Bounding Reactions in the π-calculus using Interpretations

Romain Péchoux

Université de Lorraine
Inria project Carte, Loria

FOPARA&WST,
August 29, 2013

Outline

1. Motivations
2. Process calculus
3. Interpretations
4. Resource upper bounds
5. Extensions
6. Conclusion

Implicit Computational Complexity

ICC aims at studying "programs" resource consumption.

On sequential models:

1. Function algebra (Bellantoni, Cook, Leivant ...)
2. Lambda-calculi:
   - Light logics (Baillot, Dal Lago, Gaboardi, Girard, Lafont, Ronchi Della Rocca, ...)
3. TRS:
   - Interpretations methods (Bonfante, Marion, Moyen, Péchoux, ...)
4. Imperative programs:
   - Matrices (Ben-Amram, Jones, Kristiansen, Moyen),
   - Tiered types (Hainry, Marion, Péchoux,...)

and what for concurrent models ?

Concurrent resources

Candidates for time:

- the reduction length
- the reduction length on a given set of channels
- ...

Candidates for space:

- the number of concurrent processes
- the size of sent values
- the number of channel creations
- ...

Candidates for complexity classes : Pspace, NP,...

- if all reduction length are polynomially bounded
- depending on the considered semantics

State of the art: time

If time = reduction length: a polynomial upper bound can be enforced

- using type systems
  - Demangeon, Hirschkoff and Sangiorgi
  - Deng and Sangiorgi
- using linear and light logics
  - Yoshida, Berger and Honda
  - Dal Lago, Martini and Sangiorgi
  - Madet and Amadio

State of the art: space

If space = the number of active processes: an upper bound can be obtained:

- using abstract interpretations and relational domains
  - Kobayashi et al
  - Féret
- using lattice ordered monoids
  - Konig

Other approaches (Hennessy, Pym et al) have tried to develop bisimilarity theory wrt a notion of resource process

Our approach

We make use of previous notions of interpretations on TRS to study processes.

- the pros:
  - a complete analysis wrt strongly normalizing processes
  - with a greater expressivity on both time and space
- the cons:
  - the analysis is undecidable and needs to be restricted to be automated
- Nice side effect:
  - It shows the portability of existing techniques on TRS

Methodology

A two steps static analysis:

$$\textit{Processes} \xrightarrow{[-]} (\mathcal{R}, \gtrsim) \xrightarrow{[\![-]\!]} (\mathbb{N}^{\textit{Names}\cup\{*\}}, \succeq)$$

1. Interpretation $[-]$
   - $\mathcal{R}$ is the set of resource processes, processes with no recursion and no replication abilities
   - $\gtrsim$ is a preorder on resource processes (e.g. simulation, bisimulation, ...)
2. A fixed semantics $[\![-]\!]$
   - * is a special symbol not in Names, for recursion
   - $\succeq$ is the standard order on functions of codomain $\mathbb{N}$

The considered process calculus

Standard π-calculus with inductive data:

$$
\begin{array}{lll}
e ::= & n \mid 0,1,2,\ldots \mid op(\overrightarrow{e}) & (\mathit{Expressions}) \\
v ::= & x \mid e & (\mathit{Values}) \\
P ::= & \mathbf{0} \mid x(y).P \mid \overline{x}v.P \mid P|P & (\mathit{Processes}) \\
 & \mid (\nu x)P \mid P+P \mid F(\overrightarrow{v}) &
\end{array}
$$

but with no replication : replaced by process calls and process definitions:

$$F(\overrightarrow{X}) = \texttt{Case}\ \overrightarrow{X}\ \texttt{of}\ \overrightarrow{v_1} \to P_1,\ \ldots,\ \overrightarrow{v_k} \to P_k$$

Why not considering replication ?

- There is a need of well-founded structures to apply TRS techniques.
- Replication is not suitable for resource control as:

$!P$ can be considered as equivalent to $\underbrace{P|\ldots|P}_{n}|!P,\ \forall n$

- process calls and definitions are standard :

recursion + pattern matching

- Replication $!P$ can be easily encoded in our fragment by:

$$F() = F()|P$$

Standard operational semantics

$$\frac{}{x(y).P \stackrel{xw}{\rightarrow} P\{w/y\}} \text{ (In)} \qquad \frac{}{\overline{x}w.P \stackrel{\overline{x}w}{\rightarrow} P} \text{ (Out)}$$

$$\frac{P \stackrel{\alpha}{\rightarrow} P' \quad bn(\alpha) \cap fn(Q) = \emptyset}{P \mid Q \stackrel{\alpha}{\rightarrow} P' \mid Q} \text{ (Par)} \qquad \frac{P \stackrel{\alpha}{\rightarrow} P'}{P + Q \stackrel{\alpha}{\rightarrow} P'} \text{ (Sum)}$$

$$\frac{P \stackrel{\alpha}{\rightarrow} P' \quad x \notin n(\alpha)}{(\nu x)P \stackrel{\alpha}{\rightarrow} (\nu x)P'} \text{ (Res)} \qquad \frac{P \stackrel{\overline{x}w}{\rightarrow} P' \quad x \neq w}{(\nu w)P \stackrel{(\nu w)\overline{x}w}{\rightarrow} P'} \text{ (Open)}$$

$$\frac{P \stackrel{\alpha}{\rightarrow} P' \quad Q \equiv P \quad P' \equiv Q'}{Q \stackrel{\alpha}{\rightarrow} Q'} \text{ (Var)}$$

Annotated operational semantics

We just keep into account the name of the reacting channel $\tau(x)$ or the application of a process call $x = *$.

$$\frac{P \stackrel{\overline{x}w}{\rightarrow} P' \quad Q \stackrel{xw}{\rightarrow} Q'}{P \mid Q \stackrel{\tau(x)}{\rightarrow} P' \mid Q'} \text{(Com)}$$

$$\frac{P \stackrel{(\nu w)\overline{x}w}{\rightarrow} P' \quad Q \stackrel{xw}{\rightarrow} Q' \quad w \notin fn(Q)}{P \mid Q \stackrel{\tau(x)}{\rightarrow} (\nu w)(P' \mid Q')} \text{(Close)}$$

$$\frac{\overrightarrow{v_i}\sigma = \overrightarrow{v} \quad F(\overrightarrow{v_i}) = P_i}{F(\overrightarrow{v}) \stackrel{\tau(*)}{\rightarrow} P_i\sigma} \text{(App)}$$

Example: factorial

$$
\begin{aligned}
fac(n, r) = \texttt{Case}\ \ n\ \ \texttt{of}\\
0 \rightarrow \bar{r}\langle 1 \rangle\\
m + 1 \rightarrow (\nu r')(fac(m, r')|r'(x).\bar{r}\langle x \times (m + 1)\rangle)
\end{aligned}
$$

$$
fac(3, a) \overset{\tau(*)}{\rightarrow} (\nu r')(fac(2, r')|r'(x).\bar{a}\langle x \times 3\rangle)
$$

$$
\underbrace{\overset{\tau(*)}{\rightarrow} \ldots \overset{\tau(*)}{\rightarrow}}_{3 \text{ times}} \ldots \underbrace{\overset{\tau(r')}{\rightarrow} \ldots \overset{\tau(r')}{\rightarrow}}_{3 \text{ times}} \bar{a}\langle 6 \rangle
$$

The internal channels are abstracted as a whole.

Strongly normalizing channels and processes

For any substitution (in the usual sense) σ,

$$fac(n,r)\sigma = fac(k,a) \underbrace{\xrightarrow{\tau(*)} \dots \xrightarrow{\tau(*)}}_{k+1 \text{ times}} \dots \underbrace{\xrightarrow{\tau(r')} \dots \xrightarrow{\tau(r')}}_{k \text{ times}} \overline{a}\langle k! \rangle$$

We write :

- $fac(n,r) \in SN_{\tau(*)}(k+1)$,
- $fac(n,r) \in SN_{\tau(r')}(k)$,
- $fac(n,r) \in SN_{\tau(r)}(0)$
- and $fac(n,r) \in SN$

Resource processes

- Resource processes are trivial processes with no recursion and no replication:

$$\mathcal{R} \ni R, S ::= 0 \mid \tau_\chi^n.R \mid \mu.R \mid R|R \mid R + R \mid (\nu x)R$$

  where $\tau_\chi^n.R$ will be able to perform n transitions labeled by $\tau(\chi)$
- structural congruence is extended by: $\tau_\chi^0.R \equiv R$
- operational semantics is extended by:

$$\frac{}{\tau_\chi^{n+1}.R \stackrel{\tau(\chi)}{\rightarrow} \tau_\chi^n.R} \ (n \geq 0)$$

Assignments

Let $D(P) = \{F_1, \cdots, F_m\}$ be the definitions of P

Definition (Assignment)

Given a process P s.t. $D(P) = \{F_1, \cdots, F_m\}$, an assignment $[-]$ is a total map from $D(P)$ to $\mathit{Values} \to \mathcal{R}$.

i.e. $[F_i]$ is a total function of arity n and $v_1, \cdots, v_n \in \mathit{Values}$ implies that $[F_i](v_1, \cdots, v_n) \in \mathcal{R}$.

Definition (Process assignment)

Given an assignment $[-]$ and a process P, the process assignment $[P]$ is the canonical extension of $[-]$ to P:

$$[P|Q] = [P]\|[Q] \qquad [F(\overrightarrow{v})] = [F](\overrightarrow{v}) \qquad \ldots$$

Interpretation

Definition (Interpretation)

Given a process P, an assignment $[-]$ is an interpretation of P if for each process definition $F \in D(P)$ of the shape:

$$F(\overrightarrow{X}) = \texttt{Case}\ \ \overrightarrow{X}\ \ \texttt{of}\ \ \overrightarrow{v_1} \rightarrow P_1, \ldots, \overrightarrow{v_k} \rightarrow P_k$$

for each ground substitution σ the following holds:

$$\forall i \in \{1, \ldots, k\},\ [F(\overrightarrow{v_i})\sigma] \gtrsim \tau_*^1.[P_i\sigma]$$

where the partial preorder $\gtrsim$ on resource processes is the standard simulation relation defined by:

$$R \gtrsim R' \text{ if } \forall S' \text{ s.t. } R' \xrightarrow{\alpha} S',\ \exists S \text{ s.t. } R \xrightarrow{\alpha} S \text{ and } S \gtrsim S'$$

Example of interpretation

$$
\begin{aligned}
fac(n,r) = \texttt{Case}\ \ n\ \ \texttt{of}\ & \\
& 0 \to \overline{r}\langle 1\rangle \\
& m+1 \to (\nu r')(fac(m,r')|r'(x).\overline{r}\langle x \times (m+1)\rangle)
\end{aligned}
$$

Setting $[fac](n,r) = (\nu r')\tau^{n}_{r'}|\tau^{n+1}_{*}|\overline{r}\langle n!\rangle$

$$[fac](0,r) = (\nu r')\tau^{0}_{r'}|\overline{r}\langle 0!\rangle|\tau^{1}_{*} \equiv \overline{r}\langle 1\rangle|\tau^{1}_{*}$$
$$\gtrsim \tau^{1}_{*}.\overline{r}\langle 1\rangle = \tau^{1}_{*}.[\overline{r}\langle 1\rangle]$$

$$[fac](m+1,r) = (\nu r')\tau^{m+1}_{r'}|\tau^{m+2}_{*}|\overline{r}\langle (m+1)!\rangle$$
$$\gtrsim \tau^{1}_{*}.(\nu r')(((\nu r')\tau^{m}_{r'}|\tau^{m+1}_{*}|\overline{r'}\langle m!\rangle)|(r'(x).\overline{r}\langle x \times (m+1)\rangle)$$
$$= \tau^{1}_{*}.[(\nu r')(fac(m,r')|r'(x).\overline{r}\langle x \times (m+1)\rangle)]$$

Resource algebra and semantics

Consider the max-plus algebra $(\mathbb{N}^{Names}, \otimes, \oplus)$ defined by $\forall \delta, \delta' \in \mathbb{N}^{Names}, \chi \in \mathrm{N}$:

$$(\delta \otimes \delta')(\chi) = \delta(\chi) + \delta'(\chi)$$
$$(\delta \oplus \delta')(\chi) = \max(\delta(\chi), \delta'(\chi))$$

Define $Reach^{\chi}(R) = \{S \mid R \overset{\tau(\chi)}{\rightarrow} S\}$.

Definition (Resource Process semantics)

The resource process semantics $[\![-]\!]$ is defined by:

$$[\![R]\!] = \oplus\{1_{\{\chi\}} \otimes [\![S]\!] \mid \forall \chi \in \mathbb{D},\ \forall S \in Reach^{\chi}(R)\}$$

provided that $\oplus\emptyset = \delta_0$.

Results

Define $\delta \succeq \delta'$ iff $\delta \oplus \delta' = \delta$.

Lemma (Resource consumption)

For each process P of interpretation $[-]$*, if* $P \overset{\tau(\chi)}{\rightarrow} P'$, $\chi \in \mathbb{D}$*, then* $[\![P]\!] \succeq 1_\chi \otimes [\![P']\!]$.

Theorem (Soundness)

For each process P of interpretation $[-]$*, we have:*

$$\forall \chi \; P \in SN_{\tau(\chi)}([\![P]\!](\chi))$$

Theorem (Completeness)

A process P has interpretation $[-]$ *if and only if* $P \in SN$.

Extensions

The methodology is not restricted to "finite time". We are able to infer other results by playing on $(\gtrsim, [\![-]\!])$:

- space :
  - upper bounds on the maximal value sent on a channel (see the paper)
  - upper bounds on the numbers of active channels and channel creations can be obtained (not in the paper)
- time :
  - upper bounds even for non terminating processes
  - e.g. $P = !a(x).\overline{b}x | !\overline{a}v | b(x)$
    - $\in SN_{\tau(b)}(1)$
    - even if neither $P \in SN_{\tau(a)}$,
    - nor $P \in SN_{\tau(*)}$ hold.

Conclusion

Nice adaptation of an existing technique to another domain:

- relating interpretation and simulation
- that can be adapted to control several notions of resources on processes
- that can handle all inductive data types (lists, ...)

The synthesis problem is clearly undecidable
(but guessed to be decidable on restricted spaces).