# State Transition Systems

Luís Soares Barbosa




Algebraic and Coalgebraic Methods in Software Development

MAP-i, 23.X.2017

# Reactive systems

## Reactive system

system that computes by reacting to stimuli from its environment along its overall computation

## State vs behaviour

- in contrast to sequential systems whose meaning is defined by the results of finite computations, the behaviour of reactive systems is mainly determined by interaction and mobility of non-terminating processes, evolving concurrently.
- observation $\equiv$ interaction
- behaviour $\equiv$ a structured record of interactions

# Labelled Transition System

## A model

A LTS over a set $N$ of actions is a tuple $\langle S, N, \longrightarrow \rangle$ where

- $S = \{s_0, s_1, s_2, ...\}$ is a set of states
- $\longrightarrow \subseteq S \times N \times S$ is the transition relation, often given as an $N$-indexed family of binary relations

$$s \xrightarrow{a} s' \equiv \langle s', a, s \rangle \in \longrightarrow$$

## Actions

to be regarded as transition labels or names, abstracting some observable (e.g. action name, event, input/output data, etc)

# Labelled Transition System

## Morphism

A morphism relating two LTS over $N$, $\langle S, N, \longrightarrow \rangle$ and $\langle S', N, \longrightarrow' \rangle$, is a function $h : S \longrightarrow S'$ st

$$s \xrightarrow{a} s' \quad \Rightarrow \quad h\, s \xrightarrow{a}{}' \, h\, s'$$

morphisms preserve transitions

# Behavioural equivalence

Two LTS should be equivalent if they cannot be distinguished by interacting with them.

## Equality of functional behaviour

is not preserved by parallel composition: non compositional semantics, cf,

`x:=4; x:=x+1` and `x:=5`

## Graph isomorphism

is too strong (why?)

# Trace

## Definition

Let $T = \langle S, N, \longrightarrow \rangle$ be a labelled transition system. The set of traces $\mathsf{Tr}(s)$, for $s \in S$ is the minimal set satisfying

(1) $\epsilon \in \mathsf{Tr}(s)$

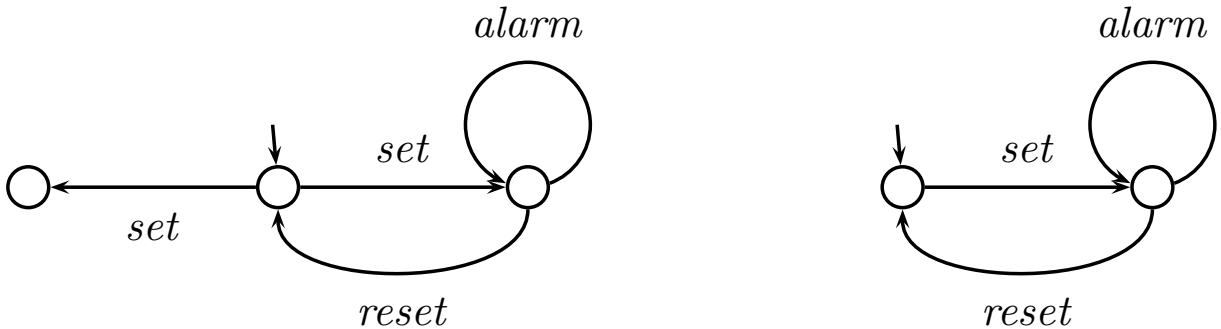
(2) $a\sigma \in \mathsf{Tr}(s) \Rightarrow \langle \exists\, s' : s' \in S : s \xrightarrow{a} s' \wedge \sigma \in \mathsf{Tr}(s') \rangle$

# Trace equivalence

## Definition

Two states $s, r$ are trace equivalent iff $\mathsf{Tr}(s) = \mathsf{Tr}(r)$
(i.e. if they can perform the same finite sequences of transitions)

## Example




Trace equivalence applies when one can neither interact with a system, nor distinguish a slow system from one that has come to a stand still.

# Simulation

the quest for a behavioural equality:
able to identify states that cannot be distinguished by any realistic form of observation

## Simulation

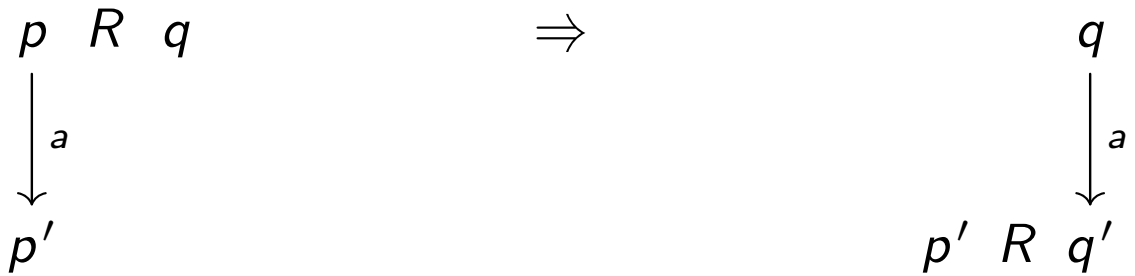
A state $q$ simulates another state $p$ if every transition from $q$ is corresponded by a transition from $p$ and this capacity is kept along the whole life of the system to which state space $q$ belongs to.

# Simulation

## Definition

Given $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$ over $N$, relation $R \subseteq S_1 \times S_2$ is a simulation iff, for all $\langle p, q \rangle \in R$ and $a \in N$,

$$p \xrightarrow{a}_1 p' \Rightarrow \langle \exists\ q' \ :\ q' \in S_2 :\ q \xrightarrow{a}_2 q' \wedge \langle p', q' \rangle \in R \rangle$$



# Similarity

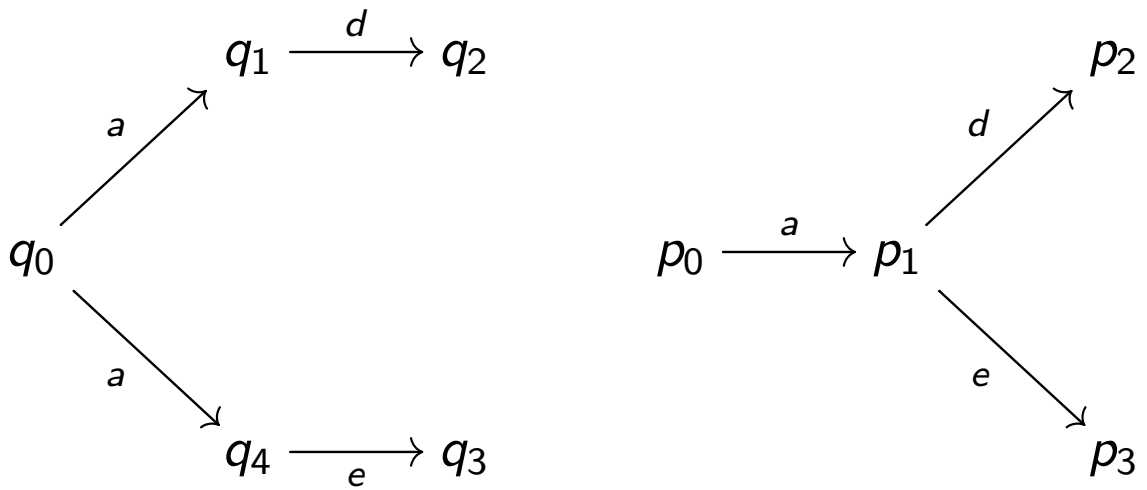
## Definition

$$p \lesssim q \equiv \langle \exists\ R\ ::\ R \text{ is a simulation and } \langle p, q \rangle \in R \rangle$$

## Lemma

The similarity relation is a preorder
(ie, reflexive and transitive)

# Example




$q_0 \lesssim p_0$ cf. $\{\langle q_0, p_0\rangle, \langle q_1, p_1\rangle, \langle q_4, p_1\rangle, \langle q_2, p_2\rangle, \langle q_3, p_3\rangle\}$

# Bisimulation

## Definition

Given $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$ over $N$, relation $R \subseteq S_1 \times S_2$ is a bisimulation iff both $R$ and its converse $R^\circ$ are simulations.
I.e., whenever $\langle p, q \rangle \in R$ and $a \in N$,

$$p \xrightarrow{a}_1 p' \;\Rightarrow\; \langle \exists\, q' \;:\; q' \in S_2 :\; q \xrightarrow{a}_2 q' \wedge \langle p', q' \rangle \in R \rangle$$

$$q \xrightarrow{a}_2 q' \;\Rightarrow\; \langle \exists\, p' \;:\; p' \in S_1 :\; p \xrightarrow{a}_1 p' \wedge \langle p', q' \rangle \in R \rangle$$

## Note

From now on instead of comparing states of two different systems $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$, we will consider a one joint system $\langle S, N, \longrightarrow \rangle$, where $S = S_1 + S_2$ and $\longrightarrow$ is the union of $\longrightarrow_1$ and $\longrightarrow_2$.

# Bisimulation

## Definition

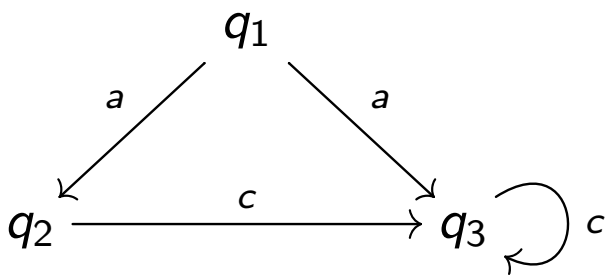
Given $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$ over $N$, relation $R \subseteq S_1 \times S_2$ is a bisimulation iff both $R$ and its converse $R^\circ$ are simulations.
I.e., whenever $\langle p, q \rangle \in R$ and $a \in N$,

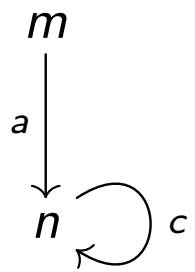
$$p \xrightarrow{a}_1 p' \;\Rightarrow\; \langle \exists\, q' \;:\; q' \in S_2 :\; q \xrightarrow{a}_2 q' \wedge \langle p', q' \rangle \in R \rangle$$
$$q \xrightarrow{a}_2 q' \;\Rightarrow\; \langle \exists\, p' \;:\; p' \in S_1 :\; p \xrightarrow{a}_1 p' \wedge \langle p', q' \rangle \in R \rangle$$

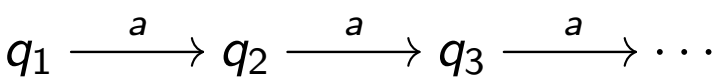
## Note

From now on instead of comparing states of two different systems $\langle S_1, N, \longrightarrow_1 \rangle$ and $\langle S_2, N, \longrightarrow_2 \rangle$, we will consider a one joint system $\langle S, N, \longrightarrow \rangle$, where $S = S_1 + S_2$ and $\longrightarrow$ is the union of $\longrightarrow_1$ and $\longrightarrow_2$.

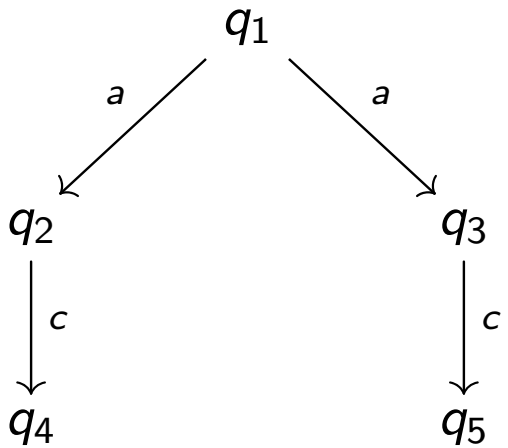
# Examples

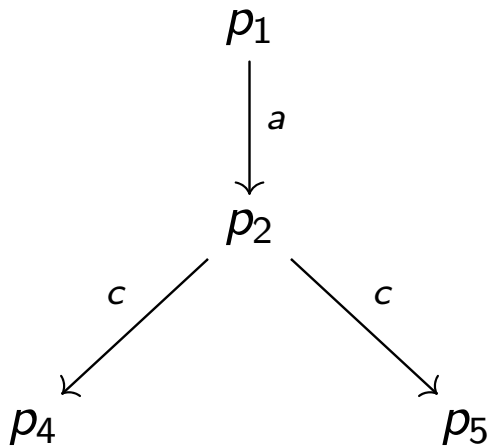


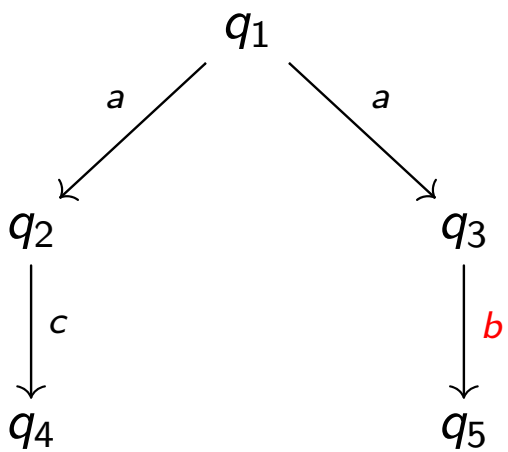



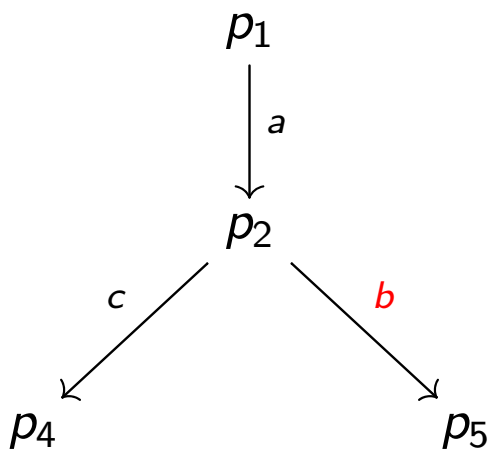




# Examples


$q_1$
$a$
$a$
$q_2$
$q_3$
$c$
$c$
$q_4$
$q_5$


$p_1$
$a$
$p_2$
$c$
$c$
$p_4$
$p_5$


$q_1$
$a$
$a$
$q_2$
$q_3$
$c$
$b$
$q_4$
$q_5$


$p_1$
$a$
$p_2$
$c$
$b$
$p_4$
$p_5$

# Bisimilarity

## Definition

$$p \sim q \equiv \langle \exists\ R\ ::\ R \text{ is a bisimulation and } \langle p, q \rangle \in R \rangle$$

## Lemma

The bisimilarity relation is an equivalence relation
(ie, reflexive, symmetric and transitive)

# ... because

## Lemma

1. The identity relation *id* is a bisimulation
2. The converse $R^\circ$ of a bisimulation is a bisimulation
3. The composition $S \cdot R$ of two bisimulations $S$ and $R$ is a bisimulation

Thus, bisimilarity can be also established by the existence of a bisimulation equivalence, i.e. an *equivalence* relation such that for all $\langle p, q \rangle \in R$ and $a \in N$,

$$p \xrightarrow{a} p' \Rightarrow \langle \exists\ q' \ :\ q' \in S :\ q \xrightarrow{a} q' \wedge \langle p', q' \rangle \in R \rangle$$

# After thoughts

- Follows a $\forall, \exists$ pattern: $p$ in all its transitions challenge $q$ which is called to find a matchh to each of those (and conversely)
- Tighter correspondence with transitions
- Based on the information that the transitions convey, rather than on the shape of the LTS
- Local checks on states
- Lack of hierarchy on the pairs of the bisimulation (no temporal order on the checks is required)

which means bisimilarity can be used to reason about infinite or circular behaviours.

# Properties

## Lemma

1. The empty relation $\perp$ is a bisimulation
2. The $\bigcup_{i \in I} R_i$ of a family of bisimulations $\{R_i | \ i \in I\}$ is a bisimulation

## Lemma

The class of all bisimulations between two LTS has the structure of a complete lattice, ordered by set inclusion, whose top is the bisimilarity relation $\sim$.

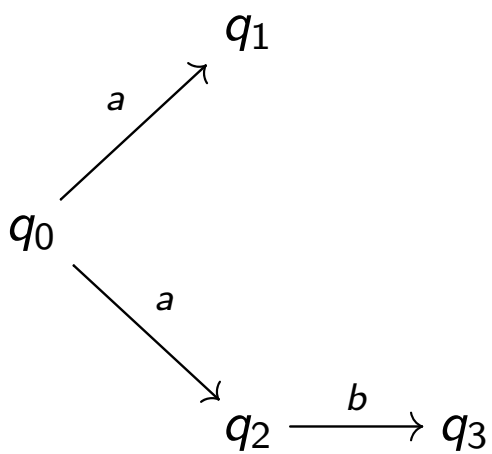
# Properties

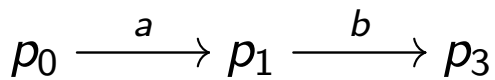
## Warning

The bisimilarity relation $\sim$ is not the symmetric closure of $\lesssim$

## Example

$$q_0 \lesssim p_0,\ p_0 \lesssim q_0 \quad \text{but} \quad p_0 \not\sim q_0$$



# Modal logic (from P. Blackburn, 2007)

*Over the years modal logic has been applied in many different ways. It has been used as a tool for reasoning about time, beliefs, computational systems, necessity and possibility, and much else besides.*

*These applications, though diverse, have something important in common: the key ideas they employ (flows of time, relations between epistemic alternatives, transitions between computational states, networks of possible worlds) can all be represented as simple graph-like structures.*

Modal logics are

- tools to talk about relational, or graph-like structures.
- fragments of classical ones, with restricted forms of quantification ...
- ... which tend to be decidable and described in a pointfree notations.

# The language

## Syntax

$$\phi ::= p \mid \mathit{true} \mid \mathit{false} \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \rightarrow \phi_2 \mid \langle m \rangle \phi \mid [m]\phi$$

where $p \in \mathrm{PROP}$ and $m \in \mathrm{MOD}$

Disjunction ($\vee$) and equivalence ($\leftrightarrow$) are defined by abbreviation. The signature of the basic modal language is determined by sets PROP of propositional symbols (typically assumed to be denumerably infinite) and MOD of modality symbols.

# The language

## Notes

- if there is only one modality in the signature (i.e., MOD is a singleton), write simply $\Diamond\phi$ and $\Box\phi$
- the language has some redundancy: in particular modal connectives are dual (as quantifiers are in first-order logic): $[m]\phi$ is equivalent to $\neg\langle m\rangle\neg\phi$

# Semantics

## Semantics

A model for the language is a pair $\mathfrak{M} = \langle \mathbb{F}, V \rangle$, where

- $\mathfrak{F} = \langle W, \{R_m\}_{m \in \mathrm{MOD}} \rangle$ is a Kripke frame, ie, a non empty set $W$ and a family of binary relations over $W$, one for each modality symbol $m \in \mathrm{MOD}$. Elements of $W$ are called points, states, worlds or simply vertices in directed graphs.
- $V : \mathrm{PROP} \longrightarrow \mathcal{P}(W)$ is a valuation.

Hennessy-Milner Theorem: Modal equivalence and bisimilarity coincide on mild conditions.

# Semantics

## Safistaction: for a model $\mathfrak{M}$ and a point $w$

$\mathfrak{M}, w \models \mathit{true}$

$\mathfrak{M}, w \not\models \mathit{false}$

| | | |
|---|---|---|
| $\mathfrak{M}, w \models p$ | iff | $w \in V(p)$ |
| $\mathfrak{M}, w \models \neg\phi$ | iff | $\mathfrak{M}, w \not\models \phi$ |
| $\mathfrak{M}, w \models \phi_1 \wedge \phi_2$ | iff | $\mathfrak{M}, w \models \phi_1$ and $\mathfrak{M}, w \models \phi_2$ |
| $\mathfrak{M}, w \models \phi_1 \rightarrow \phi_2$ | iff | $\mathfrak{M}, w \not\models \phi_1$ or $\mathfrak{M}, w \models \phi_2$ |
| $\mathfrak{M}, w \models \langle m \rangle \phi$ | iff | there exists $v \in W$ st $vR_m w$ and $\mathfrak{M}, v \models \phi$ |
| $\mathfrak{M}, w \models [m]\phi$ | iff | for all $v \in W$ st $vR_m w$ and $\mathfrak{M}, v \models \phi$ |

# Examples

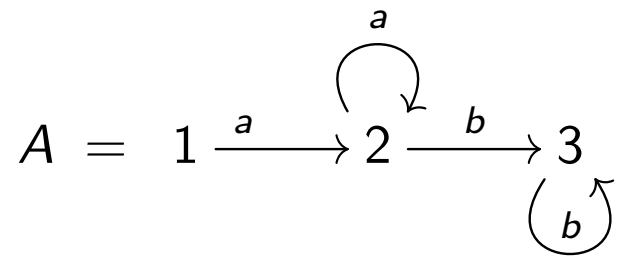
Modal formulas reflect properties of accessibility relations:

- transitive frames: $\Box\phi \rightarrow \Box\Box\phi$
- simple frames: $\Diamond\phi \rightarrow \Box\phi$
- frames consisting of isolated reflexive points: $\phi \leftrightarrow \Box\phi$
- frames consisting of isolated irreflexive points: $\Box \mathit{false}$

But there are classes of frames which are not modally definable, eg, connected, irreflexive, containing a isolated irreflexive point

# Examples

## An automaton




- two modalities $\langle a \rangle$ and $\langle b \rangle$ to explore the corresponding classes of transitions
- note that

$$1 \models \langle a \rangle \cdots \langle a \rangle \langle b \rangle \cdots \langle b \rangle t$$

  where $t$ is a proposition valid only at the (terminal) state 3.
- all modal formulas of this form correspond to the strings accepted by the automaton, i.e. in language $\mathcal{L} = \{a^m b^n | \ m, n > 0\}$

# Examples

$(P, <)$ a strict partial order with infimum 0

- $P, x \models \Box \mathit{false}$ if $x$ is a maximal element of $P$
- $P, 0 \models \Diamond \Box \mathit{false}$ iff ...
- $P, 0 \models \Box \Diamond \Box \mathit{false}$ iff ...

# Examples

## Process logic (Hennessy-Milner logic)

- PROP $= \emptyset$
- $W =$ is a set of states, typically process terms, in a labelled transition system
- each subset $K \subseteq Act$ of actions generates a modality corresponding to transitions labelled by an element of $K$

Assuming the underlying LTS $\mathfrak{F} = \langle , \{p \xrightarrow{K} p' | \ K \subseteq Act\}\rangle$ as the modal frame, satisfaction is abbreviated as

$$p \models \langle K \rangle \phi \quad \text{iff} \quad \exists_{q \in \{p' |\ p \xrightarrow{a} p' \ \wedge \ a \in K\}} \cdot q \models \phi$$

$$p \models [K] \phi \quad \text{iff} \quad \forall_{q \in \{p' |\ p \xrightarrow{a} p' \ \wedge \ a \in K\}} \cdot q \models \phi$$

# Example

Express the following properties in Process Logic

- inevitability of $a$: $\langle - \rangle \mathit{true} \wedge [-a] \mathit{false}$
- progress: $\langle - \rangle \mathit{true}$
- deadlock or termination: $[-] \mathit{false}$
- what about

$$\langle - \rangle \mathit{false} \quad \text{and} \quad [-] \mathit{true} \quad ?$$

# An alternative characterisation

The isomorphism between
relations $R \subseteq A \times B$ and functions $f : A \longrightarrow \mathcal{P}B$, given by

$$\langle a, b \rangle \in R \;\equiv\; b \in f\, a$$

supports an alternative, functional characterisation of LTS:

$$\langle S, N, \longrightarrow \rangle \;\equiv\; \alpha : S \longrightarrow \mathcal{P}(\mathbb{N} \times S)$$

given by

$$s \xrightarrow{a} s' \;\equiv\; \langle a, s' \rangle \in \alpha\, s$$

which allows us to easily draw a taxonomy of simple transition systems

# A taxonomy of simple transition systems

| | |
|---|---|
| $\alpha : S \longrightarrow \mathcal{P}(S)$ | unlabelled TS |
| $\alpha : S \longrightarrow \mathbb{N} \times S + \mathbf{1}$ | partial LTS (generative) |
| $\alpha : S \longrightarrow (S + \mathbf{1})^{\mathbb{N}}$ | partial LTS (reactive) |
| $\alpha : S \longrightarrow \mathcal{P}(\mathbb{N} \times S)$ | non deterministic LTS (generative) |
| $\alpha : S \longrightarrow \mathcal{P}(S)^{\mathbb{N}}$ | non deterministic LTS (reactive) |

## Notation for sets

$A \times B$ Cartesian product

$A + B$ disjoint union

$B^A$ function space

$\mathbf{1}$ Singular set: $\mathbf{1} \cong \{*\}$

# A zoo of transition systems

Simple transition systems can be extended with actions and suited to different sorts of behaviours (e.g. partial, non deterministic, etc).
... but the zoo is much broader, capturing

- probabilistic transitions (Prism)
- timed transitions (Uppaal, mCRL2)
- continuous evolutions (e.g. of physical processes) (KeYmaera)
- ... and several combinations thereof

(typical support tools are indicated in brown)

# Bringing probabilities into the picture

## Markov chains

$$\alpha : S \longrightarrow \mathcal{D}S$$

where $\mathcal{D}S$ is the set of all discrete probability distributions on set $S$

A Markov chain goes from a state $s$ to a state $s'$ with probability $p$ if

$$\alpha\, s = \mu \;\text{ with }\; \mu\, s' = p > 0$$

## Notation

$s \rightsquigarrow \mu$ and $s \overset{p}{\rightsquigarrow} s'$

# Bringing probabilities into the picture

## Recall

$\mu : S \longrightarrow [0,1]$ is a discrete probability distribution

- if the support of $\mu$, i.e. the set $\{s \in S | \ \mu\, s > 0\}$, is finite
- and $\sum_{s \in S} \mu\, s = 1$

## Examples

Dirac distribution $\mu_s^1 = \{s \mapsto 1\}$

Product distribution $(\mu_1 \times \mu_2)\langle s, t\rangle = (\mu_1\, s) \cdot (\mu_2\, t)$

# Bringing probabilities into the picture

## Bisimilarity for Markov chains

An equivalence relation $R \subseteq S \times S$ is a bisimulation iff for all $\langle s, t \rangle \in R$

$$\text{if } s \rightsquigarrow \mu \text{ then there is a transition } t \rightsquigarrow \mu' \text{ such that } \mu \equiv_R \mu'$$

where $\mu \equiv_R \mu'$ iff $\mu[C] = \mu'[C]$ for all equivalence class $C$ defined by relation $R$.

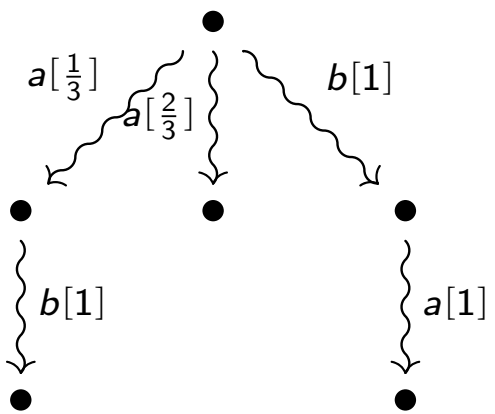
This means that the probability of getting from $s$ or $t$ to an element of $C$ is the same

... of course, any two states in a Markov chain are bisimilar! (why?)

# Reactive PTS

$$\alpha : S \longrightarrow (\mathcal{D}S + \mathbf{1})^{\mathbb{N}}$$

- $s \overset{a}{\rightsquigarrow} \mu_a$ if $\alpha\, s\, a = \mu_a$
- $s \overset{a[p]}{\rightsquigarrow} s'$ if additionally $s'$ in the support of $\mu$ and $\mu_a\, s' = p$
- $s \not\rightsquigarrow$ if $\alpha\, s\, a = *$
- Note the role of $\mathbf{1}$ (cf $\emptyset$ in the non deterministic LTS)



# Reactive PTS

## Bisimulation

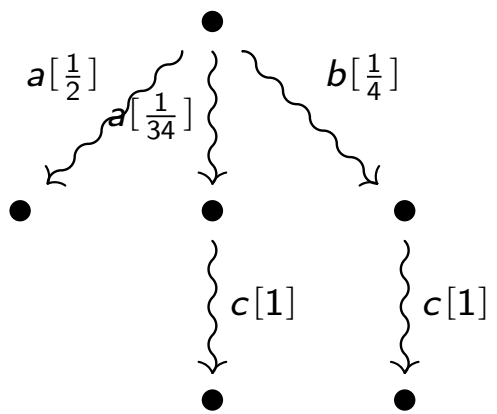
An equivalence relation $R \subseteq S \times S$ is a bisimulation iff for all $\langle s, t \rangle \in R$ and all $a \in \mathbb{N}$

if $s \overset{a}{\rightsquigarrow} \mu$ then there is a distribution $\mu'$ with $t \overset{a}{\rightsquigarrow} \mu'$ such that $\mu \equiv_R \mu'$

# Generative PTS

$$\alpha : S \longrightarrow \mathcal{D}\mathbb{N} \times S + \mathbf{1}$$

- $s \overset{a}{\rightsquigarrow} \mu_a$ if $\alpha\, s = \mu$
- $s \overset{a[p]}{\rightsquigarrow} s'$ if additionally $\langle a, s' \rangle$ in the support of $\mu$ and $\mu\langle a, s' \rangle = p$
- $s \not\rightsquigarrow$ if $\alpha\, s = *$



# Generative PTS

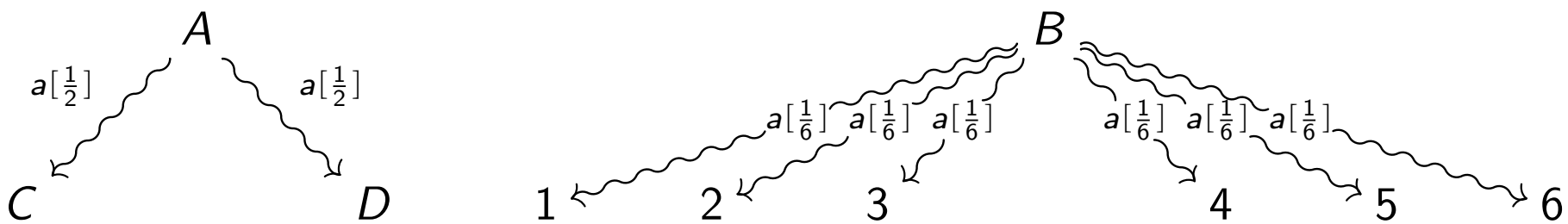
## Bisimulation

An equivalence relation $R \subseteq S \times S$ is a bisimulation iff for all $\langle s, t \rangle \in R$

if $s \rightsquigarrow \mu$ then there is a distribution $\mu'$ with $t \rightsquigarrow \mu'$ such that $\mu \equiv_{R,A} \mu'$

## Example

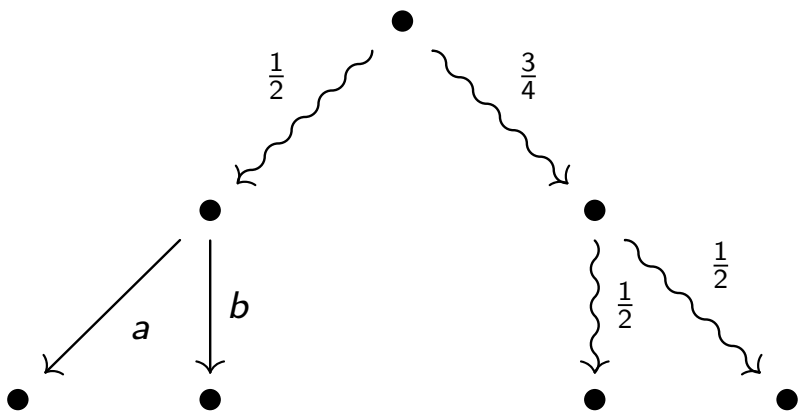
$R = \{\langle A, B \rangle, \langle C, 1 \rangle, \langle C, 2 \rangle, \langle C, 3 \rangle, \langle D, 4 \rangle, \langle D, 5 \rangle, \langle D, 6 \rangle\}$



# A taxonomy of probabilistic transition systems

| | |
|---|---|
| $\alpha : S \longrightarrow \mathcal{D}S$ | simple PTS (Markov chain) |
| $\alpha : S \longrightarrow \mathcal{D}\mathbb{N} \times S + \mathbf{1}$ | generative PTS |
| $\alpha : S \longrightarrow (\mathcal{D}S + \mathbf{1})^{\mathbb{N}}$ | reactive PTS |
| $\alpha : S \longrightarrow \mathcal{D}S + (\mathbb{N} \times S) + \mathbf{1}$ | stratified PTS |

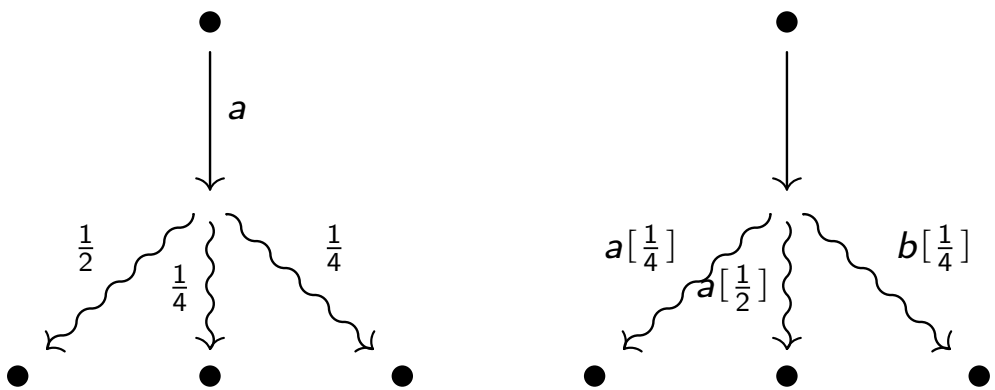
## Alternating PTS



# Adding non determinism

| | |
|---|---|
| $\alpha : S \longrightarrow \mathcal{P}(\mathcal{D}\mathbb{N} \times S)$ | strict Segala PTS |
| $\alpha : S \longrightarrow \mathcal{P}(\mathbb{N} \times \mathcal{D}S)$ | simple Segala PTS |
| $\alpha : S \longrightarrow \mathcal{P}(\mathcal{D}\mathcal{P}(\mathbb{N} \times S))$ | Pnueli-Zuck PTS |

## Transitions for simple and strict Segala PTS



# After thoughts

- The taxonomy is driven by the structure on the codomain of function $\alpha$
- The definition of bisimulation follows, in every case, the same intuition

(... we are beginning to think coalgebraically)