UNIVERSITE D'ORLEANS
Faculté des Sciences
LIFO

Laboratoire d'Informatique Fondamentale d'Orléans
4, rue Léonard de Vinci, BP 6759
F-45067 Orléans Cedex 2
FRANCE

Rapport de Recherche

www : http://www.univ-orleans.fr/SCIENCES/LIFO/

Confluence of the BSλ-calculus

Frédéric Loulergue
Université d'Orléans, LIFO

Rapport N^o **99-10**

Confluence of the BS$\boldsymbol{\lambda}$-calculus *

Frédéric Loulergue[1]

LIFO, Université d'Orléans, B.P. 6759, 45067 Orléans Cedex 2, France

Abstract. The BSλ -calculus, a formal basis for functional languages expressing bulk synchronous parallel algorithms, is presented. It is then shown to be confluent.

1 High-level languages and parallel programming

For parallel programming to become as widespread as sequential programming, the languages supporting it should incorporate all the standard abstraction mechanisms including higher-order functions and recursion. Yet for such languages to be practical scalable programming tools, abstraction should not come at the expense of predictable performance. Unfortunately many parallel languages don't describe data placement so that performance is not predictable as a function of the source program. This problem occurs when data placement depends on language implementation, instead of semantics.

On the contrary, the bulk synchronous parallel (BSP) computing paradigm [Val90, McC96] demonstrates that programs explicitly written for a static number p of processors can have predictable execution costs on a wide variety of architectures. But the combination of BSP algorithms with high-level language features is not well understood so that the evolution of BSP languages is hindered. We are investigating this combination from a functional programming perspective and introduce here an extension of the λ-calculus with BSP operations.

The *flat* part of the calculus is inspired by the paradigm of algorithmic skeletons [Col89]; it is obtained by adding operators with special rules. But it differs from the standard notion of δ-rules[Bar86] in two ways:

1. Terms are divided into two sorts. One sort describes global objects and the other describes usual "sequential" objects, prevented from containing global terms. This absence of sort-nesting is verified statically in the syntax and realizes the concept of *explicit processes* by making explicit the data-parallel structure of terms. Explicit processes allow the construction of cost measures for BSP execution.
2. The calculus is defined by rules more general than δ-rules. They involve two operators and thus provide general control structures.

As a result, the calculus presented here can be used to design portable parallel languages in which algorithmic skeletons are not predefined and can be written as librairies.

In the next section we present the (flat) BSλ calculus, which is proved to be confluent.

2 The calculus

In this section we introduce an extension of the λ-calculus called the bulk-synchronous λ-calculus, or BSλ , adapting the notion of xappings [SH86] or data fields[CCL91, Lis96, VP92] to BSP in so-called direct mode: the parallel data structures are flat and map directly to physical processors. This difference, although apparently minor is crucial: BSλ programs require no flattening [Ble90, ch. 10] and have complete control of the computation / communication ratio. The calculus introduces operations for data-parallel programming but with explicit processes in the spirit of BSP.

We now describe the BSλ syntax and its equational theory with denotational and/or operational motivations. The reader is assumed to be familiar with the elements of BSP[SHM97].

* This paper presents the proof of the confluence of the BS$\boldsymbol{\lambda}$-calculus outlined in [LHF99]

Syntax

We consider a set $\dot{V}$ of *local* variables and a set $\bar{V}$ of *global* variables. Let $\dot{x}, \dot{y}, \ldots$ denote local variables and $\bar{x}, \bar{y}, \ldots$ denote global variables from now on. x will denote a variable which can be either local or global.

The syntax of BSλ begins with *local* terms e: λ-terms representing programs or values stored in a processor's local memory. The set $\dot{\Lambda}$ of local terms is given by the following grammar:

$$e ::= \dot{x} \ \mid\ e\, e \ \mid\ \lambda \dot{x}.\, e \ \mid\ c$$

where $\dot{x}$ denotes an arbitrary local variable. We will abbreviate to $(e_1 \to e_2, e_3)$ the conditional term $e_1\ e_2\ e_3$.

The principal BSλ terms E are called *global* and represent data fields i.e. maps from a fixed set of processor names to values. The exact processor names are left unspecified but assumed to be closed normal forms so that data fields can be expressed intensionally by a special constant called π (for *parallel* object). The term $\pi\, e$ will represent a data field whose values are given by function e applied to the processor names (Fig. 1).

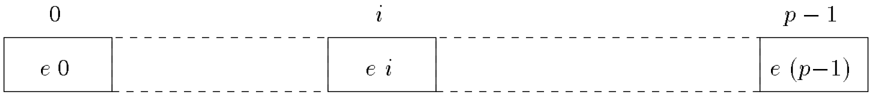

Fig. 1. The term πe

The set $\bar{\Lambda}$ of global terms is given by the following grammar[2]:

$$\begin{aligned} E ::= \bar{x} \ &\mid\ E\, E \ \mid\ E\, e \ \mid\ \lambda \bar{x}.\, E \ \mid\ \lambda \dot{x}.\, E \\ &\mid\ \pi\, e \ \mid\ E \# E \ \mid\ E\, ?\, E \ \mid\ (E \xrightarrow{e} E, E) \end{aligned}$$

and has the following denotational meaning. Let p be the *finite* number of processor names $\mathcal{N} = \{n_i \mid 0 \le i < p\}$, each being a closed λ-term. Global terms denote maps from $\mathcal{N}$ to local values, functions between them $(\lambda \bar{x}.\, E)$ or functions from local values to such maps $(\lambda \dot{x}.\, E)$. In particular the denotation of $\pi\, e$ has for value at n_i the value (normal form) of $e\ n_i$. The forms $E_1 \# E_2$ and $E_1\, ?\, E_2$ are called parallel application (*apply-par*) and *get* respectively. Apply-par represents pointwise application of a field of functions to a field of values (Fig. 2), i.e. the pure computation phase of a BSP superstep. Get represents the communication phase

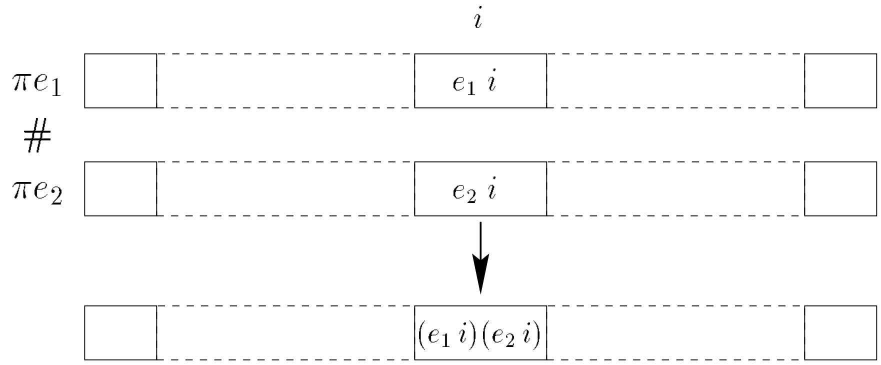

Fig. 2. The rule $(\#\pi)$

[2] Terms of the form $(\lambda \bar{x}.E)e$ (resp. $(\lambda \dot{x}.E)E'$) are not β-contracted and constitute implicit errors because they present a local argument to a global$\to$global function (resp. a global argument to a local$\to$global function). In practice a two-level type system should eliminate them, but we will not discuss this here.

of a BSP superstep: a collective data exchange with a barrier synchronization. In $E_1\,?\,E_2$, the resulting data field contains values from E_1 taken at processor names defined in E_2 (Fig. 3). The exact meanings of apply-

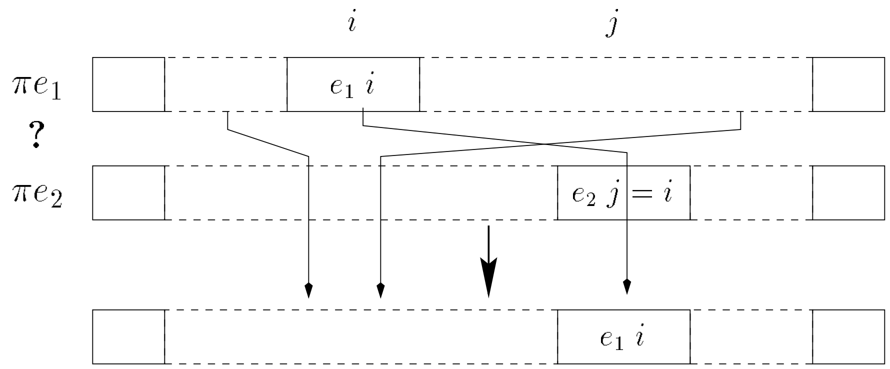

Fig. 3. The rule $(?\pi)$

par and get are defined by the BSλ equations. The last form of global terms define synchronous conditional expressions. The meaning of $(E_1 \overset{e}{\to} E_2, E_3)$ (not to be confused with $(e_1 \to e_2, e_3)$) is that of E_2 (resp. E_3) if the data field denoted by E_1 has value T (resp. F) at the processor name denoted by e (Fig. 4).

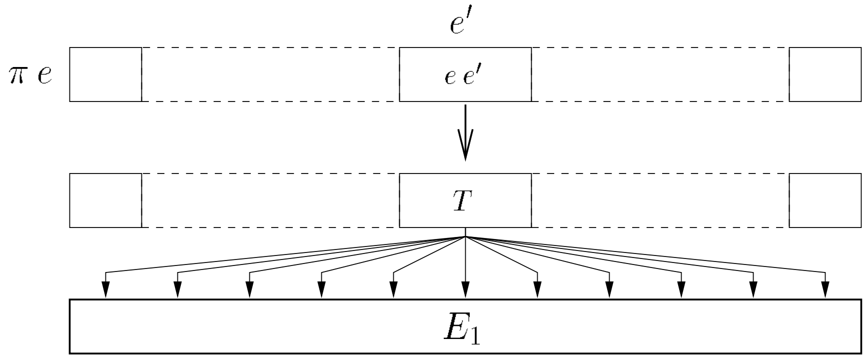

Fig. 4. The rule $(\overset{e}{\to} T)$

In the following we will identify terms modulo renaming of bound variables and we will use Barendregt's variable convention[Bar86] : if terms $t_1, \ldots t_n$ occur in a certain mathematical context then in these terms all bound variables are chosen to be different from free variables.

Equational semantics

We now define the equality theory of BSλ terms.

The equality of local terms is simply β-equivalence, obtained from the local β-contraction rule

$$(\beta) \qquad (\lambda\dot{x}.\ e)e' \doteq e[\dot{x} \leftarrow e'].$$

together with the following context rules:

$$\frac{m \doteq n}{m\,z \doteq n\,z} \qquad \frac{m \doteq n}{z\,m \doteq z\,n} \qquad \frac{m \doteq n}{\lambda\dot{x}.m \doteq \lambda\dot{x}.n}$$

The equality of global terms is defined by syntax-directed rules and context rules which determine the applicability of the former. Each rule defines an equality on global terms.

First, there are axioms for global beta-equivalence.

$$\begin{array}{ll}(B) & (\lambda \bar{x}.\ E)E' \ =_{(B)}\ E[\bar{x} \leftarrow E'] \\ (B') & (\lambda \dot{x}.\ E)e' \ =_{(B')}\ E[\dot{x} \leftarrow e']\end{array}$$

There are also axioms for the interaction of the field constructor π with the other BSP operations.

$$(?\pi)\frac{\forall n \in \mathcal{N}.\exists n' \in \mathcal{N}.e_2\, n \dot{=} n'}{(\pi\ e_1)\,?\,(\pi\ e_2)\ =_{(?\pi)}\ \pi(\lambda \dot{x}.\ e_1(e_2\ \dot{x}))},\ \text{with } \dot{x} \notin \mathrm{FV}(e_1) \cup \mathrm{FV}(e_2)$$

$$(\#\pi)\ (\pi\ e_1)\#(\pi\ e_2)\ =_{(\#\pi)}\ \pi(\lambda \dot{x}.\ (e_1\ \dot{x})(e_2\ \dot{x})),\ \text{with } \dot{x} \notin \mathrm{FV}(e_1) \cup \mathrm{FV}(e_2)$$

Equations meaning of the BSP operations on fields. In particular, get is functional composition inside the data field construction. The value of $\pi\, e_1\,?\,\pi\, e_2$ at processor name n_i is $e_1\,(e_2\, n_i)$, i.e. the value of $\pi\, e_1$ at processor name $e_2\, n_i$. Notice that, in practical terms, this represents an operation whereby every processor receives one and only one value from one and only one other processor. This restriction can be lifted[Lou99] and we will not mention it further in this paper.

Next, the global conditional is defined by two rules whose numerators refer to local computations.

$$(\overset{e}{\to} T) \qquad \frac{e\ e' \doteq T \quad e' \in \mathcal{N}}{((\pi\ e) \overset{e'}{\to} E_1, E_2)\ =_{(\overset{e}{\to} T)}\ E_1}$$

$$(\overset{e}{\to} F) \qquad \frac{e\ e' \doteq F \quad e' \in \mathcal{N}}{((\pi\ e) \overset{e'}{\to} E_1, E_2)\ =_{(\overset{e}{\to} F)}\ E_2}$$

The above two generate the following bulk-synchronous computation: first a pure computation phase where all processors evaluate local term e' yielding value v; then processor v evaluates $e\,v$ into value v'; if $v' = T$ (resp. F) then processor v broadcasts the order for global evaluation of E_1 (resp. E_2); otherwise the computation fails.

Those two rules are necessary to express algorithms of the form:

```
Repeat
  Parallel Iteration
Until Max of local errors < epsilon
```

Without those rules, the global control can not take into account data computed locally, ie global control can not depend on data.

Context rules

The BSλ rules are applicable in all context. Figure 5 gives the context rules.

3 BSλ is confluent

We orient from left to right the equalities of the denominator and the numerator of the context rules and only the denominator of the other rules of the previous sections to obtain a reduction relation.

If we orient also the numerator of the rule $(\overset{e}{\to} T)$ (resp. $(\overset{e}{\to} T)$), we will obtain the rule:

$$\frac{e\ e' \dot{\to} T \quad e' \in \mathcal{N}}{((\pi\ e) \overset{e'}{\to} E_1, E_2)\ \to\ E_1}$$

(resp. $\frac{e\ e' \dot{\to} F \quad e' \in \mathcal{N}}{((\pi\ e) \overset{e'}{\to} E_1, E_2)\ \to\ E_2}$) In this case, the term $(\pi(\lambda \dot{f}.\dot{f}\,T) \overset{\lambda \dot{x}.\dot{x}}{\to} \bar{x}, \bar{y})$ will be unreductable because

$$(\lambda \dot{f}.\dot{f}\,T)(\lambda \dot{x}.\dot{x})\ \dot{\to}\ (\lambda \dot{x}.\dot{x})T$$

$$\frac{(M,N) \in (=_{(B)} \cup =_{(B')} \cup =_{(?\pi)} \cup =_{(\#\pi)} \cup =_{(\stackrel{e}{\to} T)} \cup =_{(\stackrel{e}{\to} F)})}{M \ = \ N}$$

$$\frac{M \ = \ N}{\lambda x.M \ = \ \lambda x.N} \qquad \frac{m \doteq n}{\pi\, m \ = \ \pi\, n} \qquad \frac{M \ = \ N}{M\, Z \ = \ N\, Z} \qquad \frac{M \ = \ N}{Z\, M \ = \ Z\, N} \qquad \frac{m \doteq n}{Z\, m \ = \ Z\, n}$$

$$\frac{M \ = \ N}{Z \,?\, M \ = \ Z \,?\, N} \qquad \frac{M \ = \ N}{M \,?\, Z \ = \ N \,?\, Z} \qquad \frac{M \ = \ N}{Z \,\#\, M \ = \ Z \,\#\, N} \qquad \frac{M \ = \ N}{M \,\#\, Z \ = \ N \,\#\, Z}$$

$$\frac{M \ = \ N}{(M \stackrel{z}{\to} Z_1, Z_2) \ = \ (N \stackrel{z}{\to} Z_1, Z_2)} \qquad \frac{M_1 \ = \ N_1}{(Z \stackrel{z}{\to} M_1, Z_2) \ = \ (Z \stackrel{z}{\to} N_1, Z_2)}$$

$$\frac{M_2 \ = \ N_2}{(Z \stackrel{z}{\to} Z_1, M_2) \ = \ (Z \stackrel{z}{\to} Z_1, N_2)} \qquad \frac{m \doteq n}{(Z \stackrel{m}{\to} Z_1, Z_2) \ = \ (Z \stackrel{n}{\to} Z_1, Z_2)}$$

Fig. 5. Context rules

With one more step of reduction (which is allowed if the numerator of the rule remains $e\, e' \doteq T$), we have

$$(\lambda \dot{x}.\dot{x})T \ \dot{\to} \ T$$

and the rule for global conditional can be applied:

$$(\pi \lambda \dot{f}.\dot{f}\, T \stackrel{\lambda \dot{x}.\dot{x}}{\to} \bar{x}, \bar{y}) \ \to \ \bar{x}$$

We can define the global reduction $\to$ using the equational theory $\doteq$ (or the local reduction $\dot{\to}$) only because the BSλ calculus is composed with two levels and because the confluence of the local level is independent form the confluence of the global level. The reduction relation obtained by this choice of orientation is confluent.

The proof of the confluence of BSλ is an adaptation of the classic proof of the λ-calculus by Tait and Martin-Löf. We first define a relation $\rhd$ which includes the BSλ reduction and which is included in the transitive closure of the BSλ reduction. Then we show that this relation is strongly confluent. As the transitive closure of a strongly confluent relation is strongly confluent, the transitive closure of $\rhd$ is strongly confluent. By inclusion, this relation is in fact the transitive closure of the BSλ reduction and it ends the proof.

We first present the reduction on local terms and the associated strongly confluent relation ([Bar86]). Then we present the BSλ reduction and the associated strongly confluent relation and prove its strong confluence.

Reduction on local terms

The β reduction is written $\dot{\to}$.

Definition 1. Let be $\dot{\rhd}$ the relation on local terms defined by

$$(i)\ m \,\dot{\rhd}\, m \qquad (ii)\ \frac{m \,\dot{\rhd}\, n}{\lambda x.m \,\dot{\rhd}\, \lambda x.n}$$

$$(iii)\ \frac{m \,\dot{\rhd}\, m' \quad n \,\dot{\rhd}\, n'}{m\, n \,\dot{\rhd}\, m'\, n'} \qquad (iv)\ \frac{m \,\dot{\rhd}\, m' \quad n \,\dot{\rhd}\, n'}{(\lambda \dot{x}.m)n \,\dot{\rhd}\, m'[\dot{x} := n']}$$

We will use the following results :

Lemma 2. *If $m \,\dot{\rhd}\, m'$ and $n \,\dot{\rhd}\, n'$ then $m[\dot{x} := n] \,\dot{\rhd}\, m'[\dot{x} := n']$ [Bar86, page 60]*

Lemma 3. *$\dot{\rhd}$ is strongly confluent [Bar86, pages 61-62]*

Reduction on global terms

The BSλ reduction, written $\rightarrow$ is defined by orienting the equalities of section 2 from left to right.

Definition 4. The relation on global terms $\rhd$ is given by:

$$(I)\ M \rhd M \qquad (II)\ \frac{m \dot{\rhd} m'}{\pi m \rhd \pi m'} \qquad (III)\ \frac{M \rhd M'}{\lambda x.M \rhd \lambda x.M'}$$

$$(IV)\ \frac{M \rhd M' \quad N \rhd N'}{M\, N \rhd M'\, N'} \qquad (V)\ \frac{M \rhd M' \quad n \dot{\rhd} n'}{M\, n \rhd M'\, n'}$$

$$(VI)\ \frac{M \rhd M' \quad N \rhd N'}{M \# N \rhd M' \# N'} \qquad (VII)\ \frac{M \rhd M' \quad N \rhd N'}{M\, ?\, N \rhd M'\, ?\, N'}$$

$$(VIII)\ \frac{M \rhd M' \quad m \dot{\rhd} m' \quad M_1 \rhd M_1' \quad M_2 \rhd M_2'}{(M \overset{m}{\to} M_1, M_2) \rhd (M' \overset{m'}{\to} M_1', M_2')}$$

$$(IX)\ \frac{M \rhd M' \quad N \rhd N'}{(\lambda \bar{x}.M)N \rhd M'[\bar{x} := N']} \qquad (X)\ \frac{M \rhd M' \quad n \dot{\rhd} n'}{(\lambda \dot{x}.M)n \rhd M'[\dot{x} := n']}$$

$$(XI)\ \frac{m \dot{\rhd} m' \quad n \dot{\rhd} n' \quad \forall i \in \mathcal{N}.\exists i' \in \mathcal{N}.n\, i \dot{=} i'}{\pi\, m\, ?\, \pi\, n \rhd \pi(\lambda \dot{x}.m'(n'\, \dot{x}))} \qquad (XII)\ \frac{m \dot{\rhd} m' \quad n \dot{\rhd} n'}{\pi\, m \# \pi\, n \rhd \pi(\lambda \dot{x}.(m'\, \dot{x})(n'\, \dot{x}))}$$

$$(XIII)\ \frac{M_1 \rhd M_1' \quad m\, n \dot{=} T \quad n \in \mathcal{N}}{(\pi\, m \overset{n}{\to} M_1, M_2) \rhd M_1'} \qquad (XIV)\ \frac{M_2 \rhd M_2' \quad m\, n \dot{=} F \quad n \in \mathcal{N}}{(\pi\, m \overset{n}{\to} M_1, M_2) \rhd M_2'}$$

To prove that $\rhd$ is a strongly confluent relation we need the following results:

Lemma 5. *Let $M, N, L \in \bar{\Lambda}$, $n, l \in \dot{\Lambda}$, $\bar{x} \not\equiv \bar{y}$, $\dot{x} \not\equiv \dot{y}$, $\bar{x} \notin FV(L)$, $\dot{x} \notin FV(L)$, $\dot{x} \notin FV(l)$. Then*

1. $M[\bar{x} := N][\bar{y} := L] \equiv M[\bar{y} := L]\Big[\bar{x} := N[\bar{y} := L]\Big]$
2. $M[\dot{x} := n][\bar{y} := L] \equiv M[\bar{y} := L][\dot{x} := n]$
3. $M[\bar{x} := N][\dot{y} := l] \equiv M[\dot{y} := l]\Big[\bar{x} := N[\dot{y} := l]\Big]$
4. $M[\dot{x} := n][\dot{y} := l] \equiv M[\dot{y} := l]\Big[\dot{x} := n[\dot{y} := l]\Big]$

Lemma 6. *Let be $M, N \in \bar{\Lambda}$ such that $M \rhd M'$ and $N \rhd N'$. Then*

$$M[\bar{x} := N] \rhd M'[\bar{x} := N']$$

Proof. By induction on the structure of $M \rhd M'$.

1. $M \rhd M'$ is $M \rhd M$. So we must show $M[\bar{x} := N] \rhd M[\bar{x} := N']$. We prove it by induction on the structure of M.
   (a) M is $\bar{x}$. $N \rhd N'$ holds by hypothesis.
   (b) M is $\bar{y}$ different from $\bar{x}$. $\bar{y} \rhd \bar{y}$ holds by definition of $\rhd$.
   (c) M is $P\,Q$. By induction hypothesis the result holds.

   (d) If M is $P\,q$ we have $q[\bar{x} := N] \equiv q$. Local terms cannot contain global variables so the results holds by induction hypothesis on P.
   (e) All the other cases are similar to the two previous cases: we conclude by induction hypothesis and by the vacuity of global variable substitutions on local terms.
2. $M \rhd M'$ is $\lambda y.P \rhd \lambda y.P'$ with $P \rhd P'$. By induction hypothesis $P[\bar{x} := N] \rhd P'[\bar{x} := N']$. By rule (III) we obtain $\lambda y.P[\bar{x} := N] \rhd \lambda y.P'[\bar{x} := N']$.
3. $M \rhd M'$ is $P\,Q \rhd P'\,Q'$ with $P \rhd P'$ and $Q \rhd Q'$. Then
$$\begin{aligned} M[\bar{x} := N] &\equiv P[\bar{x} := N]Q[\bar{x} := N] \\ &\rhd P'[\bar{x} := N']Q'[\bar{x} := N'], \text{ by induction hypothesis and } (IV) \\ &\equiv M'[\bar{x} := N'] \end{aligned}$$
4. $M \rhd M'$ is the conclusion of one of the other context rules (V), (II), (VI), (VII), $(VIII)$. The proof is similar to the previous one but the argument about global variable substitution on local terms may be used for (II), $(VIII)$.
5. $M \rhd M'$ is $(\lambda\bar{y}.P)Q \rhd P'[\bar{y} := Q']$ with $P \rhd P'$ and $Q \rhd Q'$. Then
$$\begin{aligned} M[\bar{x} := N] &\equiv (\lambda\bar{y}.P[\bar{x} := N])Q[\bar{x} := N] \\ &\rhd P'[\bar{x} := N'][\bar{y} := Q'[\bar{x} := N']], \text{ by induction hypothesis and } (IX) \\ &\equiv P'[\bar{y} := Q'][\bar{x} := N'], \qquad \text{by substitution lemma 5} \\ &\equiv M'[\bar{x} := N'] \end{aligned}$$
6. $M \rhd M'$ is $(\lambda\dot{y}.P)q \rhd P'[\dot{y} := q']$ with $P \rhd P'$ and $q \mathrel{\dot{\rhd}} q'$. Then
$$\begin{aligned} M[\bar{x} := N] &\equiv (\lambda\bar{y}.P[\bar{x} := N])q \\ &\rhd P'[\bar{x} := N'][\dot{y} := q'], \text{ by induction hypothesis and } (X) \\ &\equiv P'[\dot{y} := q'][\bar{x} := N'], \text{ by substitution lemma 5} \\ &\equiv M'[\bar{x} := N'] \end{aligned}$$
7. $M \rhd M'$ is $\pi\,m\,?\,\pi\,n \rhd \pi(\lambda\dot{x}.m'(n'\,\dot{x}))$ with $m \mathrel{\dot{\rhd}} m'$ and $n \mathrel{\dot{\rhd}} n'$. $\pi\,m\,?\,\pi\,n[\bar{x} := N] \equiv \pi\,m\,?\,\pi\,n$ and $\pi(\lambda\dot{x}.m'(n'\,\dot{x}))[\bar{x} := N'] \equiv \pi(\lambda\dot{x}.m'(n'\,\dot{x}))$, so $M[\bar{x} := N] \equiv M'[\bar{x} := N']$. The proof when $M \rhd M'$ is the root of the rule (XII) is very similar to the previous case.
8. $M \rhd M'$ is $(\pi m \xrightarrow{m'} M_1, M_2) \rhd M_1'$ with $m\,m' \doteq T$, $m' \in \mathcal{N}$ and $M_1 \rhd M_1'$. we have $(\pi m \xrightarrow{m'} M_1, M_2)[\bar{x} := N] \equiv (\pi m \xrightarrow{m'} M_1[\bar{x} := N], M_2[\bar{x} := N])$. By $(XIII)$ and induction hypothesis, $(\pi m \xrightarrow{m'} M_1[\bar{x} := N], M_2[\bar{x} := N]) \equiv M_1'[\bar{x} := N']$. The other case of conditional is similar to this one.

Lemma 7. *Let be $M \in \bar{\Lambda}$ and $n \in \dot{\Lambda}$ such that $M \rhd M'$ and $n \mathrel{\dot{\rhd}} n'$. Then*
$$M[\dot{x} := n] \rhd M'[\dot{x} := n']$$

Proof. Very similar to the proof of lemma 6. We will show the differences:

1. $M \rhd M'$ is is $M \rhd M$. So we have to show $M[\dot{x} := n] \rhd M[\dot{x} := n']$. We prove it by induction on the structure of M.
   (a) M is $\bar{y}$. $\bar{y} \rhd \bar{y}$ holds by definition of $\rhd$ (rule (I)).
   (b) M is $P\,Q$. By induction hypothesis the result holds.
   (c) M is $P\,q$. By induction hypothesis, $P[\dot{x} := n] \rhd P'[\dot{x} := n']$. By lemma 2, $q[\dot{x} := n] \mathrel{\dot{\rhd}} q'[\dot{x} := n']$. Then by rule (V), $P[\dot{x} := n]q[\dot{x} := n] \rhd P'[\dot{x} := n']q'[\dot{x} := n']$.
   (d) All the other cases are similar to the two previous cases: we conclude by induction hypothesis and by lemma 2 on local terms.

We can now prove the strong confluence of $\triangleright$.

Lemma 8. *$\forall M, M_1, M_2 \in \bar{\Lambda}$ such that $M \triangleright M_1$ and $M \triangleright M_2$ there exists $M_3 \in \bar{\Lambda}$ such that $M_1 \triangleright M_3$ and $M_2 \triangleright M_3$:*

$$\begin{tikzcd}[column sep=small] & M \arrow[dl] \arrow[dr] & \\ M_1 \arrow[dr] & & M_2 \arrow[dl] \\ & M_3 & \end{tikzcd}$$

Proof. By induction on the structure of the derivation $M \triangleright M_1$. The proof is done by enumeration of the rule applied at the root of the derivation $M \triangleright M_1$

1. $M \triangleright M_1$ is $(I) M \triangleright M$. Then $M_3 \equiv M_2$. For the following cases, we will not further mention the subcase $M \triangleright M_2$ is $(I) M \triangleright M$ which is trivially resolved.
2. $M \triangleright M_1$ is $(II)\dfrac{m \dot{\triangleright} m_1}{\pi\, m \triangleright \pi\, m_1}$. $M \triangleright M_2$ is (I) or $(II)\dfrac{m \dot{\triangleright} m_2}{\pi\, m \triangleright \pi\, m_2}$. By strong confluence of $\dot{\triangleright}$ and by rule (II), there exists $m_3 \in \dot{\Lambda}$ such that $\pi\, m_1 \triangleright \pi\, m_3$ and $\pi\, m_2 \triangleright \pi\, m_3$. So $M_3 \equiv \pi\, m_3$.
3. $M \triangleright M_1$ is $(III)\dfrac{P \triangleright P'}{\lambda x.P \triangleright \lambda x.P'}$. $M \triangleright M_2$ is (I) $M \triangleright M$ or (III) $\dfrac{P \triangleright P''}{\lambda x.P \triangleright \lambda x.P''}$. By induction hypothesis, there exists $P''' \in \bar{\Lambda}$ such that $P' \triangleright P'''$ and $P'' \triangleright P'''$. By rule (III) we can conclude that $\lambda x.P' \triangleright \lambda x.P'''$ and $\lambda x.P'' \triangleright \lambda x.P'''$. So $M_3 \equiv \lambda x.P'''$.
4. $M \triangleright M_1$ is $(IV)\dfrac{N \triangleright N' \quad R \triangleright R'}{R\,N \triangleright R'\,N'}$
   There are three possible subcases :
   (a) $M \triangleright M_2$ is (I)
   (b) $M \triangleright M_2$ is $(IV)\dfrac{N \triangleright N'' \quad R \triangleright R''}{R\,N \triangleright R''\,N''}$ By induction hypothesis and by rule (IV), there exists R''' and N''' such that $R'\,N' \triangleright R'''\,N'''$ and $R''\,N'' \triangleright R'''\,N'''$. So $M_3 \equiv R'''\,N'''$
   (c) $M \triangleright M_2$ is $(IX)\dfrac{P \triangleright P'' \quad Q \triangleright Q''}{(\lambda\bar{x}.P)Q \triangleright P''[\bar{x} := Q'']}$ We have necessarily $R \equiv (\lambda\bar{x}.P)$, $N \equiv Q$, $R' \equiv (\lambda\bar{x}.P')$ and $N' \equiv Q'$. By induction hypothesis, there exists P''' and Q''' such that $P' \triangleright P'''$, $Q' \triangleright Q'''$, $P'' \triangleright P'''$ and $Q'' \triangleright Q'''$. By rule (IX), we have:
   $$(\lambda\bar{x}.P')Q' \triangleright P'''[\bar{x} := Q''']$$
   and by lemma 6 we have:
   $$P''[\bar{x} := Q''] \triangleright P'''[\bar{x} := Q''']$$
   So $M_3 \equiv P'''[\bar{x} := Q''']$.
5. $M \triangleright M_1$ is (V) $\dfrac{R \triangleright R' \quad n \dot{\triangleright} n'}{R\,n \triangleright R'\,n'}$
   There are three possible subcases:
   (a) $M \triangleright M_2$ is (I)
   (b) $M \triangleright M_2$ is (V) $\dfrac{R \triangleright R'' \quad n \dot{\triangleright} n''}{R\,n \triangleright R''\,n''}$ By induction hypothesis there exists R''' such that $R' \triangleright R'''$ and $R'' \triangleright R'''$. By strong confluence of $\dot{\triangleright}$, there exists n''' such that $n' \dot{\triangleright} n'''$ and $n'' \dot{\triangleright} n'''$. By rule (V), we have $R'\,n' \triangleright R'''\,n'''$ and $R''\,n'' \triangleright R'''\,n'''$. So $M_3 \equiv R'''\,n'''$.
   (c) $M \triangleright M_2$ is (X) $\dfrac{P \triangleright P'' \quad q \dot{\triangleright} q''}{(\lambda\dot{x}.P)q \triangleright P''[\dot{x} := q'']}$ We have necessarily $R \equiv (\lambda\dot{x}.P)$, $n \equiv q$, $R' \equiv (\lambda\dot{x}.P')$ and $n' \equiv q'$. By induction hypothesis there exists P''' such that $P' \triangleright P'''$ and $P'' \triangleright P'''$. By strong confluence of $\dot{\triangleright}$, there exists q''' such that $q' \dot{\triangleright} q'''$ and $q'' \dot{\triangleright} q'''$. By rule (X), we have:
   $$(\lambda\dot{x}.P')q' \triangleright P'''[\dot{x} := q''']$$

and by the lemma 7 we have:

$$P''[\dot{x} := q''] \ \rhd \ P'''[\dot{x} := q''']$$

So $M_3 \equiv P'''[\dot{x} := q''']$.

6. $M \ \rhd \ M_1$ is $(VI)\ \dfrac{P \ \rhd \ P' \quad Q \ \rhd \ Q'}{(P \# Q) \ \rhd \ (P' \# Q')}$

   There are three possible sub-cases:

   (a) $M \ \rhd \ M_2$ is (I)

   (b) $M \ \rhd \ M_2$ is $(VI)\dfrac{P \ \rhd \ P'' \quad Q \ \rhd \ Q''}{(P \# Q) \ \rhd \ (P'' \# Q'')}$ By induction hypothesis there exists P''' such that $P' \ \rhd \ P'''$ and $P'' \ \rhd \ P'''$ and Q''' such that $Q' \ \rhd \ Q'''$ and $Q'' \ \rhd \ Q'''$. By rule (VI), we have $(P' \# Q') \ \rhd \ (P''' \# Q''')$ and $(P'' \# Q'') \ \rhd \ (P''' \# Q''')$ So $M_3 \equiv (P''' \# Q''')$

   (c) $M \ \rhd \ M_2$ is $(XII)\dfrac{m_1 \dot{\rhd}\, m_1'' \quad m_2 \dot{\rhd}\, m_2''}{\pi\, m_1 \# \pi\, m_2 \ \rhd \ \pi(\lambda\dot{x}.(m_1''\, \dot{x})(m_2''\, \dot{x}))}$ with $P \equiv \pi m_1$, $P' \equiv \pi m_1'$, $Q \equiv \pi m_2$, $Q' \equiv \pi m_2'$,

   By strong confluence of $\dot{\rhd}$ there exists m_1''', m_2''' such that, $m_1' \ \dot{\rhd} \ m_1'''$, $m_1'' \ \dot{\rhd} \ m_1'''$, $m_2' \ \dot{\rhd} \ m_2'''$, $m_2'' \dot{\rhd}\, m_2'''$. By local context rules (iii) and (ii) and by context rule (II),

$$\pi(\lambda\dot{x}.(m_1''\, \dot{x})(m_2''\, \dot{x})) \ \rhd \ \pi(\lambda\dot{x}.(m_1'''\, \dot{x})(m_2'''\, \dot{x}))$$

   and by rule (XII),

$$(\pi\, m_1' \# \pi\, m_2') \ \rhd \ \pi(\lambda\dot{x}.(m_1'''\, \dot{x})(m_2'''\, \dot{x}))$$

   So, $M_3 \equiv \pi(\lambda\dot{x}.(m_1'''\, \dot{x})(m_2'''\, \dot{x}))$.

7. $M \ \rhd \ M_1$ is $(VII)\ \dfrac{P \ \rhd \ P' \quad Q \ \rhd \ Q'}{(P\, ?\, Q) \ \rhd \ (P'\, ?\, Q')}$

   There are three possible subcases:

   (a) $M \ \rhd \ M_2$ is (I)

   (b) $M \ \rhd \ M_2$ is $(VII)\dfrac{P \ \rhd \ P'' \quad Q \ \rhd \ Q''}{(P\, ?\, Q) \ \rhd \ (P''\, ?\, Q'')}$ By induction hypothesis there exists P''' such that $P' \ \rhd \ P'''$ and $P'' \ \rhd \ P'''$ and Q''' such that $Q' \ \rhd \ Q'''$ and $Q'' \ \rhd \ Q'''$. By rule (VII), we have $(P'\, ?\, Q') \ \rhd \ (P'''\, ?\, Q''')$ and $(P''\, ?\, Q'') \ \rhd \ (P'''\, ?\, Q''')$ So $M_3 \equiv (P'''\, ?\, Q''')$

   (c) $M \ \rhd \ M_2$ is $(XI)\dfrac{m_1 \dot{\rhd}\, m_1'' \quad m_2 \dot{\rhd}\, m_2'' \quad \forall i \in \mathcal{N}.\exists i' \in \mathcal{N}.m_2\, i \doteq i'}{\pi\, m_1\, ?\, \pi\, m_2 \ \rhd \ \pi(\lambda\dot{x}.(m_1''\, (m_2''\, \dot{x})))}$ with $P \equiv \pi m_1$, $P' \equiv \pi m_1'$, $Q \equiv \pi m_2$, $Q' \equiv \pi m_2'$,

   By strong confluence of $\dot{\rhd}$ there exists m_1''', m_2''' such that, $m_1' \ \dot{\rhd} \ m_1'''$, $m_1'' \ \dot{\rhd} \ m_1'''$, $m_2' \ \dot{\rhd} \ m_2'''$, $m_2'' \dot{\rhd}\, m_2'''$. By local context rules (iii) and (ii) and by context rule (II),

$$\pi(\lambda\dot{x}.(m_1''\, (m_2''\, \dot{x}))) \ \rhd \ \pi(\lambda\dot{x}.(m_1'''(m_2'''\, \dot{x})))$$

   For all $i \in \mathcal{N}$, there exists $i' \in \mathcal{N}$ such that $m_2 i \doteq i'$. But we have $m_2 \doteq m_2'$. By context rule (iii), $m_2\, i \doteq m_2'\, i$, so $m_2'\, i \doteq i'$ and by rule (XI),

$$(\pi\, m_1'\, ?\, \pi\, m_2') \ \rhd \ \pi(\lambda\dot{x}.(m_1'''(m_2'''\, \dot{x})))$$

   So, $M_3 \equiv \pi(\lambda\dot{x}.(m_1'''(m_2'''\, \dot{x})))$.

8. $M \ \rhd \ M_1$ is $(VIII)\dfrac{m \dot{\rhd}\, m' \quad n \doteq n' \quad N_1 \ \rhd \ N_1' \quad N_2 \ \rhd \ N_2'}{(\pi\, m \xrightarrow{n} N_1, N_2) \ \rhd \ (\pi\, m' \xrightarrow{n'} N_1', N_2')}$

   There are four possible sub-cases:

   (a) $M \ \rhd \ M_2$ is $(VIII)\ \dfrac{m \dot{\rhd}\, m'' \quad n \dot{\rhd}\, n'' \quad N_1 \ \rhd \ N_1'' \quad N_2 \ \rhd \ N_2''}{(\pi\, m \xrightarrow{n} N_1, N_2) \ \rhd \ (\pi\, m'' \xrightarrow{n''} N_1'', N_2'')}$

By induction hypothesis, there exists N_1''' and N_2''' such that $N_1' \ \triangleright\ N_1'''$, $N_1'' \ \triangleright\ N_1'''$, $N_2' \ \triangleright\ N_2'''$, $N_2'' \ \triangleright\ N_2'''$. By strong confluence of $\dot{\triangleright}$ there exists n''' and m''' such that $n' \ \dot{\triangleright}\ n'''$, $n'' \ \dot{\triangleright}\ n'''$, $m' \ \dot{\triangleright}\ m'''$, $m'' \ \dot{\triangleright}\ m'''$. By rule $(VIII)$,

$$(\pi\, m' \xrightarrow{n'} N_1', N_2') \ \triangleright\ (\pi\, m''' \xrightarrow{n'''} N_1''', N_2''')$$

and

$$(\pi\, m'' \xrightarrow{n''} N_1'', N_2'') \ \triangleright\ (\pi\, m''' \xrightarrow{n'''} N_1''', N_2''')$$

So, $M_3 \equiv (\pi\, m''' \xrightarrow{n'''} N_1''', N_2''')$

(b) $M \ \triangleright\ M_2$ is $(XIII)\dfrac{N_1 \ \triangleright\ N_1'' \quad m\, n \doteq T \quad n \in \mathcal{N}}{(\pi\, m \xrightarrow{n} N_1, N_2) \ \triangleright\ N_1''}$

By induction hypothesis, there exists N_1''' such that $N_1' \ \triangleright\ N_1'''$ and $N_1'' \ \triangleright\ N_1'''$. We have $\dot{\triangleright} \subseteq \dot{\rightarrow}^*$ and $\dot{\rightarrow}^*$ is confluent. So $m \doteq m'$, $n \doteq n'$ and finally $m'n' \doteq T$. n is a closed normal form belonging to $\mathcal{N}$, so n' too. By rule $(XIII)$,

$$(\pi\, m' \xrightarrow{n'} N_1', N_2') \ \triangleright\ N_1'''$$

So $M_3 \equiv N_1'''$.

(c) $M \ \triangleright\ M_2$ is $(XIV)\dfrac{N_2 \ \triangleright\ N_2'' \quad m\, n \doteq F}{(\pi\, m \xrightarrow{n} N_1, N_2) \ \triangleright\ N_2''}$ Similar to the previous case: $M_3 \equiv N_2'''$.

9. $M \ \triangleright\ M_1$ is $(IX)\dfrac{P \ \triangleright\ P' \quad Q \ \triangleright\ Q'}{(\lambda\bar{x}.P)Q \ \triangleright\ P'[\bar{x} := Q']}$

$M \ \triangleright\ M_2$ can be either (I), (IV) or (IX). The first case is trivial, the second case correspond to the case 4c.

If $M \ \triangleright\ M_2$ is $(IX)\dfrac{P \ \triangleright\ P'' \quad Q \ \triangleright\ Q''}{(\lambda\bar{x}.P)Q \ \triangleright\ P''[\bar{x} := Q'']}$ By induction hypothesis, there exist P''' and Q''' such that $P' \ \triangleright\ P'''$, $P'' \ \triangleright\ P'''$, $Q' \ \triangleright\ Q'''$, $Q'' \ \triangleright\ Q'''$. By lemma 6,

$$P'[\bar{x} := Q'] \ \triangleright\ P'''[\bar{x} := Q''']$$

and

$$P''[\bar{x} := Q''] \ \triangleright\ P'''[\bar{x} := Q''']$$

So $M_3 \equiv P'''[\bar{x} := Q''']$

10. $M \ \triangleright\ M_1$ is $\dfrac{P \ \triangleright\ P' \quad q \ \dot{\triangleright}\ q'}{(\lambda\dot{x}.P)q \ \triangleright\ P'[\dot{x} := q']}$

$M \ \triangleright\ M_2$ can be either (I), (V) or (X). The first case is trivial, the second case correspond to the case 5c.

If $M \ \triangleright\ M_2$ is $(X)\dfrac{P \ \triangleright\ P'' \quad q \ \dot{\triangleright}\ q''}{(\lambda\dot{x}.P)Q \ \triangleright\ P''[\dot{x} := q'']}$ By induction hypothesis, there exists P''' such that $P' \ \triangleright\ P'''$, $P'' \ \triangleright\ P'''$. By strong confluence of $\dot{\triangleright}$ there exists q''' such that $q' \ \dot{\triangleright}\ q'''$, $q' \ \dot{\triangleright}\ q''$. By lemma 7,

$$P'[\dot{x} := q'] \ \triangleright\ P'''[\dot{x} := q''']$$

and

$$P''[\dot{x} := q''] \ \triangleright\ P'''[\dot{x} := q''']$$

So $M_3 \equiv P'''[\dot{x} := q''']$

11. $M \ \triangleright\ M_1$ is $(XI)\dfrac{m_1 \ \dot{\triangleright}\ m_1' \quad m_2 \ \dot{\triangleright}\ m_2' \quad \forall i \in \mathcal{N}.\exists i' \in \mathcal{N}.m_2\, i \doteq i'}{\pi\, m_1\, ?\, \pi\, m_2 \ \triangleright\ \pi(\lambda\dot{x}.m_1'\,(m_2'\,\dot{x}))}$. $M \ \triangleright\ M_2$ can be either (I), (VII) or (XI). The first case is trivial, the second case correspond to the case 7c.

$M \ \triangleright\ M_2$ is $(XI)\dfrac{m_1 \ \dot{\triangleright}\ m_1'' \quad m_2 \ \dot{\triangleright}\ m_2'' \quad \forall i \in \mathcal{N}.\exists i' \in \mathcal{N}.m_2\, i \doteq i'}{\pi\, m_1\, ?\, \pi\, m_2 \ \triangleright\ \pi(\lambda\dot{x}.m_1''\,(m_2''\,\dot{x}))}$.

By strong confluence of $\dot{\rhd}$ there exists m_1''', m_2''' such that, $m_1' \;\dot{\rhd}\; m_1'''$, $m_1'' \;\dot{\rhd}\; m_1'''$, $m_2' \;\dot{\rhd}\; m_2'''$, $m_2'' \;\dot{\rhd}\; m_2'''$. By local context rules (iii) and (ii) and by context rule (II),

$$\pi(\lambda\dot{x}.m_1'\,(m_2'\,\dot{x})) \;\rhd\; \pi(\lambda\dot{x}.m_1'''(m_2'''\,\dot{x}))$$

$$\pi(\lambda\dot{x}.m_1''\,(m_2''\,\dot{x})) \;\rhd\; \pi(\lambda\dot{x}.m_1'''(m_2'''\,\dot{x}))$$

So, $M_3 \equiv \pi(\lambda\dot{x}.m_1'''(m_2'''\,\dot{x}))$.

12. $M \;\rhd\; M_1$ is $(XII)\dfrac{m_1 \;\dot{\rhd}\; m_1' \quad m_2 \;\dot{\rhd}\; m_2'}{\pi\, m_1 \# \pi\, m_2 \;\rhd\; \pi(\lambda\dot{x}.(m_1'\,\dot{x})(m_2'\,\dot{x}))}$. $M \;\rhd\; M_2$ can be either (I), (VI) or (XII). The first case is trivial, the second case correspond to the case 6c.
    $M \;\rhd\; M_2$ is $(XII)\dfrac{m_1 \;\dot{\rhd}\; m_1'' \quad m_2 \;\dot{\rhd}\; m_2''}{\pi\, m_1 \# \pi\, m_2 \;\rhd\; \pi(\lambda\dot{x}.(m_1''\,\dot{x})(m_2''\,\dot{x}))}$.
    By strong confluence of $\dot{\rhd}$ there exists m_1''', m_2''' such that, $m_1' \;\dot{\rhd}\; m_1'''$, $m_1'' \;\dot{\rhd}\; m_1'''$, $m_2' \;\dot{\rhd}\; m_2'''$, $m_2'' \;\dot{\rhd}\; m_2'''$. By local context rules (iii) and (ii) and by context rule (II),

$$\pi(\lambda\dot{x}.(m_1'\,\dot{x})(m_2'\,\dot{x})) \;\rhd\; \pi(\lambda\dot{x}.(m_1'''\,\dot{x})(m_2'''\,\dot{x}))$$

$$\pi(\lambda\dot{x}.(m_1''\,\dot{x})(m_2''\,\dot{x})) \;\rhd\; \pi(\lambda\dot{x}.(m_1'''\,\dot{x})(m_2'''\,\dot{x}))$$

    So, $M_3 \equiv \pi(\lambda\dot{x}.(m_1'''\,\dot{x})(m_2'''\,\dot{x}))$.
13. $M \;\rhd\; M_1$ is $(XIII)\dfrac{N_1 \;\rhd\; N_1' \quad m\,n \doteq T \quad n \in \mathcal{N}}{(\pi\, m \xrightarrow{n} N_1, N_2) \;\rhd\; N_1'}$
    $M \;\rhd\; M_2$ can be either (I), $(VIII)$ or $(XIII)$. (XIV) is impossible because $T \neq F$. The first case is trivial, the second case correspond to the case 8b.
    $M \;\rhd\; M_2$ is $(XIII)\dfrac{N_1 \;\rhd\; N_1'' \quad m\,n \doteq T \quad n \in \mathcal{N}}{(\pi\, m \xrightarrow{n} N_1, N_2) \;\rhd\; N_1''}$ By induction hypothesis, there exists N_1''' such that $N_1' \;\rhd\; N_1'''$, $N_1'' \;\rhd\; N_1'''$. $M_3 \equiv N_1'''$
14. $M \;\rhd\; M_1$ is $(XIV)\dfrac{N_2 \;\rhd\; N_2' \quad m\,n \doteq F \quad n \in \mathcal{N}}{(\pi\, m \xrightarrow{n} N_1, N_2) \;\rhd\; N_2'}$ Similar to the previous case.

BSλ is confluent

Lemma 9.

$$\rightarrow \subseteq \;\rhd\; \subseteq \rightarrow^*$$

Proof. – $\rightarrow \subseteq \;\rhd\;$: by construction.
– $\rhd\; \subseteq \rightarrow^*$: by induction on the definition of $\;\rhd\;$.

Theorem 10. *$\rightarrow$ is confluent*

Proof. Let be M such that $M \;\rightarrow^*\; M_1$ and $M \;\rightarrow^*\; M_2$. $\rightarrow^*$ is the transitive closure of $\;\rhd\;$ (lemma 9), and $\;\rhd\;$ is strongly confluent (lemma 8) therefore $\rightarrow^*$ is strongly confluent. So there exists M_3 such that $M_1 \;\rightarrow^*\; M_3$ and $M_2 \;\rightarrow^*\; M_3$

This property makes BSλ a suitable basis for functional BSP languages where the data-processor mapping and communications are explicit (the so-called *direct mode* of BSP programming). It also ensures the correctness (modulo normalization) of program-equivalence proofs built from BSλ equations.

References

[Bar86] H.P. Barendregt. *The Lambda Calculus, Its Syntax and Semantics.* North-Holland, 1986.

[BFMR96] L. Bougé, P. Fraigniaud, A. Mignotte, and Y. Robert, editors. *Euro-Par'96 Parallel Processing*, number 1123–1124 in Lecture Notes in Computer Science, Lyon, August 1996. LIP-ENSL, Springer.

[Ble90] G.E. Blelloch. *Vector Models for Data-Parallel Computing.* MIT Press, 1990.

[CCL91] M. Chen, Y.-Il Choo, and J. Li. Crystal: Theory and pragmatics of generating efficient parallel code. In B. K. Szymanski, editor, *Parallel Functional Languages and Compilers.* ACM Press, 1991.

[Col89] M. Cole. *Algorithmic Skeletons: Structured Management of Parallel Computation.* MIT Press, 1989.

[LHF99] F. Loulergue, G. Hains, and C. Foisy. A Calculus of functional BSP programs. *Science of Computer Programming*, 1999. To appear.

[Lis96] B. Lisper. Data parallelism and functional programming. In *École de Printemps sur le Data-Parallélisme*, Les Menuires, France, 1996. PRC-CNRS, Springer.

[Lou99] F. Loulergue. Extension du BSλ-calcul. In P. Weis, editor, *JFLA'99 : Journées Francophones des Langages Applicatifs*, Collection Didactique de l'INRIA, pages 93–112, Morzine-Avoriaz, February 1999.

[McC96] W.F. McColl. Universal computing. In Bougé et al. [BFMR96].

[SH86] G.H. Steele and W.D. Hillis. Connection Machine Lisp: A Fine Grained Parallel Symbolic Processing Language. In *ACM Conf. on Lisp and Funct. Progr.*, pages 276–297, Cambridge MA, 1986.

[SHM97] David Skillicorn, Jonathan M. D. Hill, and W. F. McColl. Questions and answers about bsp. *Scientific Programming*, 6(3):249–274, Fall 1997.

[Val90] Leslie G Valiant. A bridging model for parallel computation. *Communications of the ACM*, 33(8):103, August 1990.

[VP92] E. Violard and G.-R. Perrin. PEI: A language and its refinement calculus for parallel programming. *Parallel Computing*, 18(10):1167–1184, October 1992.