On Unique Decomposition of Processes in the Applied π-Calculus

Jannik Dreier, Cristian Ene, Pascal Lafourcade, Yassine Lakhnech

Université Grenoble 1, CNRS, VERIMAG
firstname.lastname@imag.fr

Foundations of Software Science and Computation Structures (FoSSaCS) 2013, Rome

March 18, 2013

(Unique) Parallel Decomposition in Process Algebras

- Suppose we have a process P.

(Unique) Parallel Decomposition in Process Algebras

- Suppose we have a process P.
- Are there processes $P_1, \ldots, P_n$ such that

$$P = P_1 | \ldots | P_n$$

where $P_1, \ldots, P_n$ are "prime", i.e. cannot be decomposed into nontrivial processes?

(Unique) Parallel Decomposition in Process Algebras

- Suppose we have a process P.
- Are there processes $P_1, \ldots, P_n$ such that

$$P = P_1 | \ldots | P_n$$

  where $P_1, \ldots, P_n$ are "prime", i.e. cannot be decomposed into nontrivial processes?
- Is this decomposition unique?

Applications

- Provides a normal form
- Gives a cancellation result, i.e.

$$P|Q = P|R \Rightarrow Q = R$$

- Provides a maximally parallelized version of a given program
- Can be used to verify the equivalence of two processes [GM92]

Previous Results

Unique decomposition results exist

- for the Calculus of Communicating Systems (CCS) [Mil89] by Moller and Milner [MM93, Mol89]:
  - finite processes w. interleaving or parallel composition w.r.t. strong bisimilarity
  - finite processes w. parallel composition w.r.t. weak bisimilarity
- for Basic Parallel Processes (BPP) [Chr93]:
  - normed processes w. interleaving or parallel composition w.r.t. strong bisimilarity
- for ordered monoids by Luttik and van Oostrom:
  - if the calculus satisfies certain properties, the result for strong bisimilarity follows directly [LvO05]
  - can be extended to weak bisimilarity [Lut12]

The Applied π-Calculus [AF01]

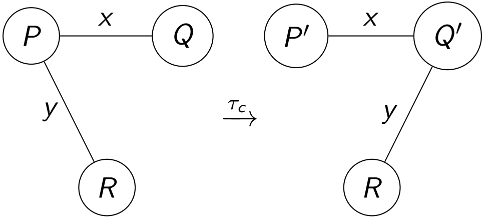
- an "impure" variant of the π-Calculus
- designed for the verification of cryptographic protocols
- features an *equational theory* to model cryptographic primitives:

$$dec(enc(m, k), k) = m$$

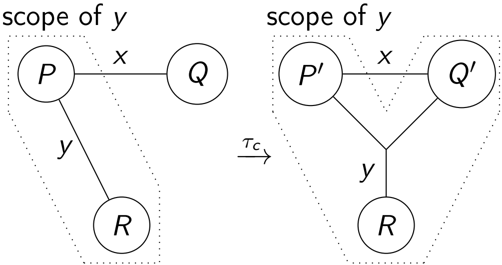
- and *active substitutions* $\{^M/_x\}$, a non-zero element that exhibits no transitions
- allows *channel* or *link passing* (sometimes also called *mobility*) and *scope extrusion*

Channel/Link passing

Consider three parallel processes P, Q and R. P and Q synchronize using an internal reduction τ_c:
$P|Q|R \xrightarrow{\tau_c} P'|Q'|R$



Scope extrusion



Plan

1. Introduction
2. The Applied π-Calculus
3. Results
   - Strong Bisimilarity
   - Weak Bisimilarity
4. Conclusion

Plan

Syntax

Plain processes:

$P, Q :=$	plain processes
0	null process
$P \mid Q$	parallel composition
$!P$	replication
$\nu n.P$	name restriction ("new")
$\texttt{if } M = N \texttt{ then } P \texttt{ else } Q$	conditional (M, N terms)
$\texttt{in}(u, x).P$	message input
$\texttt{out}(u, M).P$	message output

Syntax Cont'd

Active/extended processes:

$A, B, P, Q :=$	active processes
P	plain process
$A \vert B$	parallel composition
$\nu n.A$	name restriction
$\nu x.A$	variable restriction
$\{^M/_x\}$	active substitution

Plan

Plan

Strong Labeled Bisimilarity

Definition (Strong Labeled Bisimilarity ($\sim_l$))

Strong labeled bisimilarity is the largest symmetric relation $\mathcal{R}$ on closed active processes, such that $A \mathcal{R} B$ implies:

1. $A \approx_s B$,
2. if $A \rightarrow A'$, then $B \rightarrow B'$ and $A' \mathcal{R} B'$ for some B',
3. if $A \xrightarrow{\alpha} A'$ and $fv(\alpha) \subseteq \text{dom}(A)$ and $bn(\alpha) \cap fn(B) = \emptyset$, then $B \xrightarrow{\alpha} B'$ and $A' \mathcal{R} B'$ for some B'.

Strongly Parallel Prime

Definition (Strongly Parallel Prime)

A closed process P is *strongly parallel prime*, if

- $P \not\sim_l 0$ and
- for any two closed processes Q and R such that $P \sim_l Q|R$, we have $Q \sim_l 0$ or $R \sim_l 0$.

Example 1

Example

Consider the following process:

$$P_{ex} \quad = \quad \nu k.\nu l.\nu m.\nu d.\left(\{^{l}/_{y}\}\,|\mathtt{out}(c, enc(n,k))|\right.$$
$$\left.\mathtt{out}(d,m)|\mathtt{in}(d,x).\mathtt{out}(c,x)\right)$$

We can decompose P_{ex} as follows:

$$P_{ex} \quad \sim_l \quad (\nu l.\,\{^{l}/_{y}\})|(\nu k.\mathtt{out}(c, enc(n,k)))|$$
$$(\nu d.(\nu m.\mathtt{out}(d,m)|\mathtt{in}(d,x).\mathtt{out}(c,x)))$$

Example 2

Example

Consider $!P$ for a process $P \not\sim_l 0$.

Example 2

Example

Consider $!P$ for a process $P \not\sim_l 0$.
By definition $!P = P|!P$, hence $!P$ is not prime.

Example 2

Example

Consider $!P$ for a process $P \not\sim_l 0$.
By definition $!P = P|!P$, hence $!P$ is not prime.
There is no decomposition into prime factors.

Existence of Factorization

Theorem (Existence of Factorization)

Any closed normed process P can be expressed as the parallel product of strong parallel primes, i.e.

$$P \sim_l P_1 | \ldots | P_n$$

where for all $1 \leq i \leq n$ P_i is strongly parallel prime.

Proof by induction on the norm and the size of the domain.

Uniqueness of Factorization

Theorem (Uniqueness of Factorization)

The strong parallel factorization of a closed normed process P is unique (up to $\sim_l$).

Uniqueness of Factorization

Theorem (Uniqueness of Factorization)

The strong parallel factorization of a closed normed process P is unique (up to $\sim_l$).

Proof idea:

- *Proof by induction* on the norm of P, and inside each case by induction on the size of the domain
- Each prime factor can either perform a *transition*, or has a *non-empty domain*
- A transition might not always be *norm-reducing* since processes can be infinite, but there is always a norm-reducing one
- Suppose the existence of two different factorizations, and show that this leads to a *contradiction*

Uniqueness of Factorization, Proof Cont'd

Four cases: A process with

- no transition and empty domain: unique factorization 0.
- no transition but non-empty domain: apply a *restriction* on part of the domain to hide all factors but one. Exploit the *induction hypothesis*
- empty domain, but transitions: execute a *transition* and apply the induction hypothesis.
  - *Problem:* an internal reduction can fuse factors using scope extrusion.
  - *Solution:* Whenever possible, choose a visible transition.
  - No visible transition $\Rightarrow$ processes cannot fuse using an internal reduction, since this would mean they synchronized on a public channel $\Rightarrow$ visible transitions exist.
- non-empty domain and transitions: combine the above two

Plan

Weak Labeled Bisimilarity

Definition (Weak Labeled Bisimilarity ($\approx_l$) [AF01])

(Weak) Labeled Bisimilarity is the largest symmetric relation $\mathcal{R}$ on closed active processes, such that $A \mathcal{R} B$ implies:

1. $A \approx_s B$,
2. if $A \rightarrow A'$, then $B \rightarrow^* B'$ and $A' \mathcal{R} B'$ for some B',
3. if $A \xrightarrow{\alpha} A'$ and $fv(\alpha) \subseteq \text{dom}(A)$ and $bn(\alpha) \cap fn(B) = \emptyset$, then $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $A' \mathcal{R} B'$ for some B'.

Weakly Parallel Prime

Definition (Weakly Parallel Prime)

A closed extended process P is *weakly parallel prime*, if

- $P \not\approx_l 0$ and
- for any two closed processes Q and R such that $P \approx_l Q|R$, we have $Q \approx_l 0$ or $R \approx_l 0$.

Example 3

Example

Consider

$$P = \nu a.(out(a,m)|(in(a,x).(!in(b,y)))|in(a,x))$$

Example 3

Example

Consider

$$P = \nu a.(out(a,m)|(in(a,x).(!in(b,y)))|in(a,x))$$

We have

$$P \to \nu a.(!in(b,y)|in(a,x)) \approx_l !in(b,y)$$

and

$$P \to \nu a.(in(a,x).(!in(b,y))) \approx_l 0.$$

Example 3

Example

Consider

$$P = \nu a.(out(a, m)|(in(a, x).(!in(b, y)))|in(a, x))$$

We have

$$P \to \nu a.(!in(b, y)|in(a, x)) \approx_l !in(b, y)$$

and

$$P \to \nu a.(in(a, x).(!in(b, y))) \approx_l 0.$$

Thus $P \approx_l P|P$, hence we have no unique decomposition.

Existence of Factorization

Theorem (Existence of Factorization)

Any closed finite active process P can be expressed as the parallel product of parallel primes, i.e.

$$P \approx_l P_1 | \ldots | P_n$$

where for all $1 \leq i \leq n$ P_i is weakly parallel prime.

Uniqueness of Factorization

Theorem (Uniqueness of Factorization)

The parallel factorization of a closed finite process P is unique (up to $\approx_l$*).*

Uniqueness of Factorization

Theorem (Uniqueness of Factorization)

The parallel factorization of a closed finite process P is unique (up to $\approx_l$).

Proof idea:

- Show the following statement: Any closed finite processes P and Q with $P \approx_l Q$ have the same factorization (up to $\approx_l$)
- *Induction* on the sum of the total depth of both factorizations, and in each case on the size of the domain
- Suppose the existence of two different factorizations and show this leads to a *contradiction*

Proof of Uniqueness of Factorization, Cont'd

- *Same structure* as the proof for strong bisimilarity
- Problem:
  - each transition can be simulated using *several* internal reductions
  - can affect several factors, and prime factors could *fuse* using scope extrusion
- Solution:
  - choose transitions that decrease the visible depth by *exactly one*
  - A synchronization of two factors uses *at least two* visible actions $\Rightarrow$ the resulting processes cannot be bisimilar any more

Plan

1. Introduction
2. The Applied π-Calculus
3. Results
   - Strong Bisimilarity
   - Weak Bisimilarity
4. Conclusion

Conclusion and future work

- Two unique decomposition results for subsets of the Applied π-Calculus:
  - closed finite processes w.r.t. weak labeled bisimilarity
  - closed normed processes w.r.t. strong labeled bisimilarity
- Future work:
  - Replication (Bang) "!":
    - First result by Hirschkoff and Pous [HP10] for a subset of CCS with top-level replication: *seed* Q of a process P of least size (in terms of prefixes) whose number of replicated components is maximal
    - Similar result for the Restriction-Free-π-Calculus (i.e. no "ν") – full calculus remains an open question
  - Find an (efficient) algorithm computing the unique decomposition of a process?

Thank you for your attention!

Questions?

jannik.dreier@imag.fr

Martín Abadi and Cédric Fournet.
Mobile values, new names, and secure communication.
In *Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, POPL '01, pages 104–115, New York, 2001. ACM.

Søren Christensen.
Decidability and Decompostion in Process Algebras.
PhD thesis, School of Computer Science, University of Edinburgh, 1993.

Jan Friso Groote and Faron Moller.
Verification of parallel systems via decomposition.
In *CONCUR '92: Proceedings of the Third International Conference on Concurrency Theory*, pages 62–76, London, UK, UK, 1992. Springer-Verlag.

Daniel Hirschkoff and Damien Pous.

On bisimilarity and substitution in presence of replication.
In *37th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 6199 of *LNCS*, pages 454–465. Springer, 2010.

Bas Luttik.
Unique parallel decomposition in branching and weak bisimulation semantics.
Technical report, 2012.
Available at http://arxiv.org/abs/1205.2117v1.

Bas Luttik and Vincent van Oostrom.
Decomposition orders – another generalisation of the fundamental theorem of arithmetic.
Theoretical Computer Science, 335(2-3):147–186, 2005.

Robin Milner.
Communication and Concurrency.

International Series in Computer Science. Prentice Hall, 1989.

Robin Milner and Faron Moller.
Unique decomposition of processes.
Theoretical Computer Science, 107(2):357–363, 1993.

Faron Moller.
Axioms for Concurrency.
PhD thesis, School of Computer Science, University of Edinburgh, 1989.

Summary of results

Type of Process	Strong Bisimilarity ($\sim_l$)	Weak Bisimilarity ($\approx_l$)
finite	Theorem 5	Theorem 10
normed	Theorem 5	Counterexample 9
general	Counterexample 4	Counterexample 4

Depth of processes

Definition (Total Depth)

Let $\text{length}_t : (\mathbf{Act} \cup \mathbf{Int})^* \mapsto \mathbb{N}$ be a function where $\text{length}_t(\epsilon) = 0$ and $\text{length}_t(\mu w) = 1 + \text{length}_t(w)$. The *total depth* $|P|_t \in (\mathbb{N} \cup \{\infty\})$ of a closed process P is defined as follows:

$$|P|_t = \sup \left\{ \text{length}_t(w) : P \xrightarrow{w} P' \not\rightarrow, w \in (\mathbf{Act} \cup \mathbf{Int})^* \right\}$$

Norm of a process

Definition (Norm of a Process)

Let $\mathsf{length}_n : (\mathbf{Act} \cup \mathbf{Int})^* \mapsto \mathbb{N}$ be a function where $\mathsf{length}_n(\epsilon) = 0$ and $\mathsf{length}_n(\mu w) = \begin{cases} 1 + \mathsf{length}_n(w) & \text{if } \mu \neq \tau_c \\ 2 + \mathsf{length}_n(w) & \text{if } \mu = \tau_c \end{cases}$
The norm $\mathcal{N}(P) \in (\mathbb{N} \cup \{\infty\})$ of a closed process P is defined as follows:

$$\mathcal{N}(P) = \inf \left\{ \mathsf{length}_n(w) : P \xrightarrow{w} P' \not\rightarrow, w \in (\mathbf{Act} \cup \mathbf{Int})^* \right\}$$

Some properties

- $P = Q|R$ implies $|P|_v = |Q|_v + |R|_v$
- $P = Q|R$ implies $|P|_t = |Q|_t + |R|_t$
- $P = Q|R$ implies $\mathcal{N}(P) = \mathcal{N}(Q) + \mathcal{N}(R)$
- $|P|_v \leq |P|_t$