Switched Probabilistic I/O Automata

Ling Cheung[1] Nancy Lynch[2] Roberto Segala[3]
Frits Vaandrager[1]

[1]Nijmegen Institute for Computing and Information Sciences
University of Nijmegen, the Netherlands

[2]MIT Computer Science and Artificial Intelligence Laboratory, U.S.A.

[3]Dipartimento di Informatica, Università di Verona, Italy

ICTAC 2004, Guiyang, China

Outline

Outline

Outline

Outline

To NIII Colloquium Attendees:

Thank you all for coming to my talk!

For this talk . . .

- We need very little probability theory: *discrete distributions*. Examples:
  - fair coin: $\{\langle \text{Head}, \frac{1}{2}\rangle, \langle \text{Tail}, \frac{1}{2}\rangle\}$;
  - fair dice: $\{\langle i, \frac{1}{6}\rangle \mid 1 \leq i \leq 6\}$.

For this talk . . .

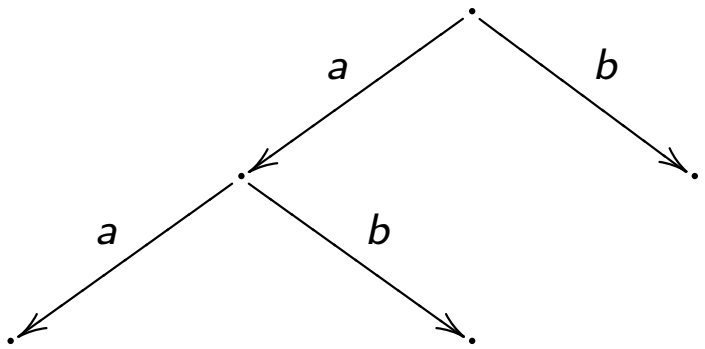
- We need very little probability theory: *discrete distributions*. Examples:
  - fair coin: $\{\langle \text{Head}, \frac{1}{2}\rangle, \langle \text{Tail}, \frac{1}{2}\rangle\}$;
  - fair dice: $\{\langle i, \frac{1}{6}\rangle \mid 1 \leq i \leq 6\}$.
- Underlying model: nondeterministic automata with asynchronous composition.
  (In our paper: input/output distinction, combination of synchronous and asynchronous compositions, etc.)

For this talk . . .

- We need very little probability theory: *discrete distributions*. Examples:
  - fair coin: $\{\langle \text{Head}, \frac{1}{2}\rangle, \langle \text{Tail}, \frac{1}{2}\rangle\}$;
  - fair dice: $\{\langle i, \frac{1}{6}\rangle \mid 1 \leq i \leq 6\}$.
- Underlying model: nondeterministic automata with asynchronous composition.
  (In our paper: input/output distinction, combination of synchronous and asynchronous compositions, etc.)
- Total order semantics: if both actions a and b occur, one must precede the other.

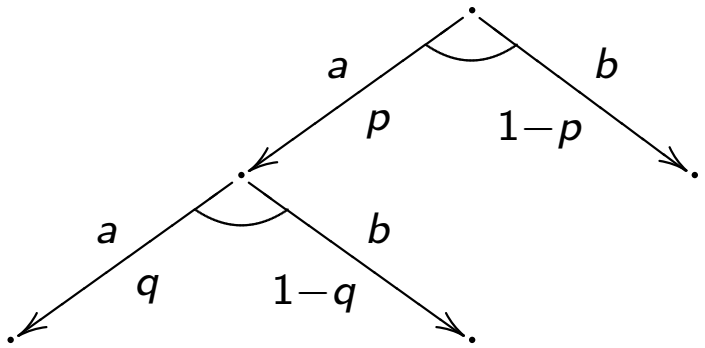
Schedulers and trace distributions

- *History-dependent*, *randomized* schedulers transform nondeterministic choices into probabilistic choices.



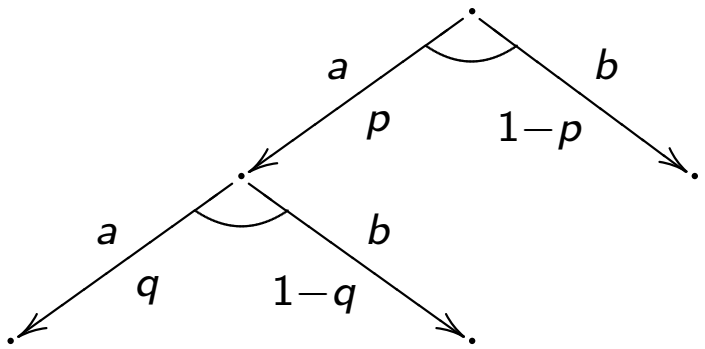
Schedulers and trace distributions

- *History-dependent*, *randomized* schedulers transform nondeterministic choices into probabilistic choices.



Schedulers and trace distributions

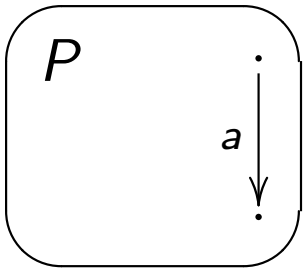
- *History-dependent, randomized* schedulers transform nondeterministic choices into probabilistic choices.

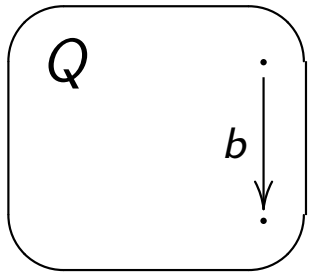


- Each scheduler induces a *trace distribution*: a discrete distributions on finite traces.

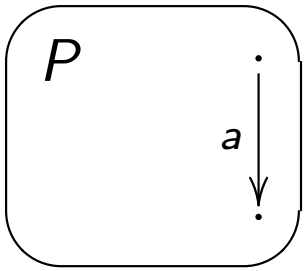
$$\{\langle aa,\ pq\rangle, \langle ab,\ p(1-q)\rangle, \langle b,\ 1-p\rangle\}$$

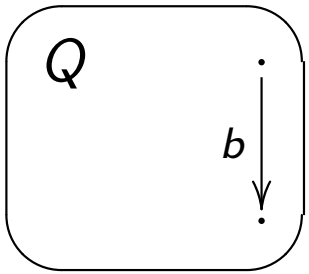
Nondeterministic parallel composition



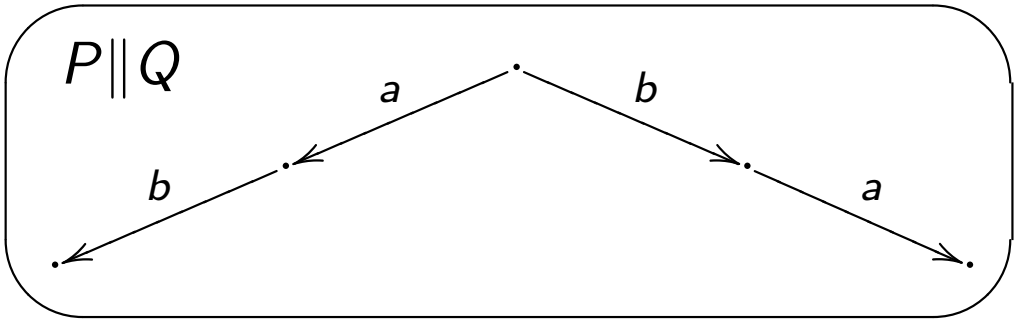


Nondeterministic parallel composition

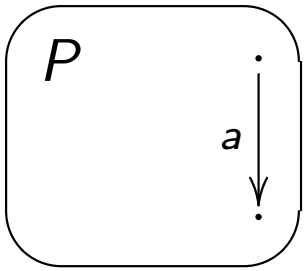


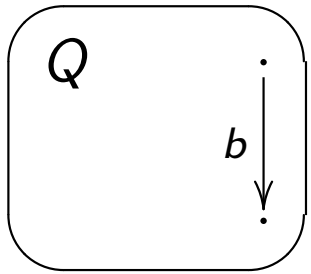


The *interleaving* axiom:

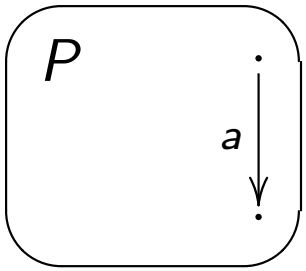


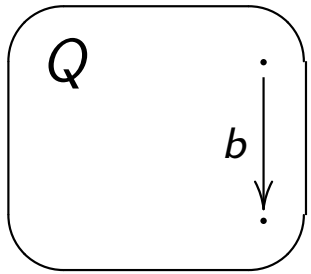
Probabilistic parallel composition





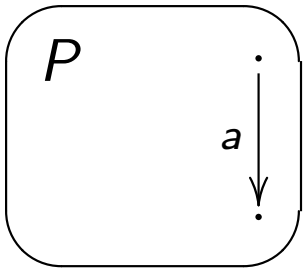
Probabilistic parallel composition

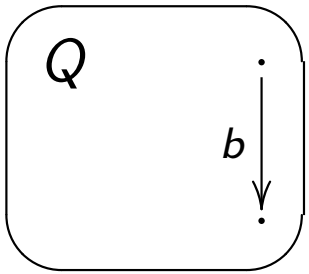




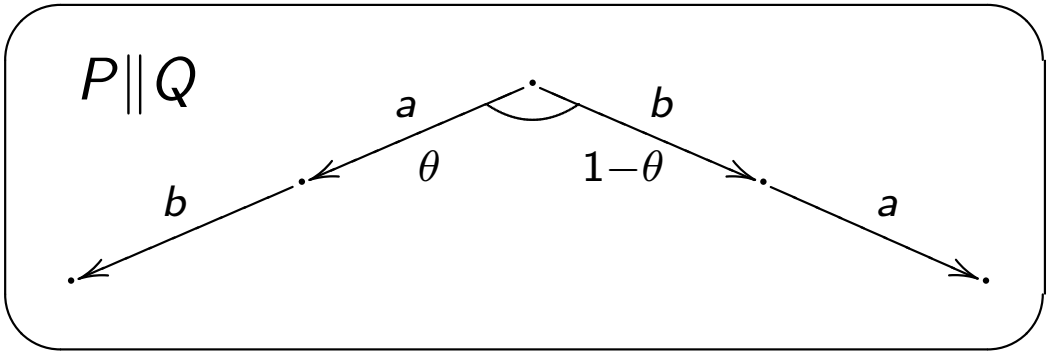
What is a *probabilistic* behavior of $P \| Q$?

Probabilistic parallel composition





What is a *probabilistic* behavior of $P \| Q$?
Quick answer: *bias factor* θ.

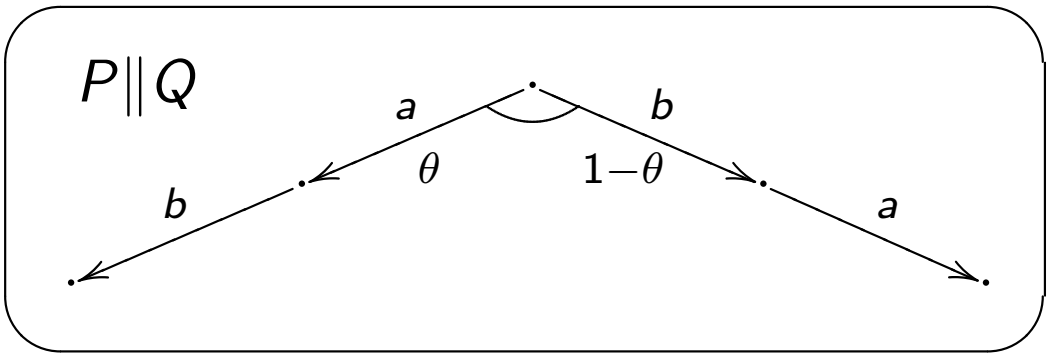


Probabilistic parallel composition

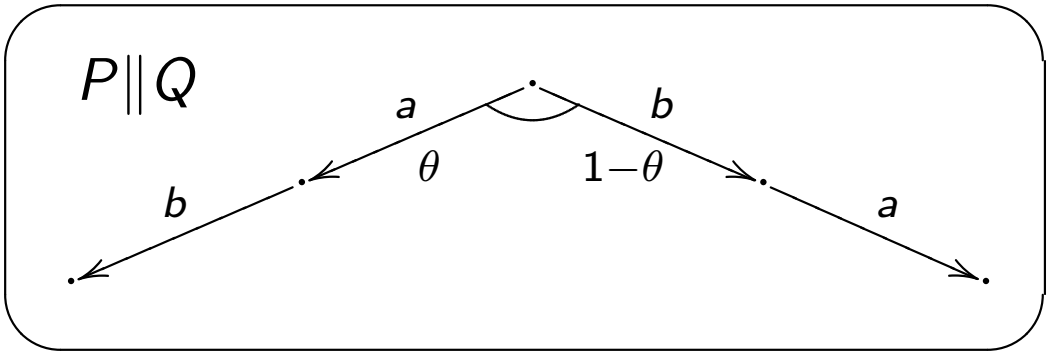
What is a *probabilistic* behavior of $P \| Q$?

Quick answer: *bias factor* θ.

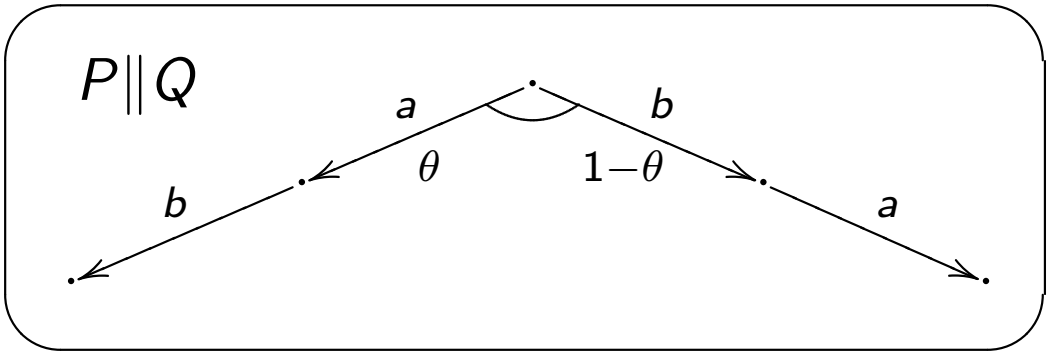
Imagine a coin-flipping daemon.



What is the value of θ?

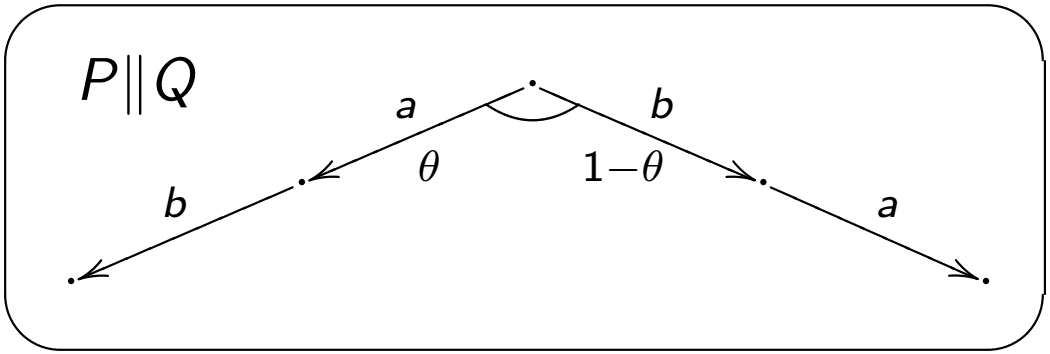


What is the value of θ?



Fixed θ: *parameterized* composition operator $\|^\theta$.

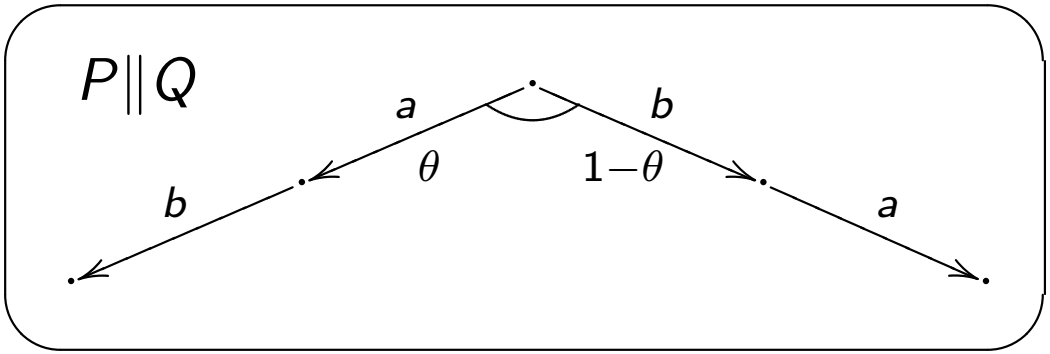
What is the value of θ?



Fixed θ: *parameterized* composition operator $\|^\theta$.

Limitations: static parameter, not commutative, not associative.

What is the value of θ?



Fixed θ: *parameterized* composition operator $\|^\theta$.
Limitations: static parameter, not commutative, not associative.

Variable θ:

- a supply of coins with different biases;
- imaginary daemon chooses a coin based on his knowledge.

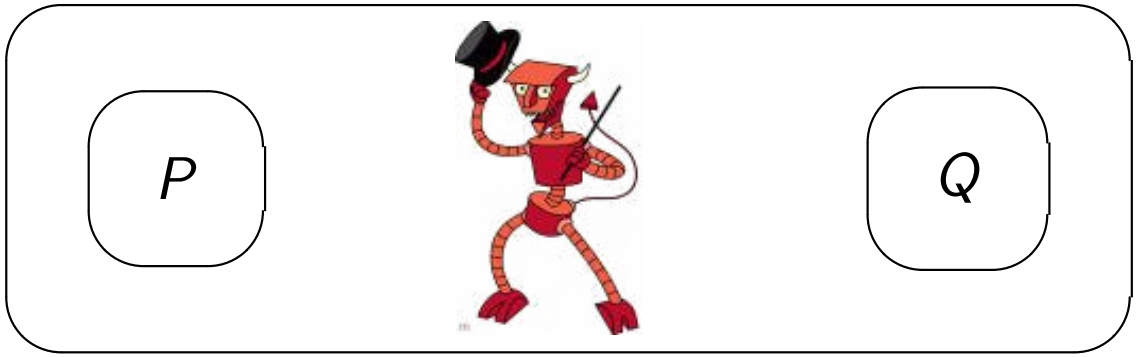
How much does the daemon know?

There are two scenarios:

How much does the daemon know?

There are two scenarios:

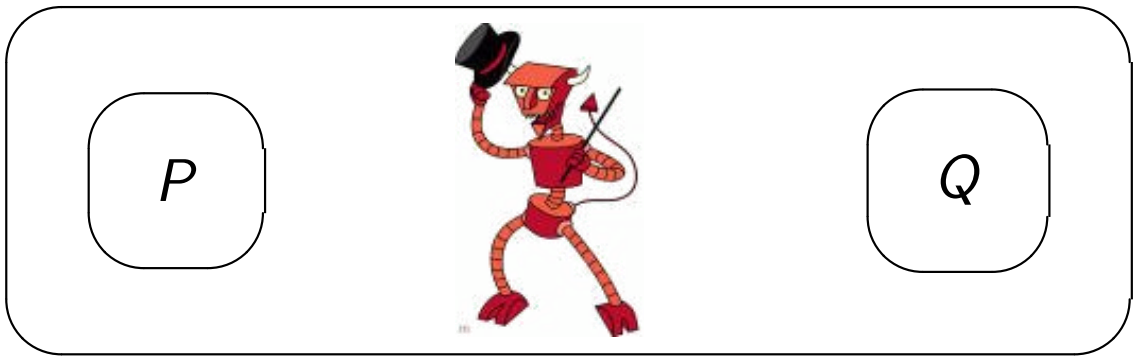
Scenario 1: *context-independent*



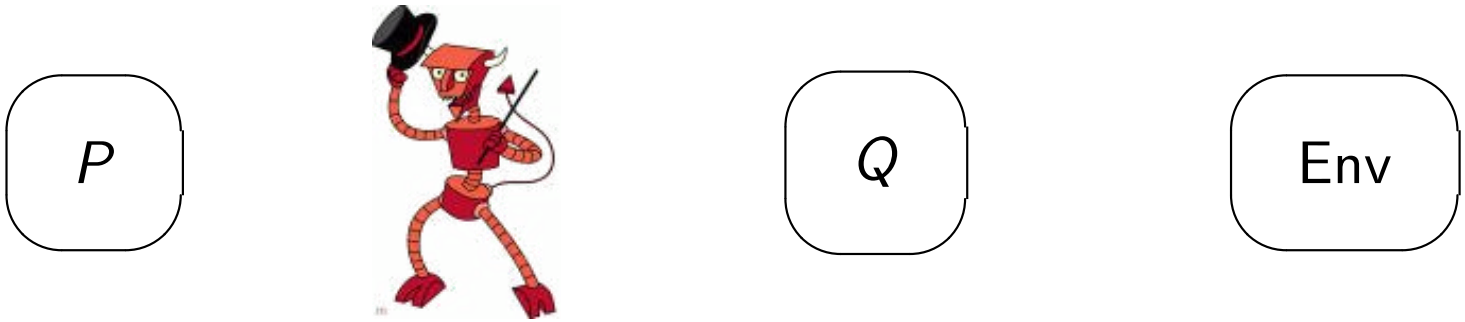
How much does the daemon know?

There are two scenarios:

Scenario 1: *context-independent*



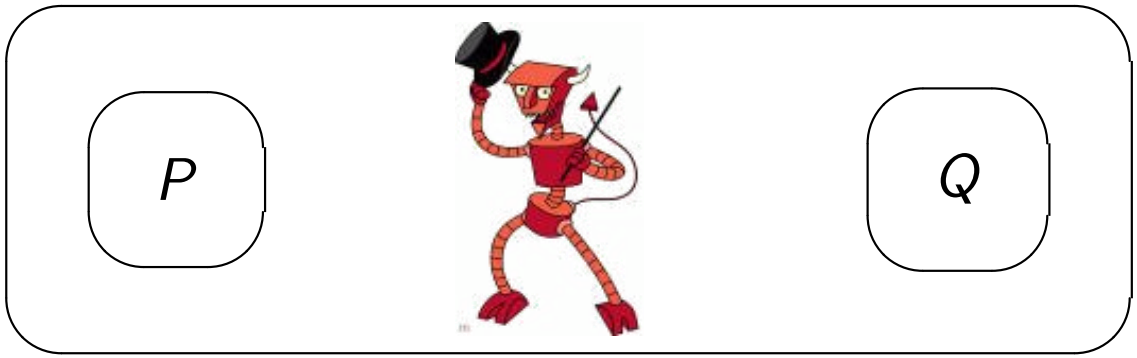
Scenario 2: *context-dependent*



Scenario 1: context-independent composition

How much does the daemon know?

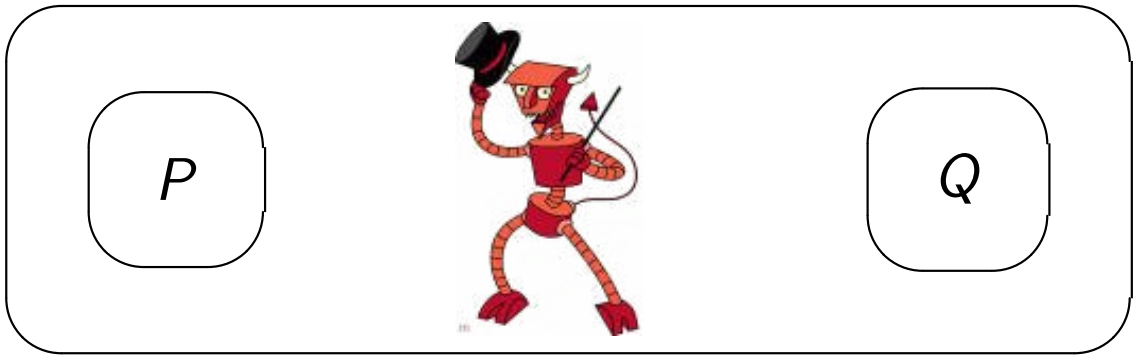
Daemon, P and Q all inside a big black box.



Scenario 1: context-independent composition

How much does the daemon know?

Daemon, P and Q all inside a big black box.

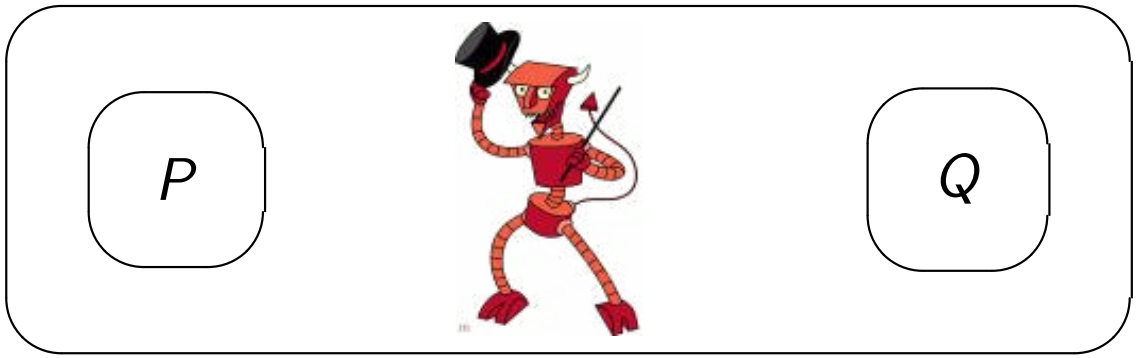


Daemon knows the histories of P and Q, but nothing about the outside world.

Scenario 1: context-independent composition

How much does the daemon know?

Daemon, P and Q all inside a big black box.

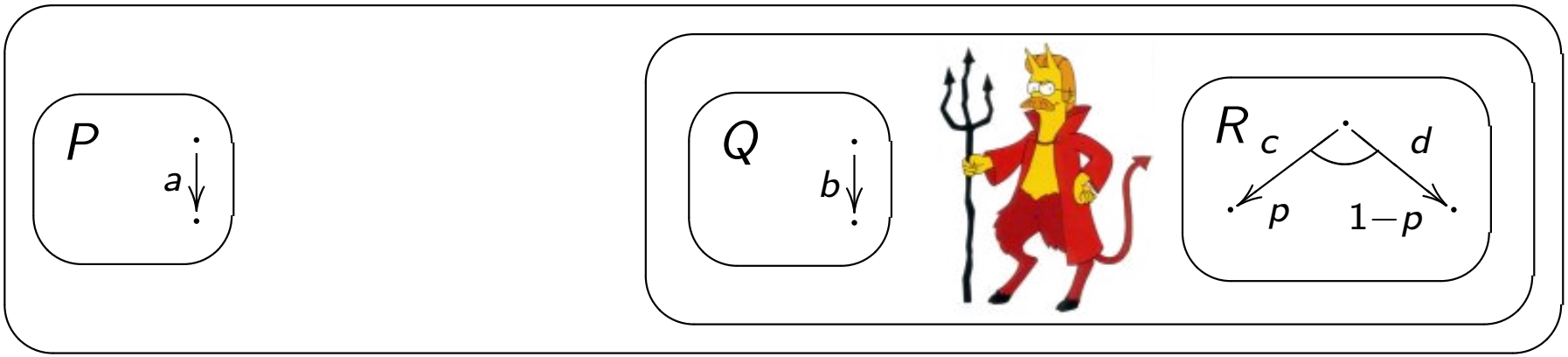


Daemon knows the histories of P and Q, but nothing about the outside world.

Problem: non-associativity.

Non-associativity: $P \| (Q \| R)$

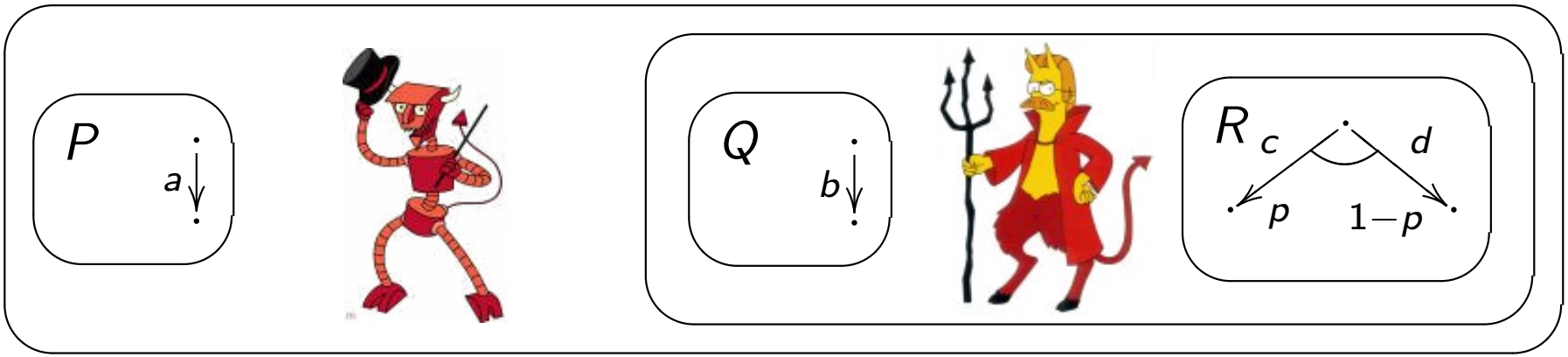
Context-independent composition



Inner daemon: $\langle R, 1 \rangle$.

Non-associativity: $P \| (Q \| R)$

Context-independent composition

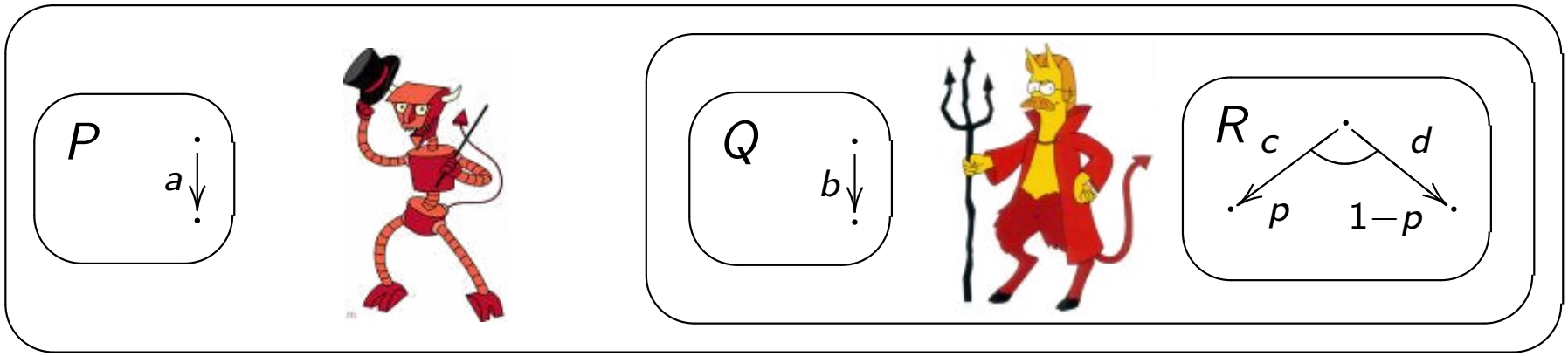


Inner daemon: $\langle R, 1 \rangle$.

Outer daemon: $\langle Q \| R, 1 \rangle$; if c, then $\langle P, 1 \rangle$, else $\langle Q \| R, 1 \rangle$.

Non-associativity: $P \| (Q \| R)$

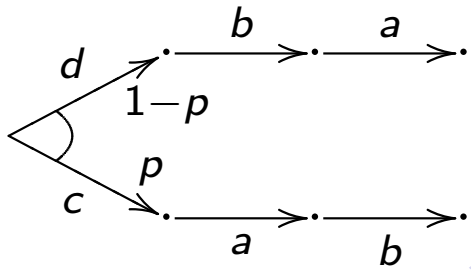
Context-independent composition



Inner daemon: $\langle R, 1 \rangle$.

Outer daemon: $\langle Q \| R, 1 \rangle$; if c, then $\langle P, 1 \rangle$, else $\langle Q \| R, 1 \rangle$.

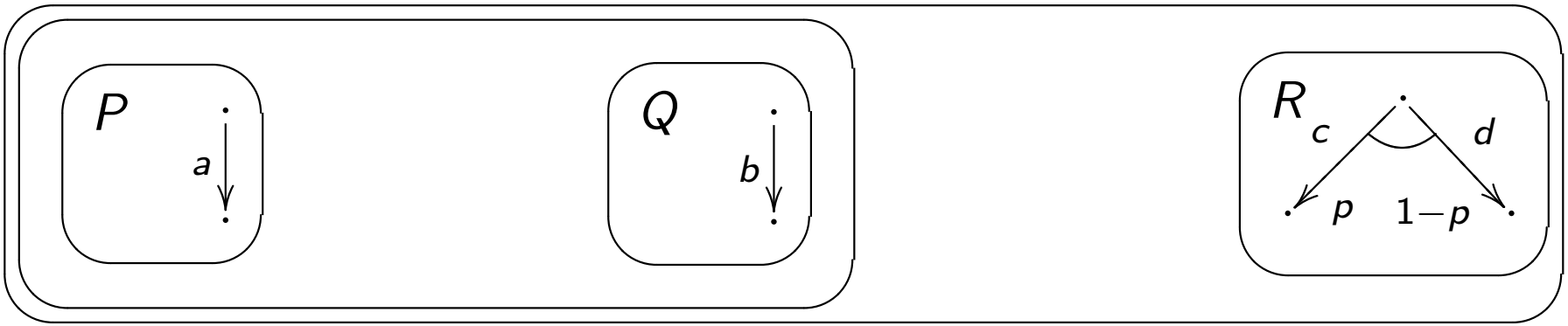
Result: $\{\langle cab, p \rangle, \langle dba, 1 - p \rangle\}$.



Non-associativity: $(P \| Q) \| R$

Context-independent composition

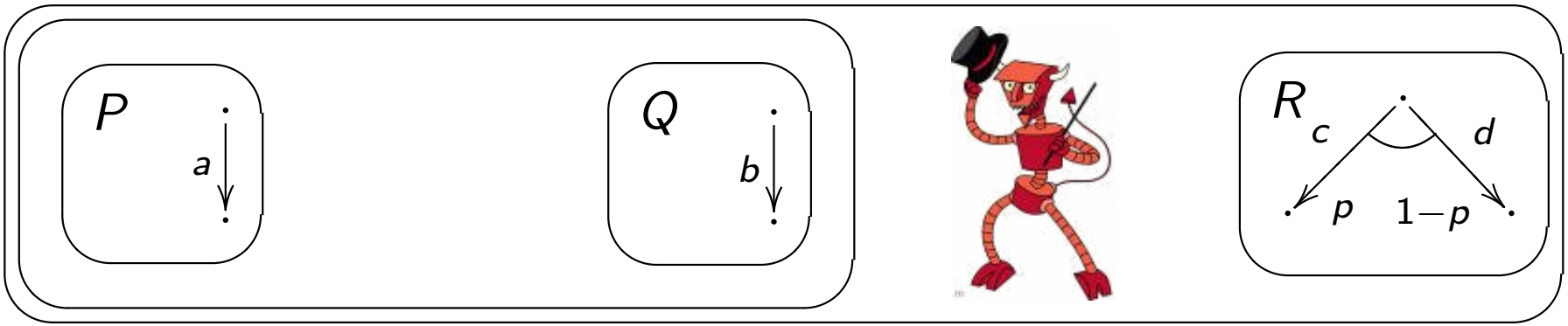
Claim: $\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$ not possible!



Non-associativity: $(P \| Q) \| R$

Context-independent composition

Claim: $\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$ not possible!

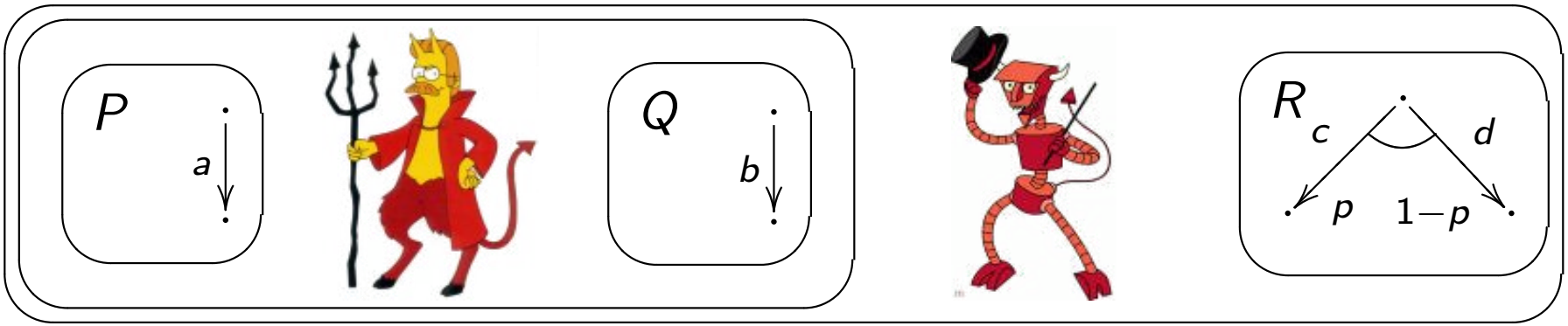


- *Outer* daemon: $\langle R,\ 1\rangle$.

Non-associativity: $(P \| Q) \| R$

Context-independent composition

Claim: $\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$ not possible!

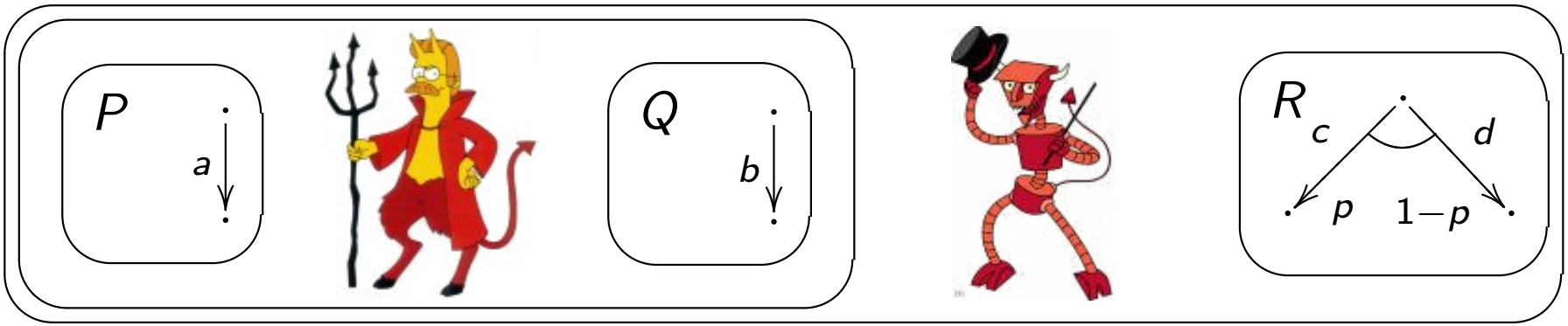


- *Outer* daemon: $\langle R,\ 1\rangle$.
- *Inner* daemon: $\{\langle P,\ q\rangle, \langle Q,\ 1-q\rangle\}$.

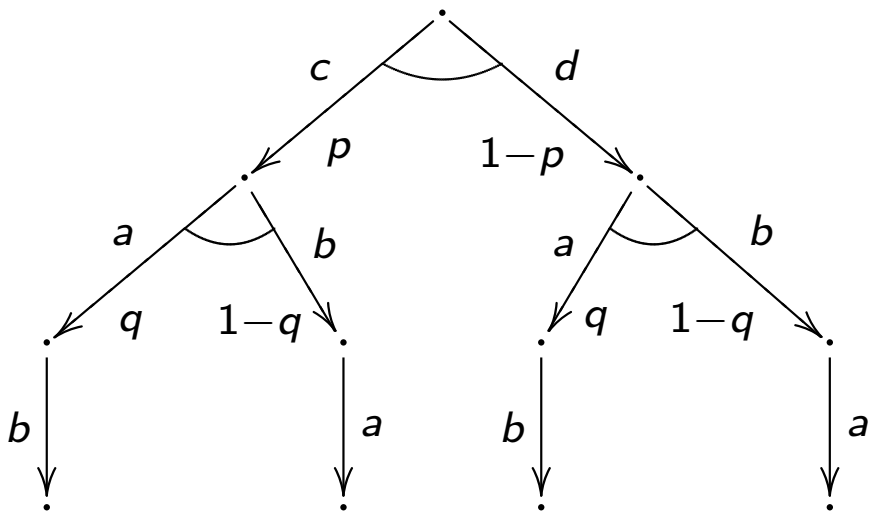
Non-associativity: $(P \| Q) \| R$

Context-independent composition

Claim: $\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$ not possible!



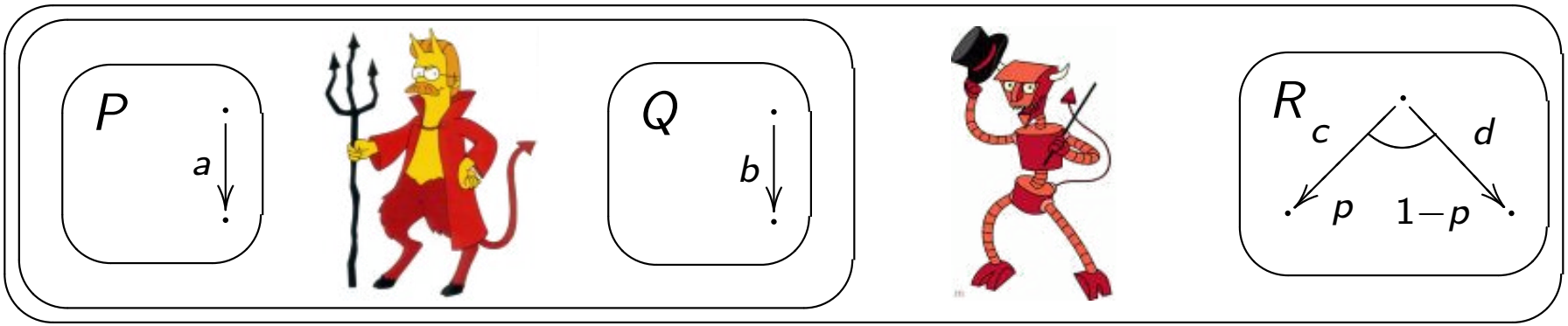
- *Outer* daemon: $\langle R,\ 1\rangle$.
- *Inner* daemon: $\{\langle P,\ q\rangle, \langle Q,\ 1-q\rangle\}$.



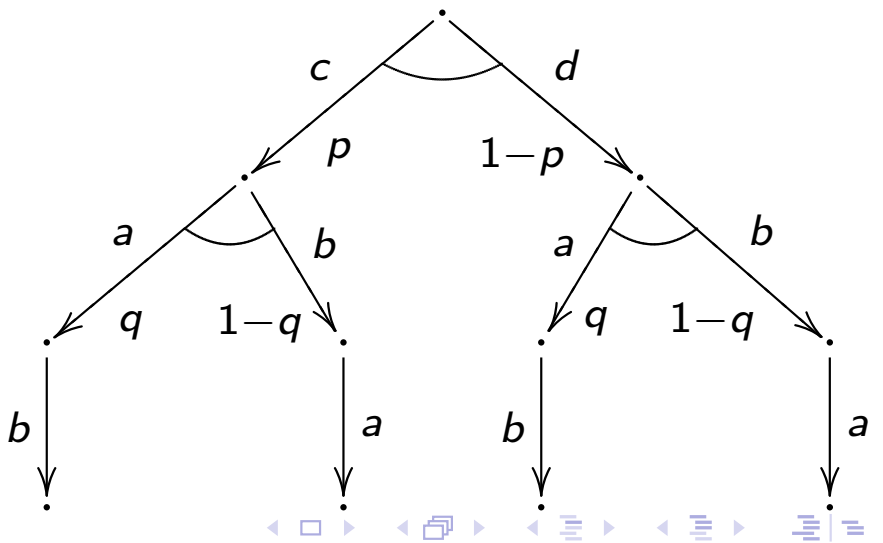
Non-associativity: $(P \| Q) \| R$

Context-independent composition

Claim: $\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$ not possible!



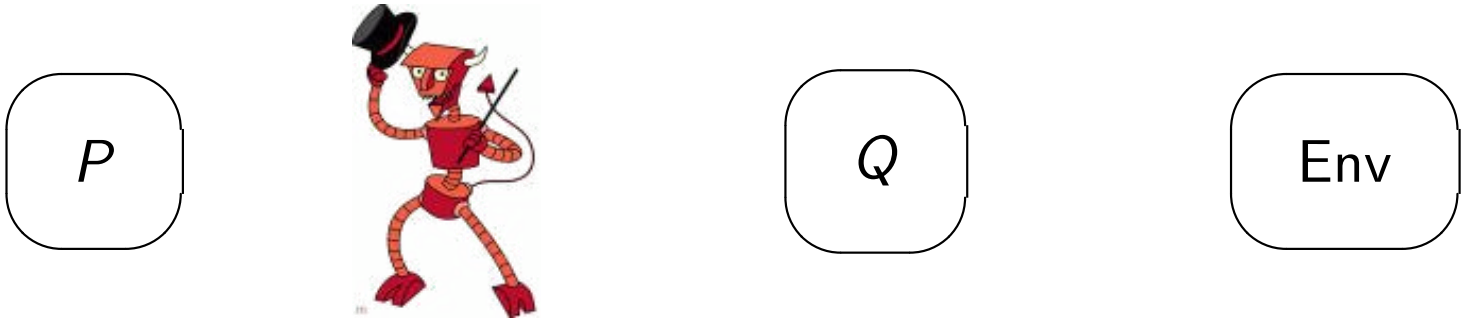
- *Outer* daemon: $\langle R,\ 1\rangle$.
- *Inner* daemon: $\{\langle P,\ q\rangle, \langle Q,\ 1-q\rangle\}$.
- Conclusion: inner daemon doesn't know enough.



Scenario 2: context-dependent composition

How much does the daemon know?

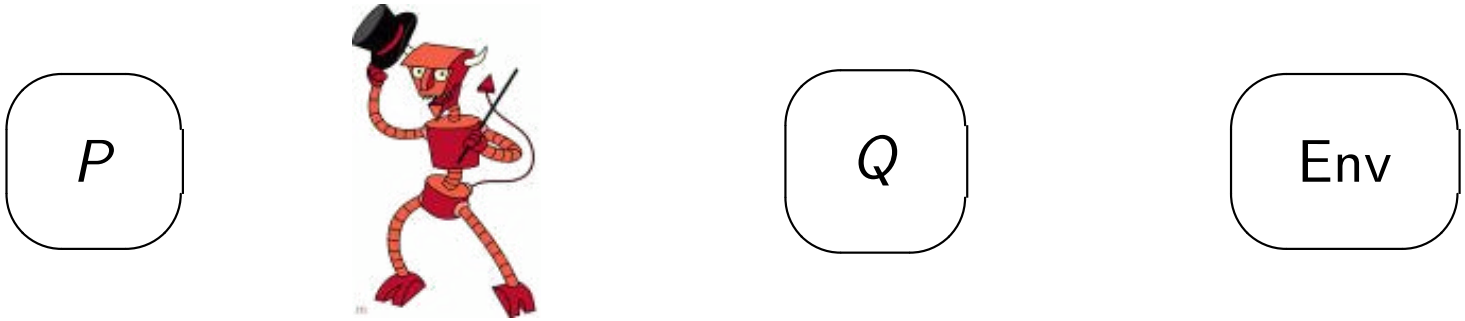
Daemon sees the outside world.



Scenario 2: context-dependent composition

How much does the daemon know?

Daemon sees the outside world.

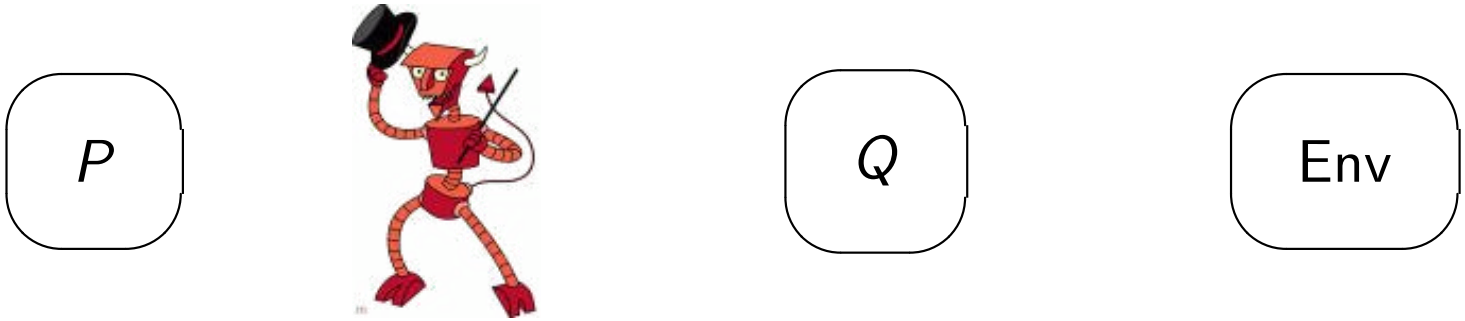


Daemon knows the histories of P, Q and Env.

Scenario 2: context-dependent composition

How much does the daemon know?

Daemon sees the outside world.



Daemon knows the histories of P, Q and Env.

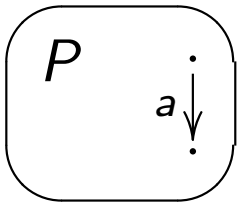
Problem: violation of the interleaving axiom!
I.e., there exists Env such that

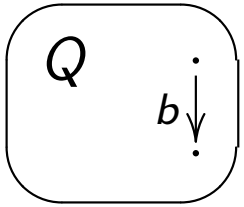
$$(a \| b) \| \text{Env} \not\cong (a.b + b.a) \| \text{Env}\,.$$

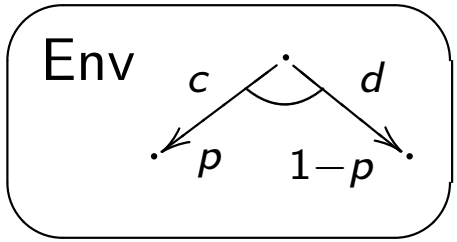
Non-interleaving semantics

Context-dependent composition

$(a \| b) \|$ Env:



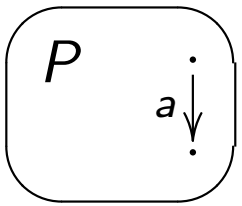


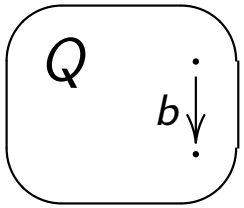


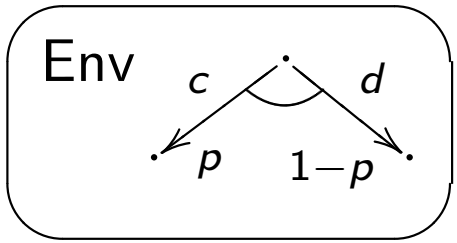
Non-interleaving semantics

Context-dependent composition

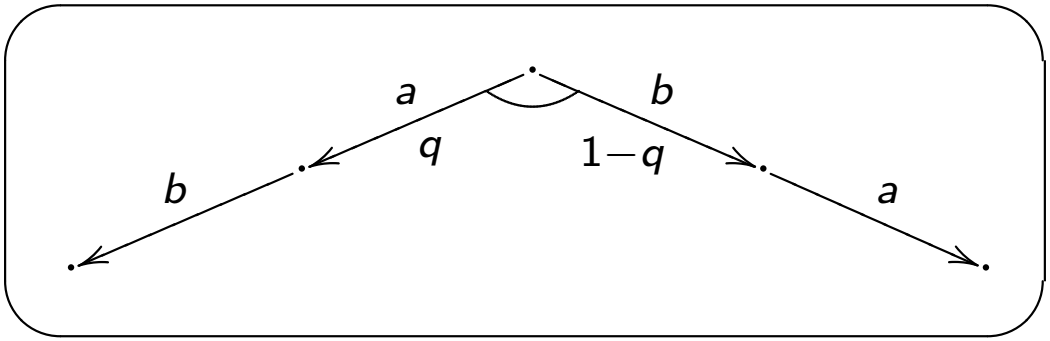
$(a \| b) \|$ Env:

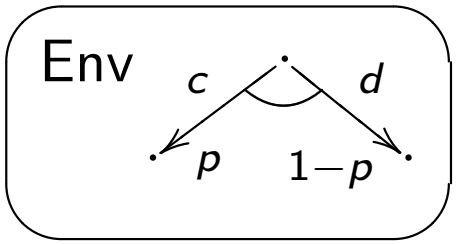






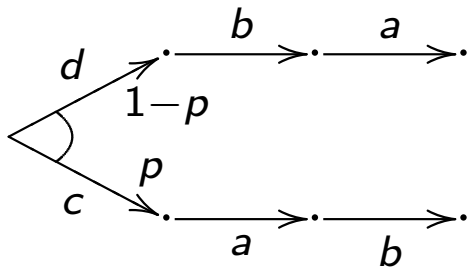
$(a.b + b.a) \|$ Env:





The (same) counterexample

Context-dependent composition



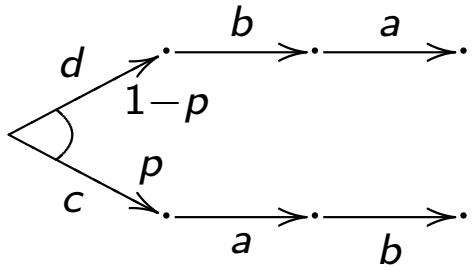
The trace distribution

$$\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$$

is possible in $(a \| b) \| \text{Env}$, but not in $(a.b + b.a) \| \text{Env}$.

The (same) counterexample

Context-dependent composition



The trace distribution

$$\{\langle cab,\ p\rangle, \langle dba,\ 1-p\rangle\}$$

is possible in $(a \| b) \| \text{Env}$, but not in $(a.b + b.a) \| \text{Env}$.

Conclusion: we have a non-interleaving, but total order semantics.

What's wrong?

Something is wrong with our understanding of parallel composition.

What's wrong?

Something is wrong with our understanding of parallel composition.

In context-independent composition, the problem shows up as non-associativity.

What's wrong?

Something is wrong with our understanding of parallel composition.

In context-independent composition, the problem shows up as non-associativity.

In context-dependent composition, the same problem leads to difference between $a \| b$ and $a.b + b.a$.

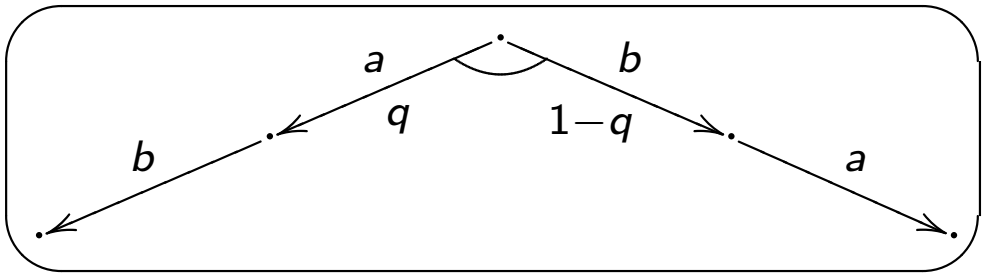
Two types of nondeterministic choices: global vs. local

Global choice: $a \| b$, resolved by a daemon.

Two types of nondeterministic choices: global vs. local

Global choice: $a \| b$, resolved by a daemon.

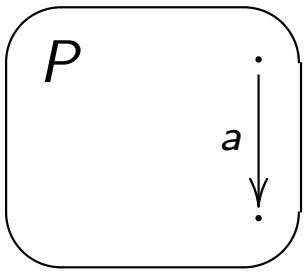
Local choice: $a.b + b.a$, resolved by a local scheduler.

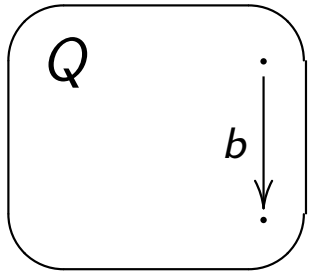


Behavior varies depending on the perspective!

Dissecting the problem, Part I: eliminate global choices.

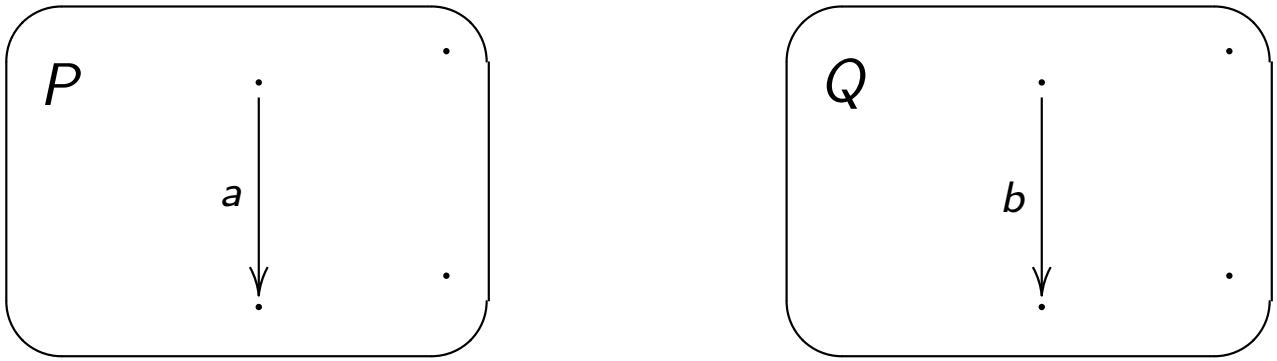
To better understand the problem, we developed the model of *Switched PIOA*.





Dissecting the problem, Part I: eliminate global choices.

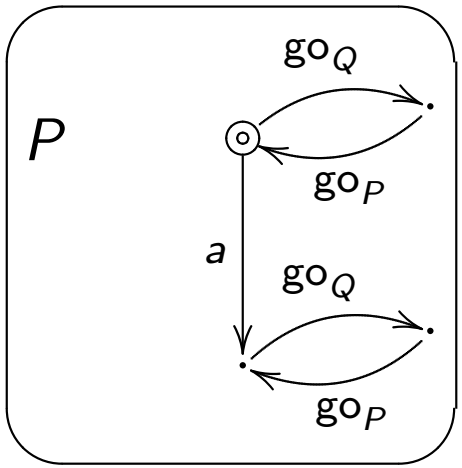
To better understand the problem, we developed the model of *Switched PIOA*.

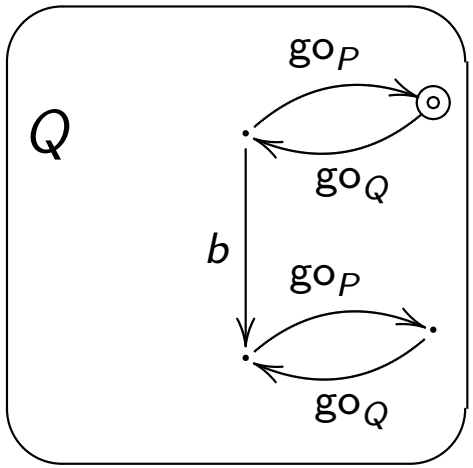


- active states (foreground) vs. inactive states (background);

Dissecting the problem, Part I: eliminate global choices.

To better understand the problem, we developed the model of *Switched PIOA*.

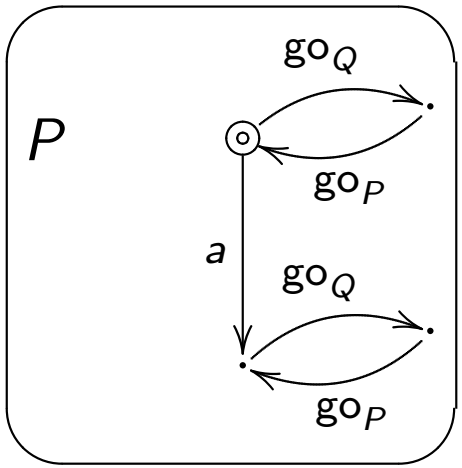


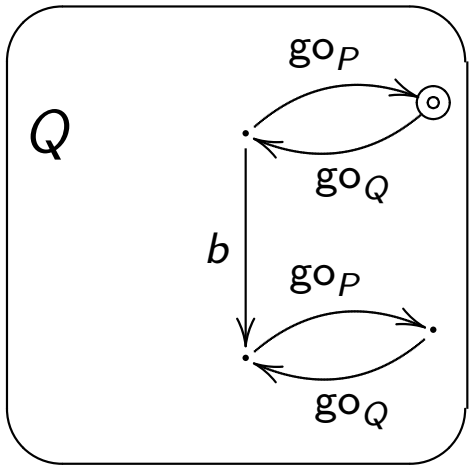


- active states (foreground) vs. inactive states (background);
- control exchange via special actions (e.g. go$_P$, go$_Q$);

Dissecting the problem, Part I: eliminate global choices.

To better understand the problem, we developed the model of *Switched PIOA*.





- active states (foreground) vs. inactive states (background);
- control exchange via special actions (e.g. go$_P$, go$_Q$);

Every decision is made locally, so no more daemons.

Due to the absence of global choices . . .

Parallel composition in Switched PIOA:

- easy to define;

Due to the absence of global choices . . .

Parallel composition in Switched PIOA:

- easy to define;
- commutative and associative;

Due to the absence of global choices . . .

Parallel composition in Switched PIOA:

- easy to define;
- commutative and associative;
- deep/semantic compositionality of trace distribution semantics;

Due to the absence of global choices . . .

Parallel composition in Switched PIOA:

- easy to define;
- commutative and associative;
- deep/semantic compositionality of trace distribution semantics;

That's all very nice, but parallel processes don't really exchange control . . .

Dissecting the problem, Part II: reintroduce global choices.

- Control-exchange should not be taken semantically.

Dissecting the problem, Part II: reintroduce global choices.

- Control-exchange should not be taken semantically.
- Switch PIOA is an implementation tool for various composition operators.

Dissecting the problem, Part II: reintroduce global choices.

- Control-exchange should not be taken semantically.
- Switch PIOA is an implementation tool for various composition operators.

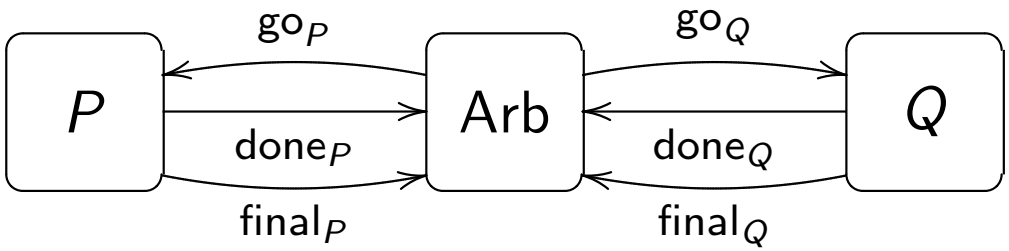
  Examples:
  - fixed bias factor θ;

Dissecting the problem, Part II: reintroduce global choices.

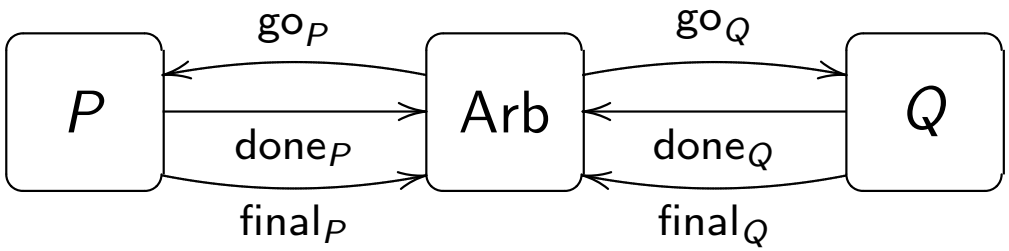
- Control-exchange should not be taken semantically.
- Switch PIOA is an implementation tool for various composition operators.

  Examples:
  - fixed bias factor θ;
  - context-independent.

Implementing biased composition

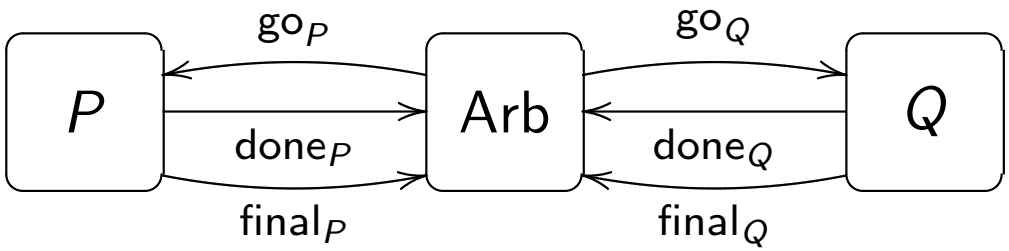


Implementing biased composition



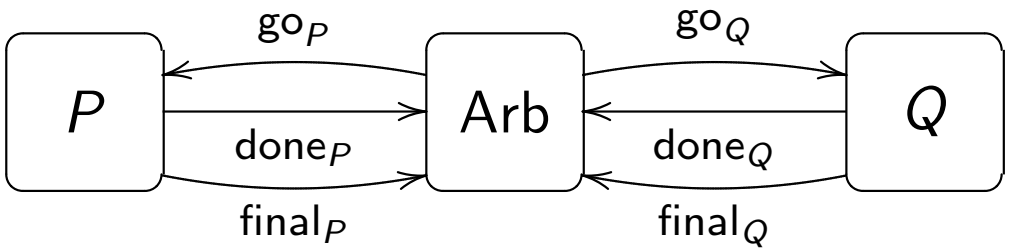
- Local schedulers: always return control after one local move.

Implementing biased composition



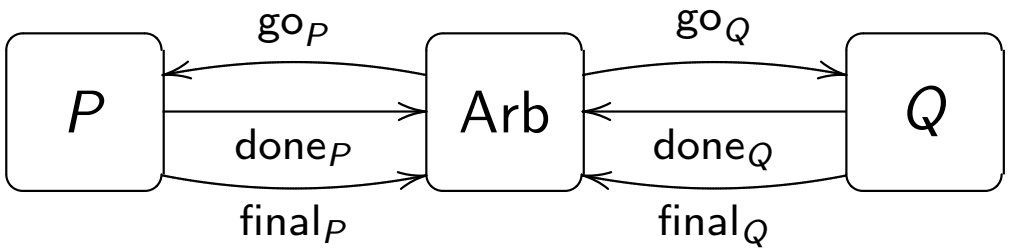
- Local schedulers: always return control after one local move.
- Arbiter:
  - usually schedule $\{\langle \text{go}_P, \theta \rangle, \langle \text{go}_Q, 1-\theta \rangle\}$;

Implementing biased composition



- Local schedulers: always return control after one local move.
- Arbiter:
  - usually schedule $\{\langle go_P, \theta\rangle, \langle go_Q, 1-\theta\rangle\}$;
  - if $final_P$ then $\langle go_Q, 1\rangle$ and vice versa.

Implementing biased composition

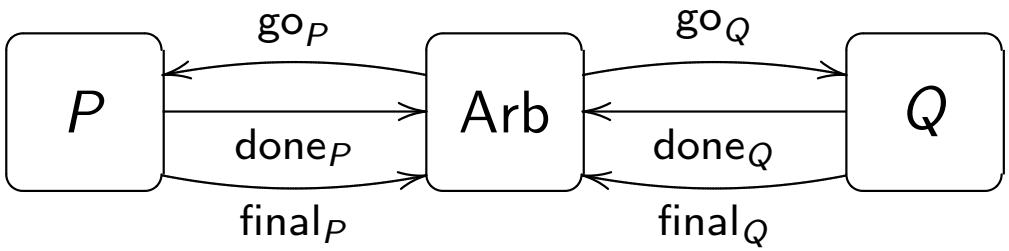


- Local schedulers: always return control after one local move.
- Arbiter:
  - usually schedule $\{\langle go_P, \theta\rangle, \langle go_Q, 1-\theta\rangle\}$;
  - if $final_P$ then $\langle go_Q, 1\rangle$ and vice versa.

Examples:

- $\langle go_P\,.\ a.\ final_P\,.\ go_Q\,.\ b.\ final_Q, \theta\rangle$;

Implementing biased composition

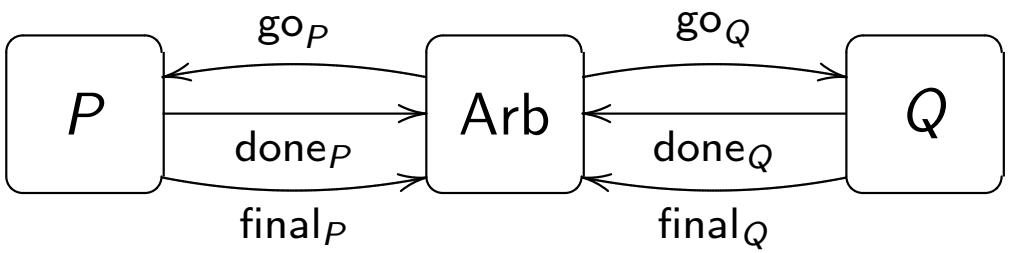


- Local schedulers: always return control after one local move.
- Arbiter:
  - usually schedule $\{\langle go_P, \theta\rangle, \langle go_Q, 1-\theta\rangle\}$;
  - if $final_P$ then $\langle go_Q, 1\rangle$ and vice versa.

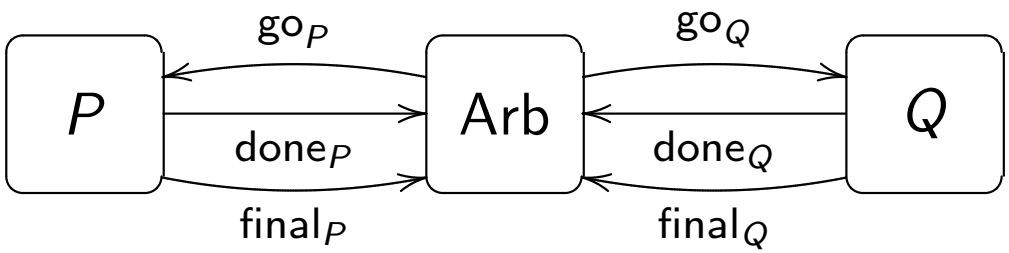
Examples:

- $\langle go_P\,.\,a.\ final_P\,.\ go_Q\,.\,b.\ final_Q, \theta\rangle$;
- $\langle go_Q\,.\,b.\ final_Q\,.\ go_P\,.\,a.\ final_P, 1-\theta\rangle$.

Implementing context-independent composition

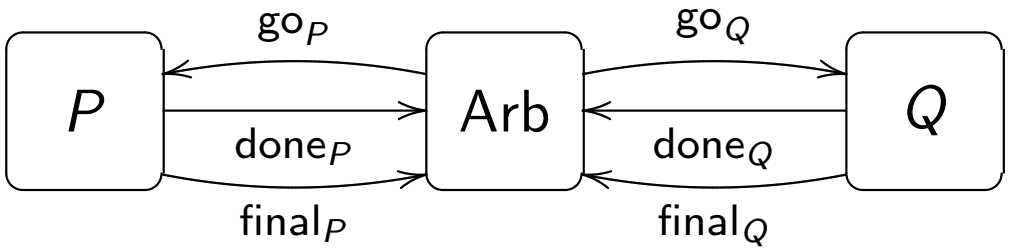


Implementing context-independent composition



- Local schedulers: no scheduling restrictions ("run to completion").

Implementing context-independent composition



- Local schedulers: no scheduling restrictions ("run to completion").
- Arbiter: if $final_P$ then $\langle go_Q, 1 \rangle$ and vice versa.

To summarize . . .

- Parallel composition is trickier than we thought.

To summarize . . .

- Parallel composition is trickier than we thought.
- Switched PIOA is a probabilistic model without global choices.

To summarize . . .

- Parallel composition is trickier than we thought.
- Switched PIOA is a probabilistic model without global choices.
- Parallel composition in Switched PIOA is well-behaved.

To summarize . . .

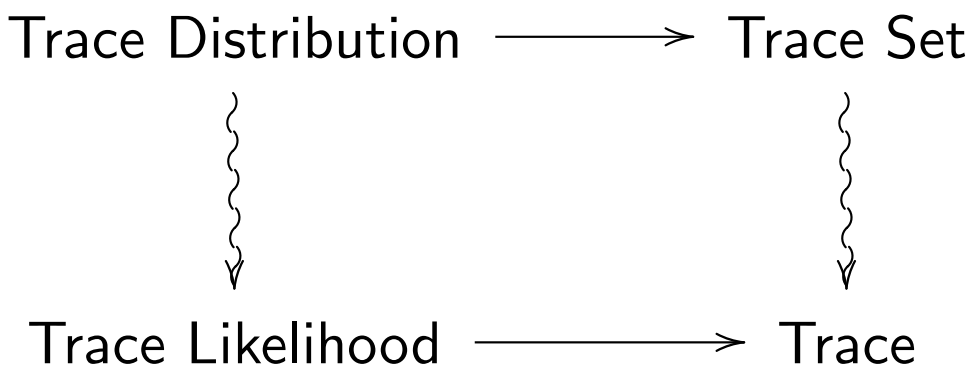
- Parallel composition is trickier than we thought.
- Switched PIOA is a probabilistic model without global choices.
- Parallel composition in Switched PIOA is well-behaved.
- Switched PIOA can be used to study various "real" parallel composition operators.

Future work

- Philosophical: is there a "most intuitive" parallel composition operator?

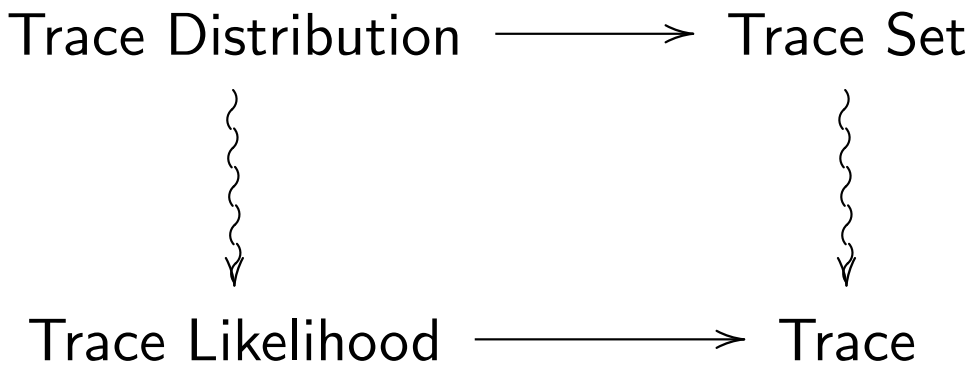
Future work

- Philosophical: is there a "most intuitive" parallel composition operator?
- Technical: "decomposing" trace distribution semantics.



Future work

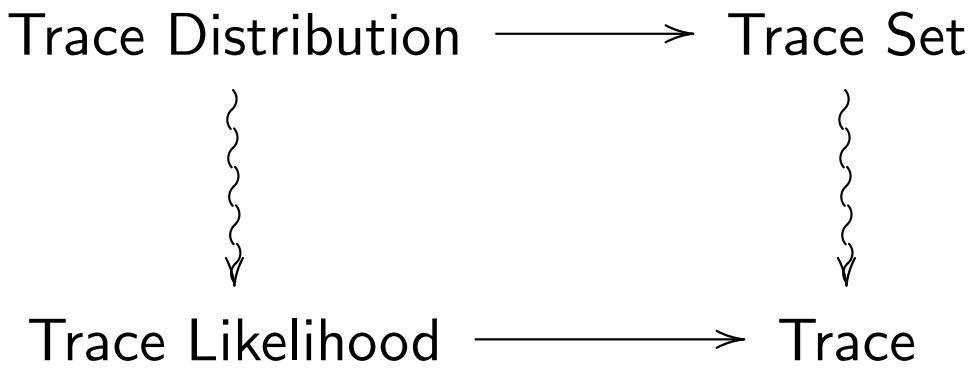
- Philosophical: is there a "most intuitive" parallel composition operator?
- Technical: "decomposing" trace distribution semantics.



- Practical: modeling communication and/or security protocols in Switched PIOA.

Future work

- Philosophical: is there a "most intuitive" parallel composition operator?
- Technical: "decomposing" trace distribution semantics.



- Practical: modeling communication and/or security protocols in Switched PIOA.

– End –

5 Appendix

- Trace Set Semantics
- Trace Likelihood Semantics
- Getting Stuck

Trace Set Semantics

Loosely speaking, a trace distribution is a discrete probability distribution over the set of finite traces.

Trace Set Semantics

Loosely speaking, a trace distribution is a discrete probability distribution over the set of finite traces.

To go from trace distribution to trace set, we forget probabilities by:

$$\text{DiscDistr}(\text{Traces}) \xrightarrow{\text{support}} \text{Powerset}(\text{Traces})$$

Trace Set Semantics

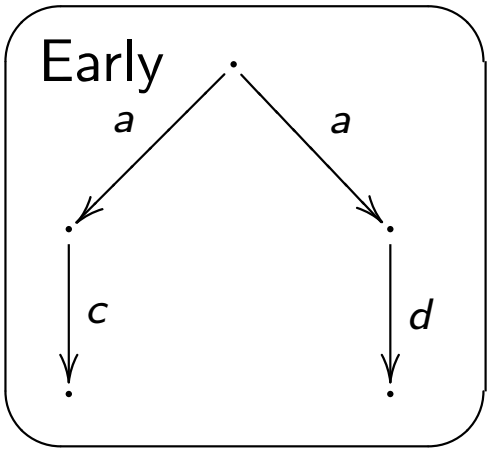
Loosely speaking, a trace distribution is a discrete probability distribution over the set of finite traces.

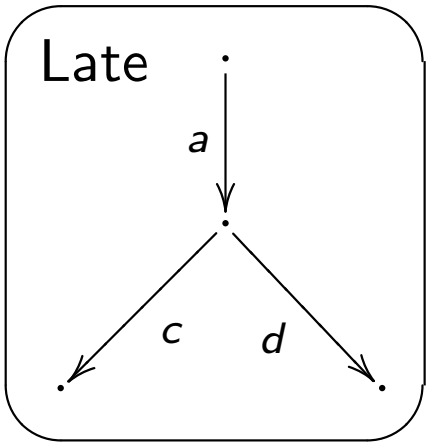
To go from trace distribution to trace set, we forget probabilities by:

$$\text{DiscDistr}(\text{Traces}) \xrightarrow{\text{support}} \text{Powerset}(\text{Traces})$$

That is, schedulers return sets of possible transitions, rather than discrete distributions over possible transitions.

Trace Set Semantics: Example

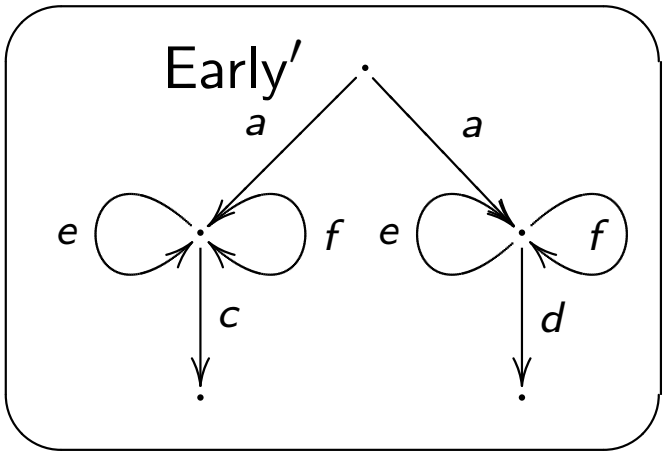


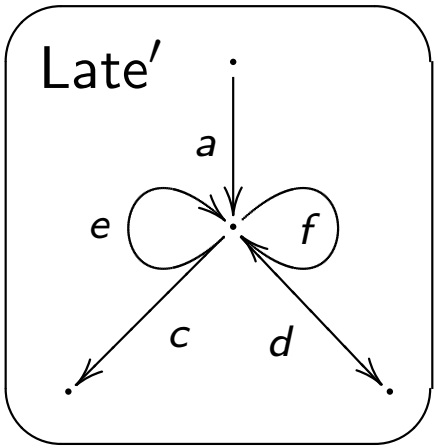


Equivalent in semantics: trace, trace set, trace distribution.

Not equivalent in semantics: bisimulation.

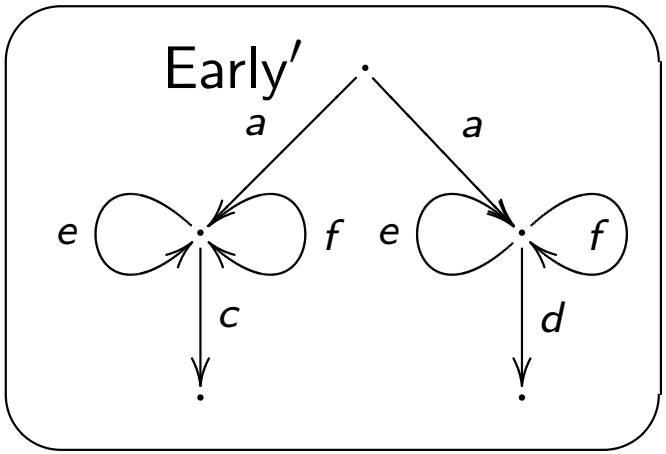
Trace Set Semantics: Example

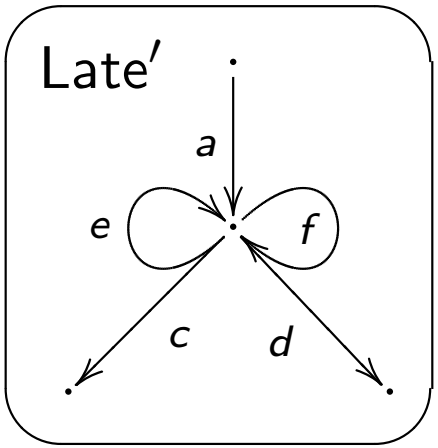




Add input e,f-loops.

Trace Set Semantics: Example

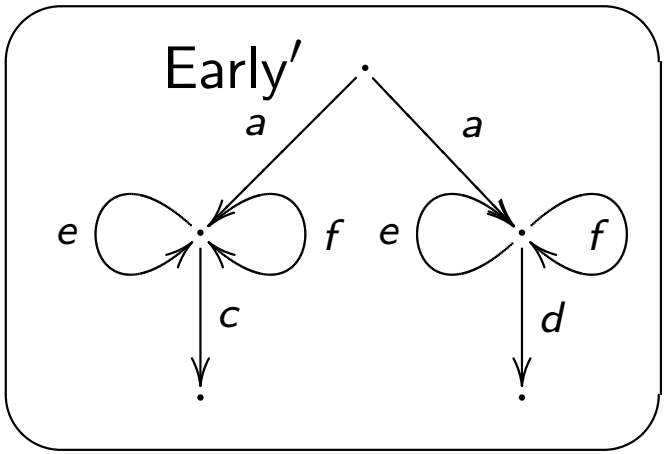


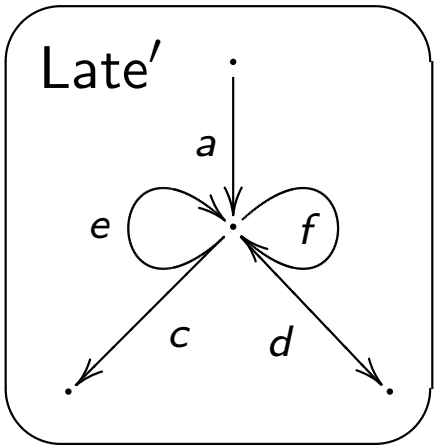


Add input e,f-loops.

Equivalent in semantics: trace.
Not equivalent in semantics: trace set, trace distribution, bisimulation.

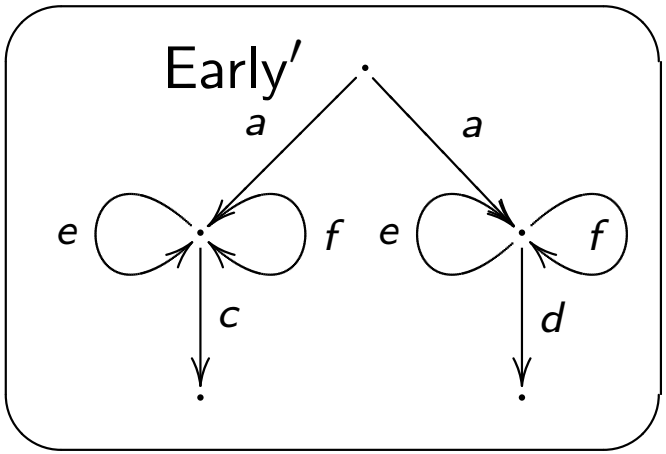
Trace Set Semantics: Example

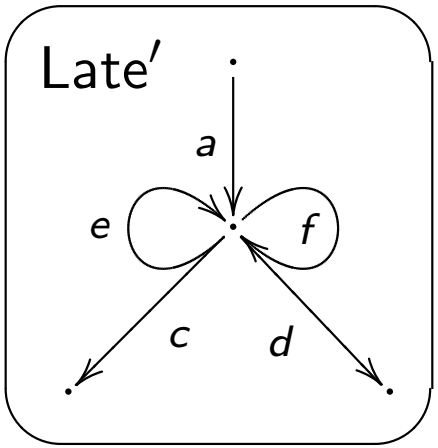




Trace set $\{aec, afd\}$ not possible in Early′.

Trace Set Semantics: Example

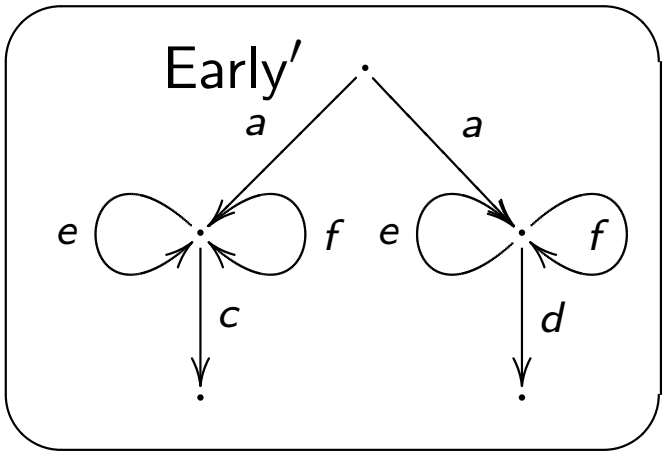


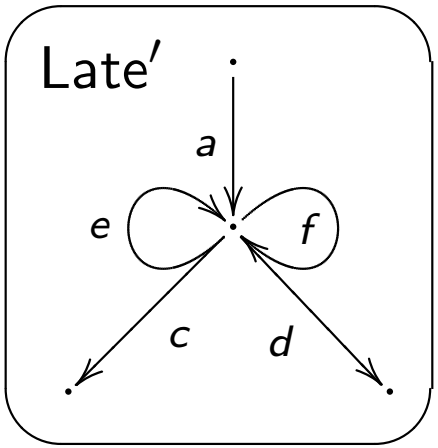


Trace set $\{aec, afd\}$ not possible in Early′.
Consider the case in which:

- Early′ chooses the left-hand branch; and

Trace Set Semantics: Example

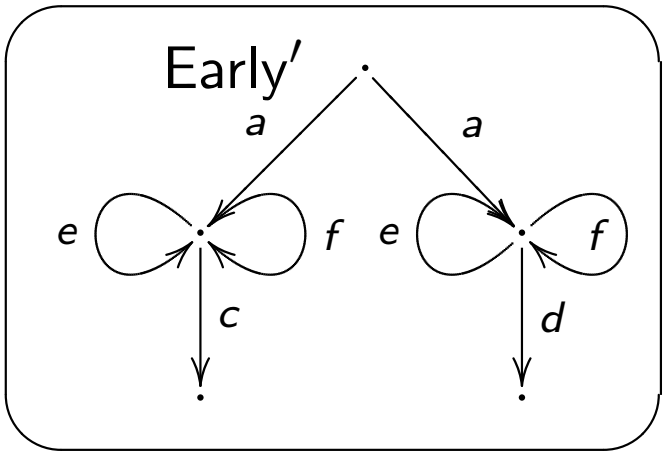


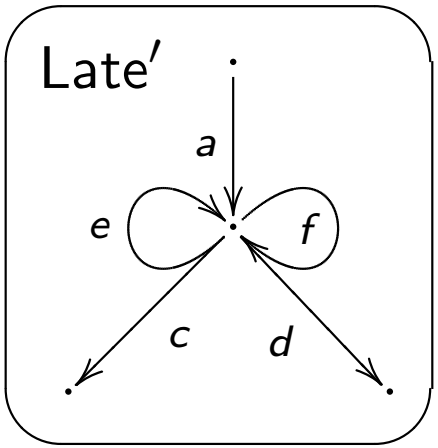


Trace set $\{aec, afd\}$ not possible in Early′.
Consider the case in which:

- Early′ chooses the left-hand branch; and
- environment performs f.

Trace Set Semantics: Example



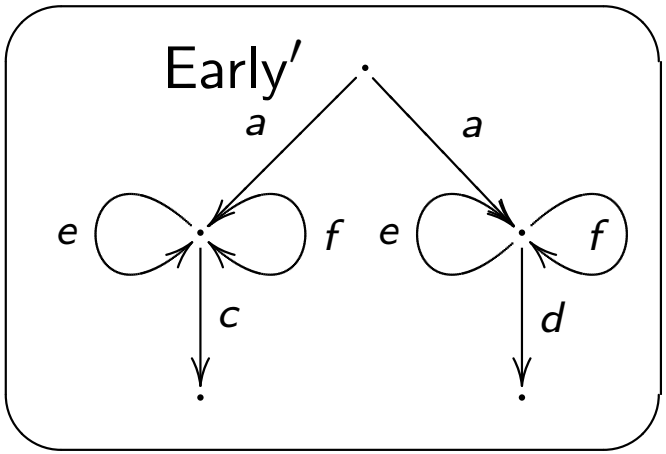


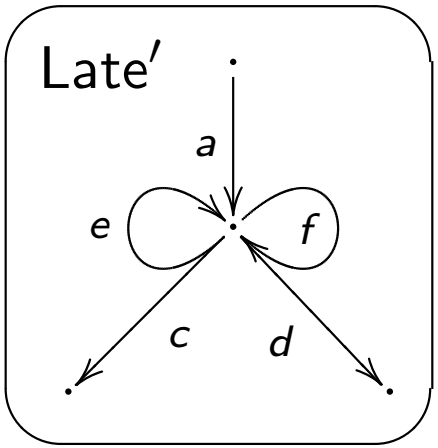
Trace set $\{aec, afd\}$ not possible in Early′.
Consider the case in which:

- Early′ chooses the left-hand branch; and
- environment performs f.

At this point, Early′ does not have the option to perform d.

Trace Set Semantics: Example





Trace set $\{aec, afd\}$ not possible in Early′.
Consider the case in which:

- Early′ chooses the left-hand branch; and
- environment performs f.

At this point, Early′ does not have the option to perform d.
Important: Early′ cannot choose between inputs e and f.

What's the lesson here?

What's the lesson here?

It's not about the numbers . . .

What's the lesson here?

It's not about the numbers . . .

Examples of "undesirable" properties of trace distribution semantics can be reproduced in trace set semantics.

What's the lesson here?

It's not about the numbers . . .

Examples of "undesirable" properties of trace distribution semantics can be reproduced in trace set semantics.

Key: each trace distribution contains a collection of traces, rather than a single trace.

In some cases, this allows us to observe branching structure.

An Alternative: Trace Likelihood Semantics

Each behavior is represented by a pair $\langle \alpha, p \rangle$.

An Alternative: Trace Likelihood Semantics

Each behavior is represented by a pair $\langle \alpha, p \rangle$.

Intended meaning: under some scenario, trace α occurs with probability p.

An Alternative: Trace Likelihood Semantics

Each behavior is represented by a pair $\langle \alpha, p \rangle$.

Intended meaning: under some scenario, trace α occurs with probability p.

Difficulty: what is a possible scenario?

An Alternative: Trace Likelihood Semantics

Each behavior is represented by a pair $\langle \alpha, p \rangle$.

Intended meaning: under some scenario, trace α occurs with probability p.

Difficulty: what is a possible scenario?

Frequentist probabilities: prediction about a large number of experiments, not about a single experiment.

Getting Stuck . . .

my current strategy:

- stop thinking, start reading;

Getting Stuck . . .

my current strategy:

- stop thinking, start reading;
- do something concrete: modeling oblivious transfer.