reduction of the λ-calculus:

$$\begin{aligned}\{(\lambda t.\lambda t.z\{t\})\{\lambda xy.x\}\} &\to_{\widetilde{\beta^1}} \{\lambda t.(\lambda xy.x)\{t\}\}\\ &\to_{\widetilde{\beta^1}} \{\lambda t.\lambda y.t\}\end{aligned}$$

The λ-term $\omega = (\lambda x.xx)\lambda x.xx$ of the λ-calculus can be encoded in the $\lambda_{\shortparallel}$-calculus and we can simulate the reduction of the λ-calculus:

$$\begin{aligned}\{(\lambda x.x\{x\})\lambda x.x\{x\}\} &\to_{\widetilde{\beta^1}} \{(\lambda x.x\{x\})\lambda x.x\{x\}\}\\ &\to_{\widetilde{\beta^1}} \cdots\end{aligned}$$

Remark 4.3 In the λ-calculus for (strict) parallel functions of G. Boudol, the application of the λ-abstraction and the application operator to *terms* is not a syntactic sugar as in the $\lambda_{\shortparallel}$-calculus introduced in this paper but is directly part of the syntax and the two following equations

$$\begin{aligned}\lambda x.\{S_i\}_i &= \{\lambda x.S_i\}_i\\ \{S_i\}_i\, M &= \{S_i\, M\}_i\end{aligned}$$

are oriented from left to right and used as evaluation rules, at the same level as the β-rule of the λ-calculus.

The result is a calculus that distinguishes all the following terms

$$(g_1 \shortparallel g_2 \shortparallel g_3)z \qquad (g_1 \shortparallel g_2)z \shortparallel g_3z \qquad g_1z \shortparallel g_2z \shortparallel g_3z$$

whereas they are all represented in the framework of the $\lambda_{\shortparallel}$-calculus by the canonical term

$$\{g_1z, g_2z, g_3z\}$$

We conclude this section by the analysis of the relation β^1 and β.

Lemma 4.4 *The relation β^1 is contextual.*

Proof Statements (i) and (ii-a) of Definition 3.7 are obtained by definition of β^1. In fact, if $S \to_{\beta^1} M'$ them there exists a k such that $S \to_{\beta^1_k} M'$. We obtain $SN \to_{\beta^1_{k+1}} M'N$ and thus $SN \to_{\beta^1} M'N$. Statement (ii-a) follows in the same way. Statement (ii-b) is an easy consequence of the definition of $\widetilde{\beta^1}$ and the property of continuity. In fact, we can remark that if $M \to_{\widetilde{\beta^1}} M'$ then there exists, by applying Rem. 3.3 an indice k such that $M \to_{\widetilde{\beta^1_k}} M'$ and then $SM \to_{\beta^1_{k+1}} SM'$ (definition of β^1).

□

Proposition 4.5 *The relation β is parallel.*

Proof By Lemma 4.4, the relation β^1 is contextual. Then we can apply Proposition 3.9. $\square$

5 The Church-Rosser property

To avoid confusions with the terminology of parallel relations in the sense of Tait and Martin-Löf and in the sense Sect. 3, the formers will be called *simultaneous* relations.

To prove the confluence the relation β of the $\lambda_{\shortparallel}$-calculus, we proceed in the same way as in the proof of the confluence of the λ-calculus based on the parallel reduction *à la* Tait and Martin-Löf that was studied in [22].

We first define a variant of the β^1-reduction (that was defined in Sect. 4) that reduce simultaneously several redexes in a single step. This relation is denoted by B. We then show that the reflexive and transitive closure of its parallel extension, that is the reflexive and transitive closure of the relation $\overline{B}$, is equal to the relation β. We finally show the confluence of the $\overline{B}$ and then deduce the confluence of β.

5.1 The simultaneous reduction $\overline{B}$

Definition 5.1 (B-reduction) *The relation on simple terms B is defined as the union of an increasing sequence $(B_k)_k$ of relations on simple terms. The relation B_0 is equal to η. The relation B_{k+1} is defined by induction as follows:*

$$\frac{S \to_{B_k} M}{S \to_{B_{k+1}} M} \qquad\qquad \frac{S \to_{B_k} N'}{\lambda x.S \to_{B_{k+1}} \lambda x.N'}$$

$$\frac{S \to_{B_k} N' \quad M \to_{\overline{B_k}} M'}{SM \to_{B_{k+1}} N'M'} \qquad\qquad \frac{S \to_{B_k} N' \quad M \to_{\overline{B_k}} M'}{(\lambda x.S)M \to_{B_{k+1}} N'[x := M']}$$

The relation B is thus defined as

$$B \triangleq \bigcup_{k=0}^{\infty} B_k \ .$$

Note that in the definition of B we use the multiplicative extension (while in the definition of β^1 we used the additive one) in order to simultaneously reduce several redexes in a single step.

We show that the relation $\overline{B}$ is parallel. This gives an example of a parallel relation that is not a reflexive and transitive closure of a contextual relation.

Proposition 5.2 *The relation $\overline{B}$ is parallel.*

Proof We first show that the relation $\overline{B}$ is reflexive, that is that the identity relation on terms denoted `id` is included in $\overline{B}$. This is true since

$$\overline{B_0} \subseteq \overline{B} \qquad \text{and} \qquad \overline{B_0} = \overline{\eta} = \texttt{id}$$

The multiplicativity of $\overline{B}$ is obvious. Let us show the parallelism w.r.t. the abstraction operator. Let M_1 and M'_1 be terms such that $M_1 \to_{\overline{B}} M'_1$. This means that $M_1 = \cup_i\{S_i\}$ and $M'_1 = \cup_i N_i$ with $S_i \to_B N_i$. Then we have $\lambda x.S_i \to_B \lambda x.N_i$ for all i, which proves $\lambda x.M_1 \to_{\overline{B}} \lambda x.M'_1$. Let us show now the parallelism w.r.t. the application operator. Let M_1, M'_1, M_2 and M'_2 be terms such that $M_1 \to_{\overline{B}} M'_1$ and $M_2 \to_{\overline{B}} M'_2$. This means that we have $M_1 = \cup_i\{S_i\}$ and $M'_1 = \cup_i N_i$ with $S_i \to_B N_i$ for all i. Then we have $S_i M_2 \to_B N_i M'_2$ for all i, which means that $M_1 M_2 \to_{\overline{B}} M'_1 M'_2$.

□

5.2 Confluence of $\overline{B}$

Lemma 5.3 *We have the following inclusions:*

$$\widetilde{\beta^1} \subseteq \overline{B} \subseteq \beta$$

The reflexive and transitive closure of $\overline{B}$ is thus β.

Proof To prove the first inclusion we prove by induction on k that $\widetilde{\beta^1_k} \subseteq \overline{B}$. The case $k = 0$ is obvious since β^1_0 is the empty relation. For the induction case, we suppose that $\widetilde{\beta^1_k} \subseteq \overline{B}$. We want to prove that $\widetilde{\beta^1_{k+1}} \subseteq \overline{B}$. Let M and N be two terms such that $M = \{S\} \cup E \to_{\widetilde{\beta^1_{k+1}}} P \cup E = N$ with $S \to_{\beta^1_{k+1}} P$. We want to prove that $\{S\} \cup E \to_{\overline{B}} P \cup E$ but by the parallelism of $\overline{B}$ and since $\overline{B}$ is reflexive (and thus in particular $E \to_{\overline{B}} E$) it is sufficient to prove that $\{S\} \to_{\overline{B}} P$, which can be done by case on the last rule used to prove $S \to_{\beta^1_{k+1}} P$:

- If $S \to_{\beta^1_{k+1}} P$ with $S \to_{\beta^1_k} P$ then $\{S\} \to_{\widetilde{\beta^1_k}} P$ and by induction hypothesis we obtain $\{S\} \to_{\overline{B}} P$.
- If $S \to_{\beta^1_{k+1}} P$ with $S = (\lambda x.S_1)M_1$, $P = S_1[x := M_1]$. Then we have $(\lambda x.S_1)M_1 \to_{\overline{B_1}} S_1[x := M_1]$ and $\{(\lambda x.S_1)M_1\} \to_{\overline{B_1}} S_1[x := M_1]$. By Rem. 3.3 applied to the relation B the result holds.
- If $S \to_{\beta^1_{k+1}} P$ with $S = \lambda x.S_1$, $P = \lambda x.P_1$ and $S_1 \to_{\beta^1_k} P_1$. By induction hypothesis, we obtain $\{S_1\} \to_{\overline{B}} P_1$ and then by the parallelism of $\overline{B}$ we conclude the case.
- If $S \to_{\beta^1_{k+1}} P$ with $S = S_1 M_1$, $P = \{S_1 M'_1\}$ and $M_1 \to_{\widetilde{\beta^1_k}} M'_1$. By induction hypothesis, we obtain $M_1 \to_{\overline{B}} M'_1$. Again by the parallelism of $\overline{B}$ we conclude $\{S_1 M_1\} \to_{\overline{B}} \{S_1 M'_1\}$.
- If $S \to_{\beta^1_{k+1}} P$ with $S = S_1 M_1$, $P = P'_1 M_1$ and $S_1 \to_{\beta^1_k} M_1$. This case is similar to the previous ones.

We now prove the second inclusion. We can prove that $\overline{\eta\beta} \subseteq \beta$ that is, the multiplicative extension of the composition of η and β is included in β. In fact, let M and N be terms such that $M \to_{\eta\beta} N$, that is there exist simple terms $S_1, \ldots, S_n$ and terms $P_1, \ldots, P_n$ such that $M = \cup_i\{S_i\}$ and $N = \cup_i P_i$ with $S_i \to_{\eta\beta} P_i$ for all i.

By definition of η this means that $\{S_i\} \mapsto\!\!\!\twoheadrightarrow_\beta P_i$ for all i. By the parallelism of β we have $\cup_i\{S_i\} \mapsto\!\!\!\twoheadrightarrow_\beta \cup_i P_i$. This proves $M \mapsto\!\!\!\twoheadrightarrow_\beta N$.

To prove that $\overline{B} \subseteq \beta$, it is sufficient to prove that

$$B_k \subseteq \eta\beta.$$

In fact if $B_k \subseteq \eta\beta$ then $\overline{B} = \cup_k \overline{B_k} \subseteq \overline{\eta\beta} \subseteq \beta$.

This is what we do in the following, by induction on k. The case $k = 0$ is trivial. Suppose that $B_k \subseteq \eta\beta$ and let us prove that $B_{k+1} \subseteq \eta\beta$. Let S be a simple term and M be a term such that $S \rightarrow_{B_{k+1}} M$. By case on the last rule used to prove $S \rightarrow_{B_{k+1}} M$:

(i) If $S \rightarrow_{B_{k+1}} M$ with $S \rightarrow_{B_k} M$ then the result holds by induction.

(ii) If $S \rightarrow_{B_{k+1}} M$ with $S = S_1 M_1$ and $M = P_1 M_2$ with $S_1 \rightarrow_{B_k} P_1$ and $M_1 \rightarrow_{\overline{B}} M_2$. Then by induction hypothesis $S_1 \rightarrow_{\eta\beta} P_1$ and $M_1 \mapsto\!\!\!\twoheadrightarrow_\beta M_2$ that is $\{S_1\} \mapsto\!\!\!\twoheadrightarrow_\beta P_1$ and $M_1 \mapsto\!\!\!\twoheadrightarrow_\beta M_2$. By the parallelism of β we have $\{S_1 M_1\} \mapsto\!\!\!\twoheadrightarrow_\beta P_1 M_2$ that is $S_1 M_1 \rightarrow_{\eta\beta} P_1 M_2$.

(iii) If $S \rightarrow_{B_{k+1}} M$ with $S = (\lambda x.T_1)M_1$, $M = P_1[x := M_2]$, $T_1 \rightarrow_{B_k} P_1$ and $M_1 \rightarrow_{\overline{B_k}} M_2$. By induction hypothesis, we have $T_1 \rightarrow_{\eta\beta} P_1$ and $M_1 \mapsto\!\!\!\twoheadrightarrow_\beta M_2$. We want to show that $(\lambda x.T_1)M_1 \rightarrow_{\eta\beta} P_1[x := M_2]$ that is $\{(\lambda x.T_1)M_1\} \rightarrow_\beta P_1[x := M_2]$, which is obviously true.

(iv) If $S \rightarrow_{B_{k+1}} P$ with $S = \lambda x.S_1$, $P = \lambda x.P_1$ and $S_1 \rightarrow_{B_k} P_1$. By induction hypothesis, we obtain $\{S_1\} \mapsto\!\!\!\twoheadrightarrow_\beta P_1$ and by the parallelism of $\mapsto\!\!\!\twoheadrightarrow_\beta$ we conclude $\{\lambda x.S_1\} \mapsto\!\!\!\twoheadrightarrow_\beta \lambda x.P_1$ that is $S \rightarrow_{\eta\beta} M$.

□

Lemma 5.4 *Let x be a variable and M_1, M_1', M_2, M_2' be terms. If $M_1 \rightarrow_{\overline{B}} M_1'$ and $M_2 \rightarrow_{\overline{B}} M_2'$, then*

$$M_1[x := M_2] \rightarrow_{\overline{B}} M_1'[x := M_2']$$

Proof We prove by induction on k that if $M_1 \rightarrow_{\overline{B_k}} M_1'$ and $M_2 \rightarrow_{\overline{B}} M_2'$, then $M_1[x := M_2] \rightarrow_{\overline{B}} M_1'[x := M_2']$. For $k = 0$ we have $M_1 = M_1'$ and then we conclude by parallelism of $\overline{B}$ (applying Lemma 3.10).

□

Proposition 5.5 *The relation $\overline{B}$ is confluent.*

Proof We prove by induction on k that the pair $(\overline{B_k}, \overline{B})$ (this clearly entail the proposition). The case $k = 0$ is trivial since $\overline{B_0}$ is just the identity function. So let us assume that $(\overline{B_k}, \overline{B})$ satisfies the multi-diamond property and let us prove that $(\overline{B_{k+1}}, \overline{B})$ satisfies also the multi-diamond property. By Lemma 3.12, it is sufficient to show that $(\widehat{B_{k+1}}, \overline{B})$ satisfies the multi-diamond property. This is what we do in the following. Let S be a simple term and $M_1, \ldots, M_n$ be terms such that $S \rightarrow_{B_{k+1}} M_i$ for all i. By case on the structure of S.

- The case $S = x$ for a variable x is not possible.

- If $S = \lambda x.T$ then we have $M_i = \lambda x.N_i$ with $T \to_{B_k} N_i$ for all i. Then by induction hypothesis, there exists a N' such that $N_i \to_{\overline{B}} N'$ and thus $\lambda x.N_i \to_{\overline{B}} \lambda x.N'$. Then we are done.
- If $S = TM$ then there are two cases
  · If for all i we have $M_i = M_i^1 M_i^2$ with $T \to_{B_k} M_i^1$ and $M \to_{\overline{B_k}} M_i^2$. Then by induction hypothesis, there exist N'_1 and N'_2 such that $M_i^1 \to_{\overline{B}} N'_1$ and $M_i^2 \to_{\overline{B}} N'_2$ for all i. The parallelism of the relation $\overline{B}$ (Prop 5.2) concludes the case.
  · Otherwise, $T = \lambda x.U$ and there exists a $1 \leq q \leq n$ such that
  (i) $M_1 = N_1[x := P_1] \ldots$ and $M_q = N_q[x := P_q]$
  (ii) $M_{q+1} = (\lambda x.N_{q+1})P_{q+1} \ldots$ and $M_n = (\lambda x.N_n)P_n$
  with $\lambda x.U \to_{B_k} \lambda x.N_i$ and $M \to_{\overline{B_k}} P_i$. By induction hypothesis, there exist a term N' and a term P' such that $\lambda x.N_i \to_{\overline{B}} \lambda x.N'$ and $P_i \to_{\overline{B}} P'$. Then we prove that $M_i \to_{\overline{B}} N'[x := P']$ by applying Lemma 5.4 for the terms $M_1, \ldots, M_q$ and by definition of ρ otherwise.

□

5.3 Confluence of β

Theorem 5.6 *The relation β over terms of the $\lambda_{\shortparallel}$-calculus enjoys the Church-Rosser property.*

Proof Since the reflexive and transitive closure of $\overline{B}$ is β (Lemma 5.3) and since the relation $\overline{B}$ is confluent (Prop. 5.5) then the result clearly holds. □

Conclusion

We have studied an extension of the λ-calculus with term collections represented by canonical sets. This provides a clear operational semantics for the λ-calculi for (strict) parallel functions and this is a first step in the study of the ρ-calculus with non-unitary matching theories.

The work of A.Reilles [20,19] on canonical abstract syntax trees is strongly related to the approach of this paper. Actually, the former provides the capability to maintain (in a very efficient way) the internal representation of data in canonical form with respect to a rewrite system. In the case of the $\lambda_{\shortparallel}$-calculus, the set of rules used to ensure that the canonical invariant is the distributivity of sets over abstractions and applications.

Acknowledgments

A. Miquel suggests to the author the reading of [14]. This was the starting point of the work. H. Cirstea gave deep feedbacks on previous versions of the paper. This was very useful. We also thank L. Vaux and C. Kirchner for useful interactions and comments on this work.

References

[1] ASF+SDF. A component-based language development environment. http://www.cwi.nl/projects/MetaEnv/.

[2] H. Barendregt. *The Lambda-Calculus, its syntax and semantics.* Studies in Logic and the Foundation of Mathematics. Elsevier Science Publishers B. V. (North-Holland), Amsterdam, 1984. Second edition.

[3] G. Barthe, H. Cirstea, C. Kirchner, and L. Liquori. Pure patterns type systems. In *Principles of Programming Languages - POPL2003, New Orleans, USA*. ACM, January 2003.

[4] P. Borovanský, C. Kirchner, H. Kirchner, P.-E. Moreau, and C. Ringeissen. An overview of ELAN. In *Proc. of WRLA*, volume 15. ENTCS, September 1998.

[5] G. Boudol. Lambda-calculi for (strict) parallel functions. *Inf. Comput*, 108(1), January 1994.

[6] H. Cirstea. *Calcul de réécriture : fondements et applications*. PhD thesis, Université Henri Poincaré - Nancy I, 2000. October 25.

[7] H. Cirstea and C. Kirchner. The rewriting calculus — Part I *and* II. *Logic Journal of the Interest Group in Pure and Applied Logics*, 9(3):427–498, May 2001.

[8] H. Cirstea, C. Kirchner, and L. Liquori. Matching Power. In *Proceedings of RTA'2001*, Lecture Notes in Computer Science. Springer-Verlag, May 2001.

[9] H. Cirstea, L. Liquori, and B. Wack. Rewriting calculus with fixpoints: Untyped and first-order systems. volume 3085. Springer, 2003.

[10] M. Dezani-Ciancaglini, U. de'Liguoro, and A. Piperno. Filter models for a parallel and non deterministic lambda-calculus. In *Mathematical Foundations of Computer Science 1993, 18th International Symposium*, volume 711 of *lncs*, 1993.

[11] Elan. The ELAN system:. http://elan.loria.fr/.

[12] G. Faure and A. Miquel. Towards a denotational semantics for the rho-calculus. Technical report, LORIA, 2005.

[13] M. Fernández, I. Mackie, and F.-R. Sinot. Interaction nets vs. the rho-calculus: Introducing bigraphical nets. In *Proceedings of EXPRESS'05, satellite workshop of Concur*, ENTCS. Elsevier, 2005.

[14] T. Herhard and L. Reigner. The differential lambda-calculus. *Theoretical Computer Science*, 309, 2003.

[15] C. Marché. Normalized rewriting: An alternative to rewriting modulo a set of equations. *Journal Symb. Comput*, 21(3), 1996.

[16] Maude. The maude system:. http://maude.cs.uiuc.edu/.

[17] Mod. Modulogic home page. http://modulogic.inria.fr.

[18] V. Prevosto. *Conception et Implantation du langage FoC pour le développement de logiciels certifiés.* Thèse de doctorat, Université Paris 6, September 2003.

[19] A. Reilles. Canonical abstract syntax trees. In *Proceedings of the 6th International Workshop on Rewriting Logic and its Applications*. Electronic Notes in Theoretical Computer Science, 2006. to appear.

[20] A. Reilles. *Réécriture et compilation de confiance*. PhD thesis, November 2006.

[21] A. Stump, A. Deivanayagam, S. Kathol, D. Lingelbach, and D. Schobel. Rogue Decision Procedures. In *1st International Workshop on Pragmatics of Decision Procedures in Automated Reasoning*, 2003.

[22] M. Takahashi. Parallel reductions in λ-calculus. *Inf. Comput.*, 118(1), 1995.

[23] Tom. The Tom language:. http://tom.loria.fr/.

[24] B. Wack. The simply-typed pure pattern type system ensures strong normalization. IFIP-WCC TCS, 2004.

[25] B. Wack. A Curry-Howard-De Bruijn Isomorphism Modulo. *Under submission*, 2006.