An introduction to process calculi: Calculus of Communicating Systems (CCS)

Lecture 2 of Modelli Matematici dei Processi Concorrenti

Paweł Sobociński

University of Southampton, UK

Introduction

- In the first lecture we discussed behavioural preorders and equivalences, in particular bisimilarity.
- This lecture is an introduction to a well-known process-calculus, the Calculus of Communicating Systems (CCS) introduced by Robin Milner in the early 1980's.
- Process calculi:
  - a pseudo-programming language which usually focuses on a small language feature;
  - usually try to eliminate syntactic sugar and extra programmer-friendly features;
  - idea is to isolate basic principles and reasoning techniques.

CCS

- focuses on a very simple paradigm of synchronous handshakes.
- processes $a?P$ and $a!Q$, when executing in parallel ($a?P \parallel a!Q$);
  - can **synchronise** their execution by synchronising on **channel** a;
  - after the synchronisation continue as P and Q, respectively;
  - $a?P \parallel a!Q \rightarrow P \parallel Q$;

CCS syntax

Assume that we have a set of names A and a countable set of process variables.

$$P \ ::= \ 0 \mid a?P \mid a!P \mid P \parallel P \mid P + P \mid \nu a P \mid X \mid \mu X.P$$

NB. sometimes $a?P$ is written aP and $a!P$ is written $\overline{a}P$. In early texts $\nu a P$ is written $P \backslash a$.

- $a?P$: input on a and proceed as P;
- $a!P$: output on a and proceed as P;
- τP : perform an internal reduction and proceed as P;
- $P_1 \parallel P_2$: put P_1 and P_2 in parallel;
- $P_1 + P_2$: act either as P_1 or as P_2;
- $\nu a P$: treat a as a local channel visible only in P;

Recursion

Infinite behaviour can be added to CCS in several (nonequivalent) ways.

$$P ::= \ldots \mid X \mid \mu X.P$$

The expression $\mu X.P$ stands for treats the X as a recursive variable in P.

Idea:

- $\mu X.a?X \quad \text{“}\equiv\text{”} \quad a?a?\ldots$;
- $\mu X.P \parallel X \quad \text{“}\equiv\text{”} \quad P \parallel P \parallel \ldots$;
- $\mu X.a!(b?X + c?X) \quad \text{“}\equiv\text{”} \quad ?$

Contexts and Congruences

Definition 1 (Contexts)**.**

$$\mathcal{C} \;::=\; - \mid P \parallel C \mid C \parallel P \mid C + P \mid P + C \mid \nu a C \mid a?C \mid a!C$$

Definition 2 (Substitution)**.** *Given a term P and a context C, let $C[P]$ denote the term obtained by substituting P for $-$. Free names may be captured!*

Definition 3. *A relation R on the terms of* CCS *is said to be a congruence when*

$$PRQ \quad \Rightarrow \quad \sigma(P)R\sigma(Q)$$

for all operations σ of CCS*. More formally, if PRQ then $C[P]RC[Q]$ for all contexts C.*

Structural congruence

- $P \parallel Q$ denotes the same system as $Q \parallel P$;
- We quotient the “raw” syntax via a relation which is a congruence with respect to the operations of CCS.

Definition 4 (SC)**.** *Let $\equiv$ be the smallest congruence which includes the following:*

$$(P \parallel Q) \parallel R \equiv P \parallel (Q \parallel R) \quad P \parallel Q \equiv Q \parallel P \quad P \parallel 0 \equiv P$$

$$(P + Q) + R \equiv P + (Q + R) \quad P + Q \equiv Q + P \quad P + 0 \equiv P$$

$$\nu a(P \parallel Q) \equiv (\nu a P) \parallel Q \;\; (a \notin Q) \quad \nu a P \equiv \nu b P[b/a] \;\; (b \notin P)$$

$$\mu X.P \equiv P[\mu X.P/X]$$

Remark 5. *Contexts are **not** quotiented by* SC*. This is because we want the contexts to have the power to bind.*

Reduction semantics 1

Idea, basic reduction:

$$a?P_1 + P_2 \parallel a!Q_1 + Q_2 \rightarrow P_1 \parallel Q_1$$

Letting

$$l_{a,P_1,P_2,Q_1,Q_2} \stackrel{\text{def}}{=} a?P_1 + P_2 \parallel a!Q_1 + Q_2$$

$$r_{a,P_1,P_2,Q_1,Q_2} \stackrel{\text{def}}{=} P_1 \parallel Q_1$$

we want, for all a, P_1, P_2, Q_1, Q_2

$$l_{a,P_1,P_2,Q_1,Q_2} \rightarrow r_{a,P_1,P_2,Q_1,Q_2}$$

Reduction semantics 2

But parallel composition shouldn't inhibit reduction, so that if $P \to P'$ then for all Q we should also have $P \parallel Q \to P' \parallel Q$. Similarly, $\nu a P \to \nu a P'$.

Evaluation contexts:

$$E ::= - \parallel P \mid \nu a -$$

Definition 6. $P \to P'$ *iff* $\exists a, P_1, P_2, Q_1, Q_2$ *and evaluation context* E *such that* $P \equiv E[l_{a,P_1,P_2,Q_1,Q_2}]$ *and* $P' \equiv E[r_{a,P_1,P_2,Q_1,Q_2}]$*.*

SOS

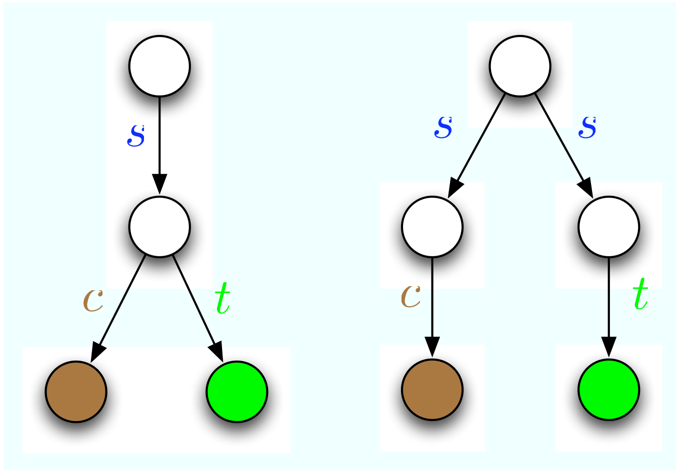
There is a useful way of presenting the reduction semantics using structural operational semantics (SOS).

$$\frac{}{a?P_1+P_2 \| a!Q_1+Q_2 \to P_1 \| Q_1}$$

$$\frac{P \to P'}{P \| Q \to P' \| Q} \qquad \frac{P \to P'}{\nu a P \to \nu a P'}$$

- The rules above generate a transition system (which we can think of as an LTS with only one label);
- we will refer to this as the reduction transition system.

Examples



$$M_1 \stackrel{\text{def}}{=} s?(c! + t!) \quad M_2 \stackrel{\text{def}}{=} s?c! + s?t!$$

- Coffee drinker $C_1 \stackrel{\text{def}}{=} s!c?0$;
- Simulation 1: $C_1 \parallel M_1 \rightarrow c?0 \parallel (c!0 + t!0) \rightarrow 0$
- Simulation 2: $C_1 \parallel M_2 \rightarrow c?0 \parallel t!0$

Process equivalence?

- What is a good notion of process equivalence?
  - we can try bisimilarity on the reduction LTS...
  - but then, for instance, $a!P \sim b!P$. In fact, all processes which do not reduce are bisimilar. Clearly, this is not sufficient.

Contextual equivalence

A canonical process equivalence for a process language is **contextual equivalence**. The idea originally arose in the theory of the λ-calculus.

Idea: Processes P and Q can be distinguished if in some context C they "behave differently".

- we can try taking the largest bisimulation which is **also a congruence**;
- this almost works, the problem is that processes which can always reduce are equated;

Reduction barbed congruence

One sensible definition of a contextual equivalence for CCS is **reduction barbed congruence**.

Definition 7. *A **barb** is a basic observation. In* CCS *it makes sense to take the instantaneous ability to input or output on a name. We say that* $P \downarrow_a$ *iff* $P \equiv \nu\overline{k}(a!P_1 + P_2 \parallel P_3)$ *or* $P \equiv \nu\overline{k}(a?P_1 + P_2 \parallel P_3)$ *and* $a \notin k$*.*

Definition 8. *Reduction barb congruence Let* $\cong$ *be the largest equivalence relation which is*

- *a congruence;*
- *barb-closed: if* $P \cong Q$ *and* $P \downarrow_a$ *then* $Q \downarrow_a$*;*
- *reduction-closed: if* $P \cong Q$ *and* $P \longrightarrow P'$ *then* $\exists Q'$*,* $Q \longrightarrow Q'$ *and* $Q \cong Q'$ *(for all contexts* C*,* $C[P] \cong C[Q]$*, bisimulation on the reduction* LTS*)*

Examples

Claim 9 (Firewall). $\nu a.a! \cong 0$.

Proof. ? $\square$

Problems with contextual equivalence

- $\cong$ is a priori defined to be a congruence;
- thus, in principle, to check that $P \cong Q$ we have to have a proof which takes into account an infinite number of arbitrarily complex contexts C

LTS characterisation

We will give a labelled transition system on which bisimulation characterises contextual equivalence, ie

- it is **sound**: $\sim \subseteq \cong$;
- it is **complete**: $\cong \subseteq \sim$.

LTS for CCS

$$\frac{}{a?P \xrightarrow{a?} P}\ (\textsc{In}) \qquad \frac{}{a!P \xrightarrow{a!} P}\ (\textsc{Out}) \qquad \frac{P \xrightarrow{a!} P' \quad Q \xrightarrow{a?} Q'}{P \| Q \xrightarrow{\tau} P' \| Q'}\ (\textsc{Tau})$$

$$\frac{P \xrightarrow{\alpha} P'}{P \| Q \xrightarrow{\alpha} P' \| Q}\ (\textsc{Par}) \qquad \frac{P \xrightarrow{\alpha} P' \quad (a \notin \alpha)}{\nu a P \xrightarrow{\alpha} \nu a P'}\ (\textsc{Nu})$$

$$\frac{P \equiv P' \quad P' \xrightarrow{\alpha} Q' \quad Q' \equiv Q}{P \xrightarrow{\alpha} Q}\ (\textsc{StrCong})$$

Example

Lemma 10 (Firewall)**.** $\nu a.a! \sim 0$

Proof. Obvious, since both sides do not have any transitions. $\square$

Lemma 11 (Expansion)**.** *If* $a \neq b$ *then* $a? \parallel b! \sim a?b! + b!a?$.

Proof. $\{(a? \parallel b!, a?b! + b!a?), (b!, b!), (a?, a?), (0, 0)\}$ is a bisimulation. $\square$

Proof of soundness

Theorem 12. $\sim$ *is a congruence.*

Proof. We'll do the case $- \parallel Q$. We will prove that
$\mathcal{R} = \{(P_1 \parallel R, P_2 \parallel R), P_1 \sim P_2\}$ is a bisimulation.
Suppose that $P_1 \parallel Q \xrightarrow{\alpha} R$. We'll do the case $\alpha = \tau$. If $P_1 \xrightarrow{\tau} P_1'$ and $R \equiv P_1' \parallel Q$ then also $P_2 \xrightarrow{\tau} P_2'$ such that $P_1' \sim P_2'$. Then $P_2 \parallel Q \xrightarrow{\tau} P_2' \parallel Q$, but $(R, P_2' \parallel Q) \in \mathcal{R}$.
The case $Q \xrightarrow{\tau} Q'$ such that $R \equiv P_1 \parallel Q'$ is similar.
The other possibility is, $P_1 \xrightarrow{a!} P_1'$ and $Q \xrightarrow{a?} Q'$ and $R \equiv P_1' \parallel Q'$. But then $P_2 \xrightarrow{a!} P_2'$ such that $P_1' \sim P_2'$ and so $P_2 \parallel Q \xrightarrow{\tau} P_2' \parallel Q'$.
(note the case $P_1 \xrightarrow{a?} P_1'$, $Q \xrightarrow{a!} Q'$ is symmetric) $\square$

Congruent bisimilarities

- Having a congruent bisimilarity is quite useful because it allows the use of a familiar algebraic principle – substituting "equal for equal".
- It can also reduce the burden of constructing bisimulations since bisimulations can be deconstructed:

Example 13. *If* $P_1 \sim Q_1$ *and* $P_2 \sim Q_2$ *then* $P_1 \parallel P_2 \sim Q_1 \parallel Q_2$*.*

Proof. $P_1 \parallel P_2 \sim Q_1 \parallel P_2 \sim Q_1 \parallel Q_2$. $\square$

Proof of soundness

Lemma 14. *$P\downarrow_a$ iff $P \xrightarrow{a!}$ or $P \xrightarrow{a?}$.*

Lemma 15. *$P \xrightarrow{\tau} P'$ iff $P \rightarrow P'$.*

Theorem 16. *$\sim \subseteq \cong$.*

Proof. Recall that $\cong$ is defined to be the largest barb and reduction closed congruence. $\sim$ is reduction and barbed closed (the two lemmas) and is a congruence. Hence $\sim \subseteq \cong$. $\square$

Proof of completeness

Theorem 17. $\cong \subseteq \sim$.

We need to make sure that the label transitions do not observe too much about the processes. Hence for each label α, we'll find a context which "observes" the label.

Weak equivalences

"Weak" in the jargon of process calculists means that internal reductions are not observable.

Definition 18 (Weak bisimulation)**.** *A relation R is a **weak bisimulation** when*

- PRQ *and* $P \xrightarrow{\tau} P'$ *then* $Q \xrightarrow{\tau}^{*} Q'$
- PRQ *and* $P \xrightarrow{\alpha} P'$ $(\alpha \neq \tau)$ *then* $Q \xrightarrow{\tau}^{*} \xrightarrow{\alpha} \xrightarrow{\tau}^{*} Q'$*;*
- *the symmetric versions of the above hold.*

Definition 19. *We will write τP for $\nu a(a!\|a?P)$ where a is fresh for P.*

Example 20.

- $\tau a! \approx a!$.

Weak bisimilarity not a congruence

The counterexample is very simple, we have $\tau a! \approx a!$, but $\tau a! + b! \not\approx a! + b!$. Weak bisimilarity behaves better with respect to other operators.