MPRI C.2.3 - Concurrency

Probabilistic models and applications
Lecture 2

Kostas Chatzikokolakis

Dec 12, 2011
Outline of the lectures

- Dec 5
- Dec 12
- Jan 9
- Jan 16
- Jan 23
Outline of the lectures

- The need for randomization
- Probabilistic automata
- Probabilistic bisimulation
- Probabilistic calculi
- Encoding of the pi-calculus into the asynchronous fragment
- Introduction to probabilistic model checking and PRISM
- Verification of anonymity protocols: Dining Cryptographers, Crowds
Exercise 1: The algorithm of Lehmann and Rabin assumes a fair scheduler.

- Why?

- Is it possible to have a probabilistic solution to the dining philosophers problem that does not depend on scheduler fairness?
Exercise 2: Give a solution of the dining philosophers problem (satisfying all constraints) in the $\pi$-calculus (or CCS). Hint: use mixed choice
And a probabilistic puzzle

- I select two real numbers in some arbitrary way
- I put them in two envelopes, you select one by tossing a fair coin
- You see the number and you have 2 options: keep it, or exchange it with the other envelope
- Your goal is to select the bigger number
- Is there any strategy that guarantees winning this game with pb higher than 1/2?
Probabilistic bisimulation

Probabilistic calculi
Probabilistic automata

\[ A = (S, q, A, D) \]

- \( S \): set of states (countable)
- \( q \in S \): initial state (or distribution on states)
- \( A \): set of actions
- \( D \subseteq S \times A \times \text{Disc}(S) \): transition relation

we write \( s \xrightarrow{a} \mu \) for \((s, a, \mu) \in D\)
Probability space of executions

- **Execution**: $\alpha = s_0a_1s_1a_2s_2 \ldots$
  such that for each $i$: $s_i \xrightarrow{a_{i+1}} \mu_i$ and $\mu_i(s_{i+1}) > 0$

- **exec***(A), exec(A)**: set of finite/all executions

- **Scheduler**: $\sigma : \text{exec}^\ast(A) \rightarrow D$
  such that $\sigma(\alpha) = (s, a, \mu)$ implies $lstate(\alpha) = s$

- **Measure $\mu_\sigma$ induced by a scheduler $\sigma$**:

$$\mu_\sigma(C_{s_0a_1s_1 \ldots a_ns_n}) = \prod_{i=1}^{n} \mu_i(s_i)$$

where $\sigma(s_0 \ldots a_is_i) = (s_i, a_{i+1}, \mu_{i+1})$, and zero for all other cones
Probabilistic automata

When can we say that two automata have the same behaviour?
Lifting relations to distributions

- Let $\sim \subseteq S \times T$ be a relation between the sets $S$, $T$
- Define $\sim_p \subseteq Disc(S) \times Disc(T)$
- $\pi \sim_p \rho$ iff there exists $\alpha \in Disc(S \times T)$ such that:
  - $\alpha(s, T) = \pi(s)$ for each $s \in S$
  - $\alpha(S, t) = \rho(t)$ for each $t \in T$
  - $\alpha(s, t) = 0$ if $s \not\sim t$
Lifting equivalence relations to distributions

Let \( \sim \subseteq S \times S \) be an equivalence relation on \( S \).

Then \( \pi \sim_p \rho \) iff \( \pi([s]) = \rho([s]) \) for all equivalence classes \([s]\) of \( \sim \).
Probabilistic bisimulation

A relation $\mathcal{R} \subseteq S \times S$ is a strong probabilistic bisimulation iff for all $s_1, s_2 \in S$ and for all $a \in A$

- if $s_1 \xrightarrow{a} \mu_1$ then $\exists \mu_2$ such that $s_2 \xrightarrow{a} \mu_2$ and $\mu_1 \mathcal{R} \mu_2$,

- if $s_2 \xrightarrow{a} \mu_2$ then $\exists \mu_1$ such that $s_1 \xrightarrow{a} \mu_1$ and $\mu_1 \mathcal{R} \mu_2$.

We write $s_1 \sim s_2$ if there is a strong bisimulation that relates them.
Example
Example

With a probabilistic choice:

Exercise: show that this is a generalization of traditional bisimulation
These processes can be made bisimilar if we allow “composed” transitions.

Exercise: define such a notion of bisimulation
Weak probabilistic bisimulation

- We include a silent action $\tau$
- $s \xrightarrow{a} \mu \quad \mu \in Disc(S) \text{ iff}\n  \cdot \text{there exits a scheduler } \sigma \text{ s.t.}\n  \cdot \text{s.t. } \mu(t) = \mu_\sigma(\tau^* a \tau^* t)\n  \cdot \text{where } \tau^* a \tau^* t \text{ is the union of all cones of the form}\n    s_1 \tau s_2 \tau \ldots s_n a \tau \ldots t$
A relation \( \mathcal{R} \subseteq S \times S \) is a weak probabilistic bisimulation iff for all \( s_1, s_2 \in S \) and for all \( a \in A \)

- if \( s_1 \xrightarrow{a} \mu_1 \) then \( \exists \mu_2 \) such that \( s_2 \xrightarrow{a} \mu_2 \) and \( \mu_1 \mathcal{R} \mu_2 \),

- if \( s_2 \xrightarrow{a} \mu_2 \) then \( \exists \mu_1 \) such that \( s_1 \xrightarrow{a} \mu_1 \) and \( \mu_1 \mathcal{R} \mu_2 \).

We write \( s_1 \approx s_2 \) if there is a weak prob bisimulation that relates them.
Outline

Probabilistic bisimulation

Probabilistic calculi
CCS with internal probabilistic choice

\[ \alpha ::= a \mid \bar{a} \mid \tau \quad \text{prefixes} \]
\[ P, Q ::= \quad \text{processes} \]
\[ \alpha.P \quad \text{prefix} \]
\[ | P \mid Q \quad \text{parallel} \]
\[ | P + Q \quad \text{nondeterministic choice} \]
\[ | \sum_i p_i P_i \quad \text{internal probabilistic choice} \]
\[ | (\nu a)P \quad \text{restriction} \]
\[ | !P \quad \text{replication} \]
\[ | 0 \quad \text{nil} \]
CCS with internal probabilistic choice

Operational semantics given by a probabilistic automaton

\[
\begin{align*}
\text{ACT} & : \quad \alpha.P \xrightarrow{\alpha} \delta(P) \\
\text{SUM1} & : \quad P \xrightarrow{\alpha} \mu \\
\quad & \Rightarrow P + Q \xrightarrow{\alpha} \mu \\
\text{PAR1} & : \quad P \xrightarrow{\alpha} \mu \\
\quad & \Rightarrow P \mid Q \xrightarrow{\alpha} \mu \mid Q \\
\text{RES} & : \quad P \xrightarrow{\alpha} \mu \\
\quad & \Rightarrow (\nu a)P \xrightarrow{\alpha} (\nu a)\mu \\
\text{SUM2} & : \quad Q \xrightarrow{\alpha} \mu \\
\quad & \Rightarrow P + Q \xrightarrow{\alpha} \mu \\
\text{PAR2} & : \quad Q \xrightarrow{\alpha} \mu \\
\quad & \Rightarrow P \mid Q \xrightarrow{\alpha} P \mid \mu
\end{align*}
\]
CCS with internal probabilistic choice

\[
\begin{align*}
\text{COM} & \quad P \xrightarrow{a} \delta(P') \quad Q \xrightarrow{\bar{a}} \delta(Q') \\
& \quad P \mid Q \xrightarrow{\tau} \delta(P' \mid Q') \\
\text{PROB} & \quad \sum_i p_i P_i \xrightarrow{\tau} \sum_i p_i \delta(P_i)
\end{align*}
\]
CCS with internal probabilistic choice

REP1

\[
P \xrightarrow{\alpha} \mu \\
\neg P \xrightarrow{\alpha} \mu | \neg P
\]

REP2

\[
P \xrightarrow{\alpha} \delta(P_1) \quad P \xrightarrow{\bar{\alpha}} \delta(P_2) \\
\neg P \xrightarrow{\tau} \delta(P_1 | P_2 | \neg P)
\]
Asynchronous $\pi$-calculus

$\pi$-calculus without output prefix (replaced by output action) and without choice (+)

$$\pi ::= x(y) \mid \tau$$

action prefixes (input, silent)

$x, y$ are channel names

$$P ::= O$$

inaction

prefix

$$\pi.P$$

output action

$$\bar{x}y$$

parallel

$$P \mid P$$

restriction, new name

$$(\nu x)P$$

replication

$$! P$$
Probabilistic asynchronous $\pi$-calculus

The input guarded choice is probabilistic.

The prefixes

$$\alpha ::= x(y) \mid \tau$$

input $\mid$ silent action

The processes

$$P ::= 0 \quad \text{inaction}$$
$$\sum_i p_i \alpha_i . P_i \quad \text{probabilistic choice}$$
$$\bar{xy} \quad \text{output}$$
$$P \mid P \quad \text{parallel}$$
$$(\nu x) P \quad \text{new name}$$
$$! P \quad \text{replication}$$

where $\sum_i p_i = 1$
Probabilistic asynchronous \( \pi \)-calculus

- To give semantics to this calculus we need generalized automata
- Transition relation: \( D \subseteq S \times Disc(A \times S) \)
- We write \( s\{ \frac{\alpha_i}{p_i} \rightarrow s_i \mid i \in I \} \) for \( (s, \mu) \in D \) with \( \mu(\alpha_i, s_i) = p_i \)
Operational semantics

Sum \[ \sum_i p_i \alpha_i . P_i \{ \frac{\alpha_i}{p_i} \rightarrow P_i \}_i \]

Res \[ P \{ \frac{\mu_i}{p_i} \rightarrow P_i \}_i \]
\[ \nu y P \{ \frac{\mu_i}{p'_i} \rightarrow \nu y P_i \}_i : y \not\in fn(\mu_i) \]

Out \[ \bar{xy} \{ \frac{\bar{xy}}{1} \rightarrow 0 \} \]

\[ \exists i. y \not\in fn(\mu_i) \text{ and } \forall i. p'_i = p_i / \sum_{j : y \not\in fn(\mu_j)} p_j \]
Operational semantics

**Open**

\[
P \{ \bar{xy} \rightarrow P' \} \quad \frac{\nu y P \{ \bar{xy} \rightarrow P' \}}{\nu y P} \quad x \neq y
\]

**Par**

\[
P \{ \mu_i \rightarrow P_i \} \quad \frac{P \mid Q \{ \mu_i \rightarrow P_i \} \mid Q \} \quad P \mid Q \{ \mu_i \rightarrow P_i \} \mid Q \}
\]

**Com**

\[
P \{ \bar{xy} \rightarrow P' \} \quad Q \{ \mu_i \rightarrow Q_i \} \quad \frac{P \mid Q \{ \tau \rightarrow P' \mid Q_i[y/z_i] \} \cup \{ \mu_i \rightarrow P \mid Q \} \quad P \mid Q \quad P \mid Q \quad P \mid Q \}
\]
Operational semantics

**Close**

\[
\frac{P \{ \bar{x}(y) \rightarrow P' \}}{Q \{ \mu_i \rightarrow Q_i \}} \quad \frac{P \{ \mu_i \rightarrow Q_i \}}{Q \{ \mu_i \rightarrow Q_i \}} \\
\]

\[
P | Q \{ \tau \rightarrow y(P' | Q_i[y/z_i]) \}_{i: \mu_i = x(z_i)} \cup \{ \mu_i \rightarrow P | Q_i \}_{i: \mu_i \neq x(z_i)} \\
\]

**Cong**

\[
P \equiv P' \\
P' \{ \mu_i \rightarrow Q'_i \}_{i} \\
\forall i. Q'_i \equiv Q_i \\
\]

\[
P \{ \mu_i \rightarrow Q_i \}_{i} \\
\]