# Concurrency 5 = CCS (3/4)

## Examples, and axiomatization

Pierre-Louis Curien (CNRS – Université Paris 7)

MPRI concurrency course 2004/2005 with :

Jean-Jacques Lévy (INRIA-Rocquencourt)
Eric Goubault (CEA)
James Leifer (INRIA - Rocq)
Catuscia Palamidessi (INRIA - Futurs)

———————

(`http://pauillac.inria.fr/~leifer/teaching/mpri-concurrency-2004`)

# Specification and weak bisimulation

| HAMMER | JOBBER | STRONG JOBBER |
|---|---|---|
| $H = g \cdot H' \quad H' = p \cdot H$ | $J = in \cdot S \quad S = \overline{g} \cdot U$ <br> $U = \overline{p} \cdot F \quad F = \overline{out} \cdot J$ | $K = in \cdot D \quad D = \overline{out} \cdot K$ |

We have : $(\nu g, h)(J \mid J \mid H) \approx K \mid K$. Their first actions are the same :

$$(\nu g, h)(J \mid J \mid H) \;\mathcal{R}\; K \mid K \qquad (\nu g, h)(S \mid J \mid H) \;\mathcal{R}\; D \mid K$$

$$(\nu g, h)(J \mid S \mid H) \;\mathcal{R}\; K \mid D \qquad (\nu g, h)(S \mid S \mid H) \;\mathcal{R}\; D \mid D$$

The only possible sequence of actions out of, say, $(\nu g, h)(S \mid S \mid H)$ is :

$$(\nu g, h)(S \mid S \mid H) \xrightarrow{\tau} (\nu g, h)(S \mid U \mid H') \xrightarrow{\tau} (\nu g, h)(S \mid U \mid H') \xrightarrow{\overline{out}} (S \mid J \mid H)$$

Hence we complete $\mathcal{R}$ with :

$$(\nu g, h)(S \mid U \mid H') \;\mathcal{R}\; D \mid D \qquad (\nu g, h)(S \mid F \mid H) \;\mathcal{R}\; D \mid D$$

$$(\nu g, h)(J \mid U \mid H') \;\mathcal{R}\; K \mid D \qquad (\nu g, h)(J \mid F \mid H) \;\mathcal{R}\; K \mid D$$

$$(\nu g, h)(U \mid J \mid H') \;\mathcal{R}\; D \mid K \qquad (\nu g, h)(F \mid J \mid H) \;\mathcal{R}\; D \mid K$$

# CCS encodings (1/4)

(Thanks to Catuscia Palamidessi for the encodings of this lecture).

Here is a specification $P$ of (up to) $n$ readers in parallel and (at most) one writer :

$$R = \overline{p_R} \cdot \mathsf{read} \cdot \overline{v_R} \qquad S_0 = p_R \cdot S_1 + p_W \cdot v_W \cdot S_0$$

$$W = \overline{p_W} \cdot \mathsf{write} \cdot \overline{v_W} \qquad S_k = p_R \cdot S_{k+1} + v_R \cdot S_{k-1} \quad (0 < k < n)$$

$$S_n = v_R \cdot S_{n-1}$$

in

$(\nu p_R, v_R, p_W, v_W)(S_0|R|\cdots|R|W|\cdots|W)$ (arbitrarily many readers and writers)

If $P \xrightarrow{s} (\nu p_R, v_R, p_W, v_W)P'$, then there are two cases :

- $P' = S_i|Q$ : then up to $i$ threads of $Q$ can perform read and no thread can perform write.

- $P' = (v_W \cdot S_0)|Q$ : then no thread of $Q$ can perform read and at most one thread can perform write.

# CCS encodings (2/4)

The dining philosophers can be encoded by a closed linking (cf. previous lecture) of $n$ copies of the following process $\mathsf{Phil}_{n,p,a}$ (each philosopher holds its left fork at the beginning)

$$\begin{aligned}
\mathsf{Phil}_{n,p,a} &= \tau \cdot \mathsf{Phil}_{h,p,a} + \tau \cdot \mathsf{Phil}_{n,p,a} + \overline{c_L} \cdot \mathsf{Phil}_{n,a,a} \\
\mathsf{Phil}_{n,a,p} &= \text{ symmetric} \\
\mathsf{Phil}_{n,a,a} &= \tau \cdot \mathsf{Phil}_{n,a,a} + \tau \cdot \mathsf{Phil}_{h,a,a} \\
\mathsf{Phil}_{h,a,a} &= c_L \cdot \mathsf{Phil}_{h,p,a} + c_R \cdot \mathsf{Phil}_{h,a,p} \\
\mathsf{Phil}_{h,p,a} &= \overline{c_L}\mathsf{Phil}_{h,a,a} + c_R \cdot \mathsf{Phil}_{h,p,p} \\
\mathsf{Phil}_{h,a,p} &= \text{ symmetric} \\
\mathsf{Phil}_{h,p,p} &= \mathsf{eat} \cdot \mathsf{Phil}_{n,p,p} \\
\mathsf{Phil}_{n,p,p} &= \overline{c_L} \cdot \mathsf{Phil}_{n,a,p} + \overline{c_R} \cdot \mathsf{Phil}_{n,p,a}
\end{aligned}$$

- $n/h$ stand for "not hungry" / "hungry", $a/p$ for absent / present (second and third index for first and second fork, respectively)

- under the linking, $c_R$ (resp. $c_L$) is (privately) identified with the $c_L$ (resp. $c_R$) of the right (resp. left) neighbour

# CCS encodings (3/4)

We show, at any stage : Fairness $\Rightarrow$ Progress

Fairness A hungry philosopher, or a philosopher who has just eaten, is not ignored forever.

Progress If at least one philosopher is hungry, then eventually one of the hungry philosophers will eat.

By contradiction : Suppose $P$ is the state of the system in which one philosopher at least is hungry, and suppose that there is an infinite fair evolution $P \xrightarrow{\tau}^{\star} \cdots$ that makes no progress. Then :

Step 1 : Eventually, all philosophers hold at most one fork. No philosopher at any stage can be in state $(h, p, p)$, since by fairness eventually this philosopher will eat. If at some stage a philosopher is in state $(n, p, p)$, then by fairness this philosopher will eventually give one of his forks. No philosopher at any styage can be in state $(n, p, p)$ unless it was already in this state in $P$, since the only way to enter this state is after eating. Hence all the $(n, p, p)$ states will eventually disappear.

# CCS encodings (4/4)

Step 2 : Eventually, all philosophers hold exactly one fork. This is because if one philosopher had no fork, then another one would hold two ($n$ forks for $n-1$ philosophers).

Step 3 : When this happens, our philosopher is still hungry (he cannot revert to non-hungry unless he eats), say it is in state $(h, p, a)$, and eventually by Fairness it is his turn. The transition $(h, p, p)$ is forbidden. Hence he gives his fork to the left neighbour. Only a hungry philosopher receives forks, hence the neighbour is in state $(h, p, a)$, but then makes the transition $(h, p, p)$ which is also forbidden.

**Exercice 1** Show that the system can never deadlock.

# Strong axiomatization (1/4)

For finitary CCS (no recursion, finite guarded sums), $P \sim Q$ iff $\mathcal{A}_1 \vdash P = Q$, where $\mathcal{A}_1$ is :

$$
\begin{array}{ll}
(1) & \Sigma_{i\in I}\mu_i \cdot P_i = \Sigma_{i\in I}\mu_{f(i)} \cdot P_{f(i)} \quad \text{(permutation)} \\
(2) & \Sigma_{i\in I}\mu_i \cdot P_i + \mu_j \cdot P_j = \Sigma_{i\in I}\mu_i \cdot P_i \quad (j \in I) \quad \text{(idempotency)} \\
(3) & P \mid Q = \Sigma\{\mu \cdot (P' \mid Q) \mid P \xrightarrow{\mu} P'\} + \Sigma\{\mu \cdot (P \mid Q') \mid Q \xrightarrow{\mu} Q'\} \\
 & \qquad + \Sigma\{\tau \cdot (P' \mid Q') \mid P \xrightarrow{\alpha} P' \text{ and } Q \xrightarrow{\overline{\alpha}} Q'\} \quad \text{(expansion)} \\
(4) & (\nu a)\,(\Sigma_{i\in I}\mu_i \cdot P_i) = \Sigma_{\{j\in I \mid \mu_j \neq a,\overline{a}\}}\, \mu_j \cdot (\nu a)P_j
\end{array}
$$

**Exercice 2** Show that $\mathcal{A}_1 \vdash (\nu b)(a\cdot(b|c) + \tau\cdot(b|\overline{b}\cdot c)) = \tau\cdot\tau\cdot c\cdot 0 + a\cdot c\cdot 0$.

# Strong axiomatization (2/4)

First step : each process is provably equal to a synchronization tree (guarded sums only), using only

$$
\begin{array}{ll}
(3) & P \mid Q = \Sigma\{\mu \cdot (P' \mid Q) \mid P \xrightarrow{\mu} P'\} + \Sigma\{\mu \cdot (P \mid Q') \mid Q \xrightarrow{\mu} Q'\} \\
& \qquad +\Sigma\{\tau \cdot (P' \mid Q') \mid P \xrightarrow{\alpha} P' \text{ and } Q \xrightarrow{\overline{\alpha}} Q'\} \\
(4) & (\nu a)\,(\Sigma_{i \in I} \mu_i \cdot P_i) = \Sigma_{\{j \in I \mid \mu_j \neq a, \overline{a}\}}\, \mu_j \cdot (\nu a) P_j
\end{array}
$$

We associate with a process $P$ the multi-set of the sizes of all its subterms $(\nu a)Q$ and $Q_1 \mid Q_2$. This multi-set decreases at each application of rules (3)-(4).

# Strong axiomatization (3/4)

Second step : If $P = \Sigma_{i=1\ldots m}\alpha_i \cdot P_i$ and $Q = \Sigma_{j=m+1\ldots n}\alpha_j \cdot P_j$, and if $P \sim Q$, then $P$ and $Q$ are provably equal, using only

$$(1) \quad \Sigma_{i\in I}\mu_i \cdot P_i = \Sigma_{i\in I}\mu_{f(i)} \cdot P_{f(i)} \quad (f \text{ permutation})$$

$$(2) \quad \Sigma_{i\in I}\mu_i \cdot P_i + \mu_j \cdot P_j = \Sigma_{i\in I}\mu_i \cdot P_i \quad (j \in I)$$

Induction on size($P$)+size($Q$) : Let $\leftrightharpoons$ be the equivalence relation on $\{1,\ldots n\}$ defined by $i \leftrightharpoons j$ iff $\alpha_i = \alpha_j$ and $P_i \sim P_j$.

By strong bisimilarity, each $\leftrightharpoons$ equivalence class contains at least one element of $[1,m]$ and at least one element of $[m+1,n]$. Now for each of the equivalence classes we pick one representative in $[1,m]$ and one in $[m+1,n]$. Call them $p_1,\ldots,p_k$ and $q_1,\ldots,q_k$, respectively. Then we have :

$$\vdash \Sigma_{i=1\ldots m}\alpha_i.P = \Sigma_{l=1\ldots k}\alpha_{p_l}\cdot P_{p_l} \quad \text{and} \quad \vdash \Sigma_{j=m+1\ldots n}\alpha_j\cdot P_j = \Sigma_{l=1\ldots k}\alpha_{q_l}\cdot P_{q_l}$$

with $P_{p_l} \sim P_{q_l}$ for all $l$, so we can apply induction.

(Note that the finiteness of sums is crucial.)

# Weak axiomatization (1/6)

For finitary CCS, $P \approx Q$ iff $\mathcal{A}_1 + \mathcal{A}_2 \vdash P = Q$, where $\mathcal{A}_2$ is :

$$
\begin{array}{ll}
(\tau_0) & P = \tau \cdot P \\
(\tau_1) & \tau \cdot P + R = P + \tau \cdot P + R \\
(\tau_2) & \alpha \cdot (\tau \cdot P + Q) + R = \alpha \cdot (\tau \cdot P + Q) + \alpha \cdot P + R
\end{array}
$$

(In general, we do not have $\vdash P + Q = \tau \cdot P + Q$.)

# Weak axiomatization (2/6)

We can limit ourselves to synchronization trees (ST).

There is a notion of ST in fully standard form such that :

- each ST $P$ is provably equal (by $\mathcal{A}_2$) to a ST in fully standard form

- if $P, Q$ are in fully standard form and $P \approx Q$, then $P$ and $Q$ are provably equal

.

# Weak axiomatization (3/6)

Definition : $P = \Sigma_{i \in I} \mu_i \cdot P_i$ is in fully standard form if and only if

each $P_i$ is in fully standard form and

$\forall\, \mu, P' \ (P \stackrel{\mu}{\Rightarrow} P'$ and $P' \neq P) \Rightarrow P \stackrel{\mu}{\rightarrow} P'$

# Weak axiomatization (4/6)

Lemma : For any ST $P$, if $P \stackrel{\mu}{\Rightarrow} P'$ and $P \neq P'$, then $\vdash P = P + \mu.P'$.

Then, given $P = \Sigma_{i\in I}\mu_i \cdot P_i$, first convert each $P_i$ to a fully standard form $P'_i$. Next, consider all $(\nu_j, P''_j)$ such that $P' = \Sigma_{i\in I}\mu_i \cdot P'_i \stackrel{\nu_j}{\Rightarrow} P''_j$. Then

$$\vdash P = \Sigma_{i\in I}\mu_i \cdot P'_i = \Sigma_{i\in I}\mu_i \cdot P'_i + \Sigma_j \nu_j \cdot P''_j = Q'$$

and $Q'$ is in fully standard form :

- Each $P''_j$, being a subterm of some $P'_i$, is in fully standard form.

- Suppose $Q' \stackrel{\nu}{\Rightarrow} Q''$, passing through $P''_{j_0}$ :

1. $\nu = \nu_{j_0} = \alpha$ and $P''_{j_0} \stackrel{\tau}{\Rightarrow} Q''$. Then

$$(P' \stackrel{\nu_{j_0}}{\Rightarrow} P''_{j_0} \text{ and } P''_{j_0} \stackrel{\tau}{\Rightarrow} Q'') \Rightarrow P' \stackrel{\nu}{\Rightarrow} Q''$$

2. $\nu_{j_0} = \tau$ and $P'_{j_0} \stackrel{\nu}{\Rightarrow} P''$. Then we get also $P' \stackrel{\nu}{\Rightarrow} Q''$.

Then by definition of $Q'$ we have $\nu = \nu_{j_1}$ and $Q'' = P''_{j_1}$ for some $j_1$.

# Weak axiomatization (5/6)

Proof of the lemma (by induction on size($P$)) :

(1) $P \xrightarrow{\mu} P'$. Then $P = P_1 + \mu \cdot P'$ and $\vdash P = P + \mu \cdot P'$ by idempotency.

(2) $P \xrightarrow{\tau} P'' \stackrel{\mu}{\Rightarrow} P'$ and $P' \neq P''$. Then $P = P_1 + \tau \cdot P''$, and hence $\vdash P = P + P''$ by $(\tau_1)$. By induction we have $\vdash P'' = P'' + \mu \cdot P'$, so we conclude :

$$\vdash P = P + P'' = P + (P'' + \mu \cdot P') = (P + P'') + \mu \cdot P' = P + \mu \cdot P'$$

(3) $\mu = \alpha$, $P \xrightarrow{\alpha} P'' \stackrel{\tau}{\Rightarrow} P'$, and $P' \neq P''$. Then $P = P_1 + \alpha \cdot P''$, and by induction $\vdash P'' = P'' + \tau \cdot P'$. Hence, by $(\tau_2)$ :

$$\begin{aligned} \vdash P = P_1 + \alpha \cdot P'' &= P_1 + \alpha \cdot (P'' + \tau \cdot P') \\ &= P_1 + \alpha \cdot (P'' + \tau \cdot P') + \alpha \cdot P' = P + \alpha \cdot P' \end{aligned}$$

# Weak axiomatization (6/6)

If $P = \Sigma_{i \in I} \mu_i \cdot P_i$ and $Q = \Sigma_{j \in J} \nu_j \cdot Q_j$ are in fully standard form and $P \approx Q$, then we have "almost" $P \sim Q$.

Indeed, for every $P \xrightarrow{\mu_i} P_i$ there exists $Q'$ such that $Q' \approx P_i$ and $Q \overset{\mu_i}{\Rightarrow} Q'$, and hence $Q \xrightarrow{\mu_i} Q'$, the only possible exception being when $\mu_i = \tau$ and $Q' = Q$.

We prove $\vdash P = Q$ by induction on size($P$)+size($Q$). If the exceptional case does not apply, we proceed as for strong bisimulation and apply induction. Otherwise :

$$\exists\, i_0 \;\; (\mu_{i_0} = \tau \text{ and } P_{i_0} \approx Q \text{ and } \;\nexists\, j \;\; (\mu_j = \tau \text{ and } Q_j \approx P_{i_0}))$$

Now, we have :

$$(Q \approx \Sigma_{i \in I} \mu_i \cdot P_i \text{ and } \;\nexists\, j \;\; (\mu_j = \tau \text{ and } Q_j \approx P_{i_0})) \;\Rightarrow Q \approx \Sigma_{i \in I \setminus \{i_0\}} \mu_i \cdot P_i$$

Hence by induction $\vdash P_{i_0} = Q$ and $\vdash Q = \Sigma_{i \in I \setminus \{i_0\}} \mu_i \cdot P_i$, and we conclude with $(\tau_1)$ and $(\tau_0)$ :

$$\vdash Q = \tau \cdot Q = Q + \tau.Q = \Sigma_{i \in I \setminus \{i_0\}} \mu_i \cdot P_i + \tau.P_{i_0} = P$$