Concurrency 4

CCS - Simulation and bisimulation. Coinduction.

Catuscia Palamidessi
INRIA Futurs and LIX - Ecole Polytechnique

The other lecturers for this course:

Jean-Jacques Lévy (INRIA Rocquencourt)
James Leifer (INRIA Rocquencourt)
Eric Goubault (CEA)

http://pauillac.inria.fr/~leifer/teaching/mpri-concurrency-2005/

Announcement

The class of Wednesday 26 October will follow the usual schedule (16h15 - 19h15).

Outline

The semaphore

Define in CCS a semaphore with initial value n

First Solution

$$\mathrm{rec}_{S_n} down.\mathrm{rec}_{S_{n-1}}(up.S_n + down.\mathrm{rec}_{S_{n-2}}(\ldots(up.S_2 + down.\mathrm{rec}_{S_0}\, up.S_1)\ldots))$$

Second solution

- Let $S = \mathrm{rec}_X down.up.X$
- Then $S_n = S \mid S \mid \ \ldots \mid S \quad n$ times

The semaphore

Define in CCS a semaphore with initial value n

First Solution

$$\text{rec}_{S_n} down.\text{rec}_{S_{n-1}}(up.S_n + down.\text{rec}_{S_{n-2}}(\ldots(up.S_2 + down.\text{rec}_{S_0}\, up.S_1)\ldots))$$

Second solution

- Let $S = \text{rec}_X down.up.X$
- Then $S_n = S \mid S \mid \ldots \mid S \quad n$ times

The semaphore

Define in CCS a semaphore with initial value n

First Solution

$$\mathrm{rec}_{S_n} down.\mathrm{rec}_{S_{n-1}}(up.S_n + down.\mathrm{rec}_{S_{n-2}}(\ldots(up.S_2 + down.\mathrm{rec}_{S_0}\, up.S_1)\ldots))$$

Second solution

- Let $S = \mathrm{rec}_X down.up.X$
- Then $S_n = S \mid S \mid \ \ldots \mid S \quad n$ times

Maximal trace equivalence is not a congruence

Consider the following processes

- $P = a.(b.0 + c.0)$
- $Q = a.b.0 + a.c.0$
- $R = \bar{a}.\bar{b}.\bar{d}.0$

P and Q have the same maximal traces, but $(\nu a)(\nu b)(\nu c)(P \mid R)$ and $(\nu a)(\nu b)(\nu c)(Q \mid R)$ have different maximal traces.

Maximal trace equivalence is not a congruence

Consider the following processes

- $P = a.(b.0 + c.0)$
- $Q = a.b.0 + a.c.0$
- $R = \bar{a}.\bar{b}.\bar{d}.0$

P and Q have the same maximal traces, but $(\nu a)(\nu b)(\nu c)(P \mid R)$ and $(\nu a)(\nu b)(\nu c)(Q \mid R)$ have different maximal traces.

Outline

Syntax of “modern” CCS

- (channel, port) names: $a, b, c, \ldots$
- co-names: $\bar{a}, \bar{b}, \bar{c}, \ldots$ Note: $\bar{\bar{a}} = a$
- silent action: τ
- actions, prefixes: $\mu ::= a \mid \bar{a} \mid \tau$
- processes:

$$
\begin{array}{rcll}
P, Q & ::= & 0 & \text{inaction} \\
& \mid & \mu.P & \text{prefix} \\
& \mid & P \mid Q & \text{parallel} \\
& \mid & P + Q & \text{(external) choice} \\
& \mid & (\nu a)P & \text{restriction} \\
& \mid & K(\vec{a}) & \text{process name with parameters}
\end{array}
$$

- Process definitions:

$D ::= K(\vec{x}) \stackrel{def}{=} P$ where P may contain only the $\vec{x}$ as channel names

Syntax of "modern" CCS

- (channel, port) names: $a, b, c, \ldots$
- co-names: $\bar{a}, \bar{b}, \bar{c}, \ldots$ Note: $\bar{\bar{a}} = a$
- silent action: τ
- actions, prefixes: $\mu ::= a \mid \bar{a} \mid \tau$
- processes: P, Q

$::=$	0	inaction
$\mid$	$\mu.P$	prefix
$\mid$	$P \mid Q$	parallel
$\mid$	$P + Q$	(external) choice
$\mid$	$(\nu a)P$	restriction
$\mid$	$K(\vec{a})$	process name with parameters

- Process definitions:
  $D ::= K(\vec{x}) \stackrel{def}{=} P$ where P may contain only the $\vec{x}$ as channel names

Outline

Labeled transition system for “modern” CCS

We assume a given set of definitions D

$$[\text{Act}] \quad \frac{}{\mu.P \xrightarrow{\mu} P} \qquad [\text{Res}] \quad \frac{P \xrightarrow{\mu} P' \quad \mu \neq a, \bar{a}}{(\nu a)P \xrightarrow{\mu} (\nu a)P'}$$

$$[\text{Sum1}] \quad \frac{P \xrightarrow{\mu} P'}{P+Q \xrightarrow{\mu} P'} \qquad [\text{Sum2}] \quad \frac{Q \xrightarrow{\mu} Q'}{P+Q \xrightarrow{\mu} Q'}$$

$$[\text{Par1}] \quad \frac{P \xrightarrow{\mu} P'}{P|Q \xrightarrow{\mu} P'|Q} \qquad [\text{Par2}] \quad \frac{Q \xrightarrow{\mu} Q'}{P|Q \xrightarrow{\mu} P|Q'}$$

$$[\text{Com}] \quad \frac{P \xrightarrow{a} P' \quad Q \xrightarrow{\bar{a}} Q'}{P|Q \xrightarrow{\tau} P'|Q'} \qquad [\text{Rec}] \quad \frac{P[\vec{a}/\vec{x}] \xrightarrow{\mu} P' \quad K(\vec{x}) \stackrel{def}{=} P \in D}{K(\vec{a}) \xrightarrow{\mu} P'}$$

The reason for moving to “modern” CCS was to get static scope (thanks to the presence of the parameters). The old version had dynamic scope.

Outline

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup \{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \; \mathcal{R} \; Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \; \mathcal{R} \; Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Simulation

- **Definition** We say that a relation R on processes is a *simulation* if

$$P \,\mathcal{R}\, Q \quad \text{implies that} \quad \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \,\mathcal{R}\, Q'$$

- Note that this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\lesssim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a simulation}\}$
- **Theorem** $\lesssim$ is a bisimulation (Proof: Exercise)
- $P \lesssim Q$ intuitively means that Q can do everything that P can do. Q simulates P.

Outline

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a bisimulation}\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a bisimulation}\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R}$ is a bisimulation$\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \stackrel{\mu}{\rightarrow} P' \textit{ then } \exists Q' \textit{ s.t. } Q \stackrel{\mu}{\rightarrow} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \stackrel{\mu}{\rightarrow} Q' \textit{ then } \exists P' \textit{ s.t. } P \stackrel{\mu}{\rightarrow} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R}$ is a bisimulation$\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \ = \ \bigcup\{\mathcal{R} \mid \mathcal{R} \text{ is a bisimulation}\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Bisimulation

- **Definition** We say that a relation R on processes is a *bisimulation* if

$$P \mathcal{R} Q \quad \text{implies that} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \mathcal{R} Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \mathcal{R} Q' \end{array}$$

- Again, this property does not uniquely defines $\mathcal{R}$. There may be several relations that satisfy it.
- Define $\sim \; = \; \bigcup\{\mathcal{R} \mid \mathcal{R}$ is a bisimulation$\}$
- **Theorem** $\sim$ is a bisimulation (Proof: Exercise)
- $P \sim Q$ intuitively means that Q can do everything that P can do, and vice versa, at every step of the computation. Q is bisimilar to P.

Outline

Proof methods

- Simulation and bisimulation are coinductive definitions.
- In order to prove that $P \lesssim Q$ it is sufficient to find a simulation $\mathcal{R}$ such that $P \mathrel{\mathcal{R}} Q$
- Similarly, in order to prove that $P \sim Q$ it is sufficient to find a bisimulation $\mathcal{R}$ such that $P \mathrel{\mathcal{R}} Q$

Proof methods

- Simulation and bisimulation are coinductive definitions.
- In order to prove that $P \lesssim Q$ it is sufficient to find a simulation $\mathcal{R}$ such that $P \mathcal{R} Q$
- Similarly, in order to prove that $P \sim Q$ it is sufficient to find a bisimulation $\mathcal{R}$ such that $P \mathcal{R} Q$

Proof methods

- Simulation and bisimulation are coinductive definitions.
- In order to prove that $P \lesssim Q$ it is sufficient to find a simulation $\mathcal{R}$ such that $P \mathrel{\mathcal{R}} Q$
- Similarly, in order to prove that $P \sim Q$ it is sufficient to find a bisimulation $\mathcal{R}$ such that $P \mathrel{\mathcal{R}} Q$

Proof methods

- Simulation and bisimulation are coinductive definitions.
- In order to prove that $P \lesssim Q$ it is sufficient to find a simulation $\mathcal{R}$ such that $P \mathcal{R} Q$
- Similarly, in order to prove that $P \sim Q$ it is sufficient to find a bisimulation $\mathcal{R}$ such that $P \mathcal{R} Q$

Outline

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Examples and exercises

- Consider the following processes
  - $P = a.(b.0 + c.0)$
  - $Q = a.b.0 + a.c.0$

  Prove that $Q \lesssim P$ but $P \not\lesssim Q$ and $Q \not\sim P$
- Assume that $Q \lesssim P$ and $P \lesssim Q$ (for two generic P and Q). Does it follow that $P \sim Q$?
- Consider the following processes
  - $R = a.(b.0 + b.0)$
  - $S = a.b.0 + a.b.0$

  Prove that $Q \sim P$
- Consider the two definitions of semaphore given at the beginning of this lecture. Prove that they are bisimilar.
- Consider the processes $H(a)$ and $K(a)$ defined by $H(x) \stackrel{def}{=} x.H(x)$ and $K(x) \stackrel{def}{=} x.K(x) \mid x.K(x)$. Are they bisimilar?
- What is the smallest bisimulation?

Outline

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.
- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$P \; \mathcal{F}(\mathcal{R}) \; Q \quad \text{iff} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P' \,\mathcal{R}\, Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P' \,\mathcal{R}\, Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic
- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$\mathit{lfp}(\mathcal{F}) = \bigcap \{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\}$$

$$\mathit{gfp}(\mathcal{F}) = \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\}$$

- **Corollary** $\sim \; = \; \mathit{gfp}(\mathcal{F})$
- A similar characterization, of course, holds for $\lesssim$ as well.

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.

- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$\begin{array}{lll} P\ \mathcal{F}(\mathcal{R})\ Q & \text{iff} & \text{if } P \xrightarrow{\mu} P' \text{ then } \exists Q' \text{ s.t. } Q \xrightarrow{\mu} Q' \text{ and } P'\ \mathcal{R}\ Q' \\ & & \text{if } Q \xrightarrow{\mu} Q' \text{ then } \exists P' \text{ s.t. } P \xrightarrow{\mu} P' \text{ and } P'\ \mathcal{R}\ Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic

- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$lfp(\mathcal{F}) = \bigcap \{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\}$$

$$gfp(\mathcal{F}) = \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\}$$

- **Corollary** $\sim\ =\ gfp(\mathcal{F})$

- A similar characterization, of course, holds for $\lesssim$ as well.

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.
- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$P\ \mathcal{F}(\mathcal{R})\ Q \quad \text{iff} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P'\ \mathcal{R}\ Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P'\ \mathcal{R}\ Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic
- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$\begin{aligned} \mathit{lfp}(\mathcal{F}) &= \bigcap \{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\} \\ \mathit{gfp}(\mathcal{F}) &= \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\} \end{aligned}$$

- **Corollary** $\sim\, = \mathit{gfp}(\mathcal{F})$
- A similar characterization, of course, holds for $\lesssim$ as well.

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.

- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$P\ \mathcal{F}(\mathcal{R})\ Q \quad \text{iff} \quad \begin{array}{l} \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P'\ \mathcal{R}\ Q' \\ \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P'\ \mathcal{R}\ Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic

- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$\begin{aligned} lfp(\mathcal{F}) &= \bigcap\{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\} \\ gfp(\mathcal{F}) &= \bigcup\{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\} \end{aligned}$$

- **Corollary** $\sim\ =\ gfp(\mathcal{F})$

- A similar characterization, of course, holds for $\lesssim$ as well.

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.

- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$\begin{array}{lll} P\ \mathcal{F}(\mathcal{R})\ Q & \text{iff} & \textit{if } P \xrightarrow{\mu} P' \textit{ then } \exists Q' \textit{ s.t. } Q \xrightarrow{\mu} Q' \textit{ and } P'\ \mathcal{R}\ Q' \\ & & \textit{if } Q \xrightarrow{\mu} Q' \textit{ then } \exists P' \textit{ s.t. } P \xrightarrow{\mu} P' \textit{ and } P'\ \mathcal{R}\ Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic

- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$\begin{aligned} \textit{lfp}(\mathcal{F}) &= \bigcap \{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\} \\ \textit{gfp}(\mathcal{F}) &= \bigcup \{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\} \end{aligned}$$

- **Corollary** $\sim\ =\ \textit{gfp}(\mathcal{F})$

- A similar characterization, of course, holds for $\lesssim$ as well.

Bisimulation as greatest fixpoint

- Consider the set of relations on processes (that is, on the powerset of the cartesian product on processes) ordered by set inclusion. Obviously, this is a complete lattice.

- **Definition** Let $\mathcal{F}$ be a function on relation defined in the following way:

$$P\ \mathcal{F}(\mathcal{R})\ Q \quad \text{iff} \quad \begin{array}{l} \textit{if } P \stackrel{\mu}{\rightarrow} P' \textit{ then } \exists Q' \textit{ s.t. } Q \stackrel{\mu}{\rightarrow} Q' \textit{ and } P'\ \mathcal{R}\ Q' \\ \textit{if } Q \stackrel{\mu}{\rightarrow} Q' \textit{ then } \exists P' \textit{ s.t. } P \stackrel{\mu}{\rightarrow} P' \textit{ and } P'\ \mathcal{R}\ Q' \end{array}$$

- **Lemma** $\mathcal{F}$ is monotonic

- **Theorem (Knaster-Tarski)** F has (unique) least and greatest fixpoints, and

$$\begin{aligned} \mathit{lfp}(\mathcal{F}) &= \bigcap\{\mathcal{R} \mid \mathcal{F}(\mathcal{R}) \subseteq \mathcal{R}\} \\ \mathit{gfp}(\mathcal{F}) &= \bigcup\{\mathcal{R} \mid \mathcal{R} \subseteq \mathcal{F}(\mathcal{R})\} \end{aligned}$$

- **Corollary** $\sim\ =\ \mathit{gfp}(\mathcal{F})$

- A similar characterization, of course, holds for $\lesssim$ as well.

Outline

Bisimulation in CCS is a congruence

- **Definition** A relation R on a language is called *congruence* if
  - $\mathcal{R}$ is an equivalence relation (i.e. it is reflexive, symmetric, and transitive), and
  - $\mathcal{R}$ is preserved by all the operators of the language, namely if $P \mathcal{R} Q$ then $op(P, \vec{R}) \mathcal{R} op(P, \vec{R})$
- **Theorem** $\sim$ is a congruence relation

Bisimulation in CCS is a congruence

- **Definition** A relation R on a language is called *congruence* if
  - $\mathcal{R}$ is an equivalence relation (i.e. it is reflexive, symmetric, and transitive), and
  - $\mathcal{R}$ is preserved by all the operators of the language, namely if $P \mathcal{R} Q$ then $op(P, \vec{R}) \mathcal{R} op(P, \vec{R})$
- **Theorem** $\sim$ is a congruence relation

Exercises

- Complete the proof that bisimulation in CCS is a congruence
- Prove that if $P \lesssim Q$ then the traces of P are contained in the traces of Q
- Prove that if $P \sim Q$ then $P \lesssim Q$ and $Q \lesssim P$
- Prove that
    - $P + 0 \sim P$ and $P|0 \sim P$
    - $P + P \sim P$ but (in general) $P|P \not\sim P$
    - $P + Q \sim Q + P$ and $P|Q \sim Q|P$
    - $(P + Q) + R \sim P + (Q + R)$ and $(P|Q)|R \sim P|(Q|R)$