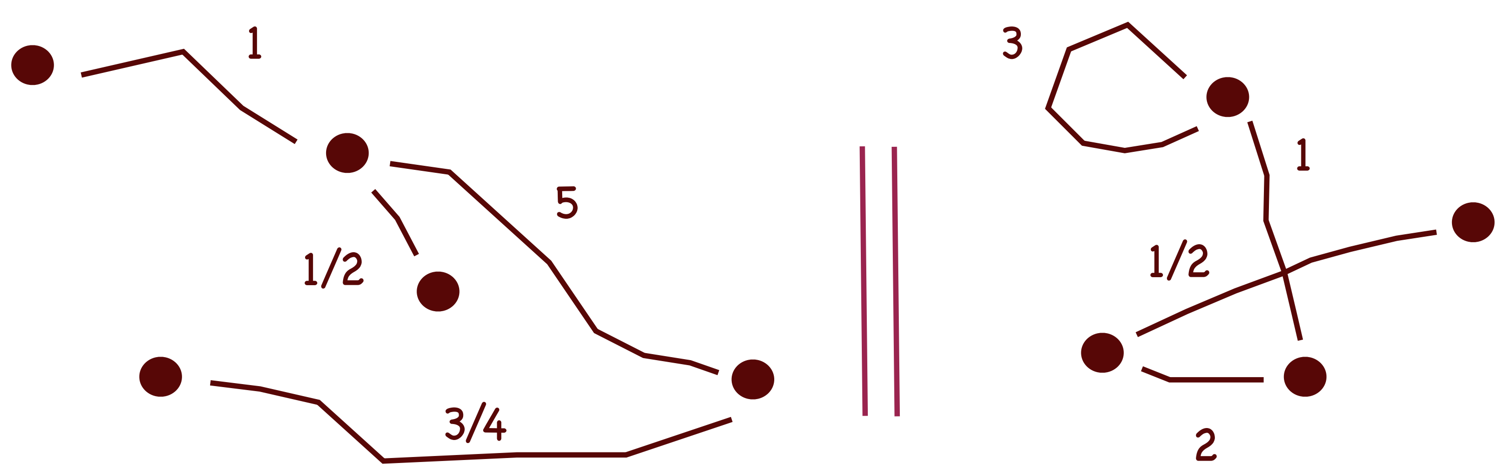
# Probabilistic Systems Semantics via Coalgebra

Ana Sokolova UNIVERSITY of SALZBURG




TbiLLC'17 tutorial, part 3

Plan:

**Part 1.** Modelling probabilistic systems for branching-time semantics

bisimilarity

**Part 2.** Traces, linear-time semantics

trace equivalence

**Part 3.** Belief-state-transformer semantics via convexity

distribution bisimilarity

Mathematical framework based on category theory for state-based systems semantics

all with help of coalgebra

Plan:

we still have something to discuss here

**Part 1.** Modelling probabilistic systems for branching-time semantics

bisimilarity

**Part 2.** Traces, linear-time semantics

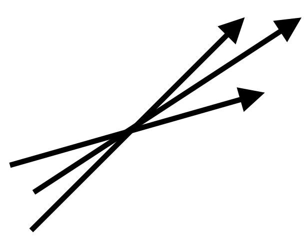
trace equivalence

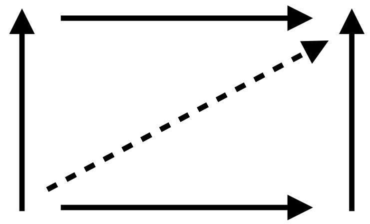
**Part 3.** Belief-state-transformer semantics via convexity

distribution bisimilarity

Mathematical framework based on category theory for state-based systems semantics

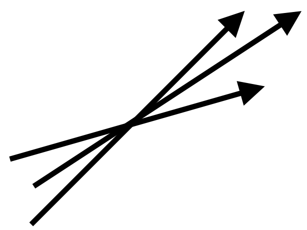
all with help of coalgebra

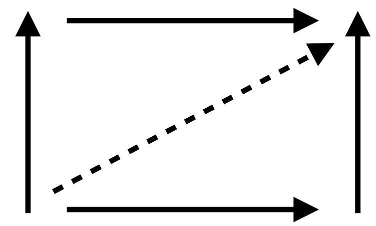






# Probabilistic systems
# are
# coalgebras

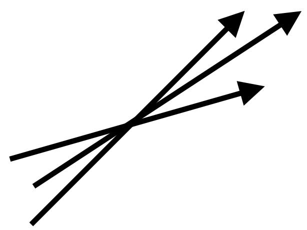


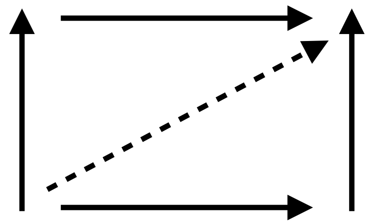


Source Wikipedia, by Diacritica - Own work


# Probabilistic systems are coalgebras

$\mathcal{D}$ on **Sets**



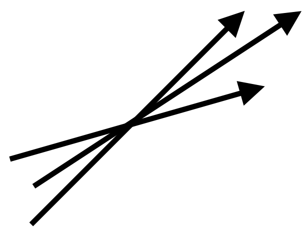


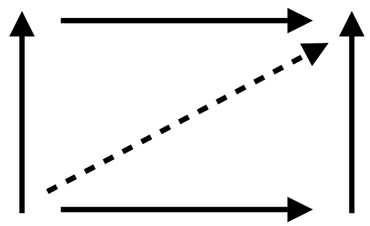


# Probabilistic systems are coalgebras

$\mathcal{D}$ on **Sets**

$\mathcal{G}$ on **Meas**





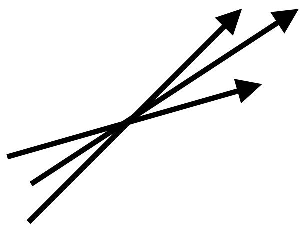


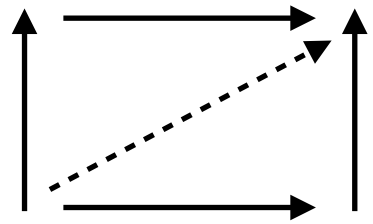
# Probabilistic systems are coalgebras

$\mathcal{D}$ on **Sets**

$\mathcal{G}$ on **Meas**

generic notion of behavioural equivalence







# Probabilistic systems are coalgebras

$\mathcal{D}$ on **Sets**

$\mathcal{G}$ on **Meas**

generic notion of behavioural equivalence

strong, branching-time semantics

# Trace semantics coalgebraically

we need to move out of **Sets**

trace equivalence is behaviour equivalence

# Trace semantics coalgebraically

we need to move out of **Sets**

Two approaches:

(1) modelling in a Kleisli category

(2) modelling in an Eilenberg-Moore category

of a monad $T$

there is a way to connect (1) and (2)

trace equivalence is behaviour equivalence

# Main ideas in both approaches

# Main ideas in both approaches

NFA / LTS

Two ideas:

(1) unfold branching + transitions on words

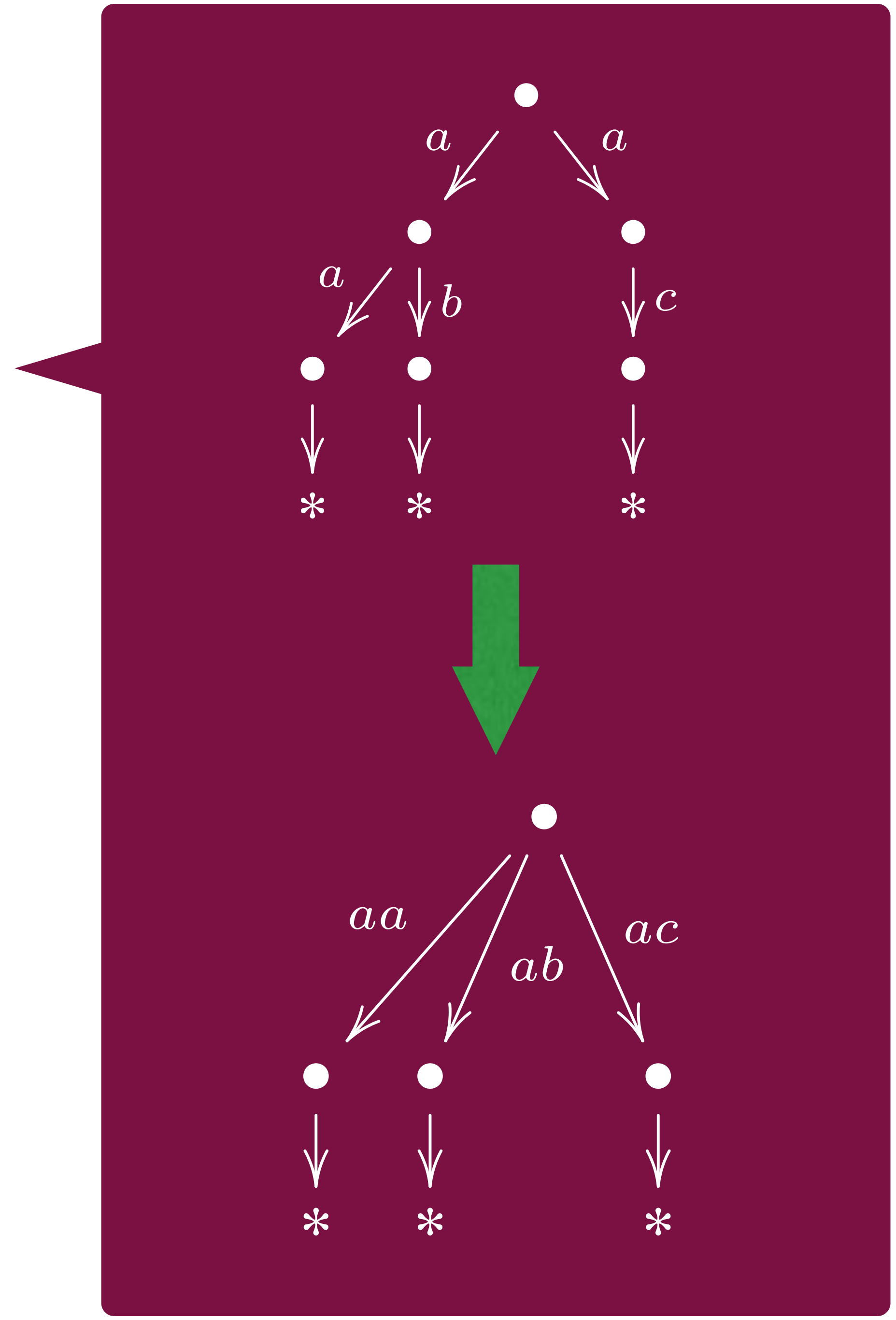
(2) trace = bisimilarity after determinisation

# Main ideas in both approaches

Two ideas: NFA / LTS

(1) unfold branching + transitions on words

(2) trace = bisimilarity after determinisation

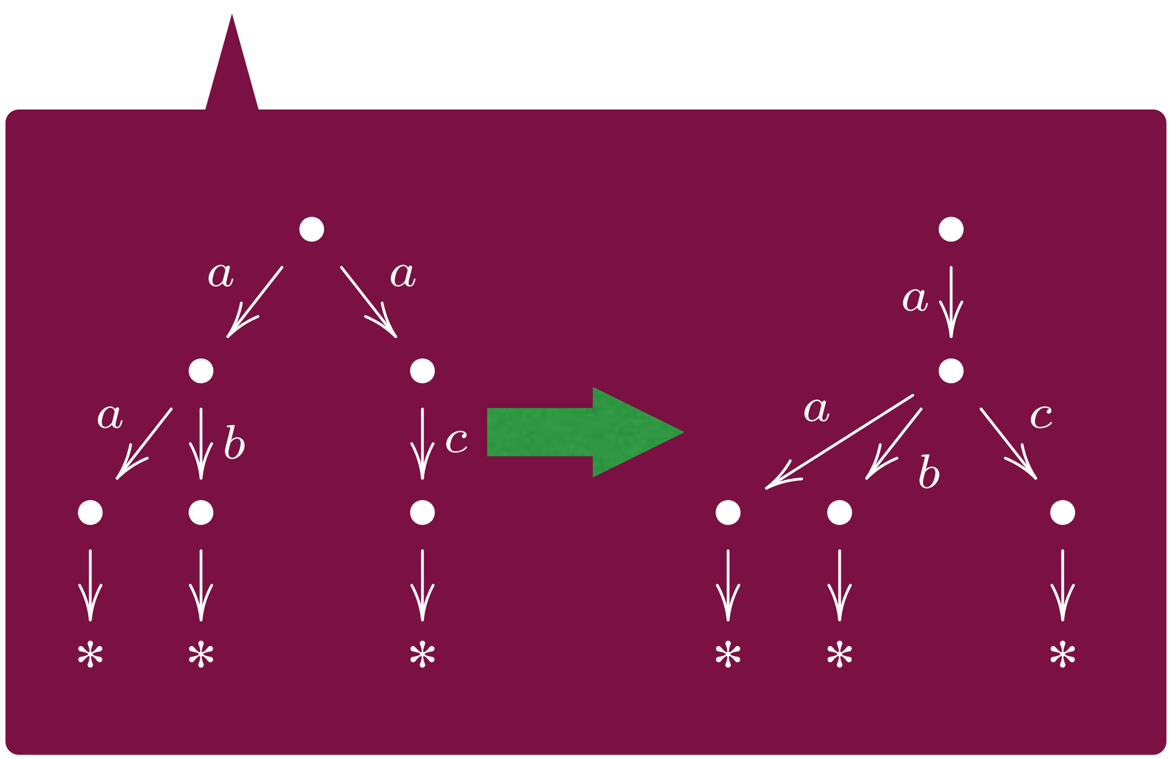


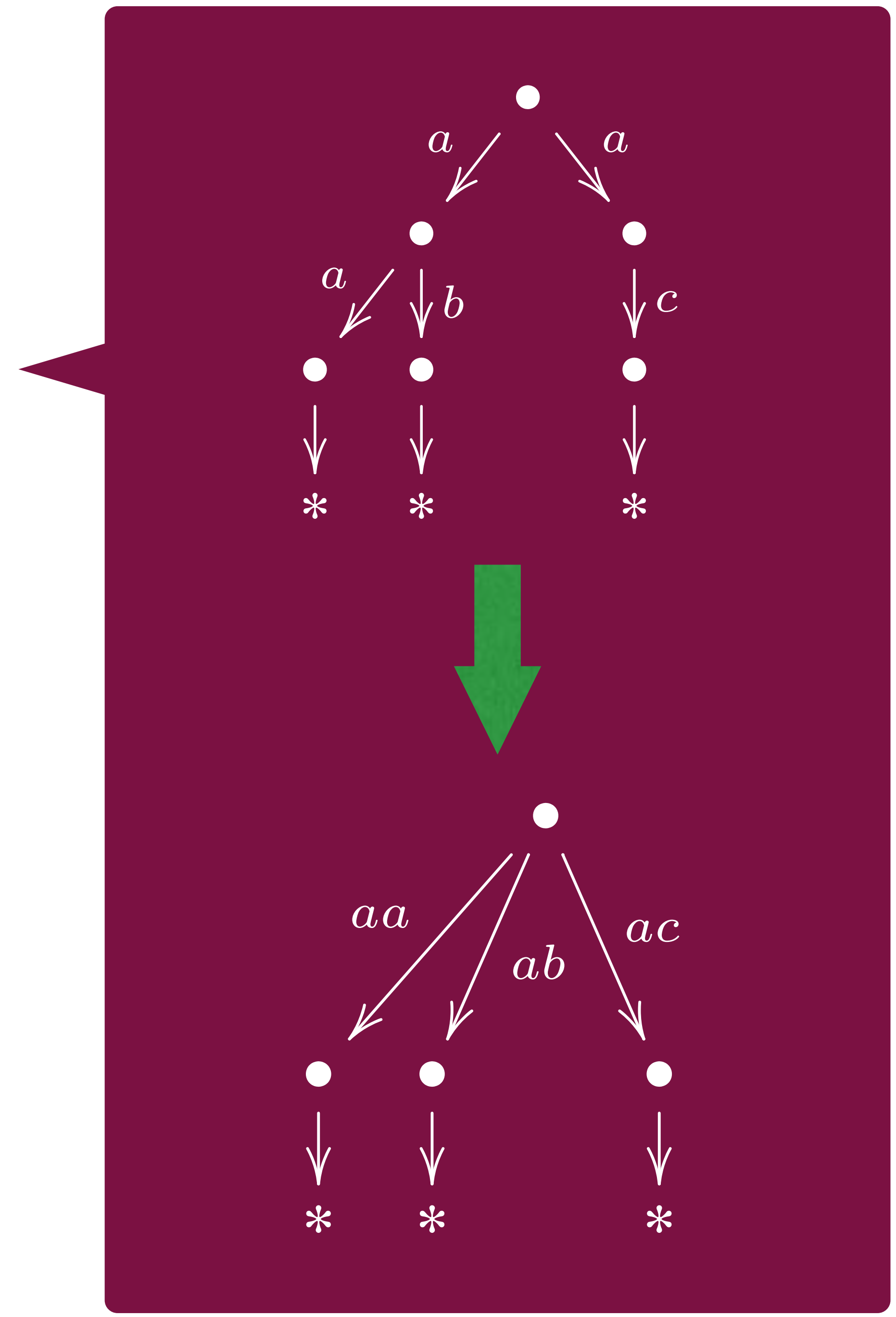
# Main ideas in both approaches

Two ideas:


(1) unfold branching + transitions on words

(2) trace = bisimilarity after determinisation



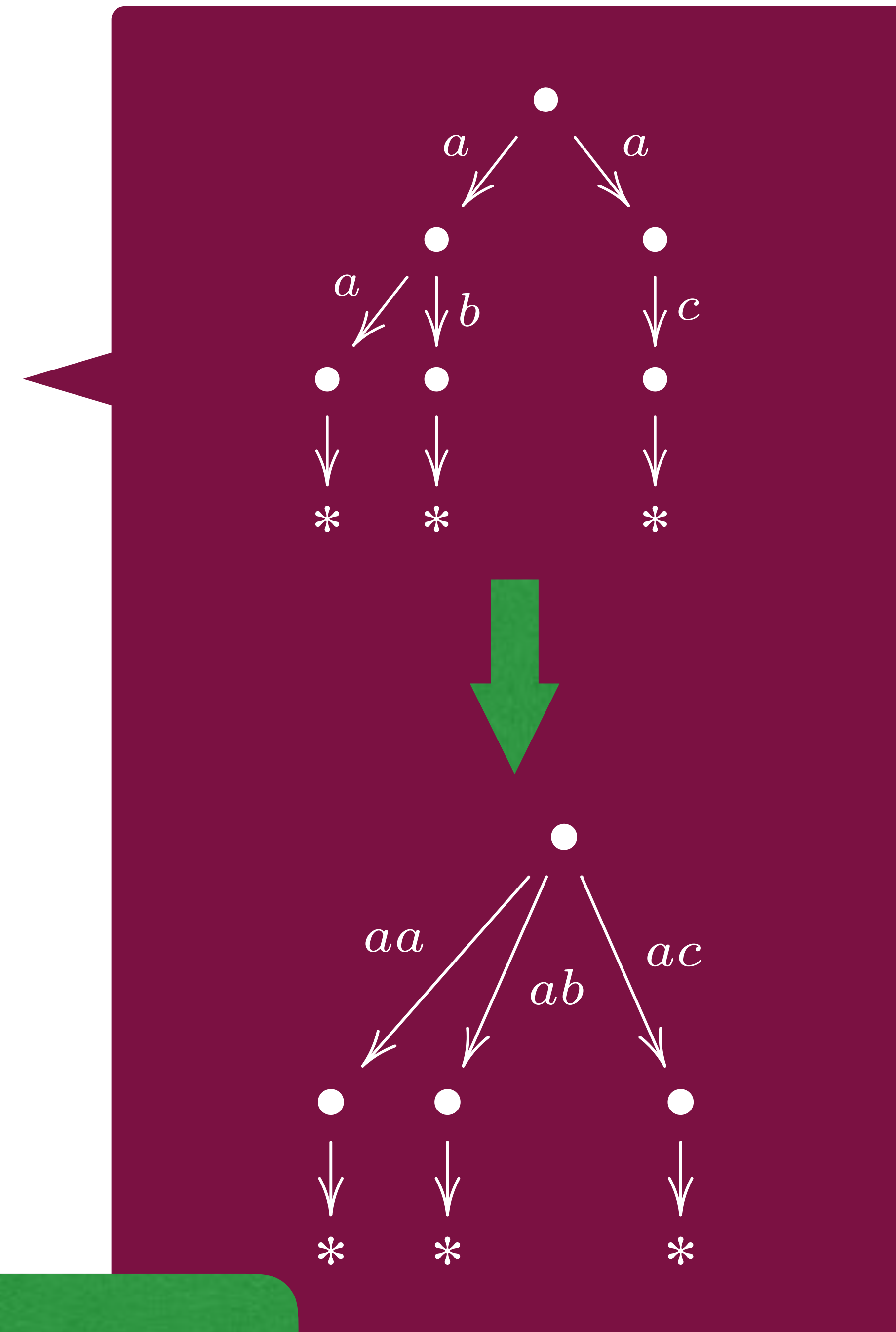



# Main ideas in both approaches

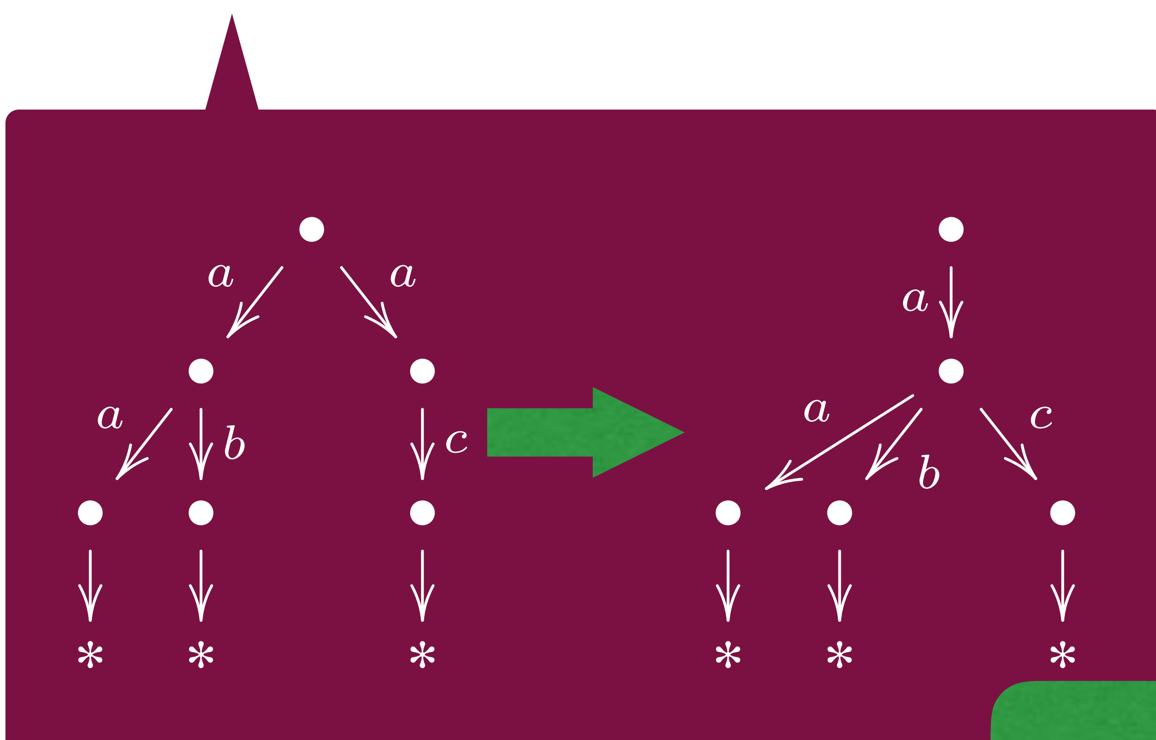
NFA / LTS

Two ideas:

(1) unfold branching + transitions on words




(2) trace = bisimilarity after determinisation




monads !

# Eilenberg-Moore Algebras

convex algebras

$\mathcal{EM}(\mathcal{D})$

finitely supported

- objects

$$\begin{array}{c} \mathcal{D}A \\ \downarrow a \\ A \end{array}$$

satisfying

$$\begin{array}{ccc} A & \xrightarrow{\eta} & \mathcal{D}A \\ & {\scriptstyle a} \searrow\!\!\!\!\searrow & \downarrow a \\ & & A \end{array} \qquad \begin{array}{ccc} \mathcal{D}\mathcal{D}A & \xrightarrow{\mu} & \mathcal{D}A \\ \mathcal{D}a \downarrow & & \downarrow a \\ \mathcal{D}A & \xrightarrow{a} & A \end{array}$$

- morphisms

$$\begin{array}{c} \mathcal{D}A \\ \downarrow a \\ A \end{array} \xrightarrow{h} \begin{array}{c} \mathcal{D}B \\ \downarrow b \\ B \end{array}$$

$$\begin{array}{ccc} \mathcal{D}A & \xrightarrow{\mathcal{D}h} & \mathcal{D}B \\ a \downarrow & & \downarrow b \\ A & \xrightarrow{h} & B \end{array}$$

# Convex Algebras

- algebras
- convex (affine) maps

# Convex Algebras

- algebras

$$(A, \sum_{i=1}^{n} p_i(-)_i)$$

- convex (affine) maps

# Convex Algebras

- algebras $(A, \sum_{i=1}^{n} p_i(-)_i)$ $p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$

- convex (affine) maps

# Convex Algebras

infinitely many
finitary operations

- algebras $(A, \sum_{i=1}^{n} p_i(-)_i)$ $p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$

- convex (affine) maps

# Convex Algebras

infinitely many finitary operations

convex combinations

- algebras $\left(A, \sum_{i=1}^{n} p_i(-)_i\right)$ $\quad p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$

- convex (affine) maps

# Convex Algebras

infinitely many finitary operations

convex combinations

binary ones "suffice"

- algebras $\left(A, \sum_{i=1}^{n} p_i(-)_i\right)$ $\quad p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$

- convex (affine) maps

# Convex Algebras

infinitely many finitary operations

convex combinations

binary ones "suffice"

- algebras

$$(A, \sum_{i=1}^{n} p_i(-)_i)$$

$$p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$$

- convex (affine) maps

$$h\left(\sum_{i=1}^{n} p_i a_i\right) = \sum_{i=1}^{n} p_i h(a_i)$$

# Convex Algebras

infinitely many finitary operations

convex combinations

binary ones "suffice"

- algebras

$$\left(A, \sum_{i=1}^{n} p_i(-)_i\right)$$

$$p_i \in [0,1], \sum_{i=1}^{n} p_i = 1$$

- convex (affine) maps

$$h\left(\sum_{i=1}^{n} p_i a_i\right) = \sum_{i=1}^{n} p_i h(a_i)$$

satisfying

- Projection $\sum_{i=1}^{n} p_i a_i = a_k, \quad p_k = 1$
- Barycenter $\sum_{i=1}^{n} p_i \left(\sum_{j=1}^{m} p_{i,j} a_j\right) = \sum_{j=1}^{m} \left(\sum_{i=1}^{n} p_i p_{i,j}\right) a_j$

# Trace axioms for generative PTS

# Trace axioms for generative PTS

Silva, S.
MFPS '11

# Trace axioms for generative PTS

Silva, S.
MFPS '11

Axioms for bisimilarity

+

# Trace axioms for generative PTS

Silva, S.
MFPS '11

Axioms for bisimilarity

+

$$p \cdot a \cdot (p_1 \mathsf{E}_1 \oplus p_2 \mathsf{E}_2) \quad \equiv \quad p_1 \cdot a \cdot p\mathsf{E}_1 \oplus p_2 \cdot a \cdot p\mathsf{E}_2 \qquad (D)$$

# Trace axioms for generative PTS

Silva, S.
MFPS '11

Axioms for bisimilarity

+

$$p \cdot a \cdot (p_1 \mathsf{E}_1 \oplus p_2 \mathsf{E}_2) \quad \equiv \quad p_1 \cdot a \cdot p\mathsf{E}_1 \oplus p_2 \cdot a \cdot p\mathsf{E}_2 \qquad (D)$$

soundness and
completeness

# Trace axioms for generative PTS

Silva, S.
MFPS '11

Axioms for bisimilarity

+

$$p \cdot a \cdot (p_1 \mathsf{E}_1 \oplus p_2 \mathsf{E}_2) \quad \equiv \quad p_1 \cdot a \cdot p\mathsf{E}_1 \oplus p_2 \cdot a \cdot p\mathsf{E}_2 \quad (D)$$

soundness and completeness

Happens in $\mathcal{EM}(\mathcal{D})$

# Trace axioms for generative PTS

Silva, S.
MFPS '11

Axioms for bisimilarity

+

$$p \cdot a \cdot (p_1 \mathsf{E}_1 \oplus p_2 \mathsf{E}_2) \quad \equiv \quad p_1 \cdot a \cdot p\mathsf{E}_1 \oplus p_2 \cdot a \cdot p\mathsf{E}_2 \qquad (D)$$

soundness and completeness

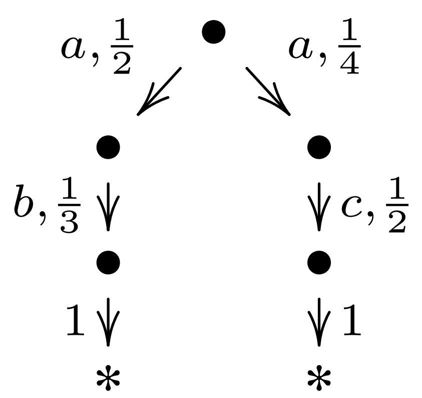
Happens in $\mathcal{EM}(\mathcal{D})$

positive convex algebras

# Trace axioms for generative PTS

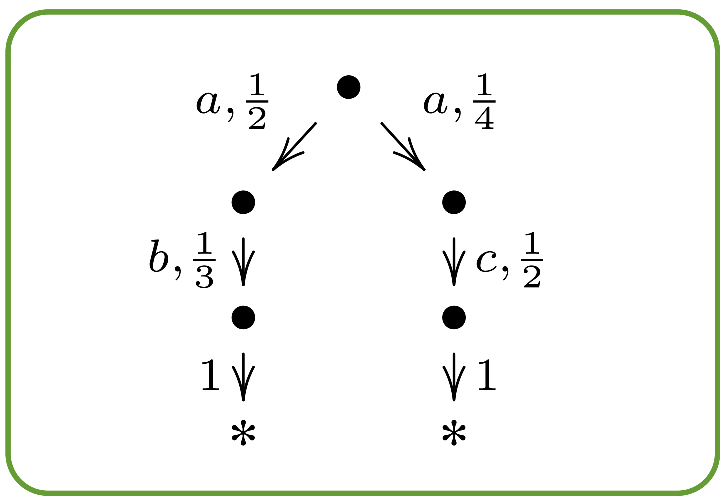
# Trace axioms for generative PTS

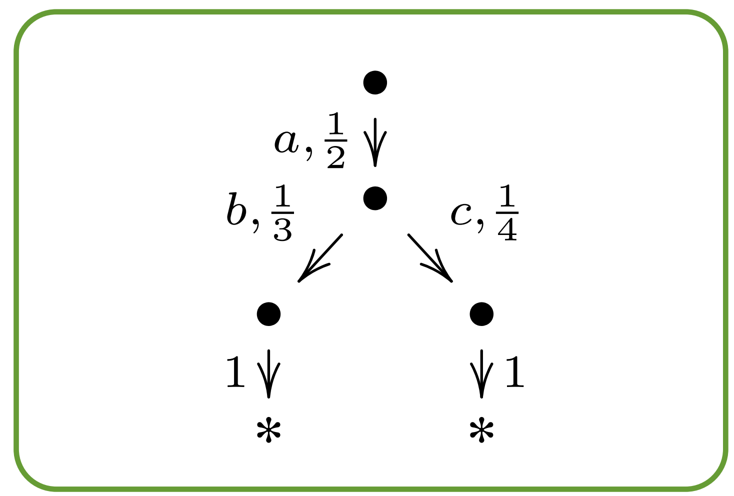
$\mathcal{D}\,(1 + A \times (-))$



# Trace axioms for generative PTS

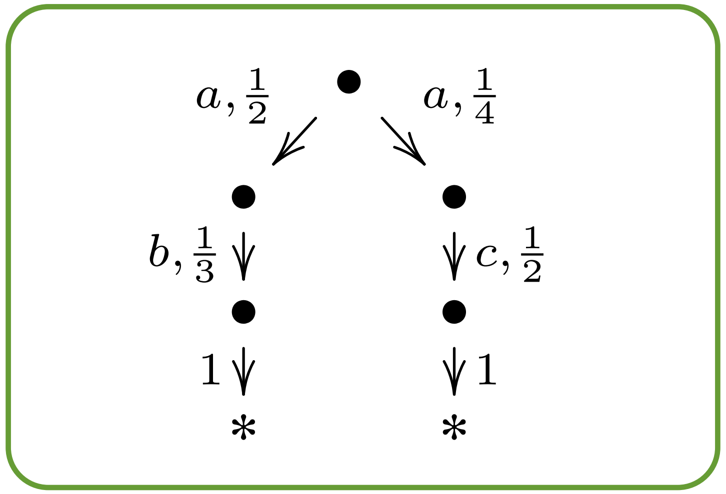
$\mathcal{D}(1 + A \times (-))$

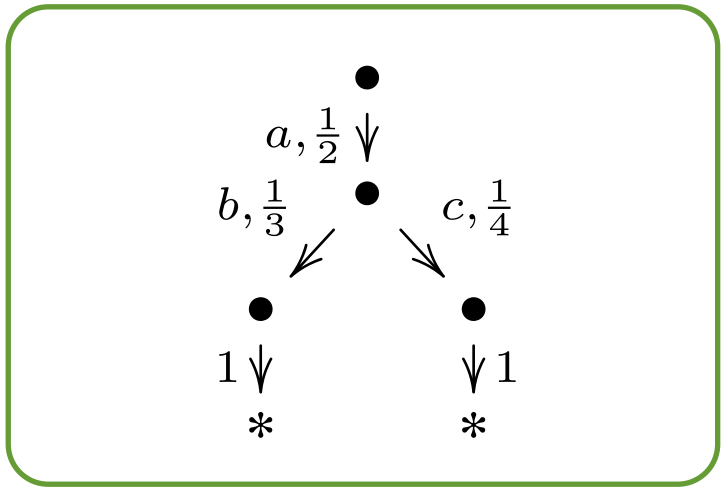


# Trace axioms for generative PTS

$\mathcal{D}(1 + A \times (-))$



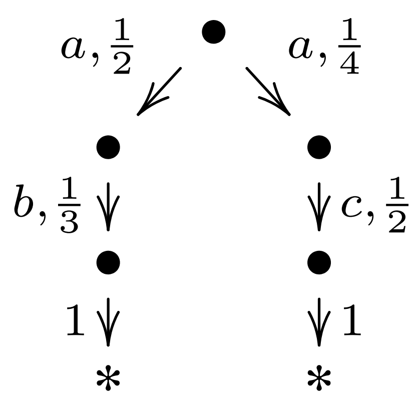


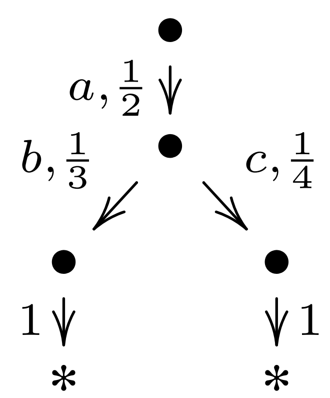
$$\left(\frac{1}{2}\cdot a\cdot\frac{1}{3}\cdot b\cdot 1\cdot *\right)\oplus\left(\frac{1}{4}\cdot a\cdot\frac{1}{2}\cdot c\cdot 1\cdot *\right)\overset{(Cong)}{\equiv}\left(\frac{1}{2}\cdot a\cdot\frac{1}{3}\cdot b\cdot 1\cdot *\right)\oplus\left(\frac{1}{2}\cdot a\cdot\frac{1}{4}\cdot c\cdot 1\cdot *\right)$$

$$\overset{(D)}{\equiv}\frac{1}{2}\cdot a\cdot\left(\frac{1}{3}\cdot b\cdot 1\cdot *\oplus\frac{1}{4}\cdot c\cdot 1\cdot *\right)$$

# Trace axioms for generative PTS

$\mathcal{D}$ (1 + A x (-))





$$\frac{1}{4}\cdot a\cdot\frac{1}{2}\cdot c\cdot 1\cdot * \stackrel{(D)}{\equiv} \frac{1}{2}\cdot a\cdot\frac{1}{4}\cdot c\cdot 1\cdot *$$

$$\left(\frac{1}{2}\cdot a\cdot\frac{1}{3}\cdot b\cdot 1\cdot *\right)\oplus\left(\frac{1}{4}\cdot a\cdot\frac{1}{2}\cdot c\cdot 1\cdot *\right) \stackrel{(Cong)}{\equiv} \left(\frac{1}{2}\cdot a\cdot\frac{1}{3}\cdot b\cdot 1\cdot *\right)\oplus\left(\frac{1}{2}\cdot a\cdot\frac{1}{4}\cdot c\cdot 1\cdot *\right)$$

$$\stackrel{(D)}{\equiv} \frac{1}{2}\cdot a\cdot\left(\frac{1}{3}\cdot b\cdot 1\cdot *\oplus\frac{1}{4}\cdot c\cdot 1\cdot *\right)$$

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras

- Proper functors

# The quest for completeness

Inspired lots of new research:

S., Woracek
JPAA '15

- Congruences of convex algebras

- Proper functors

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA ’15

Milius
CALCO’17

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA ’15

f.p. = f.g.
for (positive)
convex algebras

Milius
CALCO’17

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA '15

f.p. = f.g.
for (positive)
convex algebras

Milius
CALCO'17

if f.p. = f.g. and ★

then completeness

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA ’15

f.p. = f.g.
for (positive)
convex algebras

Milius
CALCO’17

if f.p. = f.g. and ★

then completeness

does not hold

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA '15

f.p. = f.g.
for (positive)
convex algebras

Milius
CALCO'17

our axiomatisation is complete since one particular functor Ĝ on $\mathcal{EM}(\mathcal{D})$ is proper

if f.p. = f.g. and ★ then completeness

does not hold

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek
JPAA '15

f.p. = f.g.
for (positive)
convex algebras

Milius
CALCO'17

our axiomatisation is complete since one particular functor $\hat{G}$ on $\mathcal{EM}(\mathcal{D})$ is proper

if f.p. = f.g. and ★ then completeness

does not hold

Woracek, S.'17

# The quest for completeness

Inspired lots of new research:

- Congruences of convex algebras
- Proper functors

S., Woracek JPAA '15

f.p. = f.g. for (positive) convex algebras

Milius CALCO'17

our axiomatisation is complete since one particular functor Ĝ on $\mathcal{EM}(\mathcal{D})$ is proper

if f.p. = f.g. and ★ then completeness

does not hold

NEW!

Woracek, S.'17

# Coalgebraic traces

need a move out of **Sets**

and monads

# Coalgebraic traces

need a move out of **Sets**

and monads

Trace semantics

- is also behaviour semantics, in a category of algebras
- the general approach pays off

# Coalgebraic traces

need a move out of **Sets**

and monads

Trace semantics

$\mathcal{EM}(\mathcal{D})$ convex algebras

$\mathcal{EM}(\mathcal{P})$ join semilattices

- is also behaviour semantics, in a category of algebras
- the general approach pays off

# Coalgebraic traces

need a move out of **Sets**

and monads

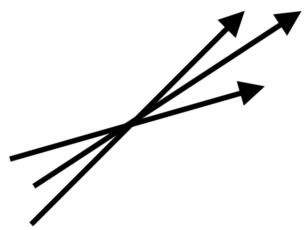
Trace semantics

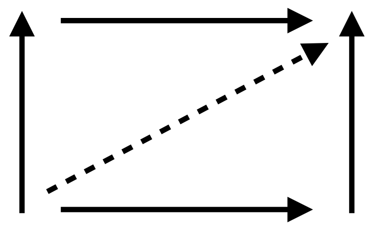
$\mathcal{EM}(\mathcal{D})$
convex algebras

$\mathcal{EM}(\mathcal{P})$
join semilattices

- is also behaviour semantics, in a category of algebras
- the general approach pays off

no concrete proof of completeness of the axiomatisation found

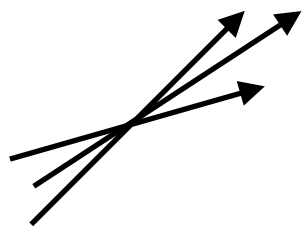


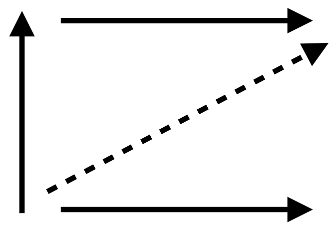


Source Wikipedia, by Diacritica - Own work


# Part III

## Modelling probabilistic systems
## for
## distribution semantics







# Part III

## Modelling probabilistic systems
## for
## distribution semantics

coalgebraically

probabilistic automata

The true nature of probabilistic systems

as transformers of belief states

probabilistic automata

The true nature of probabilistic systems as transformers of belief states

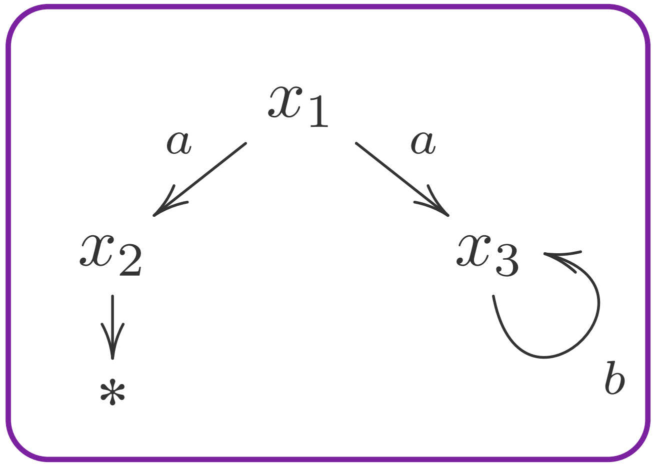
Bonchi, Silva, S.
CONCUR '17

# Determinisations

# Determinisations

NFA

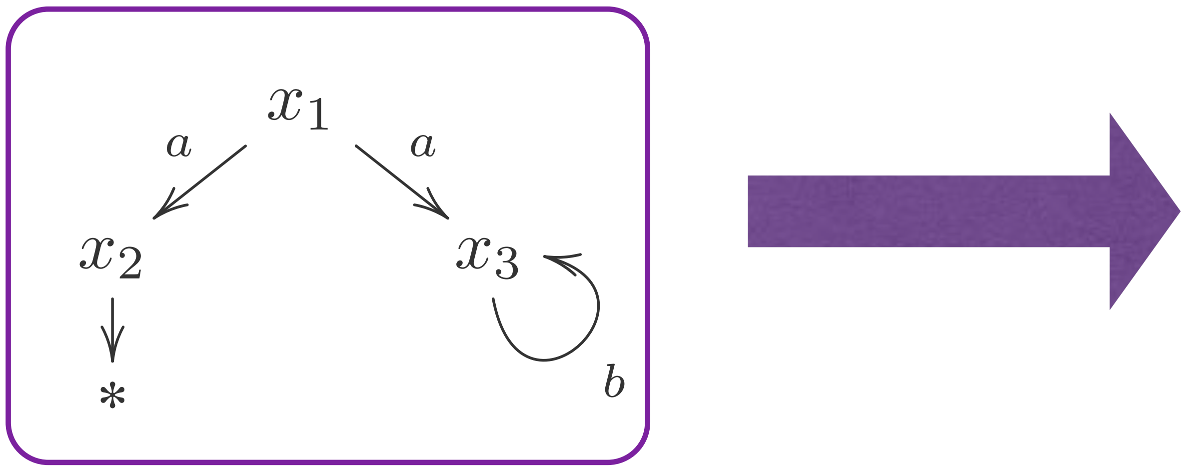
$X \rightarrow 2 \times (\mathcal{P}(X))^A$



# Determinisations

NFA

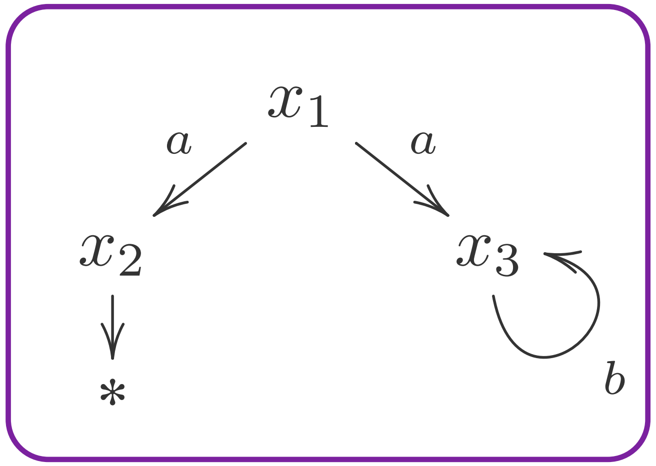
$X \to 2 \times (\mathcal{P}(X))^A$

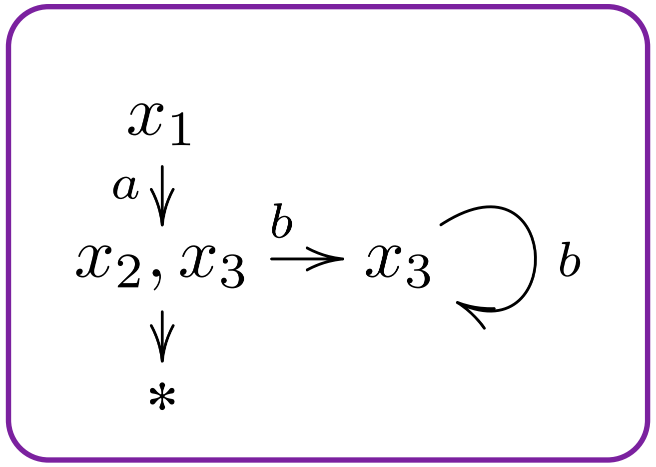


# Determinisations

NFA

$X \to 2 \times (\mathcal{P}(X))^A$

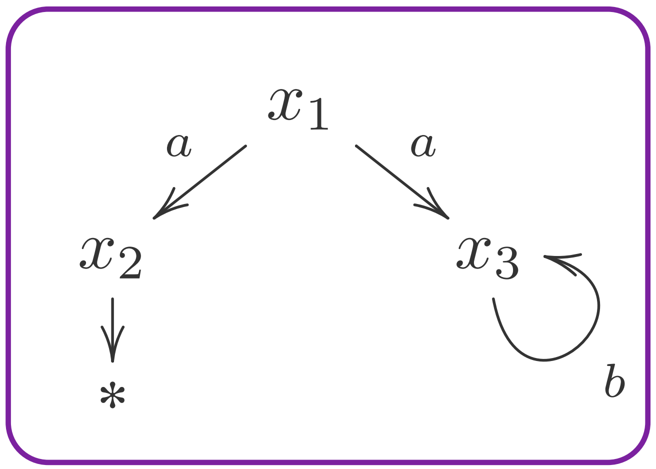


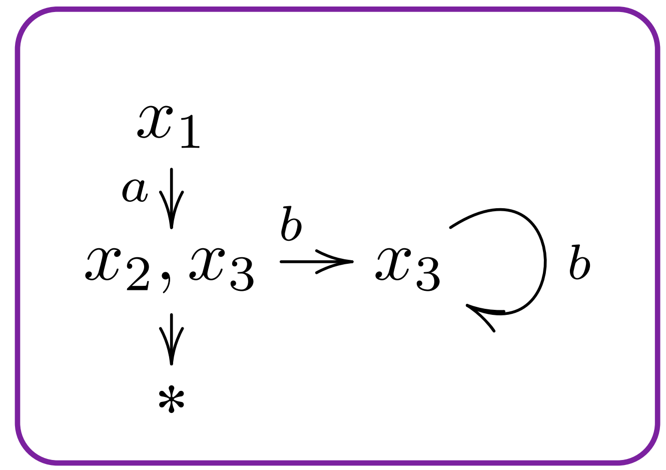



# Determinisations

NFA

$X \to 2 \times (\mathcal{P}(X))^A$



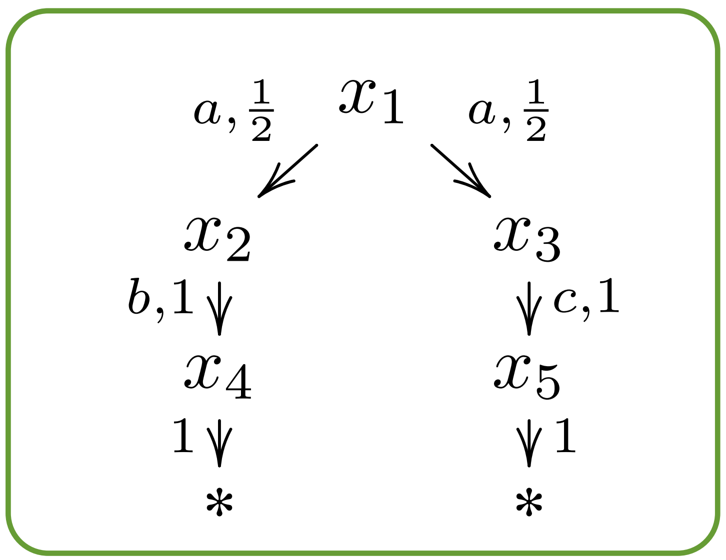

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations

# Determinisations

## Generative PTS

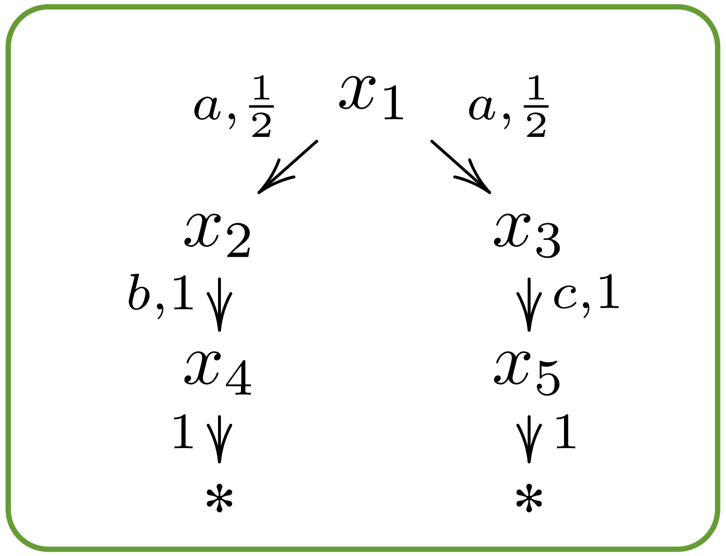
$X \to \mathcal{D}(1 + A \times X)$



# Determinisations

## Generative PTS

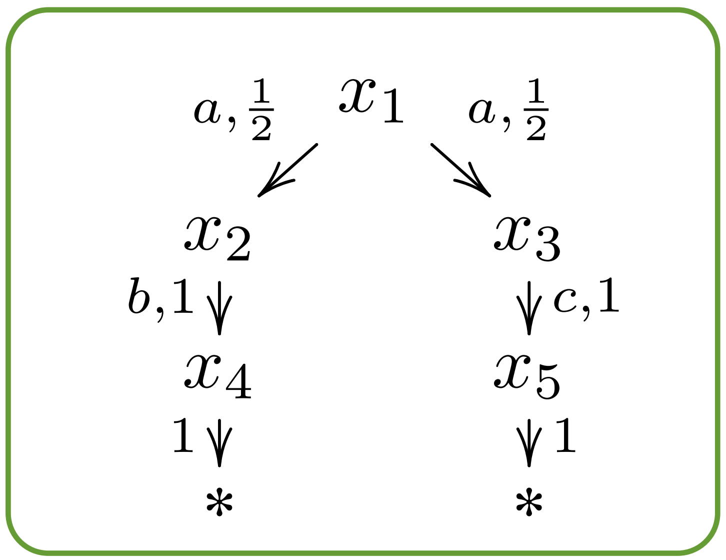
$X \to \mathcal{D}(1 + A \times X)$

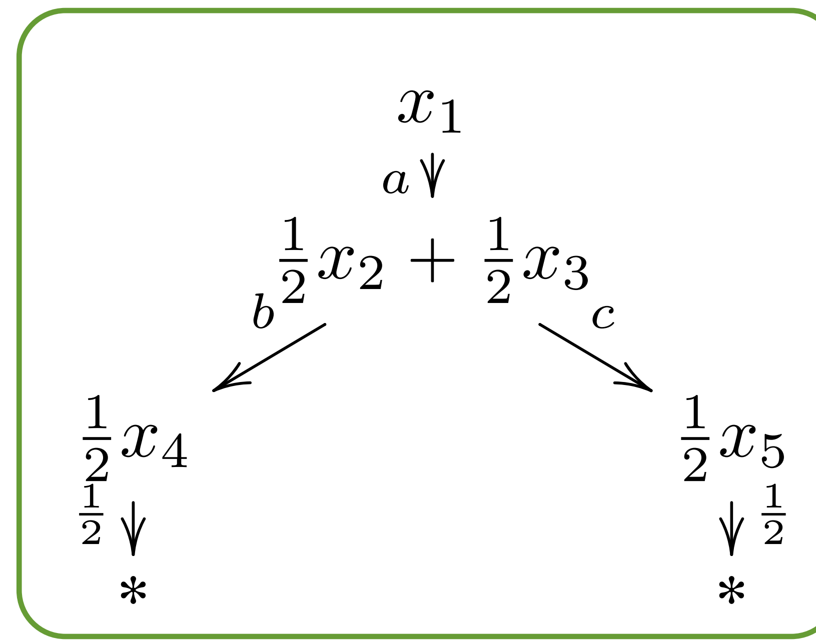


# Determinisations

## Generative PTS

$X \to \mathcal{D}(1 + A \times X)$

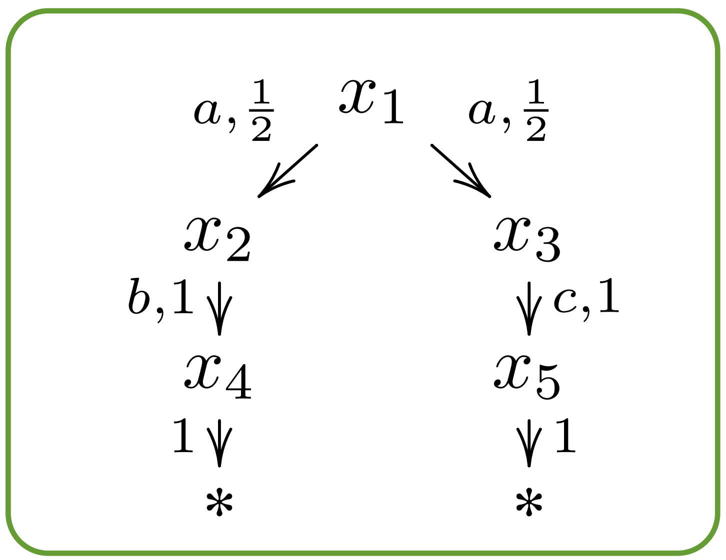


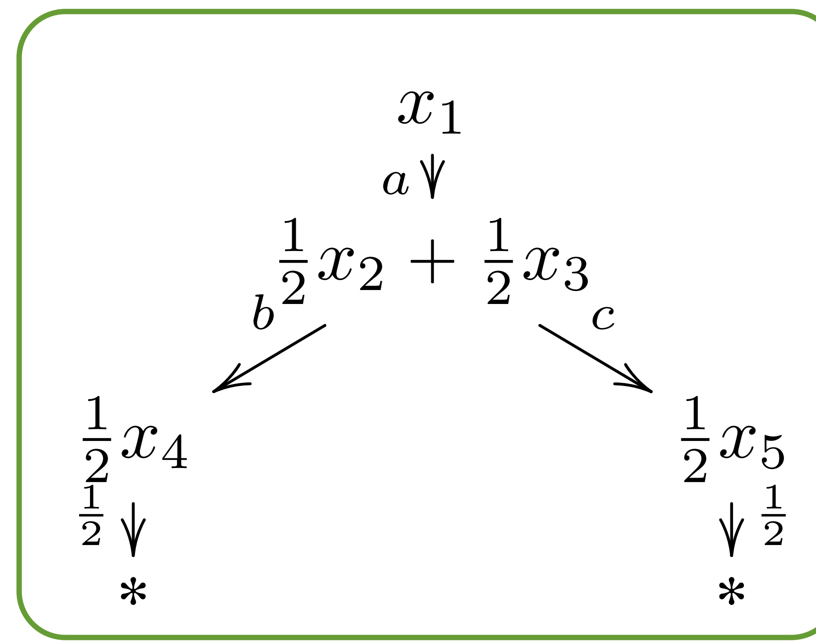



# Determinisations

## Generative PTS

$X \to \mathcal{D}(1 + A \times X)$




[Silva, S. MFPS'11]

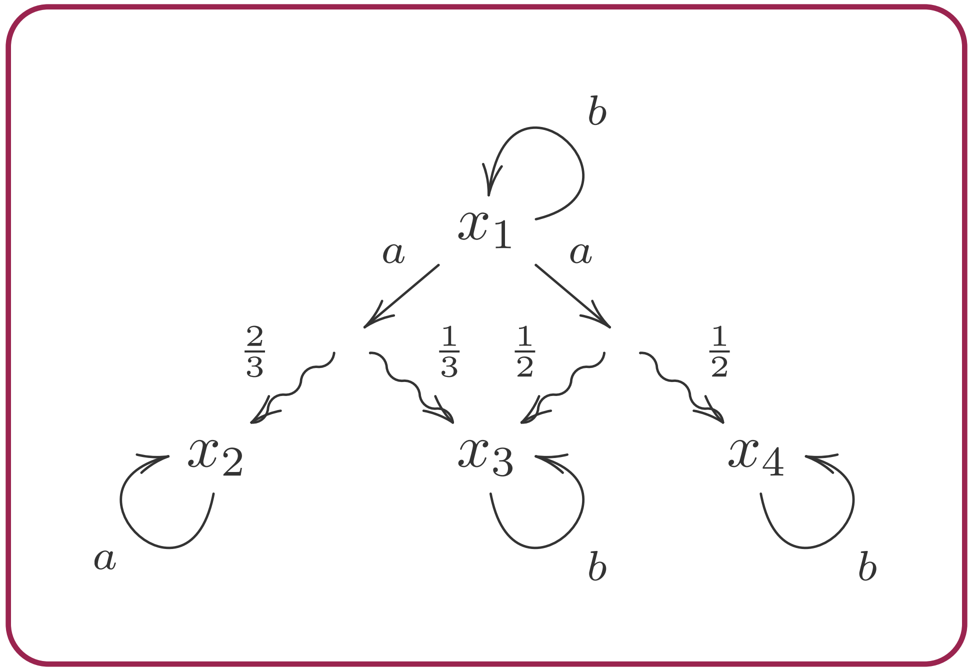
[Jacobs, Silva, S. JCSS'15]

# Determinisations

# Determinisations

PA

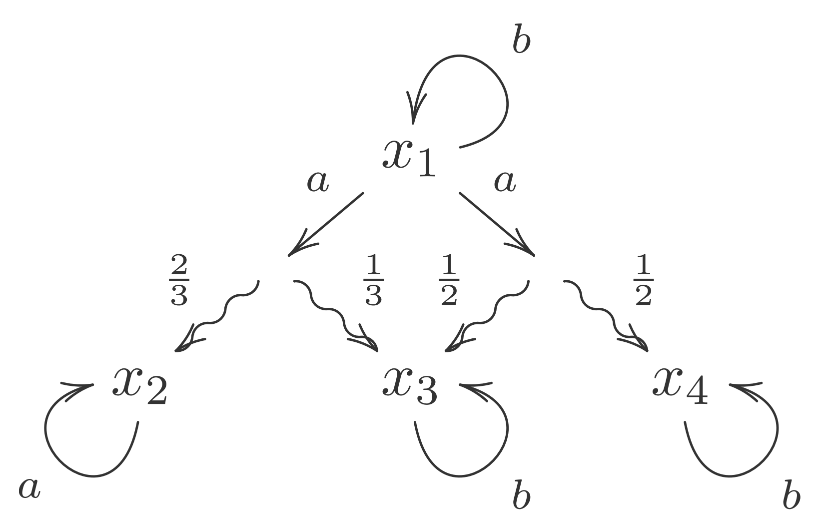
$X \to (\mathcal{PD}(X))^A$



# Determinisations

PA

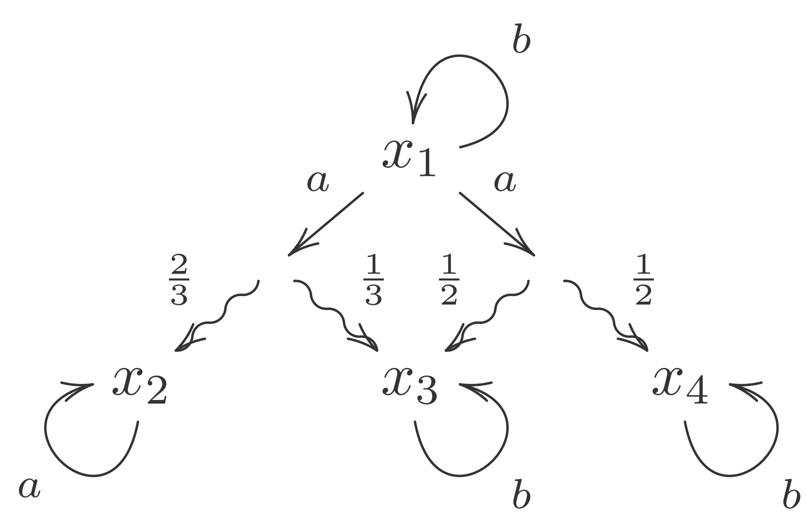
$X \to (\mathcal{PD}(X))^A$

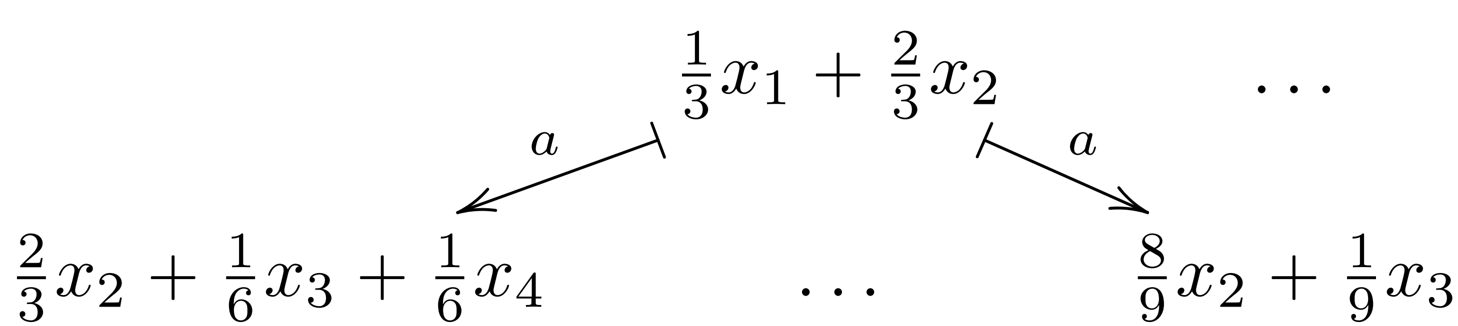



# Determinisations

PA

$X \to (\mathcal{PD}(X))^A$

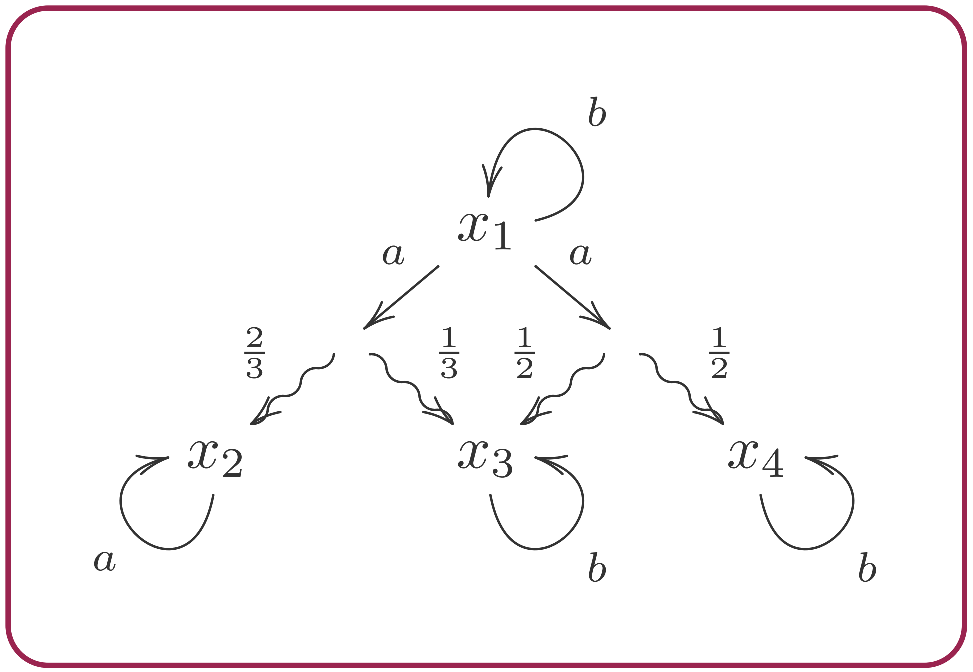


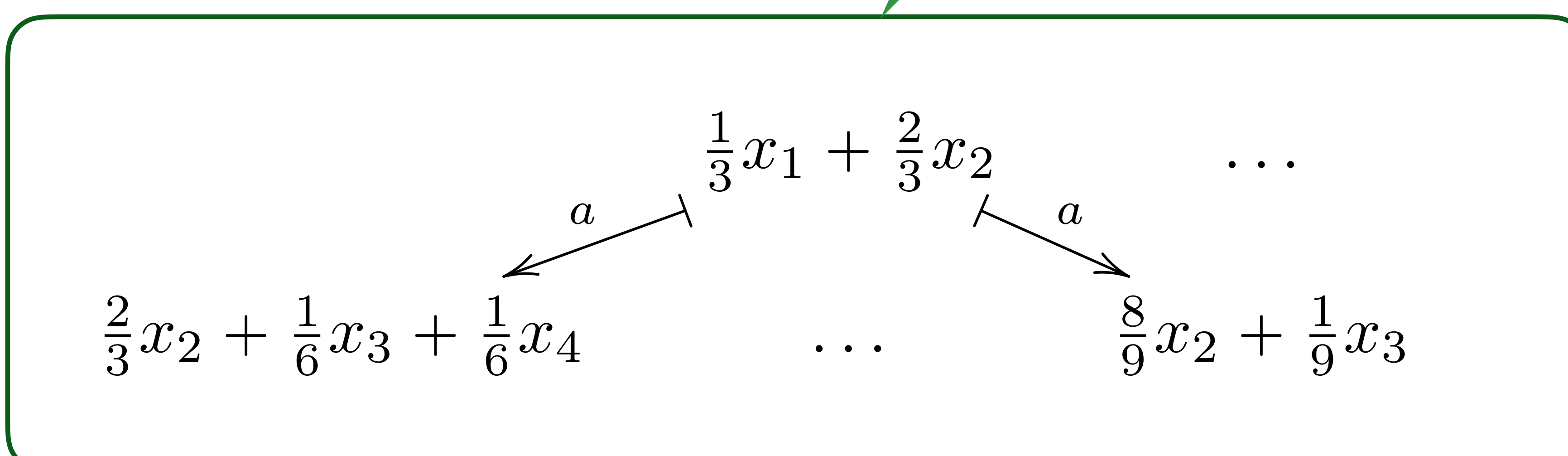
# Determinisations

PA

$X \to (\mathcal{PD}(X))^A$



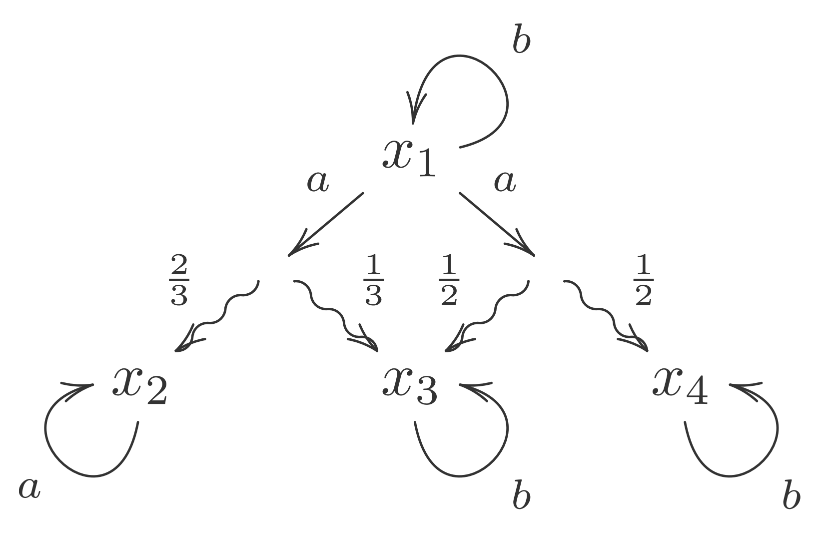

belief-state transformer



# Determinisations

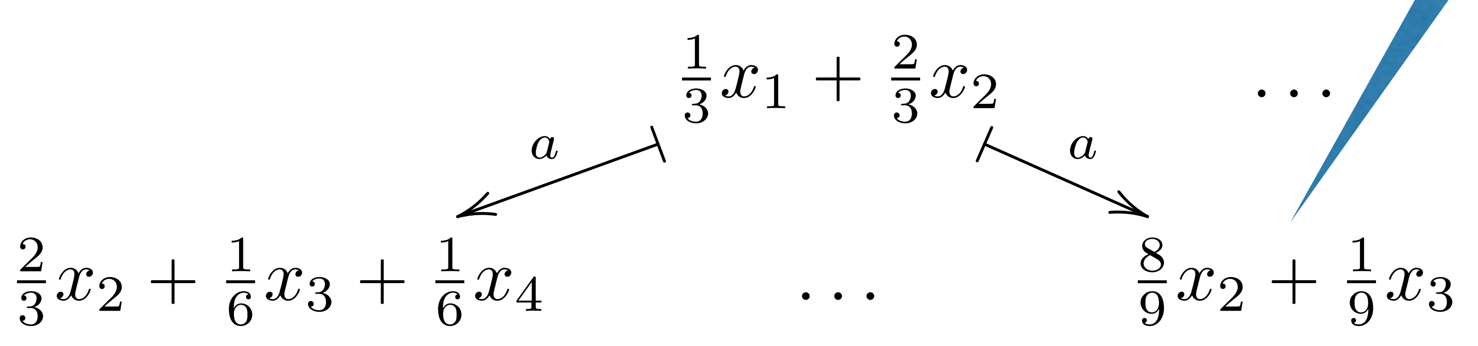
PA

$X \to (\mathcal{PD}(X))^A$




belief-state transformer

belief state

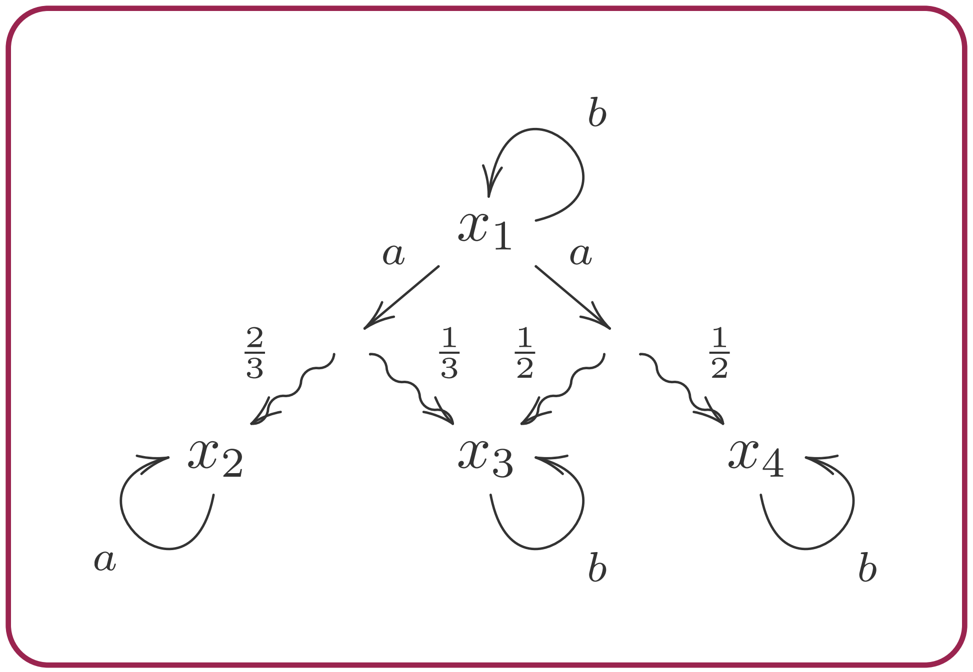


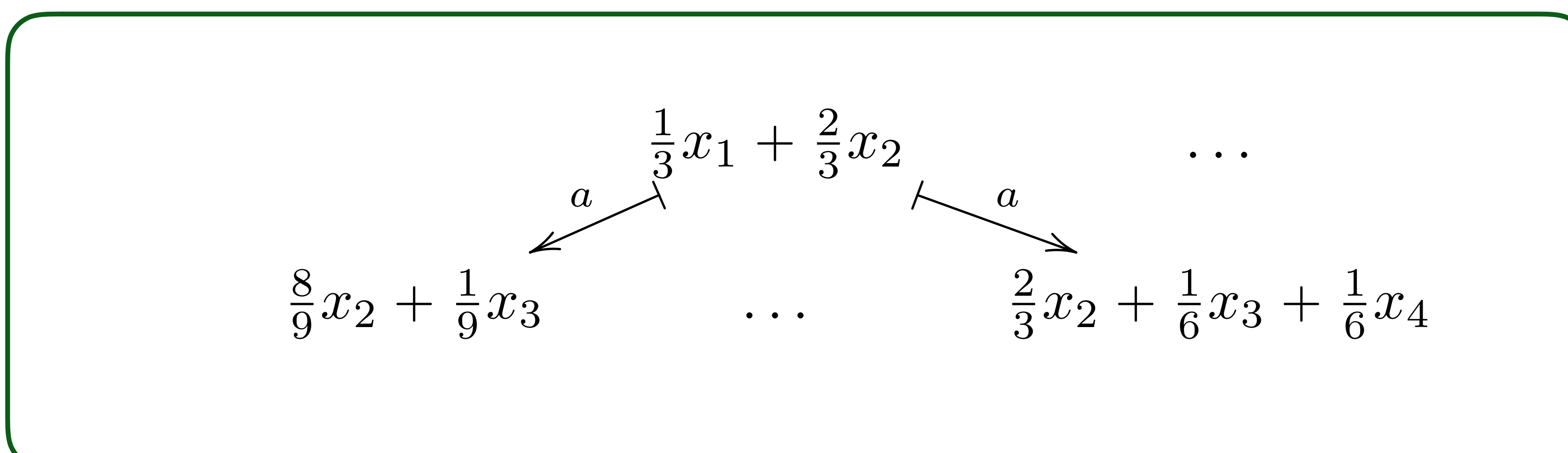
# Belief-state transformer

foundation ?

PA

X → $(\mathcal{PD}(X))^A$



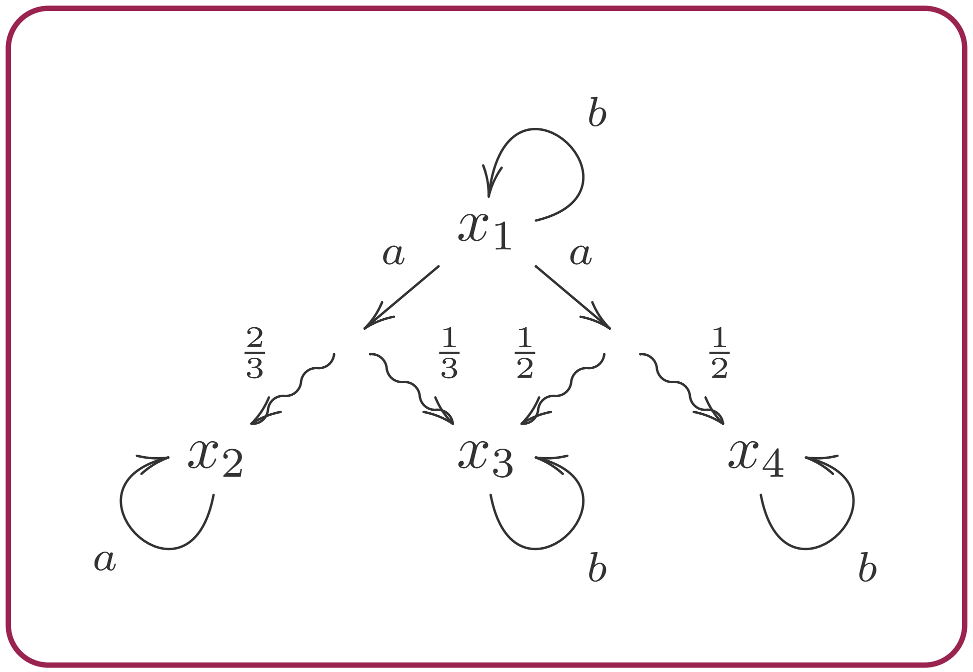



# Belief-state transformer

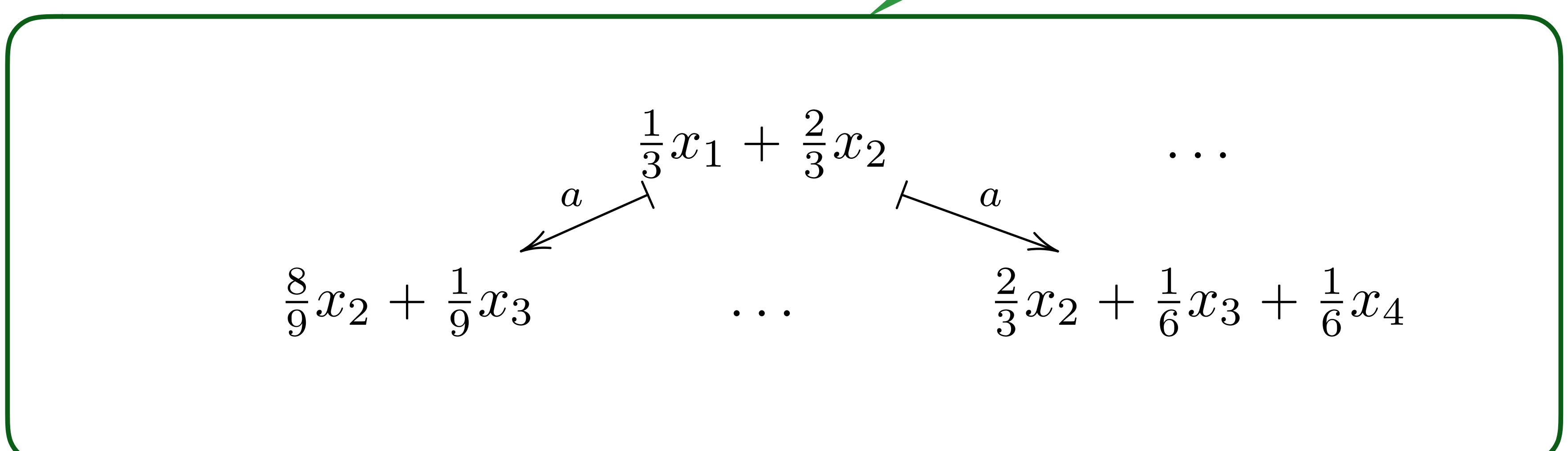
foundation ?

PA

$X \to (\mathcal{PD}(X))^A$




what is it?

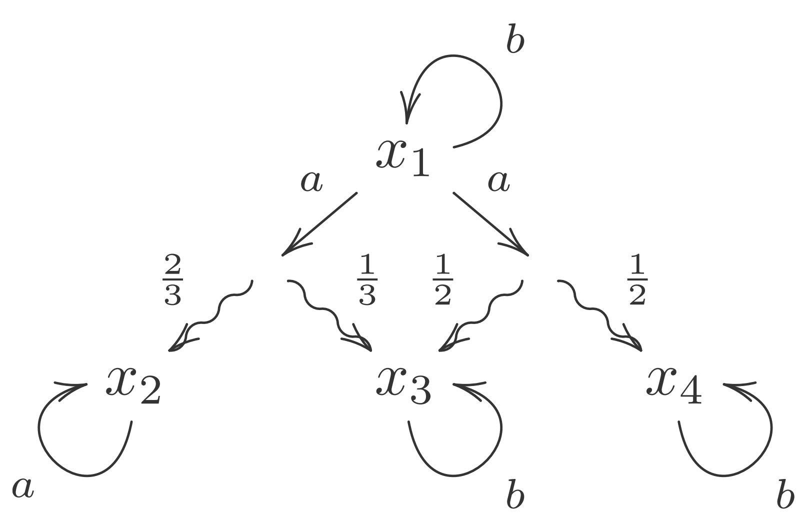


# Belief-state transformer

foundation ?

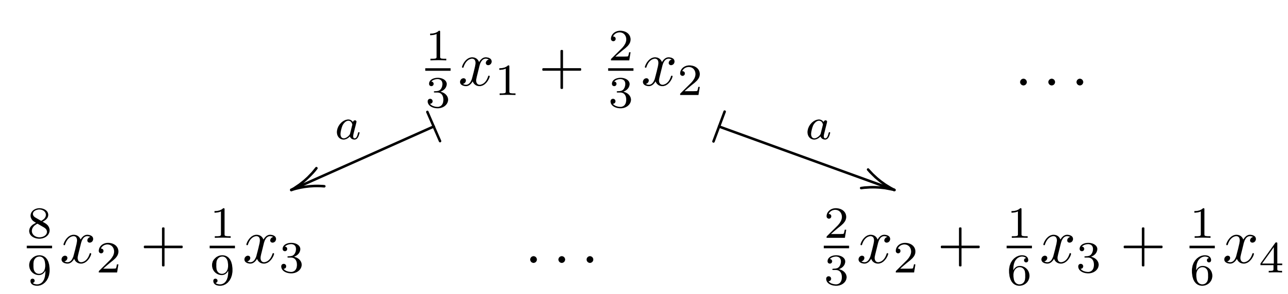
PA

$X \to (\mathcal{PD}(X))^A$

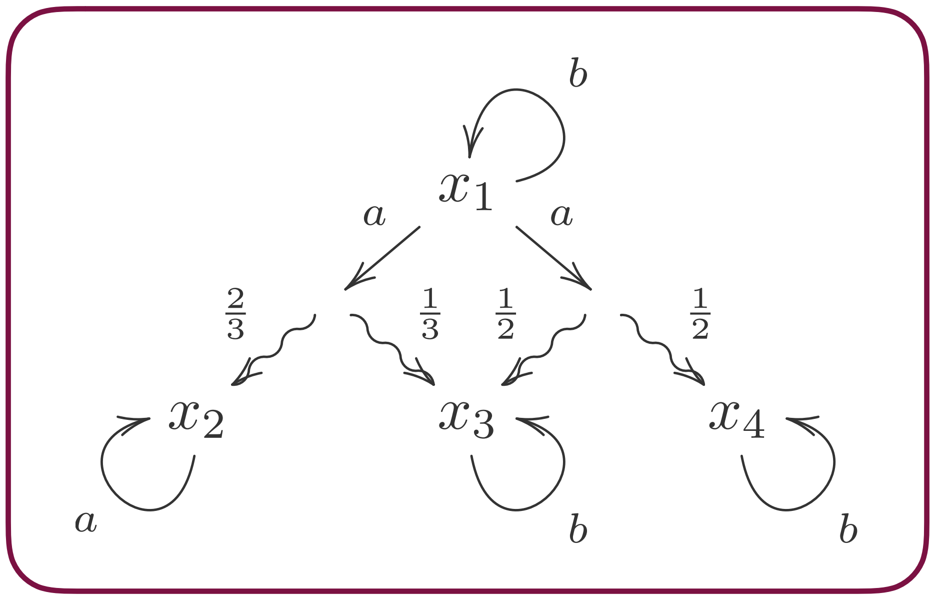


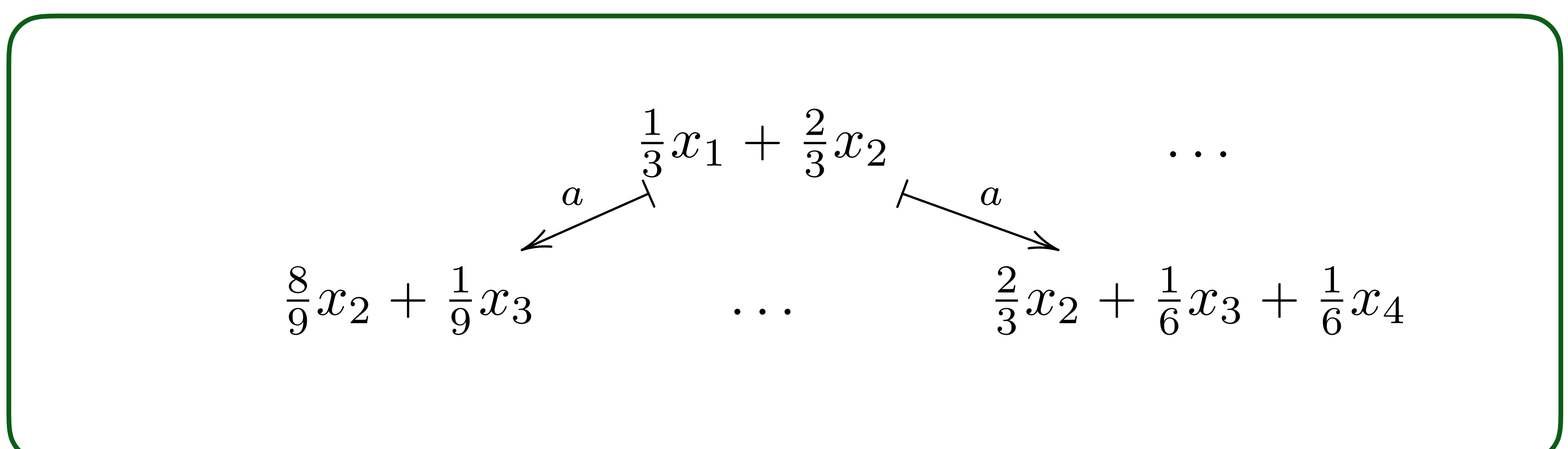

how does it emerge?

what is it?

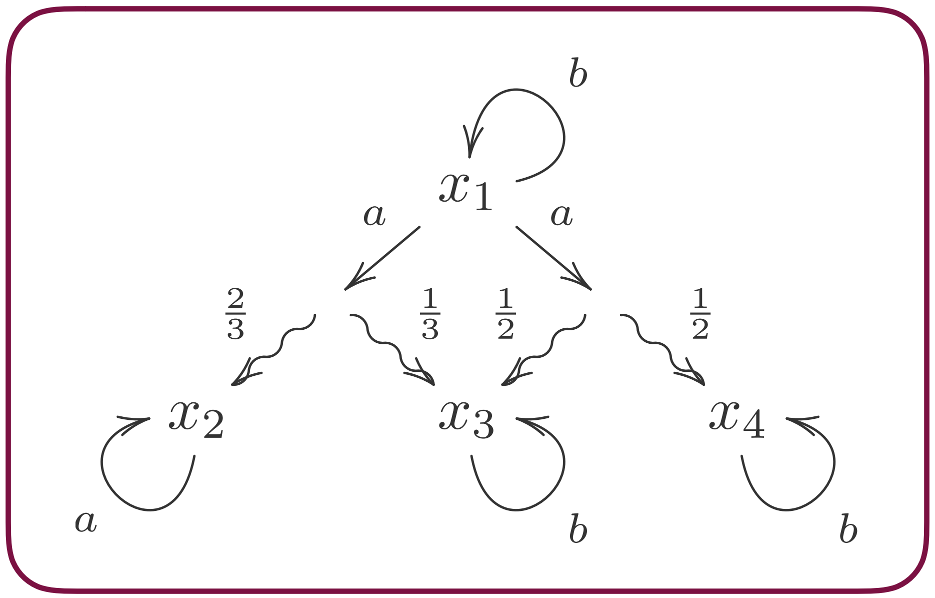


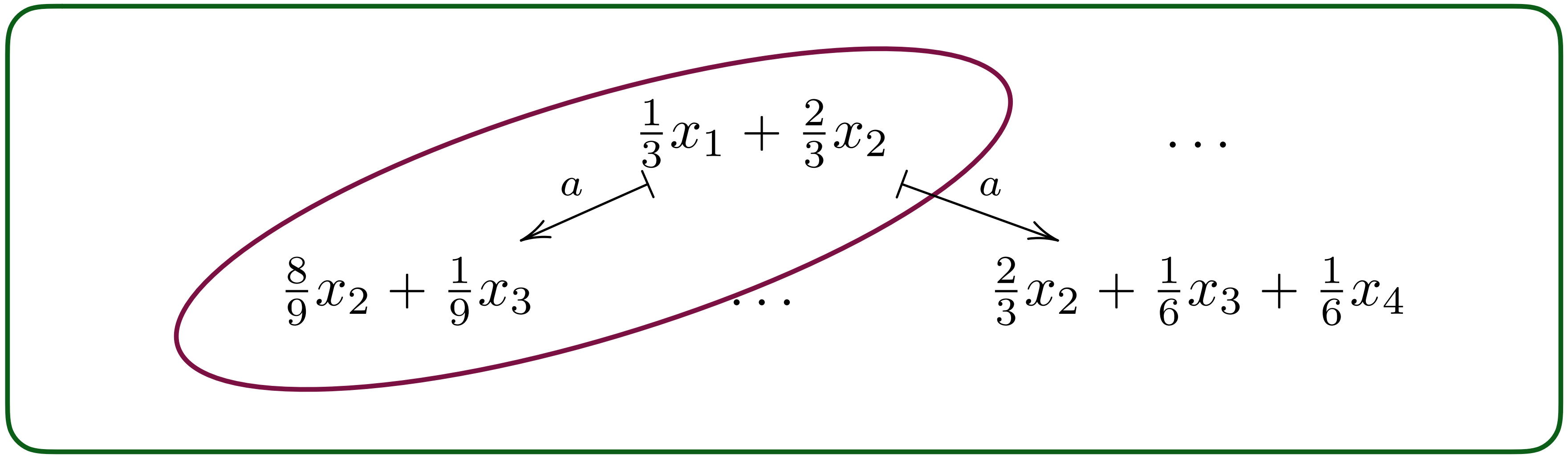
# Belief-state transformer



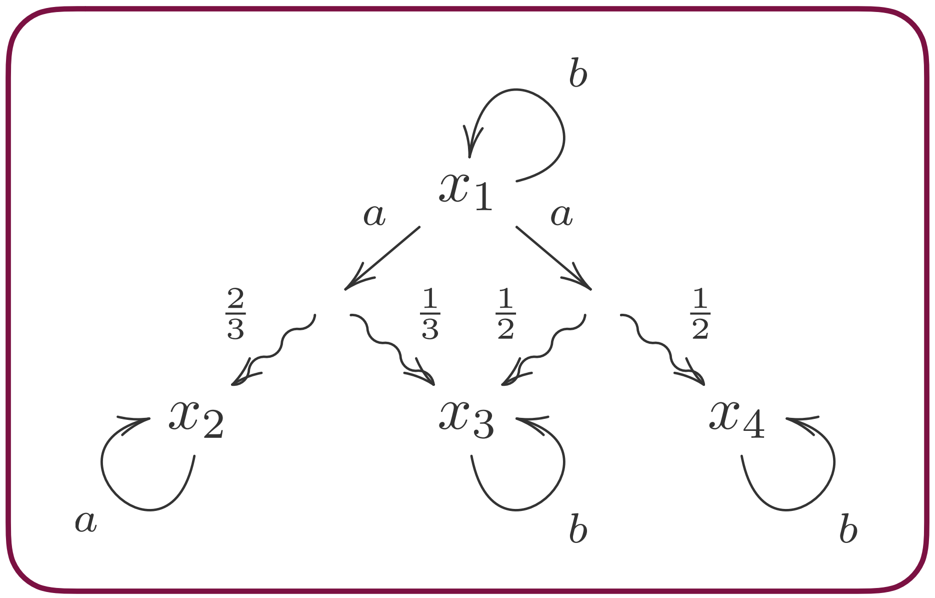



# Belief-state transformer

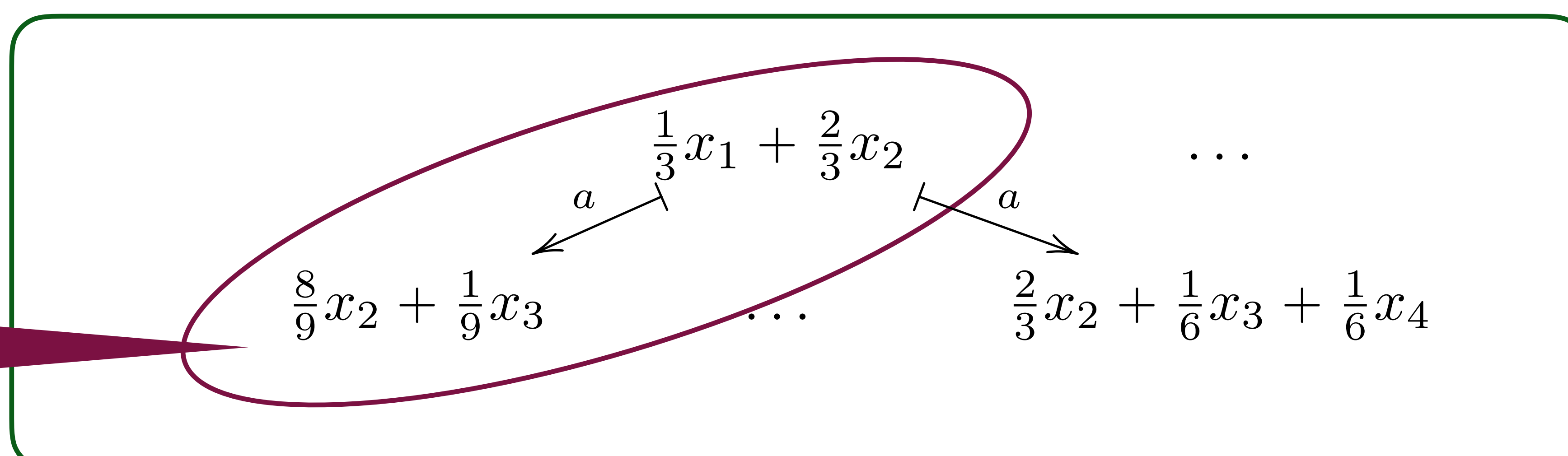


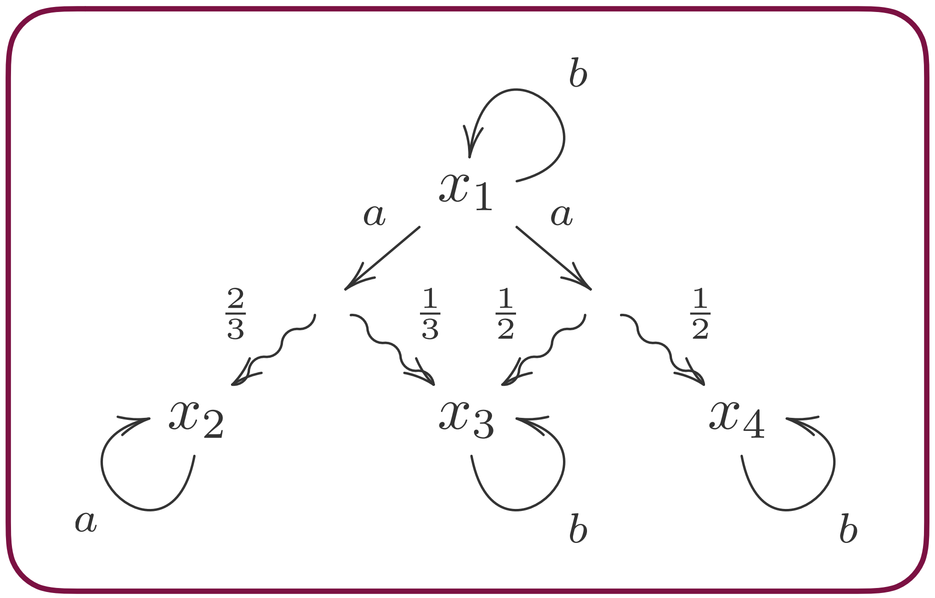
# Belief-state transformer

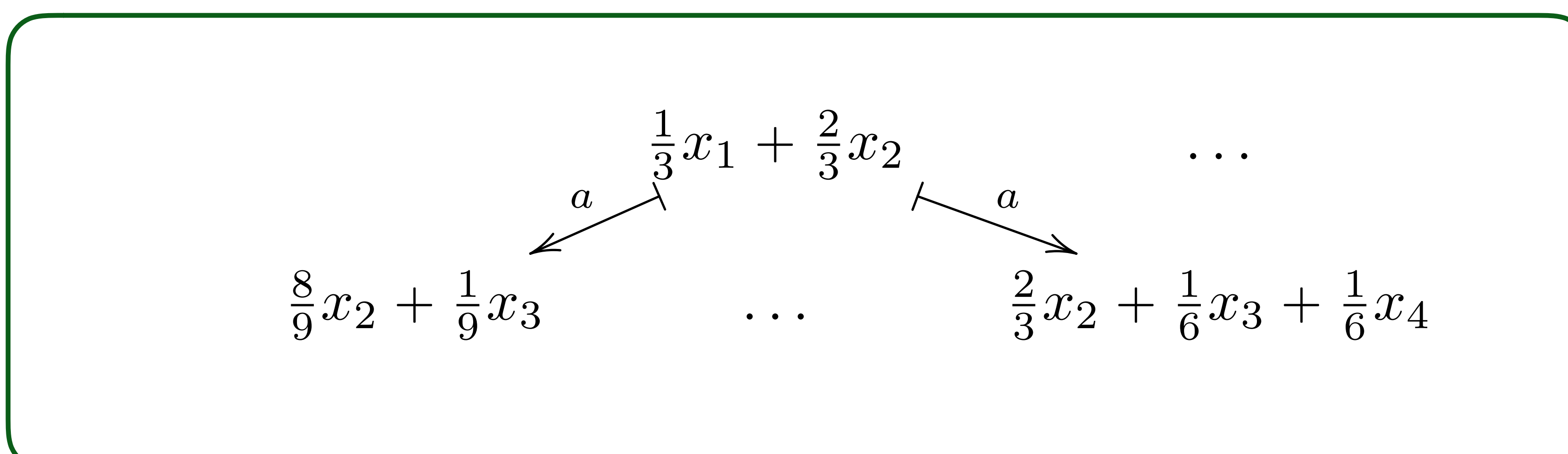



$$\frac{1}{3}\left(\frac{2}{3}x_2 + \frac{1}{3}x_3\right) + \frac{2}{3}(1x_2)$$

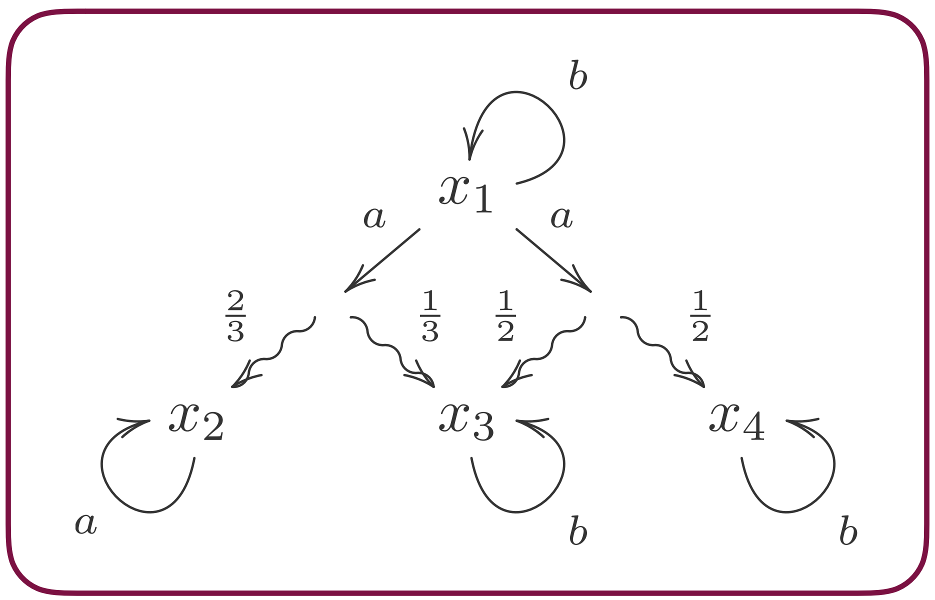


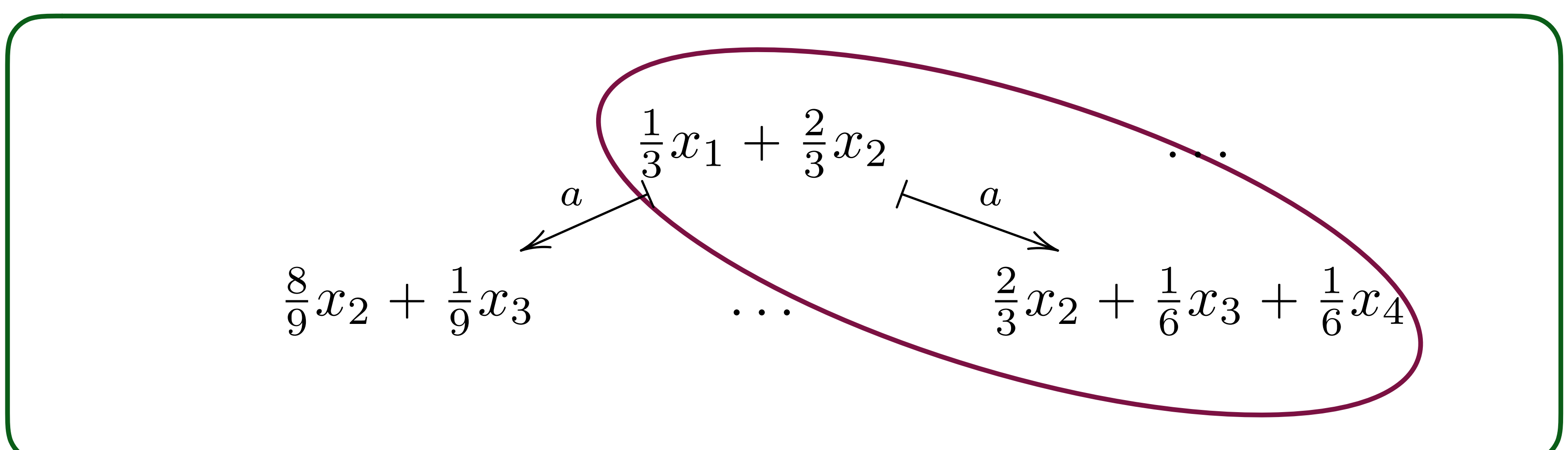
# Belief-state transformer



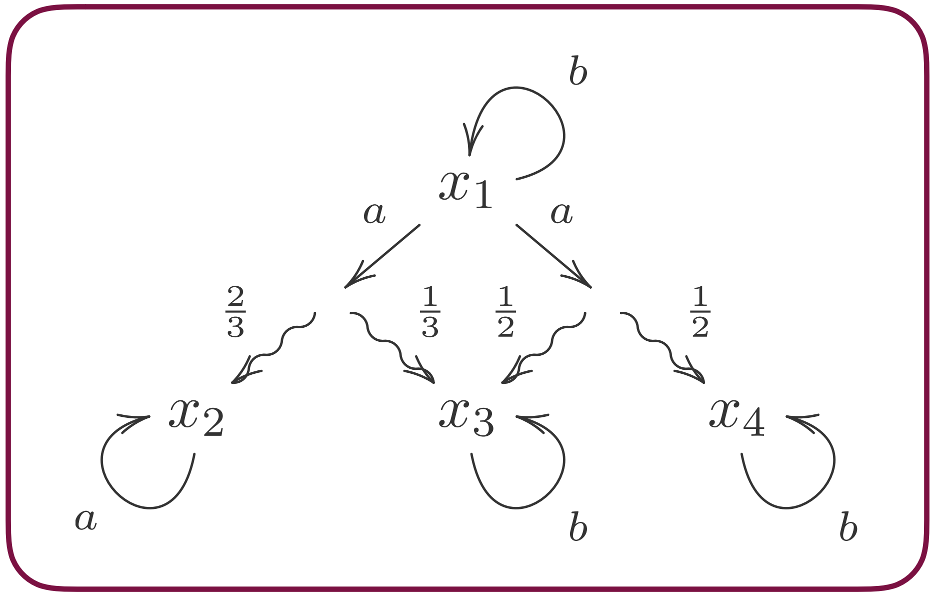



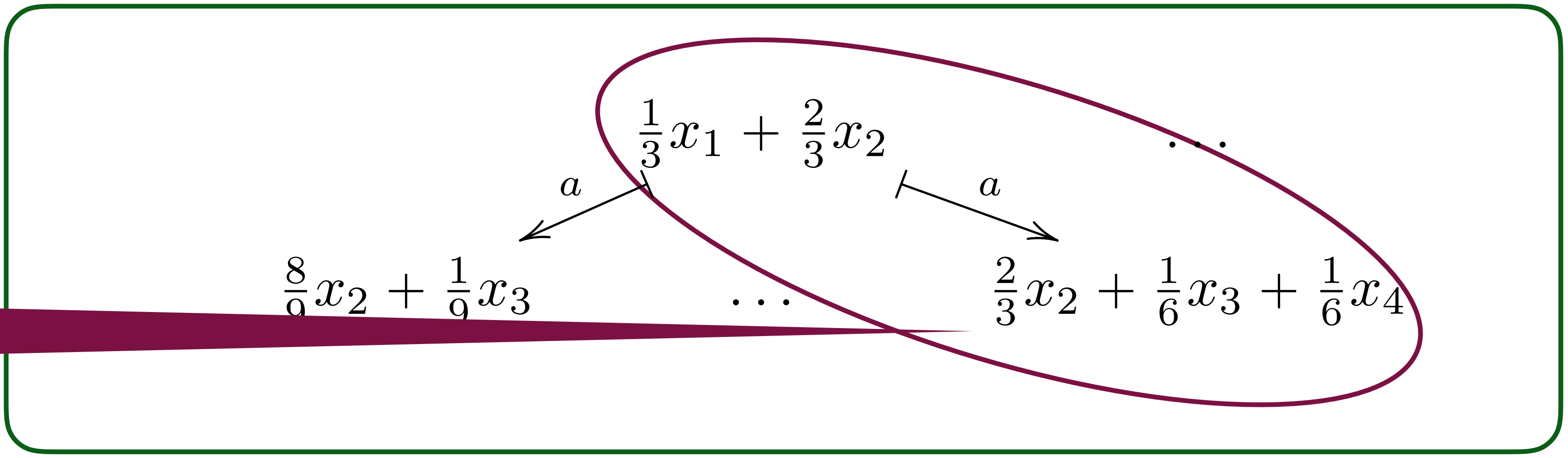
# Belief-state transformer



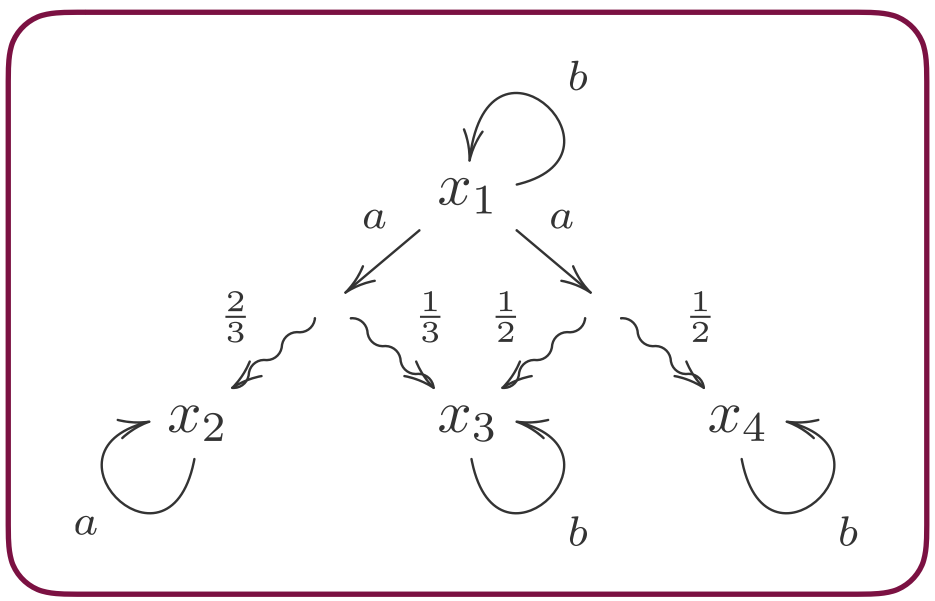
# Belief-state transformer

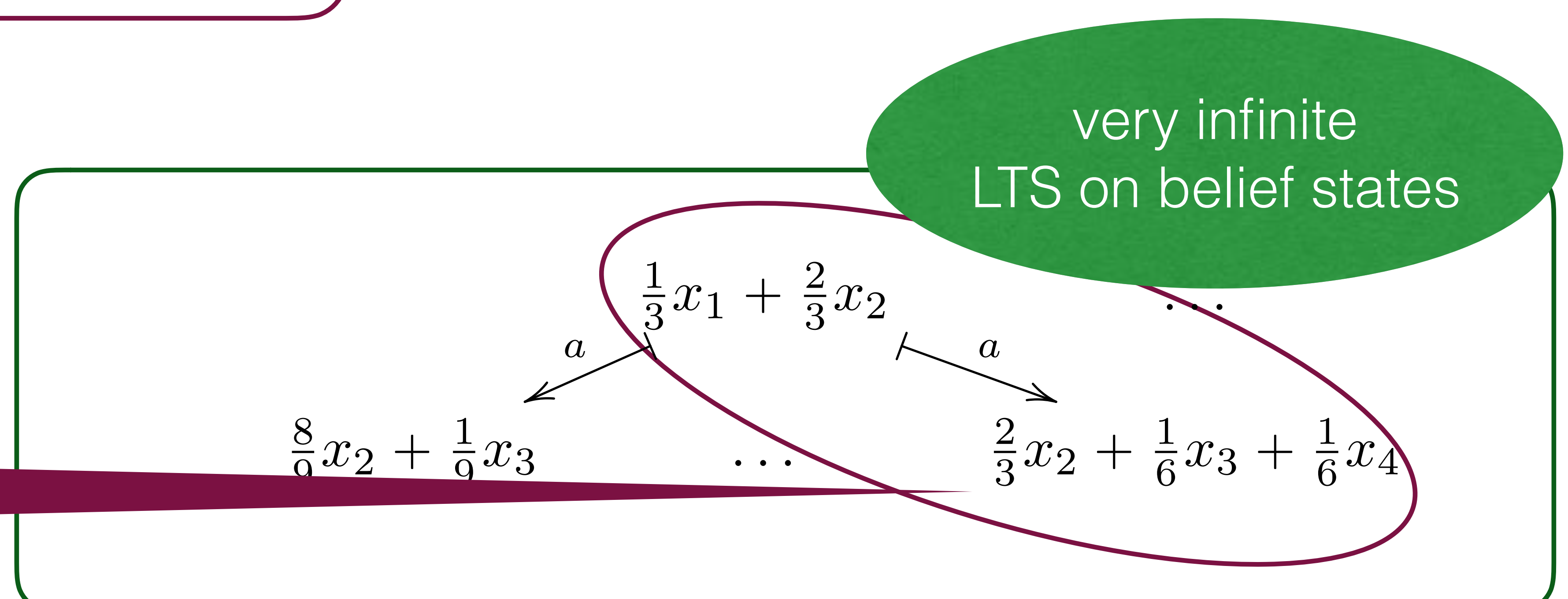



$$\frac{1}{3}\left(\frac{1}{2}x_3 + \frac{1}{2}x_4\right) + \frac{2}{3}(1x_2)$$

# Belief-state transformer




$$\frac{1}{3}\left(\frac{1}{2}x_3 + \frac{1}{2}x_4\right) + \frac{2}{3}(1x_2)$$

# Probabilistic Automata

# Probabilistic Automata

Can be given different semantics:

1. Bisimilarity
2. Convex bisimilarity
3. Distribution bisimilarity

# Probabilistic Automata

Can be given different semantics:

1. Bisimilarity
2. Convex bisimilarity
3. Distribution bisimilarity

strong bisimilarity

# Probabilistic Automata

Can be given different semantics:

1. Bisimilarity
2. Convex bisimilarity
3. Distribution bisimilarity

strong bisimilarity

probabilistic / combined bisimilarity

# Probabilistic Automata

Can be given different semantics:

1. Bisimilarity — strong bisimilarity
2. Convex bisimilarity — probabilistic / combined bisimilarity
3. Distribution bisimilarity — belief-state bisimilarity

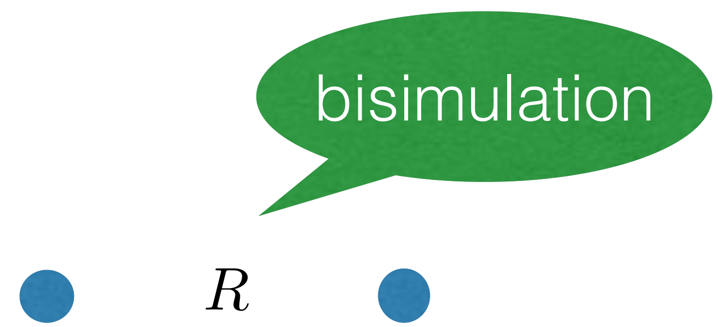
# Bisimilarity

# Bisimilarity

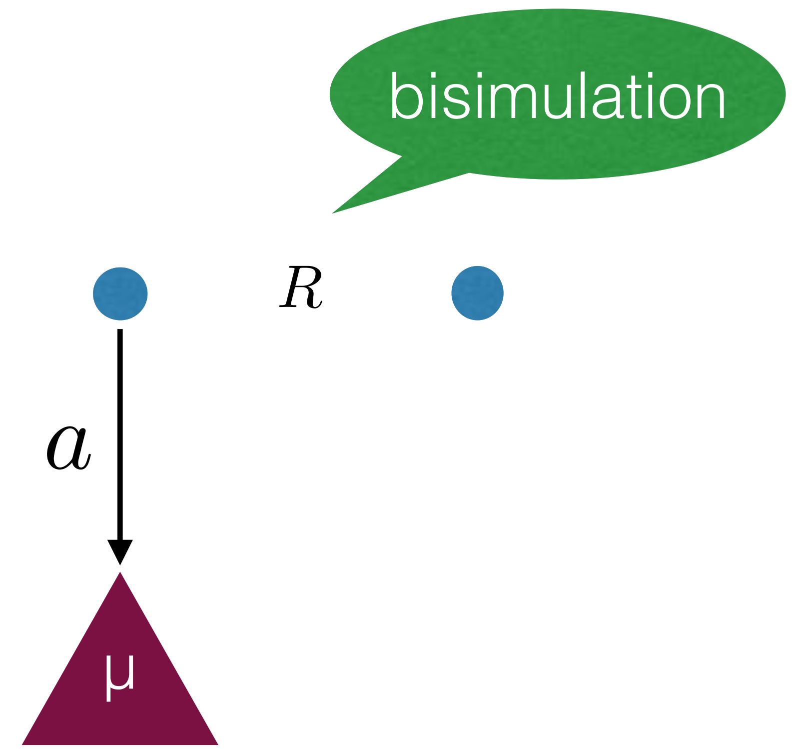
bisimulation

$R$

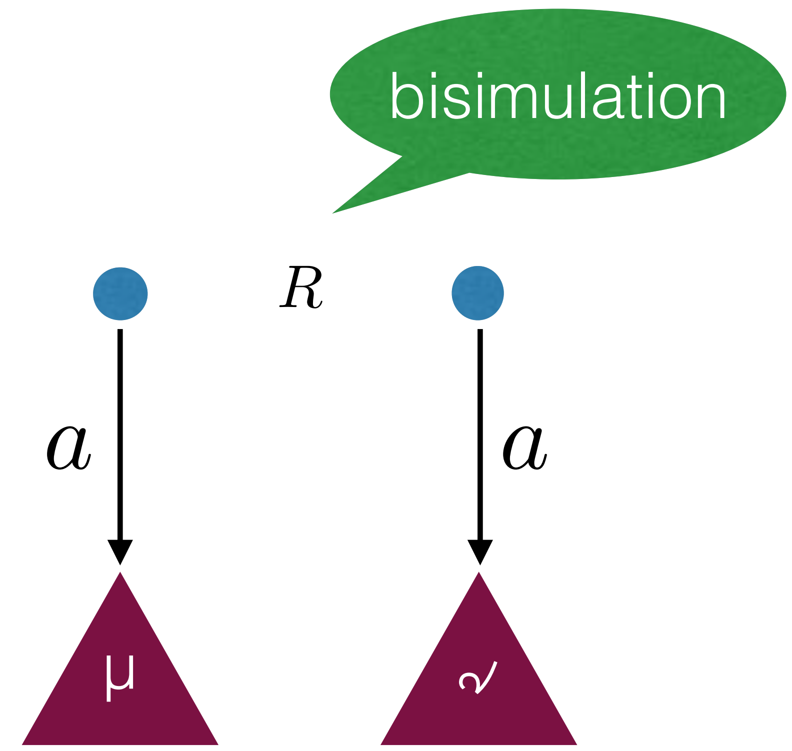
# Bisimilarity



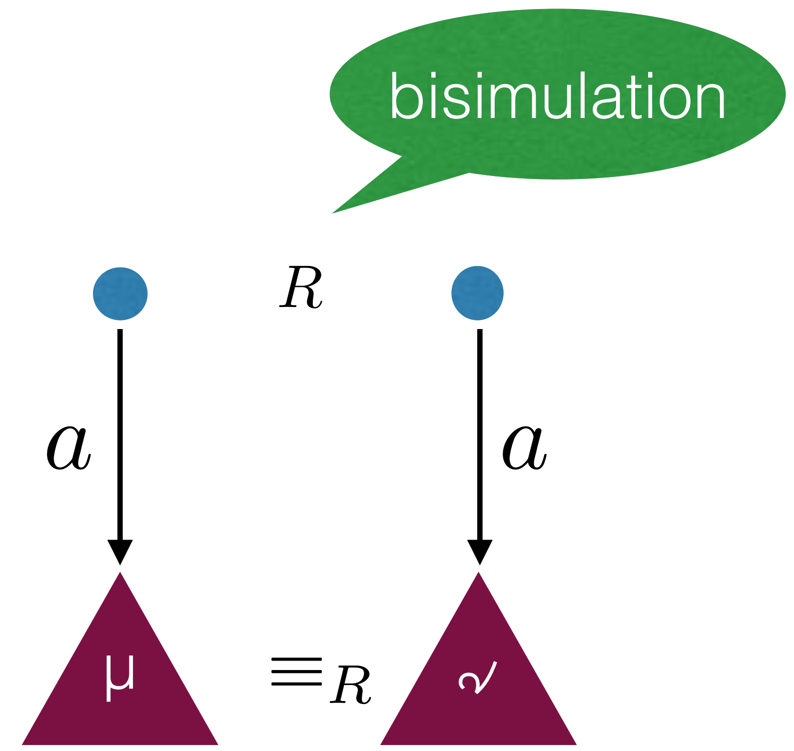
# Bisimilarity



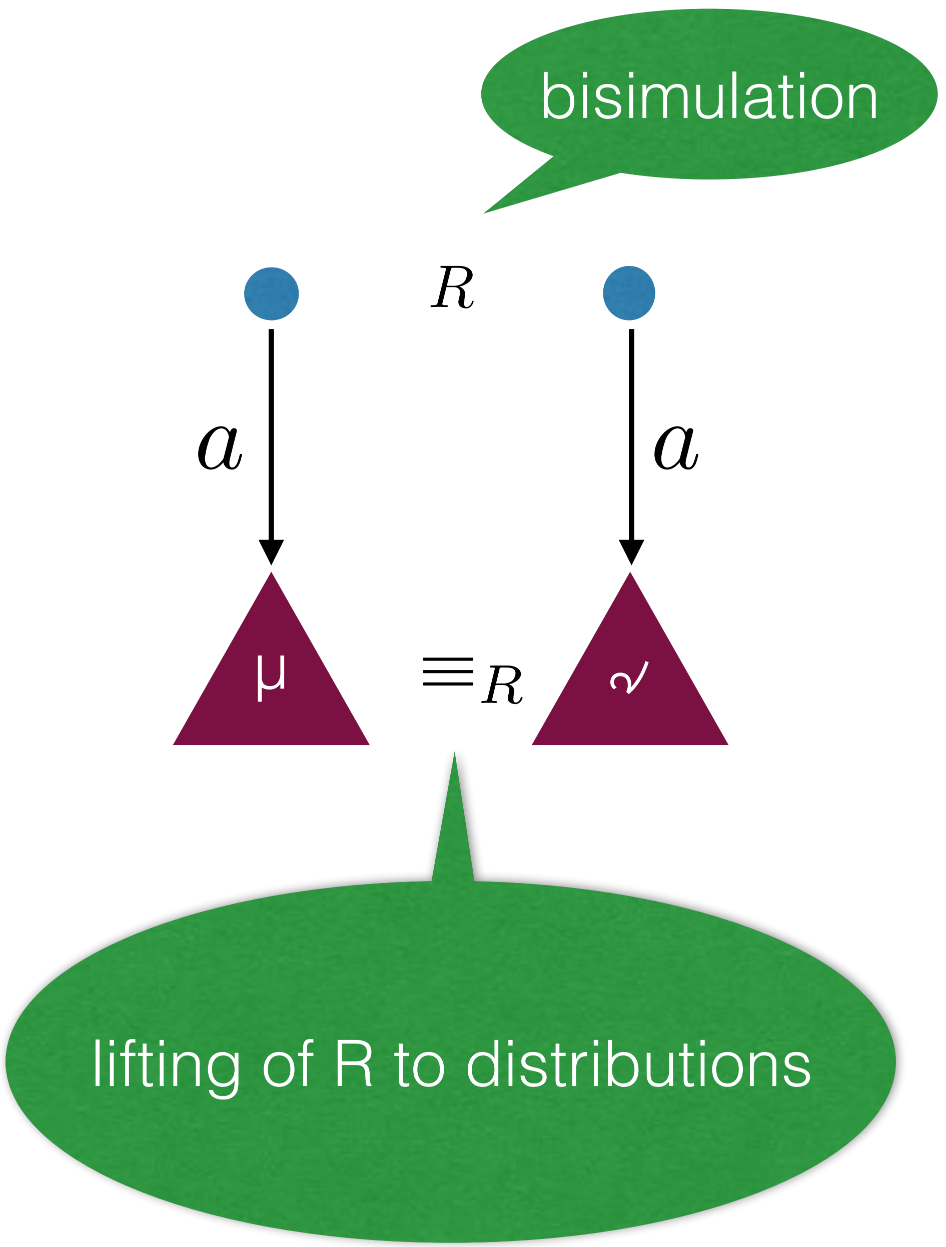
# Bisimilarity



# Bisimilarity

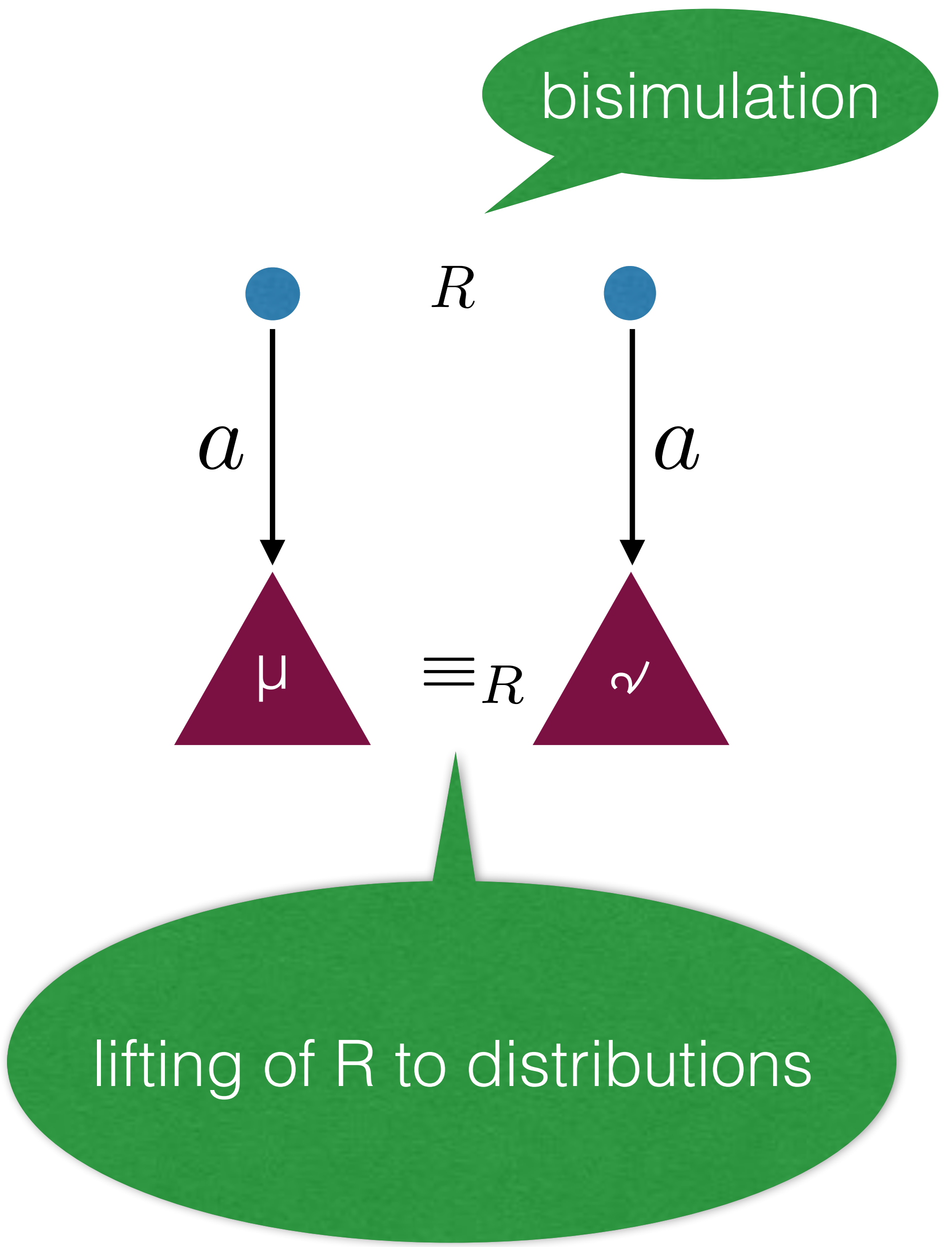


# Bisimilarity



# Bisimilarity

$\sim$ largest bisimulation

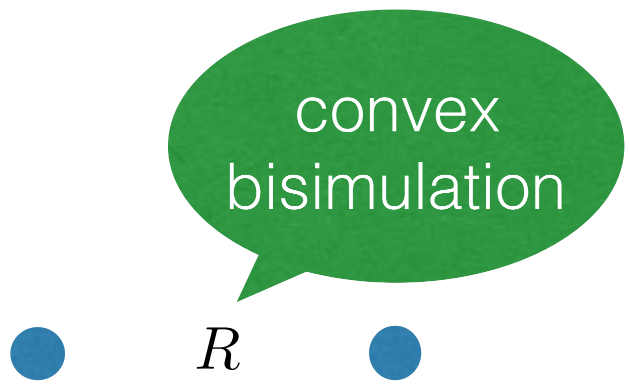


# Convex bisimilarity

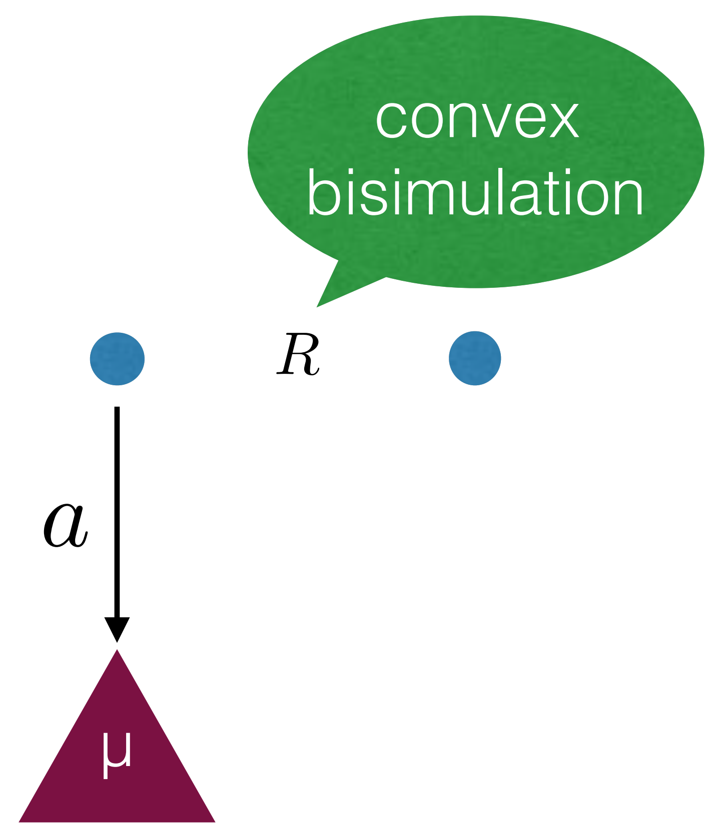
# Convex bisimilarity


$R$

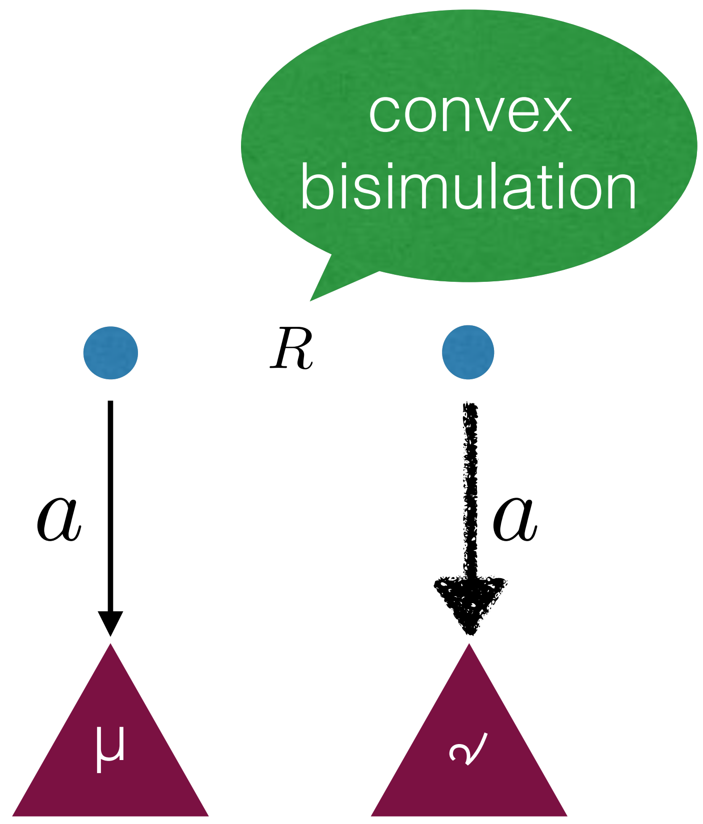
# Convex bisimilarity



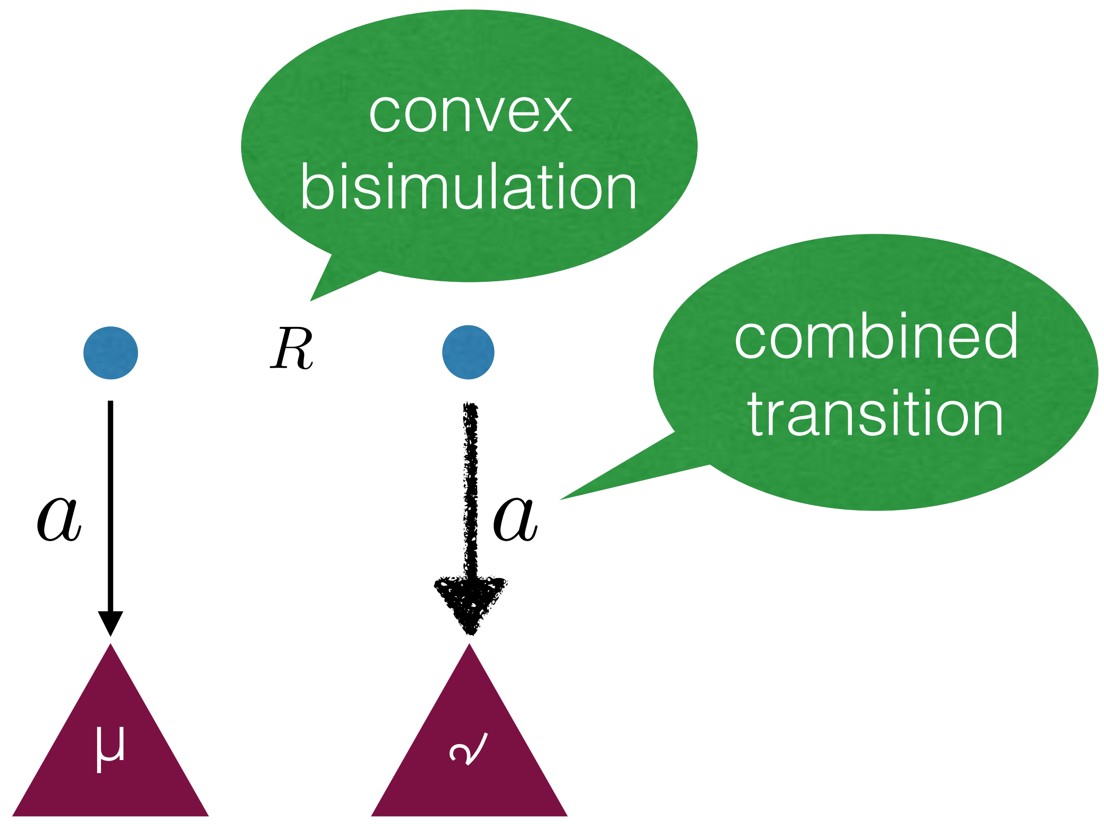
# Convex bisimilarity



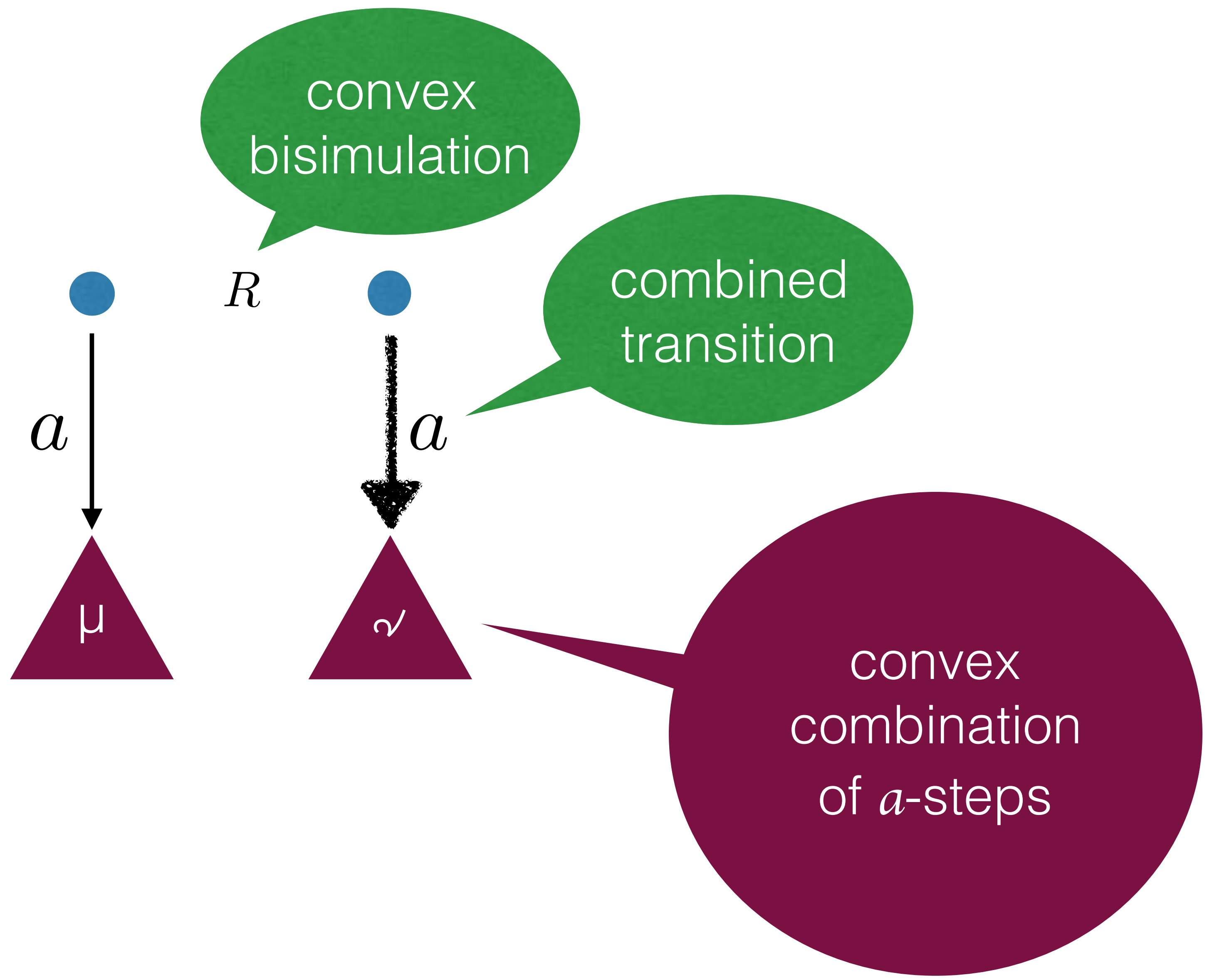
# Convex bisimilarity



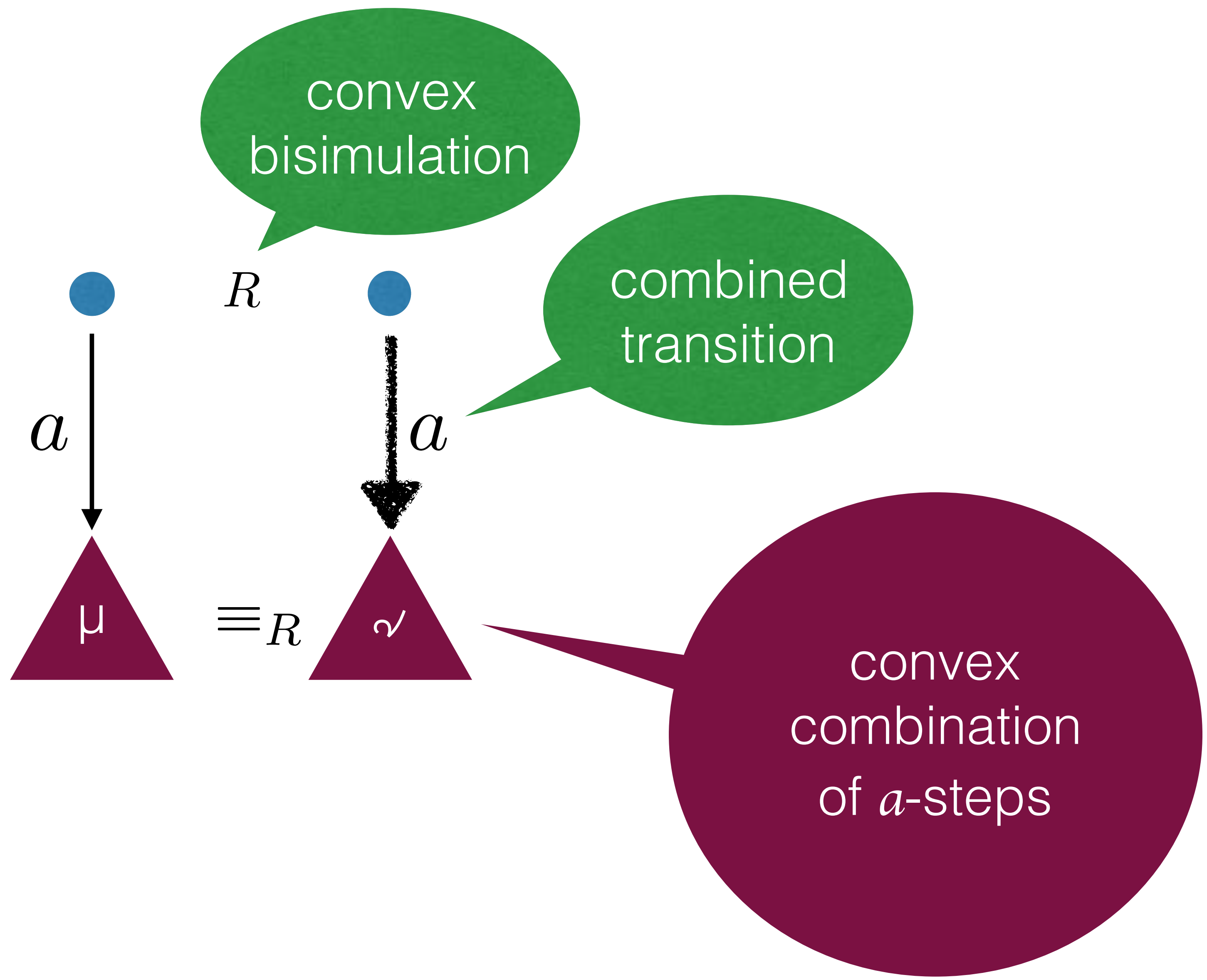
# Convex bisimilarity



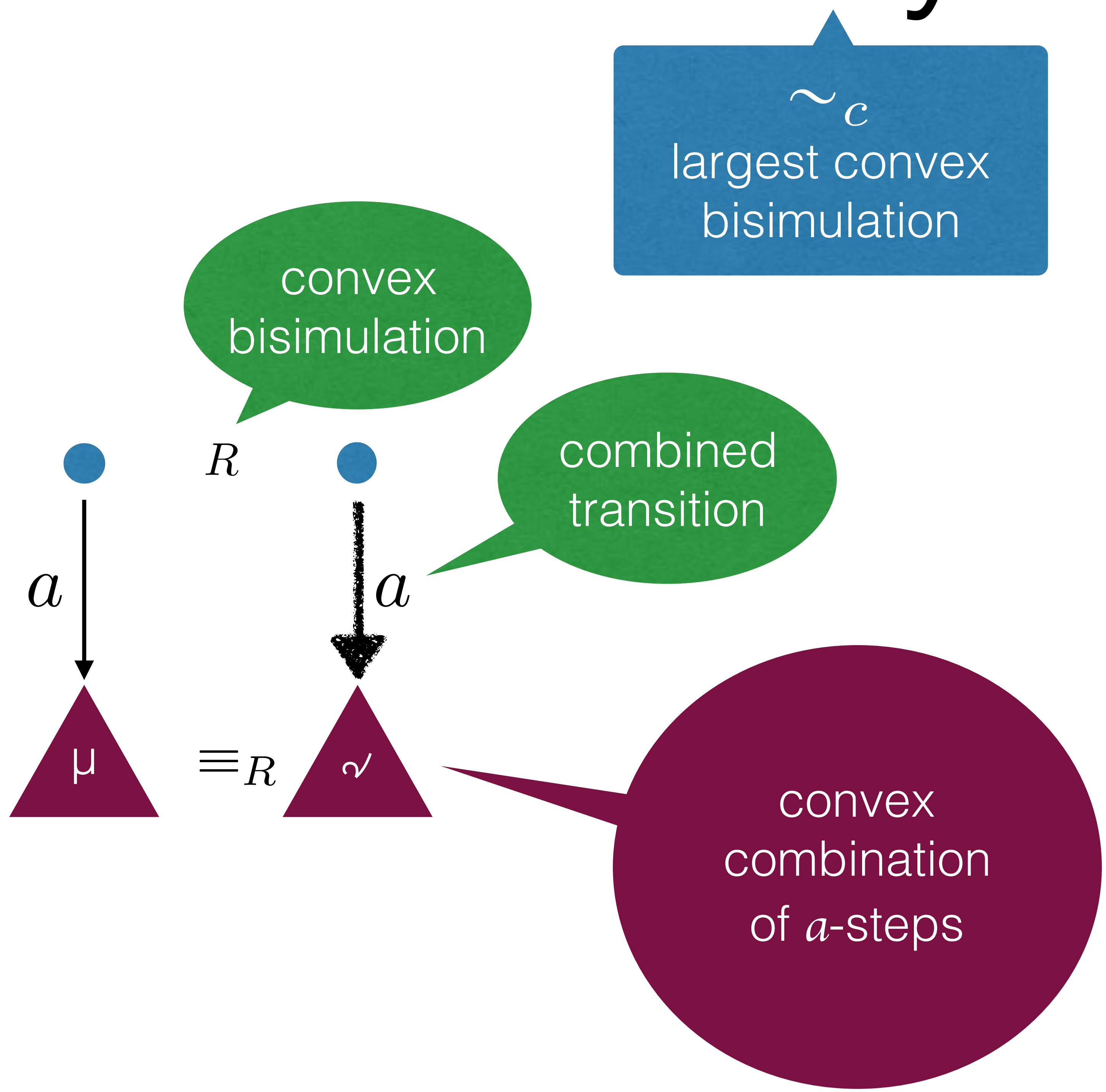
# Convex bisimilarity



# Convex bisimilarity



# Convex bisimilarity



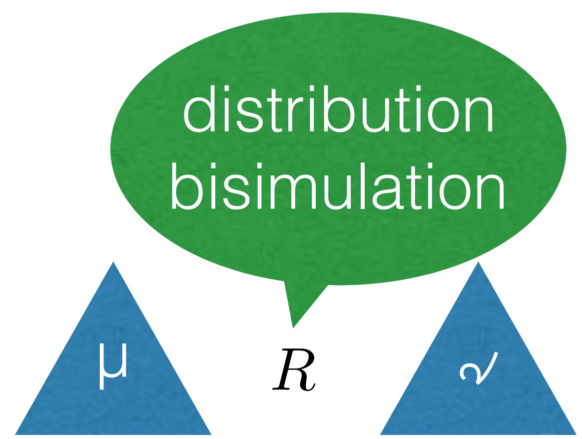
# Distribution bisimilarity

# Distribution bisimilarity

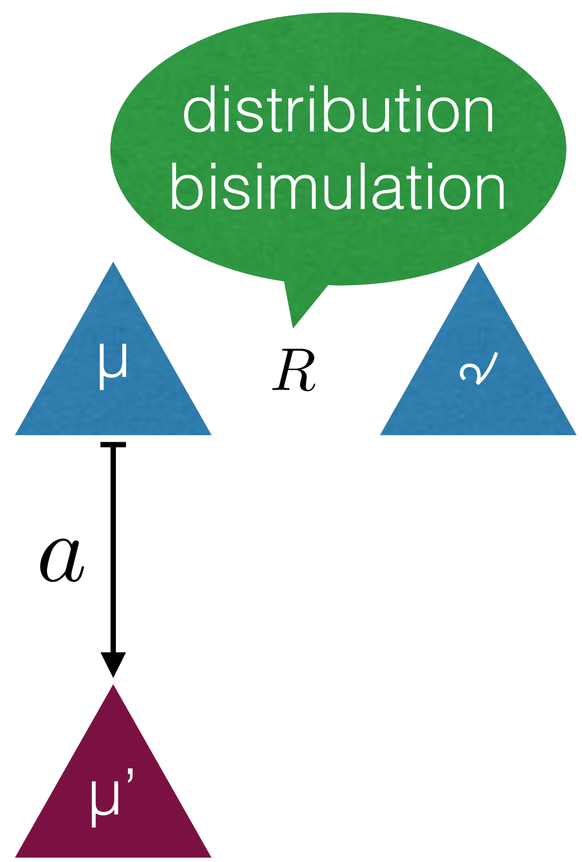
distribution bisimulation

$R$

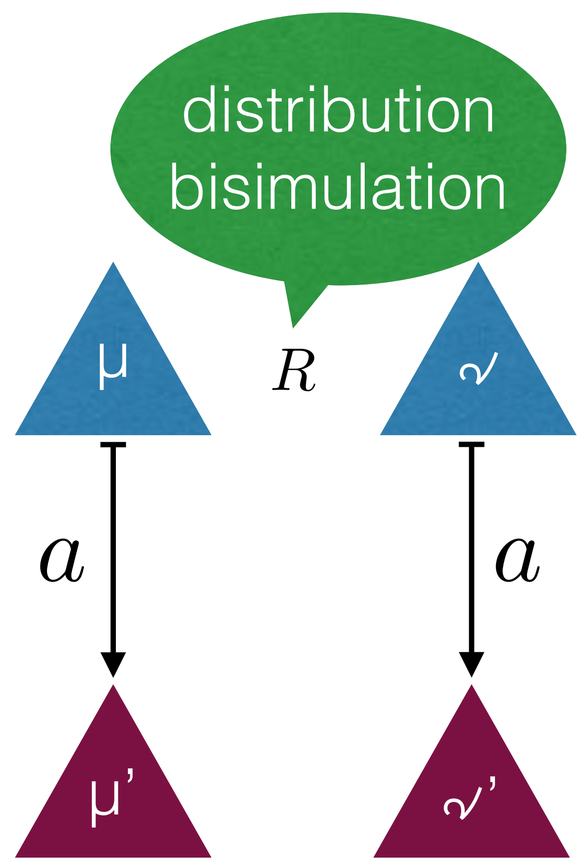
# Distribution bisimilarity



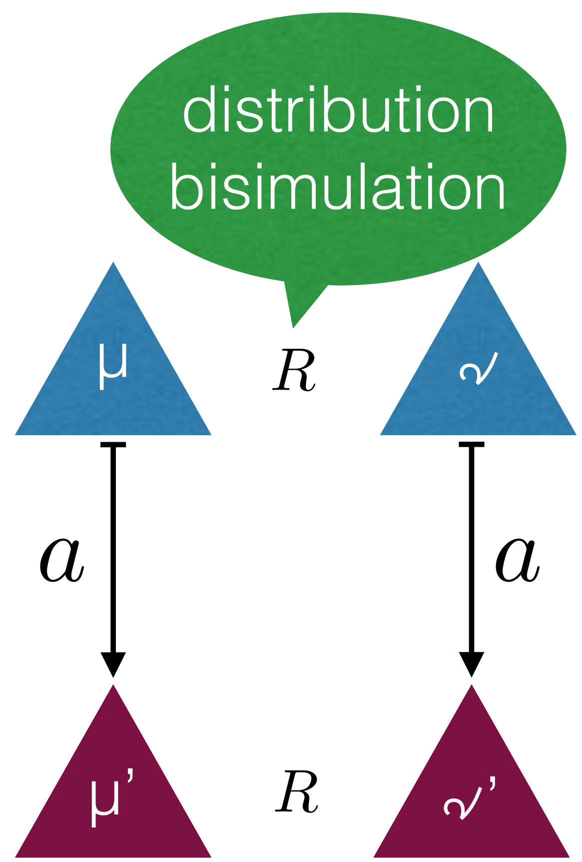
# Distribution bisimilarity



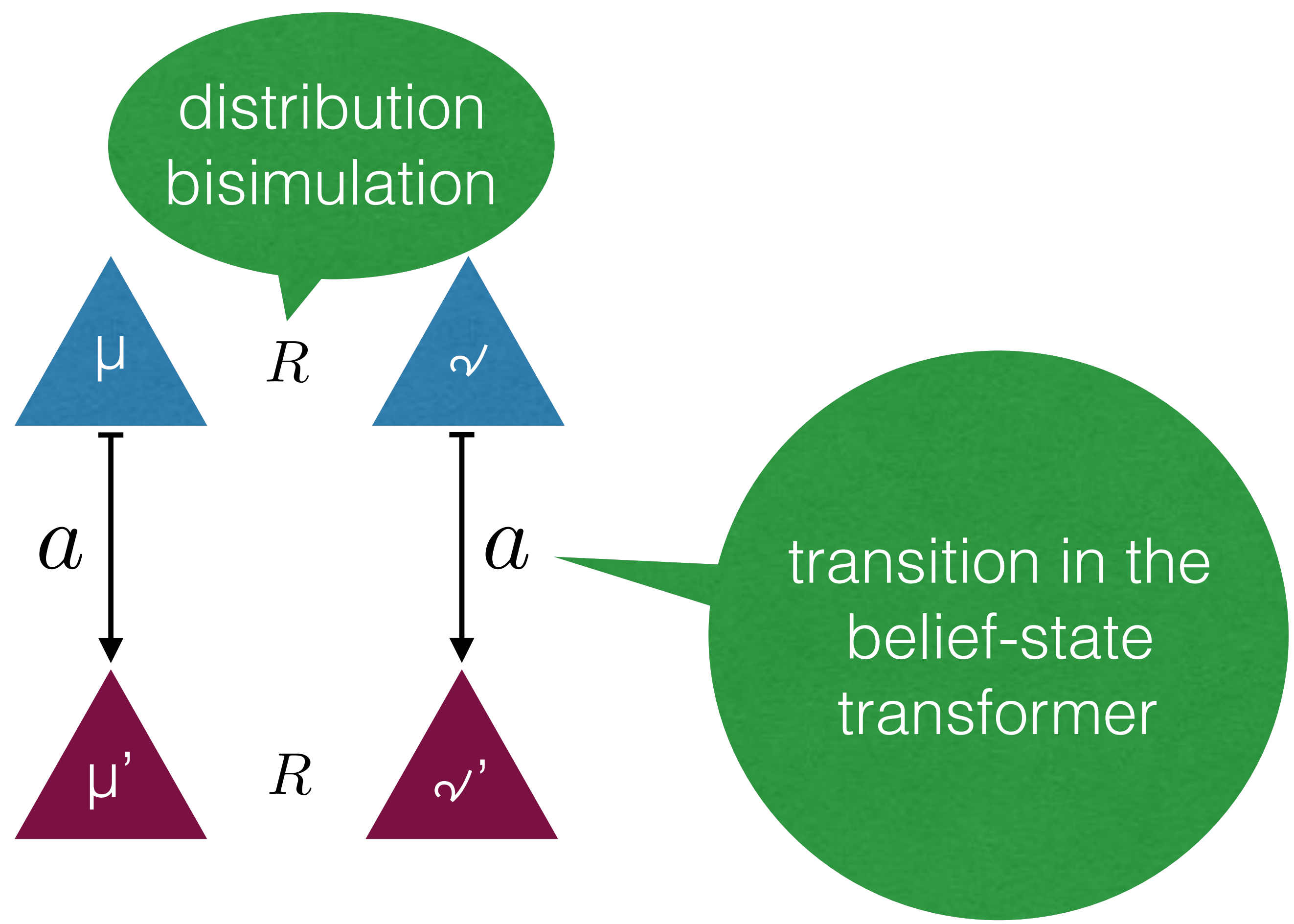
# Distribution bisimilarity



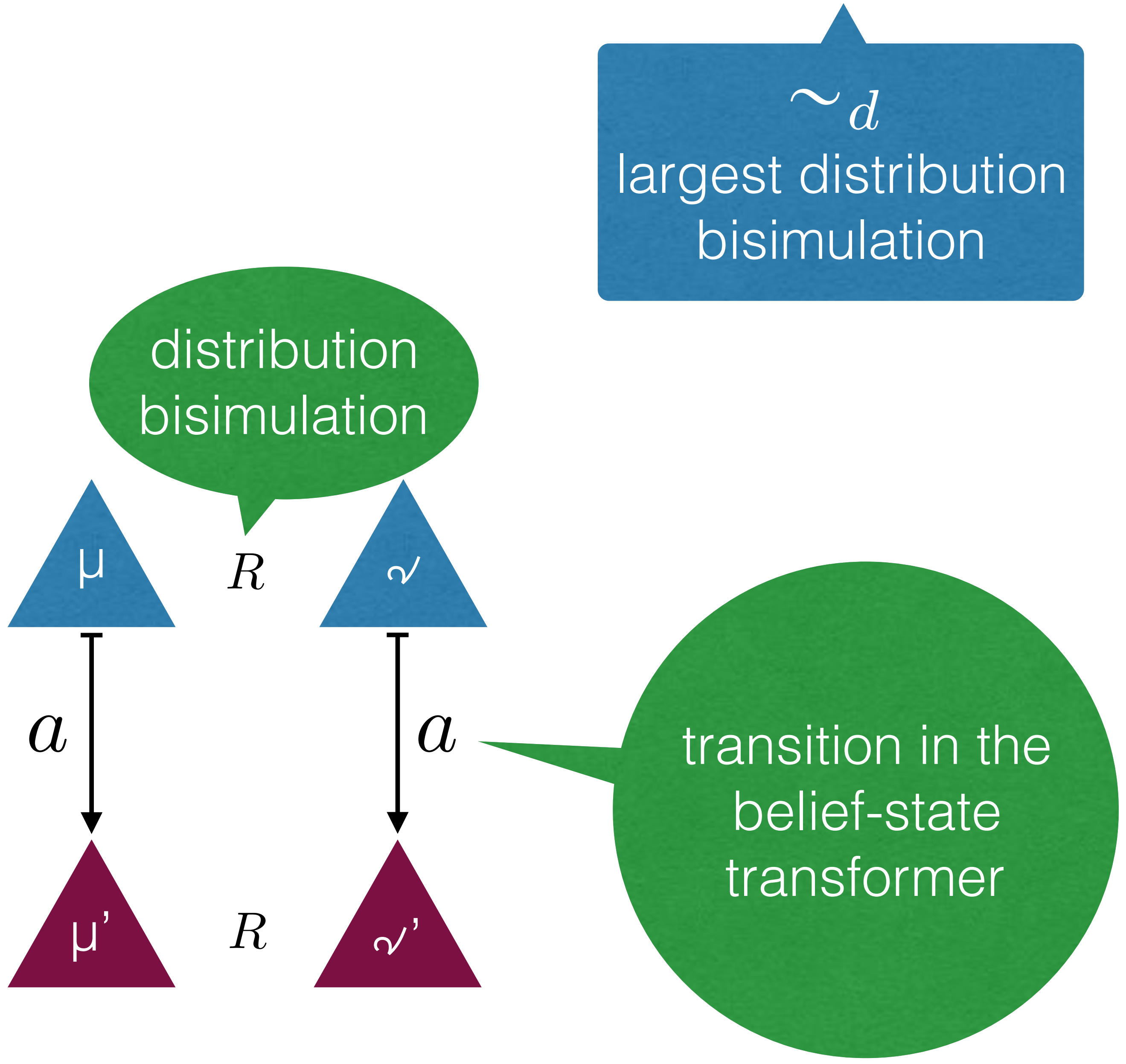
# Distribution bisimilarity



# Distribution bisimilarity



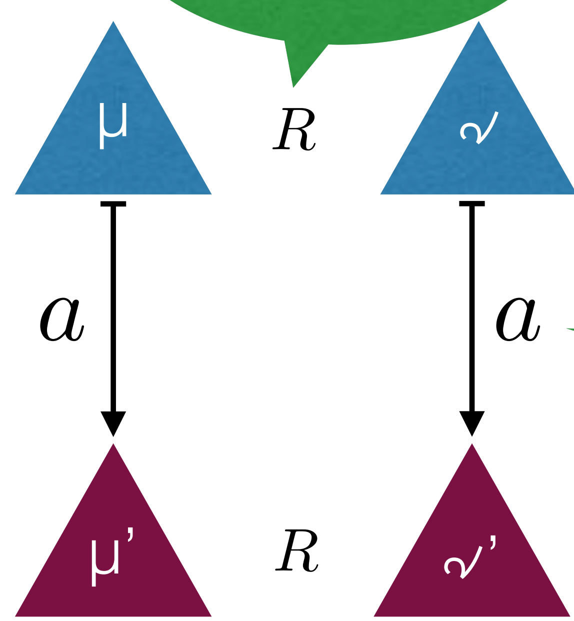
# Distribution bisimilarity



# Distribution bisimilarity

$\sim_d$
largest distribution bisimulation

distribution bisimulation




transition in the belief-state transformer

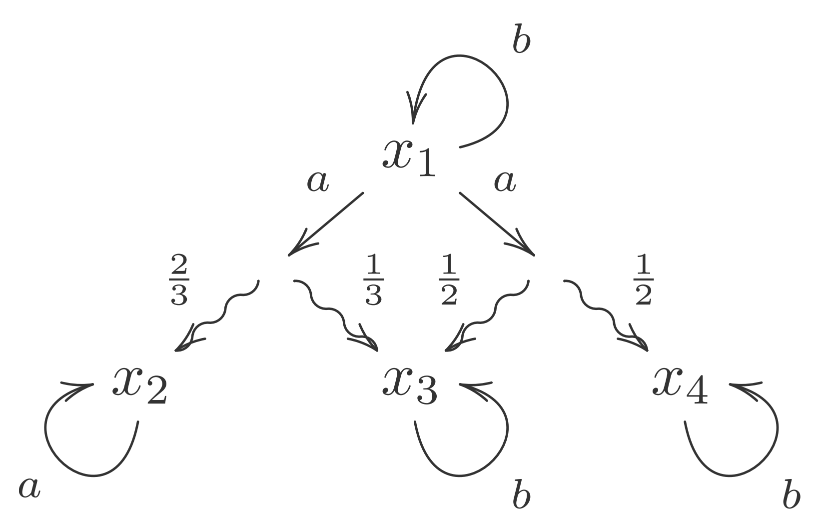
$\sim_d$
is LTS bisimilarity on the belief-state transformer

# Belief-state transformer

foundation ?

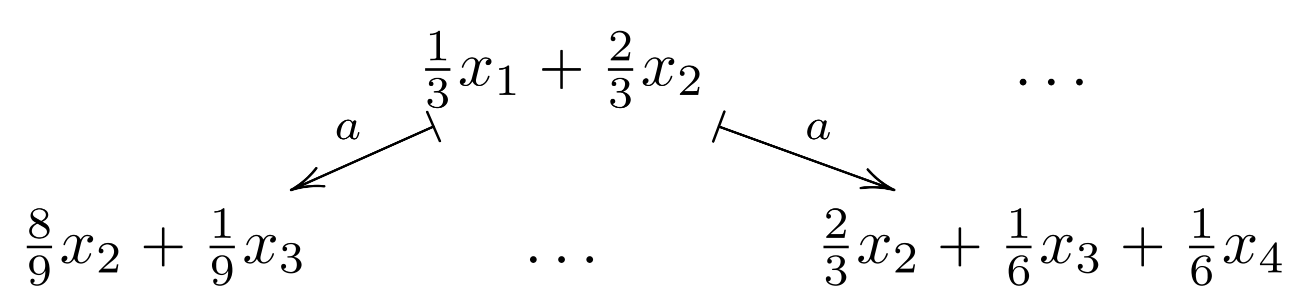
PA

$X \to (\mathcal{PD}(X))^A$




how does it emerge?

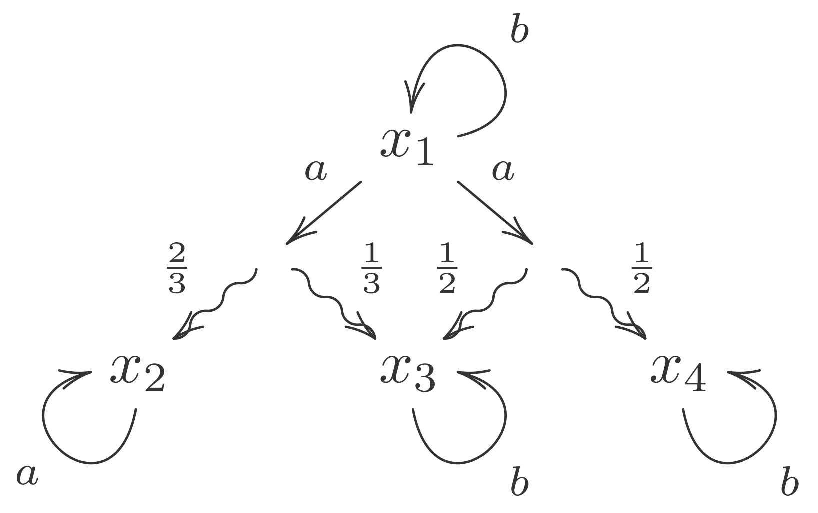
what is it?



# PA coalgebraically

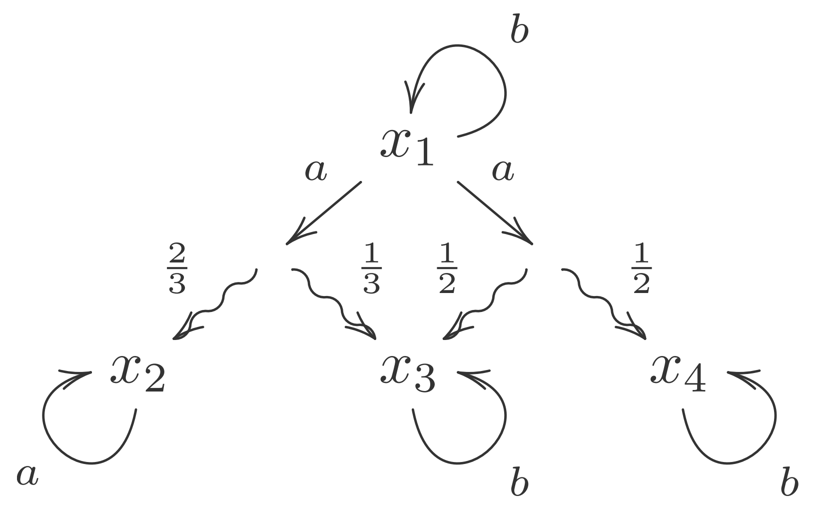
# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$



# PA coalgebraically

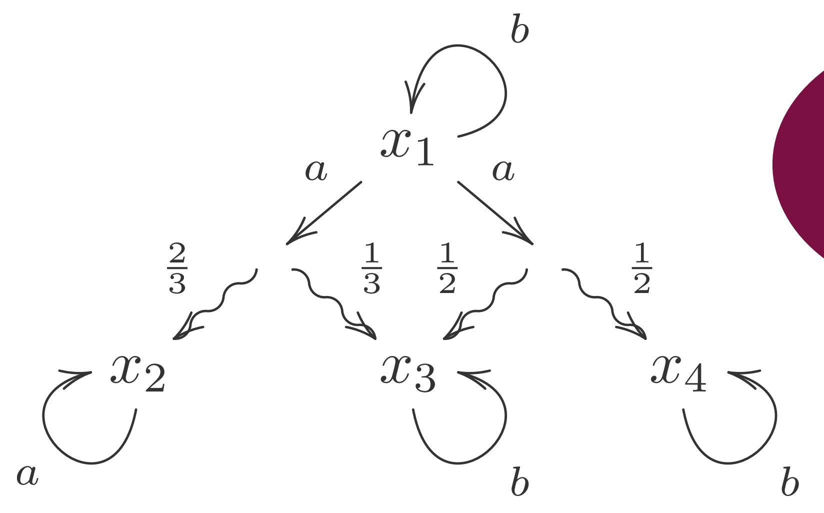
$X \to (\mathcal{PD}(X))^A$




on **Sets**

# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$

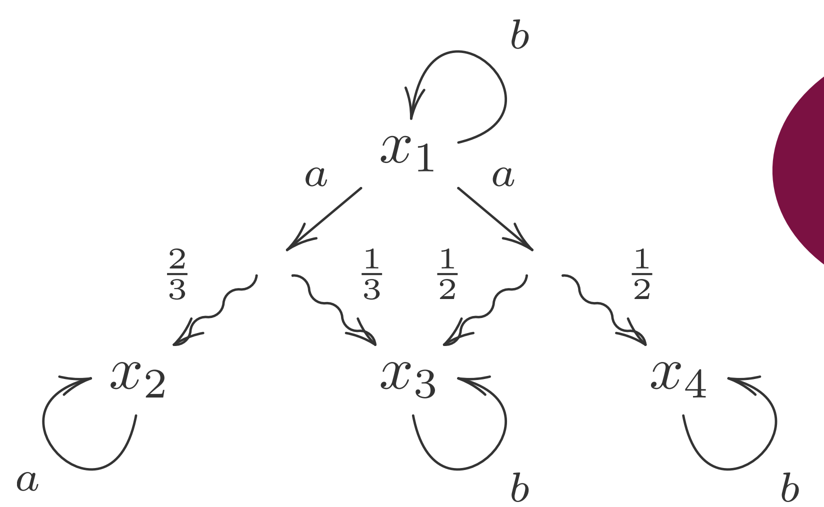



on **Sets**

$\sim \; = \; \approx$

# PA coalgebraically

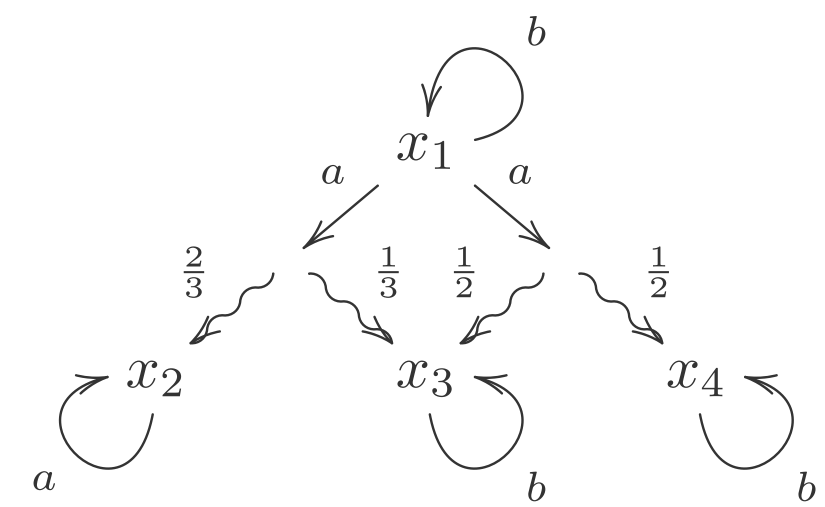
$X \to (\mathcal{PD}(X))^A$




on **Sets**

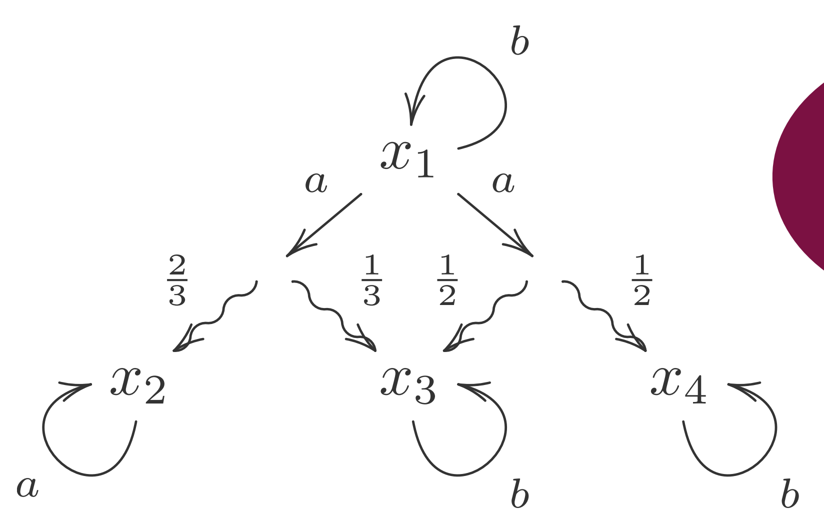
$\sim\ =\ \approx$

$X \to (\mathcal{C}(X))^A$



# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$

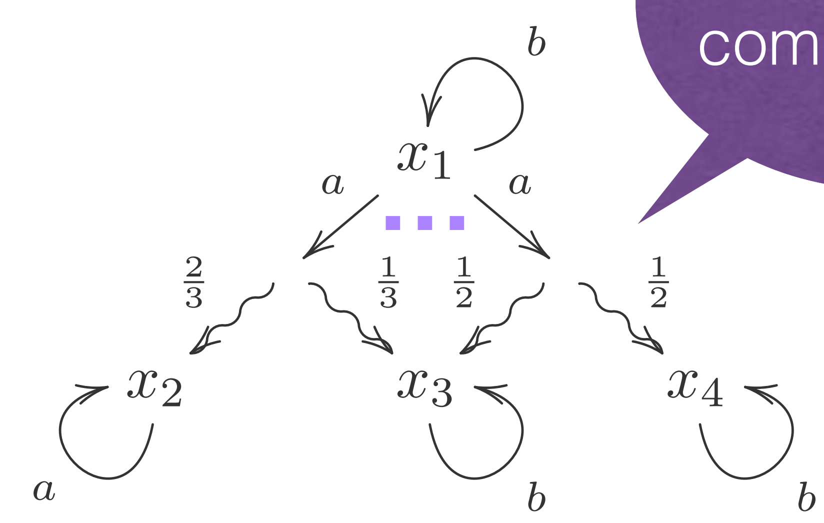



on **Sets**

$\sim \; = \; \approx$

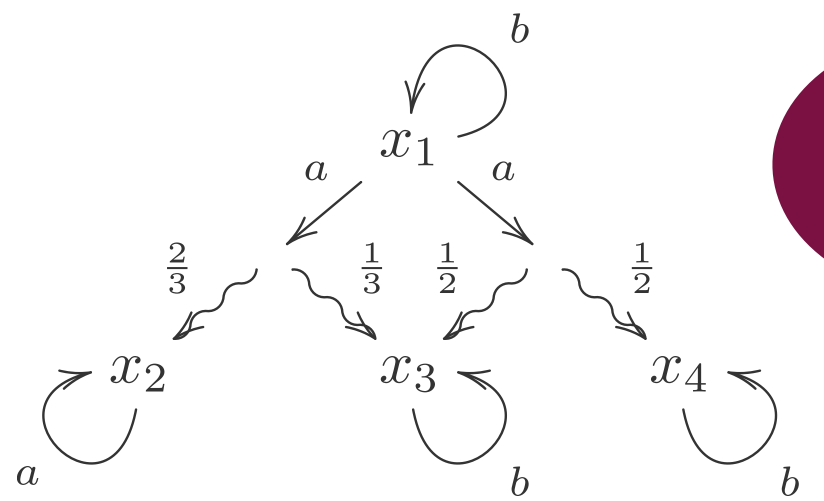
$X \to (\mathcal{C}(X))^A$

and all convex combinations



# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$



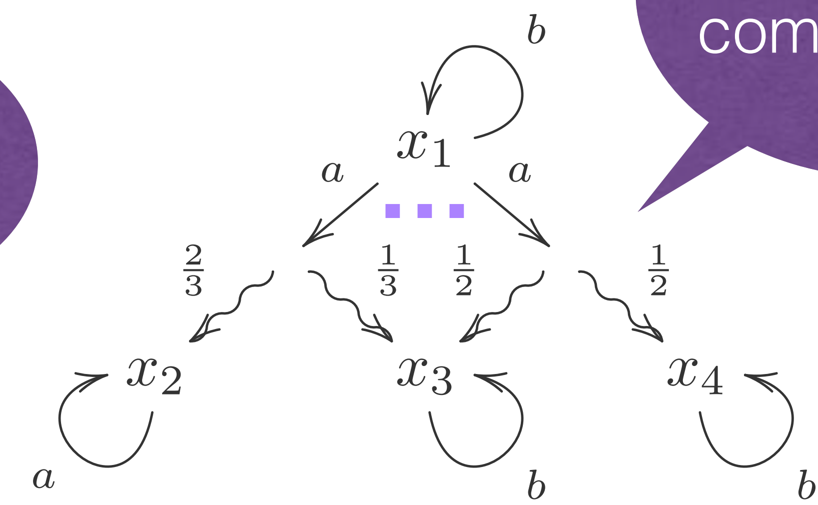

on **Sets**

$\sim \; = \; \approx$

$\sim_c \; = \; \approx$

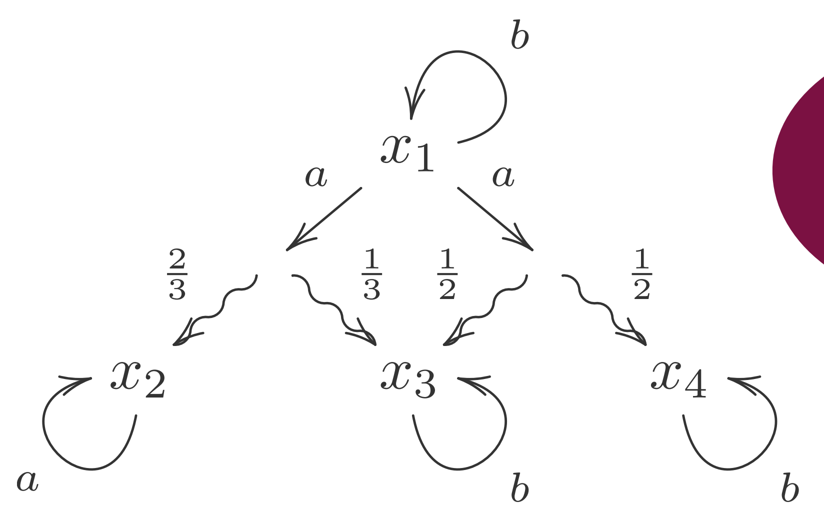
$X \to (\mathcal{C}(X))^A$

and all convex combinations



# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$




on **Sets**

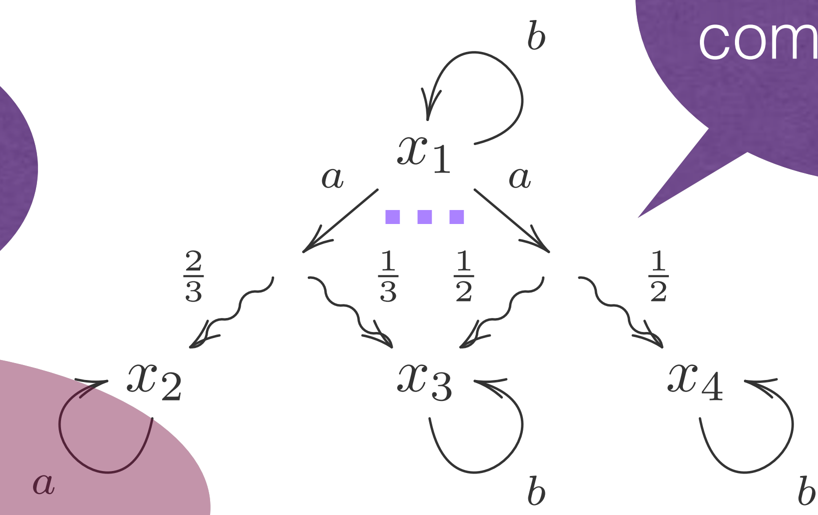
$\sim \,=\, \approx$

$\sim_c \,=\, \approx$

Mio FoSSaCS '14

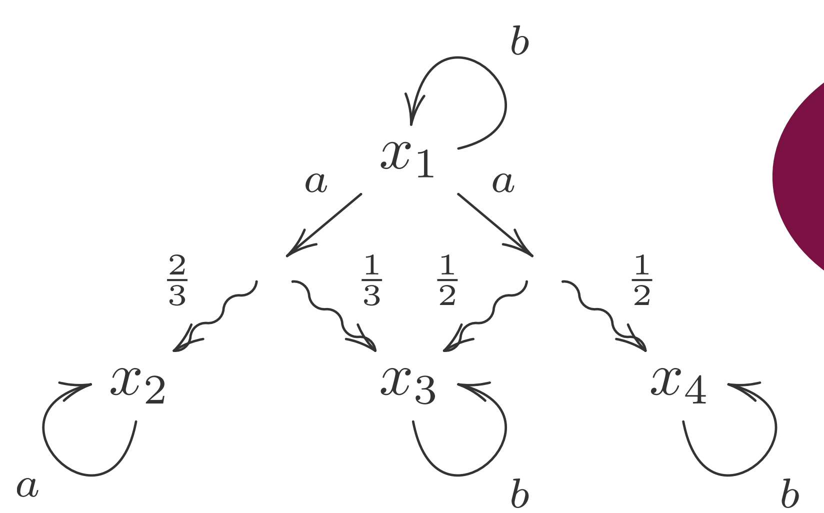
$X \to (\mathcal{C}(X))^A$

and all convex combinations



# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$



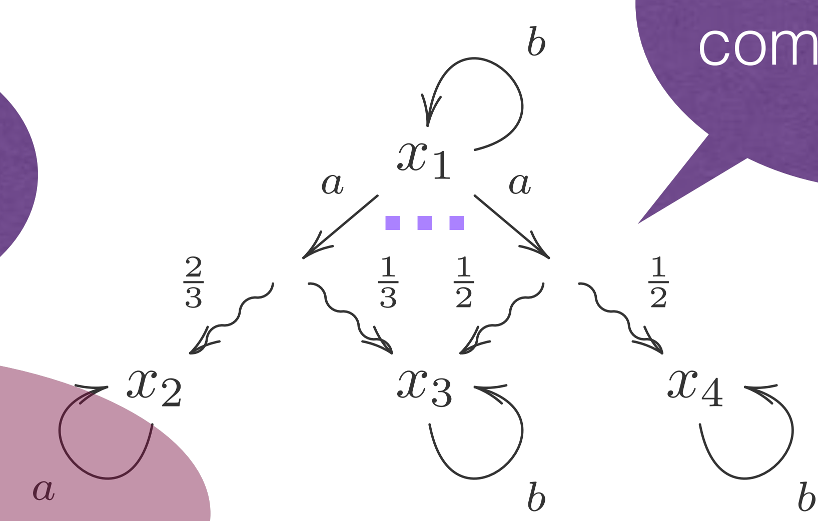

on **Sets**

$\sim \, = \, \approx$

$\sim_c \, = \, \approx$

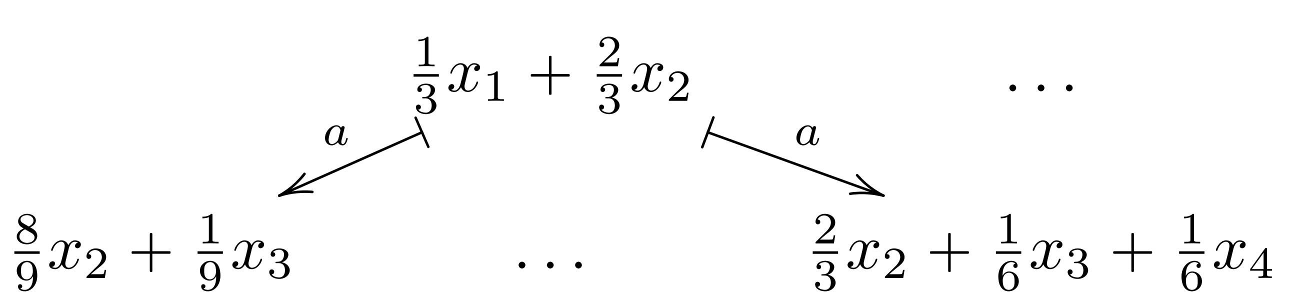
Mio FoSSaCS '14

$X \to (\mathcal{C}(X))^A$



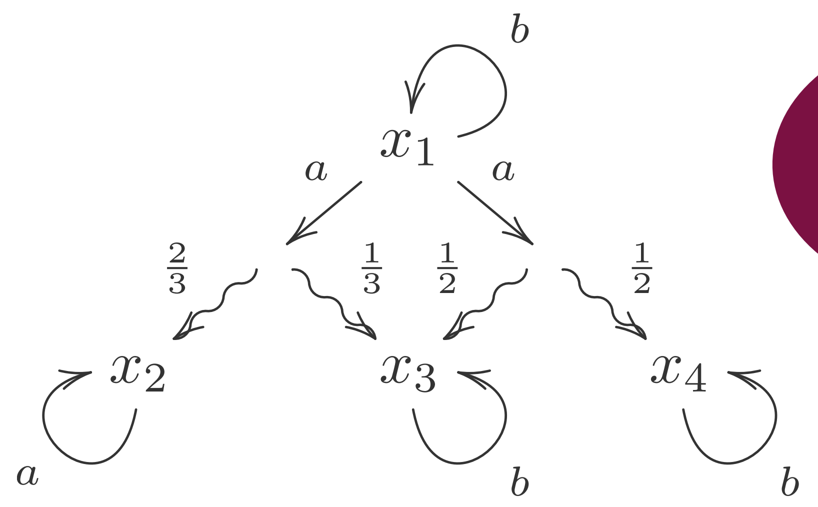

and all convex combinations

$X \to (\mathcal{P}_c(X)+1)^A$



# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$



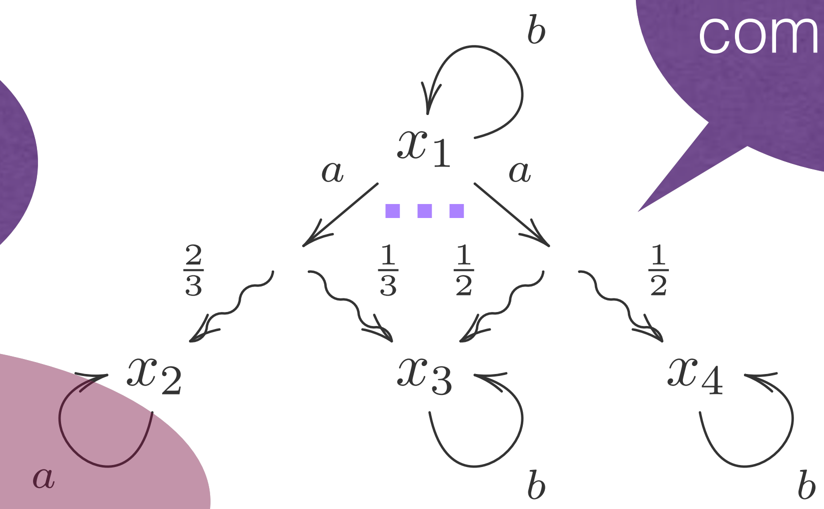

on **Sets**

$\sim \;=\; \approx$

$\sim_c \;=\; \approx$

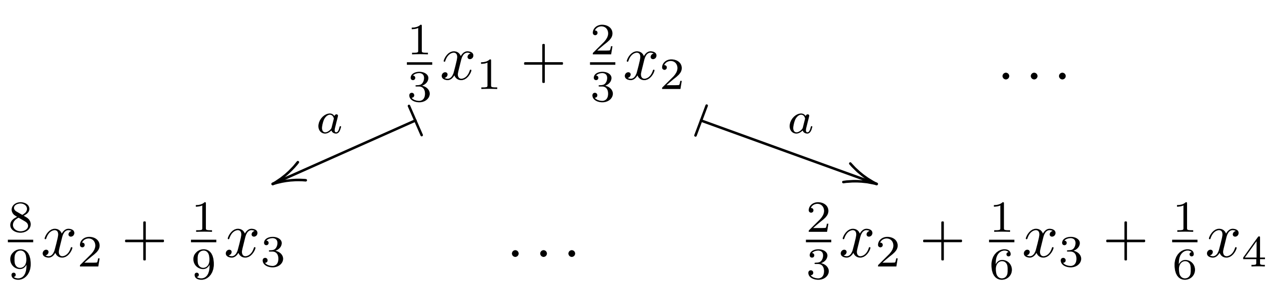
Mio FoSSaCS '14

$X \to (\mathcal{C}(X))^A$




and all convex combinations

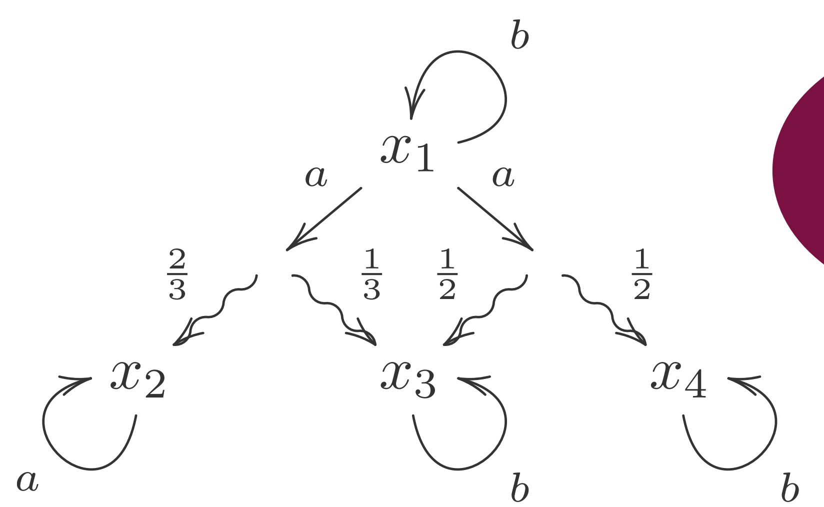
$X \to (\mathcal{P}_c(X)+1)^A$




on convex algebras

# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$



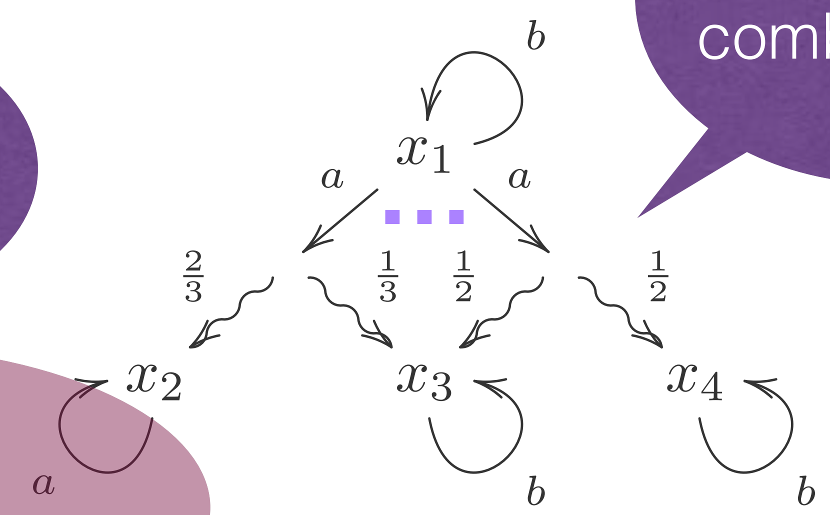

on **Sets**

$\sim \, = \, \approx$

$\sim_c \, = \, \approx$

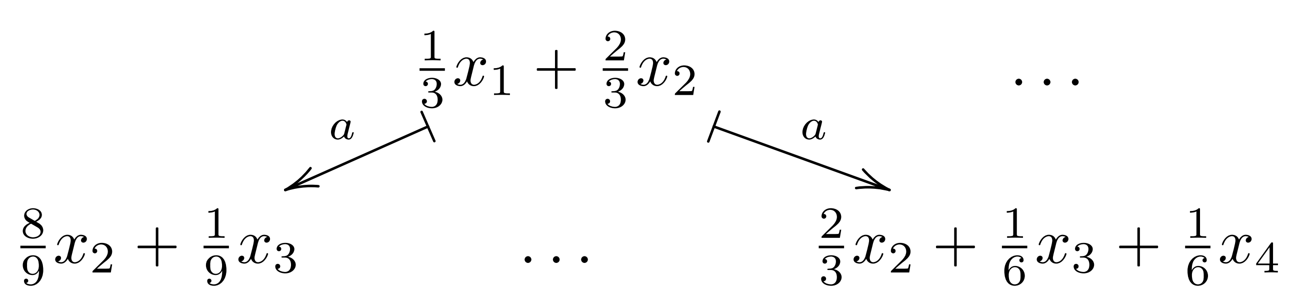
Mio FoSSaCS '14

$X \to (\mathcal{C}(X))^A$




and all convex combinations

$X \to (\mathcal{P}_c(X)+1)^A$

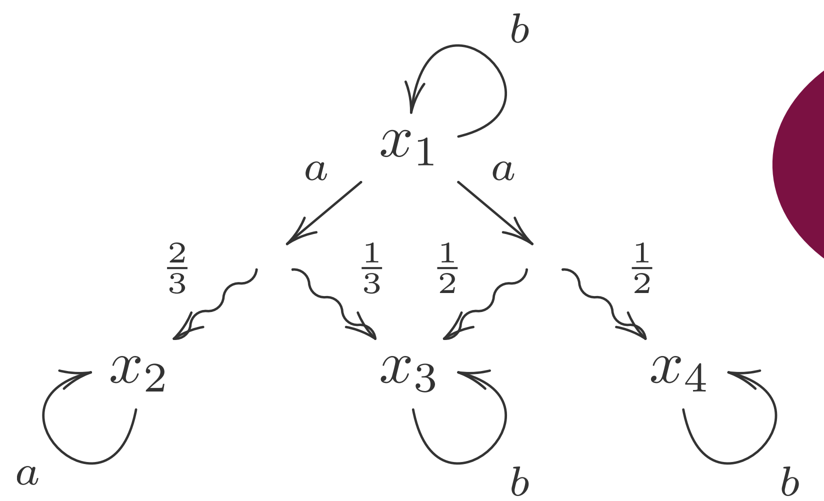



on convex algebras

$\mathcal{EM}(\mathcal{D})$

# PA coalgebraically

$X \to (\mathcal{PD}(X))^A$




on **Sets**

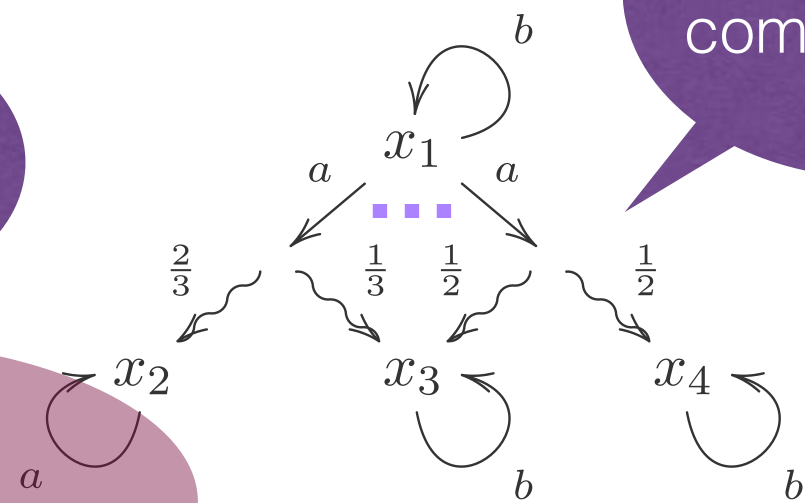
$\sim \, = \, \approx$

$\sim_c \, = \, \approx$

Mio FoSSaCS '14

$X \to (\mathcal{C}(X))^A$

and all convex combinations




$X \to (\mathcal{P}_c(X)+1)^A$

$\frac{1}{3}x_1 + \frac{2}{3}x_2$ $\xrightarrow{a}$ $\frac{8}{9}x_2 + \frac{1}{9}x_3$ ... $\frac{2}{3}x_2 + \frac{1}{6}x_3 + \frac{1}{6}x_4$ ...

on convex algebras

$\mathcal{EM}(\mathcal{D})$

$\sim_d \, = \, \approx$

# Belief-state transformers

coalgebras on
free convex algebras

# Belief-state transformers

$$\mathbb{D}_X = \begin{array}{c}\mathcal{D}\mathcal{D}X\\ \big\downarrow\mu\\ \mathcal{D}X\end{array}$$

free convex algebra

convex combinations

coalgebras on free convex algebras

# Belief-state transformers

$$\mathbb{D}_X = \begin{array}{c}\mathcal{D}\mathcal{D}X\\ \downarrow \mu\\ \mathcal{D}X\end{array}$$

free convex algebra

convex combinations

coalgebras on free convex algebras

$$\mathbb{D}_X \to (\mathcal{P}_c(\mathbb{D}_X) + 1)^A$$

constant exponent

nonempty convex powerset

termination

# Belief-state transformers

$$\mathbb{D}_X = \begin{array}{c}\mathcal{D}\mathcal{D}X \\ \downarrow \mu \\ \mathcal{D}X\end{array}$$

free convex algebra

convex combinations

coalgebras on free convex algebras

$$\mathbb{D}_X \to (\mathcal{P}_c(\mathbb{D}_X) + 1)^A$$

constant exponent

nonempty convex powerset

termination

$$pA_1 + (1-p)A_2 = \{pa_1 + (1-p)a_2 \mid a_1 \in A_1, a_2 \in A_2\}$$

# Belief-state transformers

$$\mathbb{D}_X = \begin{array}{c}\mathcal{D}\mathcal{D}X \\ \downarrow \mu \\ \mathcal{D}X\end{array}$$

convex combinations

free convex algebra

coalgebras on free convex algebras

$$\mathbb{D}_X \to (\mathcal{P}_c(\mathbb{D}_X) + 1)^A$$

constant exponent

nonempty convex powerset

termination

$$pA_1 + (1-p)A_2 = \{pa_1 + (1-p)a_2 \mid a_1 \in A_1, a_2 \in A_2\}$$

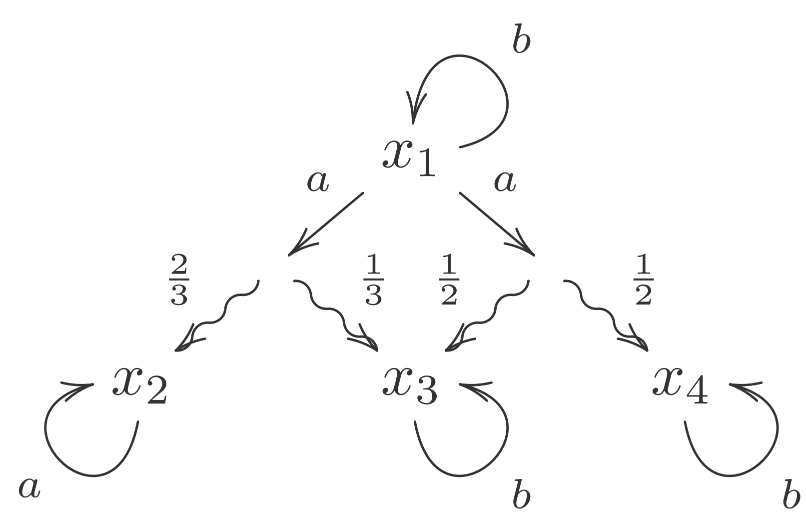
Minkowski sum

# Belief-state transformer

foundation ?

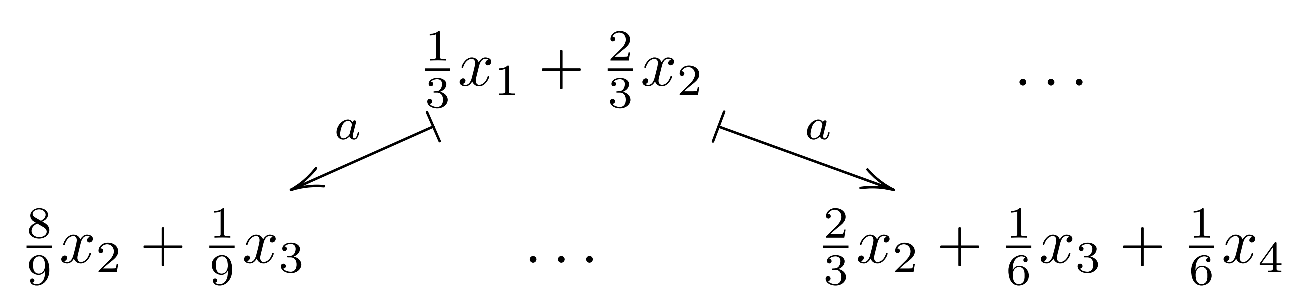
PA

$X \to (\mathcal{PD}(X))^A$




how does it emerge?

what is it?

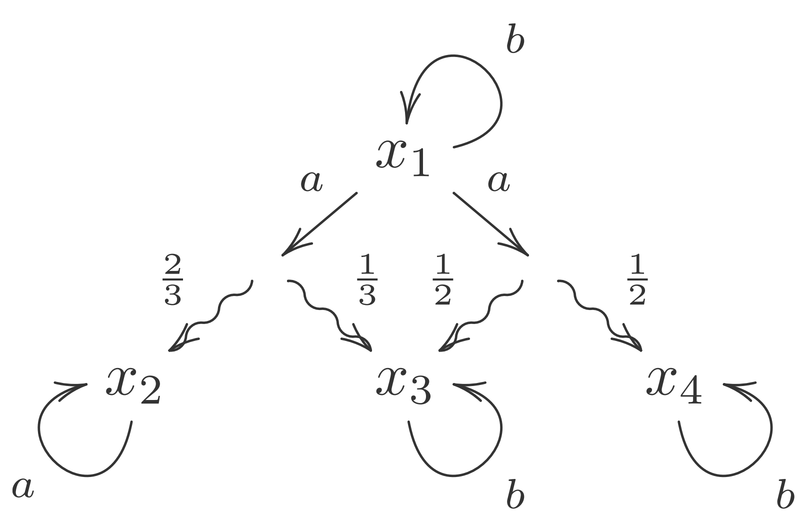


# Belief-state transformer

foundation ?

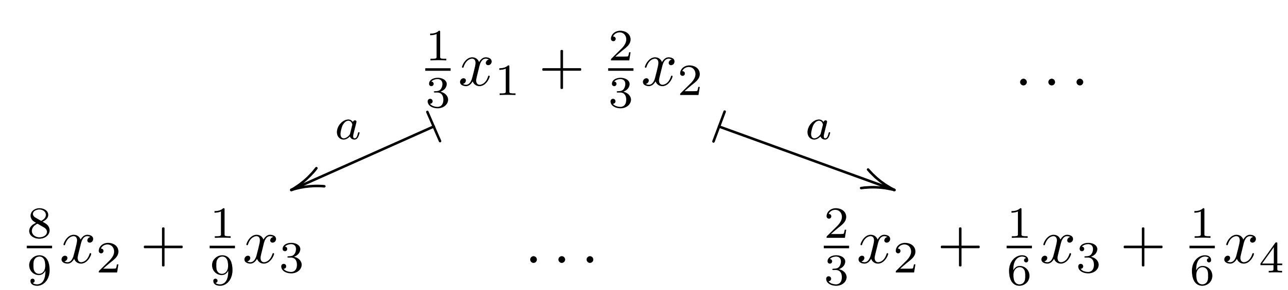
PA

$X \to (\mathcal{PD}(X))^A$




how does it emerge?

coalgebra over free convex algebra



# Determinisations I

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations I

$$\mathbf{CoAlg}_{\mathbf{C}}(FT) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{F}) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(F)$$

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations I

determinise

$$\mathbf{CoAlg}_{\mathbf{C}}(FT) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{F}) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(F)$$

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations I

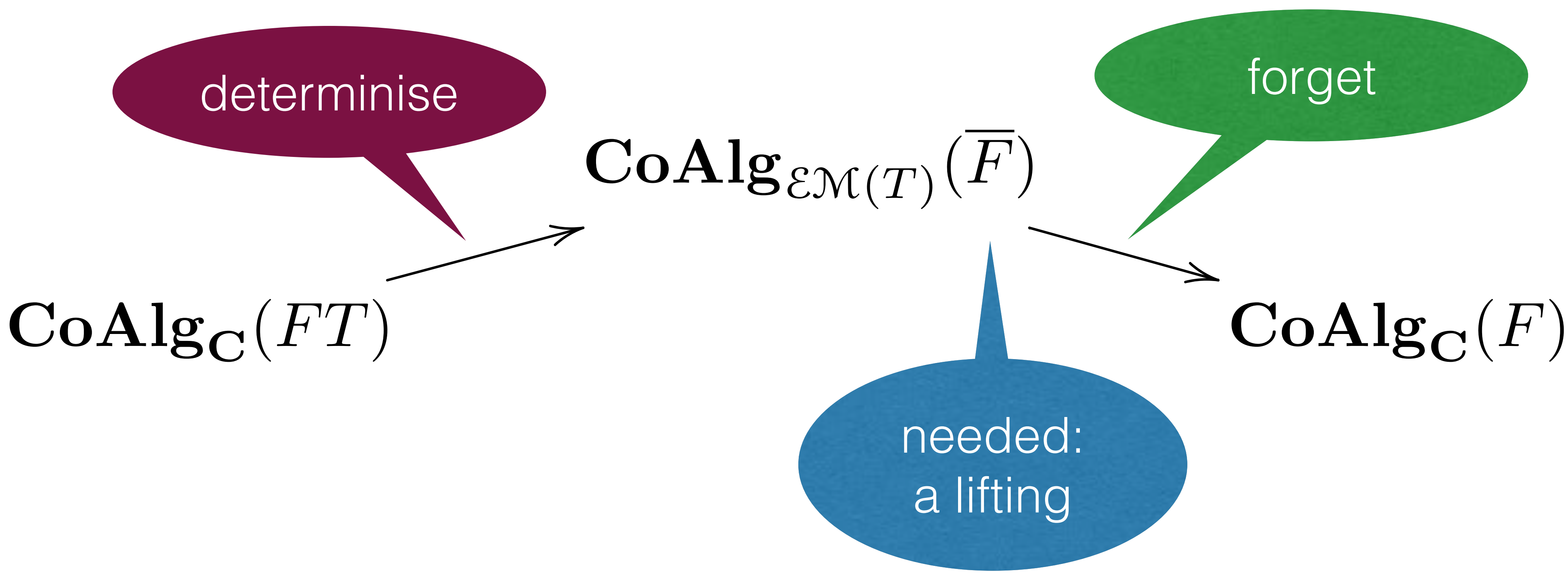
determinise

forget

$$\mathbf{CoAlg}_{\mathbf{C}}(FT) \rightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{F}) \rightarrow \mathbf{CoAlg}_{\mathbf{C}}(F)$$

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations I




[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations I

determinise

$\mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{F})$

forget

$$\mathbf{CoAlg}_{\mathbf{C}}(FT) \rightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{F}) \rightarrow \mathbf{CoAlg}_{\mathbf{C}}(F)$$

needed:
a lifting

works for NFA

not for generative PTS
not for PA / belief-state transformer

[Silva, Bonchi, Bonsangue, Rutten, FSTTCS'10]

# Determinisations II

[Silva, S.  MFPS'11]

[Jacobs, Silva, S  JCSS'15]

# Determinisations II

$$\mathbf{CoAlg}_{\mathbf{C}}(TF) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G}) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

[Silva, S. MFPS'11]

[Jacobs, Silva, S JCSS'15]

# Determinisations II

determinise

$$\mathbf{CoAlg}_{\mathbf{C}}(TF) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G}) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

[Silva, S. MFPS'11]

[Jacobs, Silva, S JCSS'15]

# Determinisations II

determinise

$\mathbf{CoAlg}_{\mathbf{C}}(TF) \rightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G}) \rightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$

forget

[Silva, S. MFPS’11]

[Jacobs, Silva, S JCSS’15]

# Determinisations II

determinise

$\mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G})$

forget

$\mathbf{CoAlg}_{\mathbf{C}}(TF)$ $\rightarrow$ $\mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G})$ $\rightarrow$ $\mathbf{CoAlg}_{\mathbf{C}}(G)$

needed:
a lifting

[Silva, S. MFPS'11]

[Jacobs, Silva, S JCSS'15]

# Determinisations II

determinise

$\mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G})$

forget

$\mathbf{CoAlg}_{\mathbf{C}}(TF) \rightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(\overline{G}) \rightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$

needed:
a lifting

works for generative PTS

not for PA / belief-state transformer

[Silva, S. MFPS'11]

[Jacobs, Silva, S JCSS'15]

# Determinisations III

# Determinisations III

$$\mathbf{CoAlg}_{\mathcal{EM}(T)}(H)$$

$$\mathbf{CoAlg}_{\mathbf{C}}(F) \nearrow \qquad \searrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

# Determinisations III

determinise

$$\mathbf{CoAlg}_{\mathcal{EM}(T)}(H)$$

$$\mathbf{CoAlg}_{\mathbf{C}}(F) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(H) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

# Determinisations III

determinise

$$\mathbf{CoAlg}_{\mathbf{C}}(F) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(H) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

forget

# Determinisations III

determinise

$$\mathbf{CoAlg}_{\mathbf{C}}(F) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(H) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

forget

needed:
a quasi lifting
and
a lax lifting

# Determinisations III

determinise

$$\mathbf{CoAlg}_{\mathcal{EM}(T)}(H)$$

forget

$$\mathbf{CoAlg}_{\mathbf{C}}(F) \longrightarrow \mathbf{CoAlg}_{\mathcal{EM}(T)}(H) \longrightarrow \mathbf{CoAlg}_{\mathbf{C}}(G)$$

needed:
a quasi lifting
and
a lax lifting

works for PA /belief-state transformer

# Determinisations III

determinise

forget

$\mathbf{CoAlg}_{\mathcal{EM}(T)}(H)$

$\mathbf{CoAlg}_{\mathbf{C}}(F)$ → $\mathbf{CoAlg}_{\mathcal{EM}(T)}(H)$ → $\mathbf{CoAlg}_{\mathbf{C}}(G)$

needed:
a quasi lifting
and
a lax lifting

works for PA /belief-state transformer

$(\mathcal{P}_c+1)^A$ on $\mathcal{EM}(\mathcal{D})$

is a quasi lifting and lax lifting of

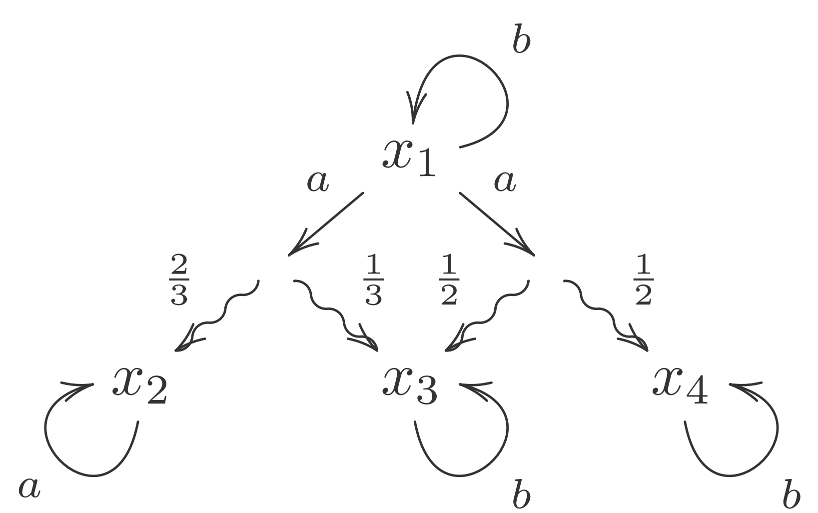
$\mathcal{C}^A$ and $\mathcal{P}^A$ on $\mathbf{Sets}$

# Belief-state transformer

foundation ?

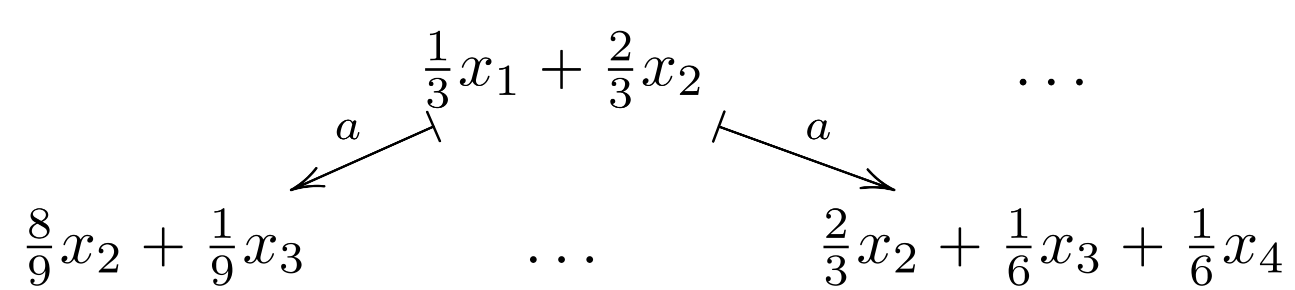
PA

$X \to (\mathcal{PD}(X))^A$




how does it emerge?

coalgebra over free convex algebra

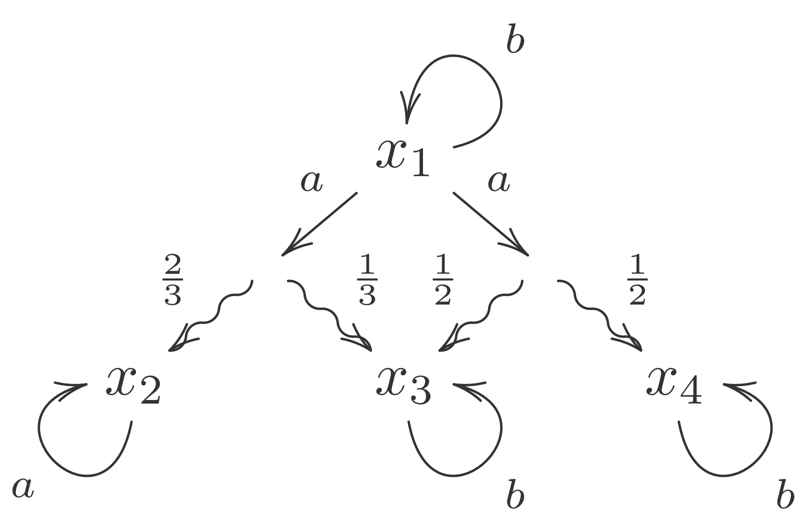


# Belief-state transformer

foundation ?

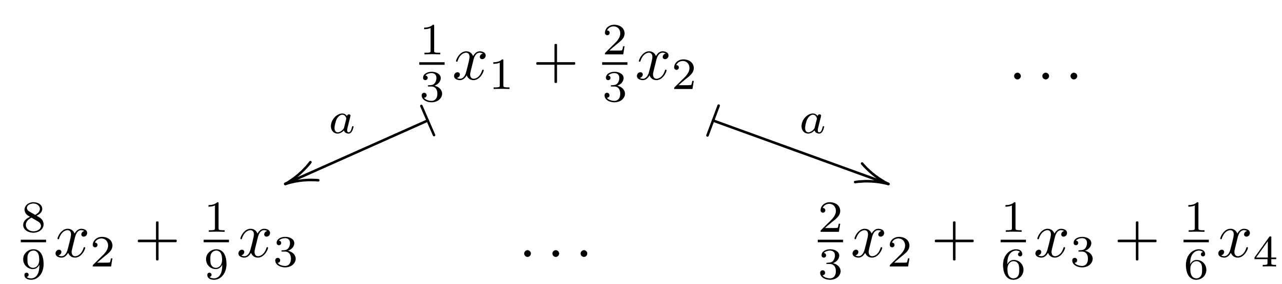
PA

$X \to (\mathcal{PD}(X))^A$




via a generalised[3] determinisation

coalgebra over free convex algebra

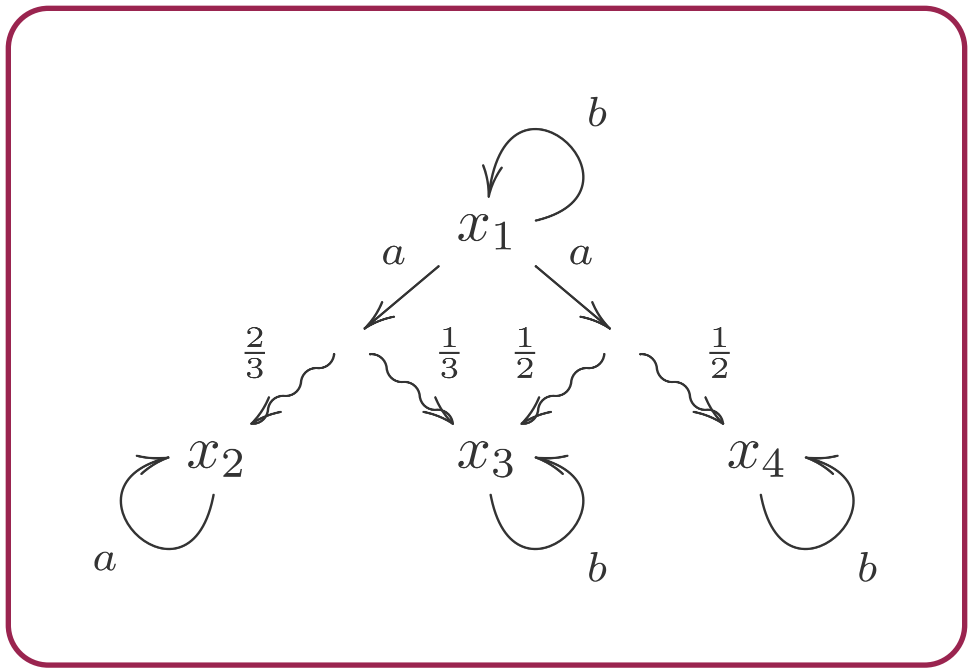


# Belief-state transformer

are natural indeed

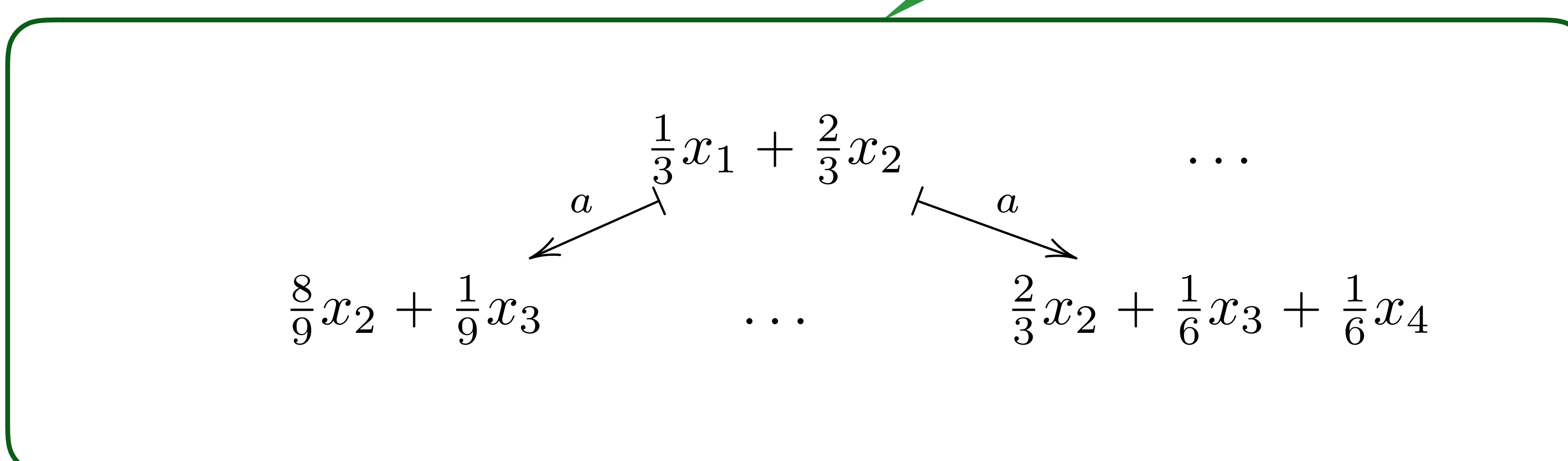
PA

$X \to (\mathcal{PD}(X))^A$




via a generalised[3] determinisation

coalgebra over free convex algebra



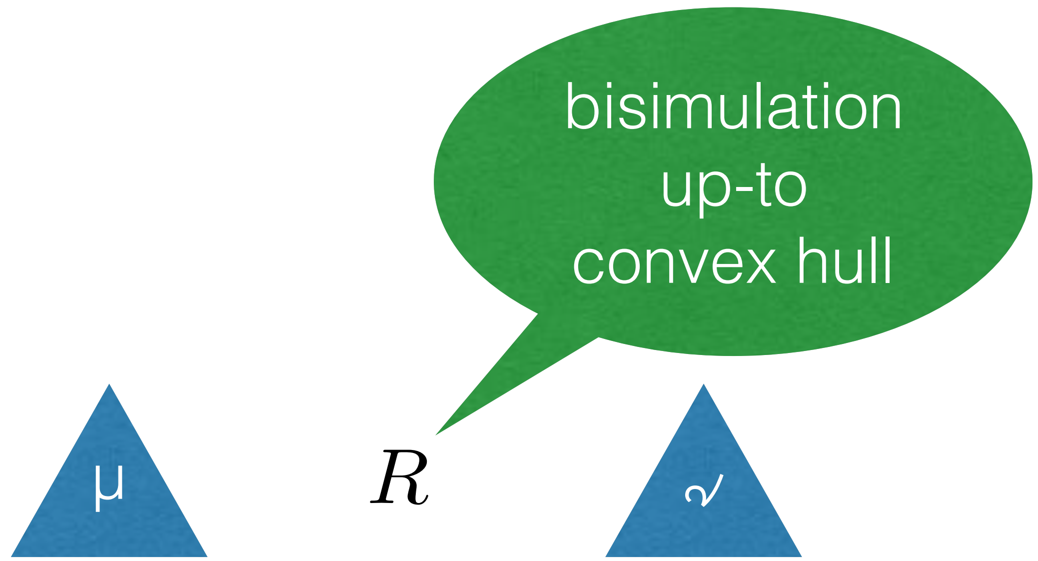
# Coinductive proof method for distribution bisimilarity

# Coinductive proof method for distribution bisimilarity

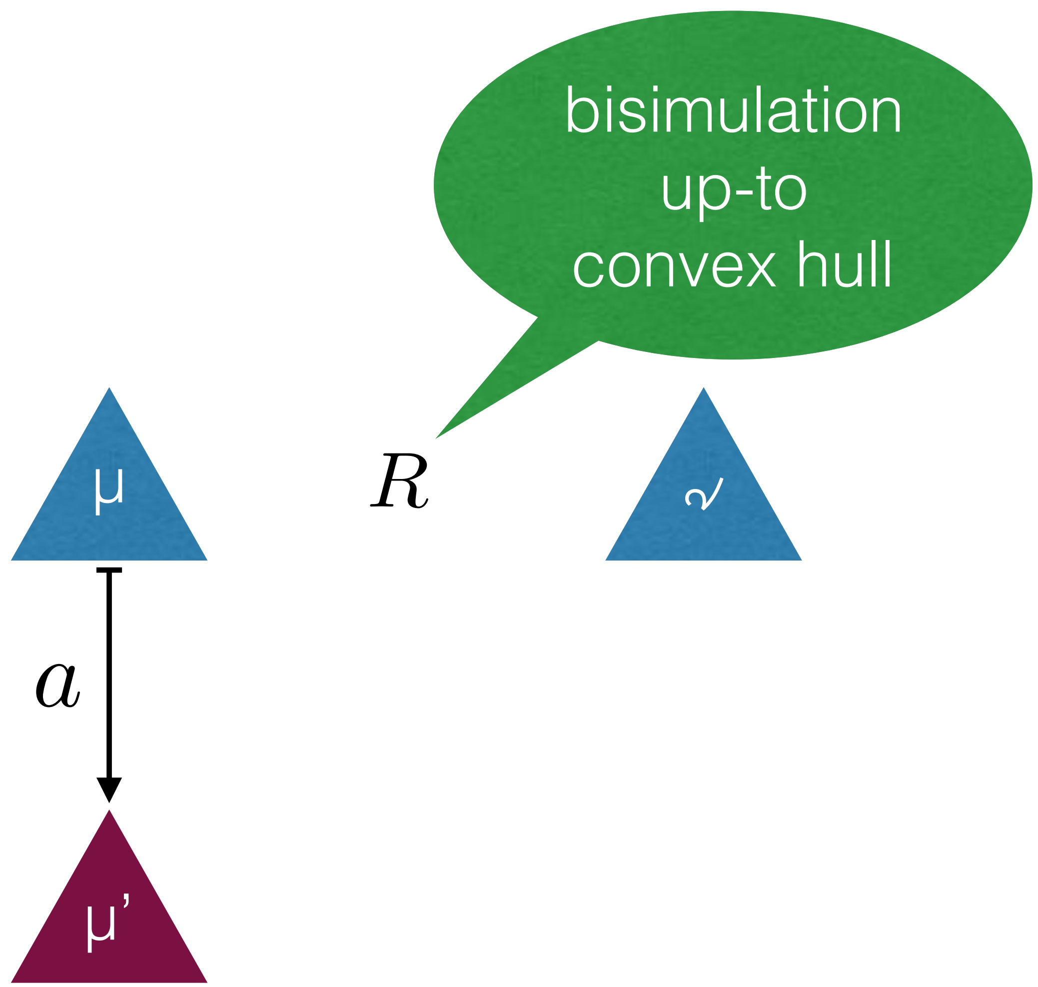
bisimulation up-to convex hull

$R$

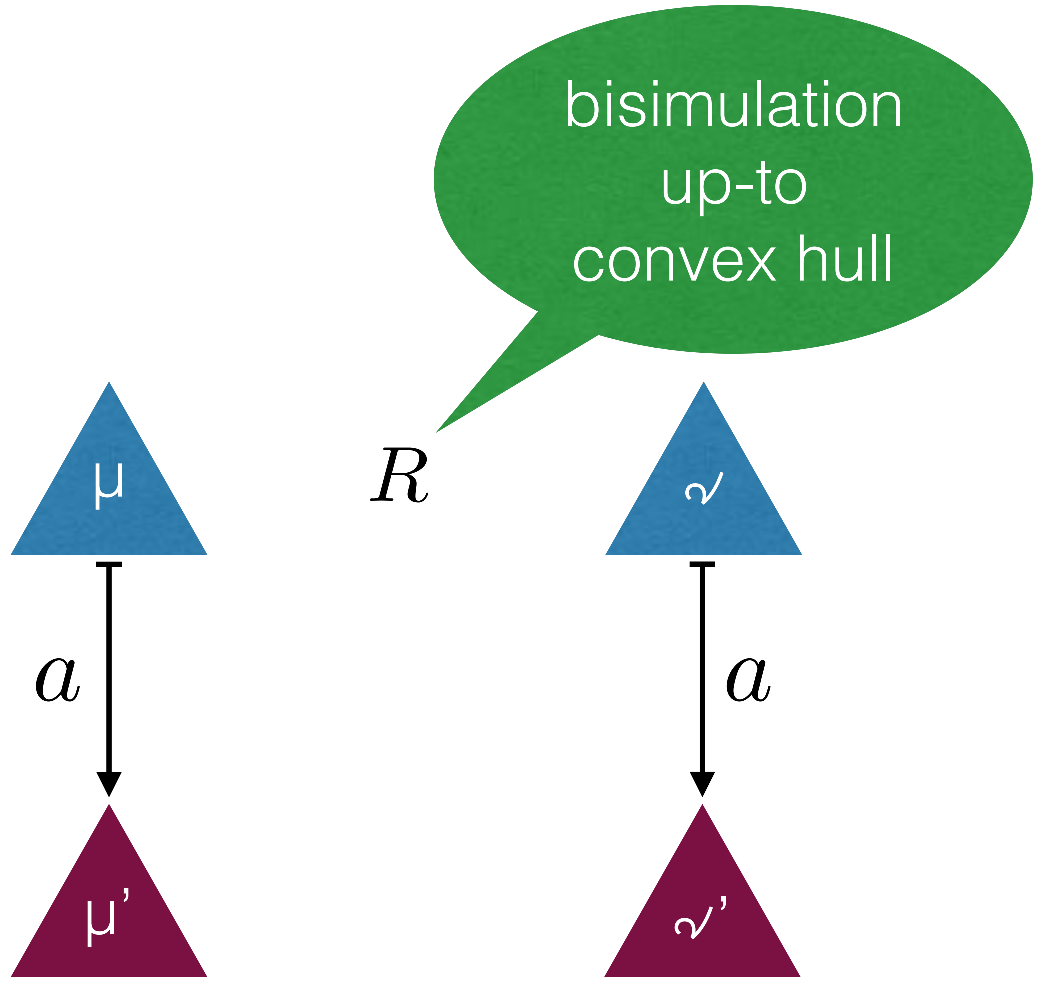
# Coinductive proof method for distribution bisimilarity



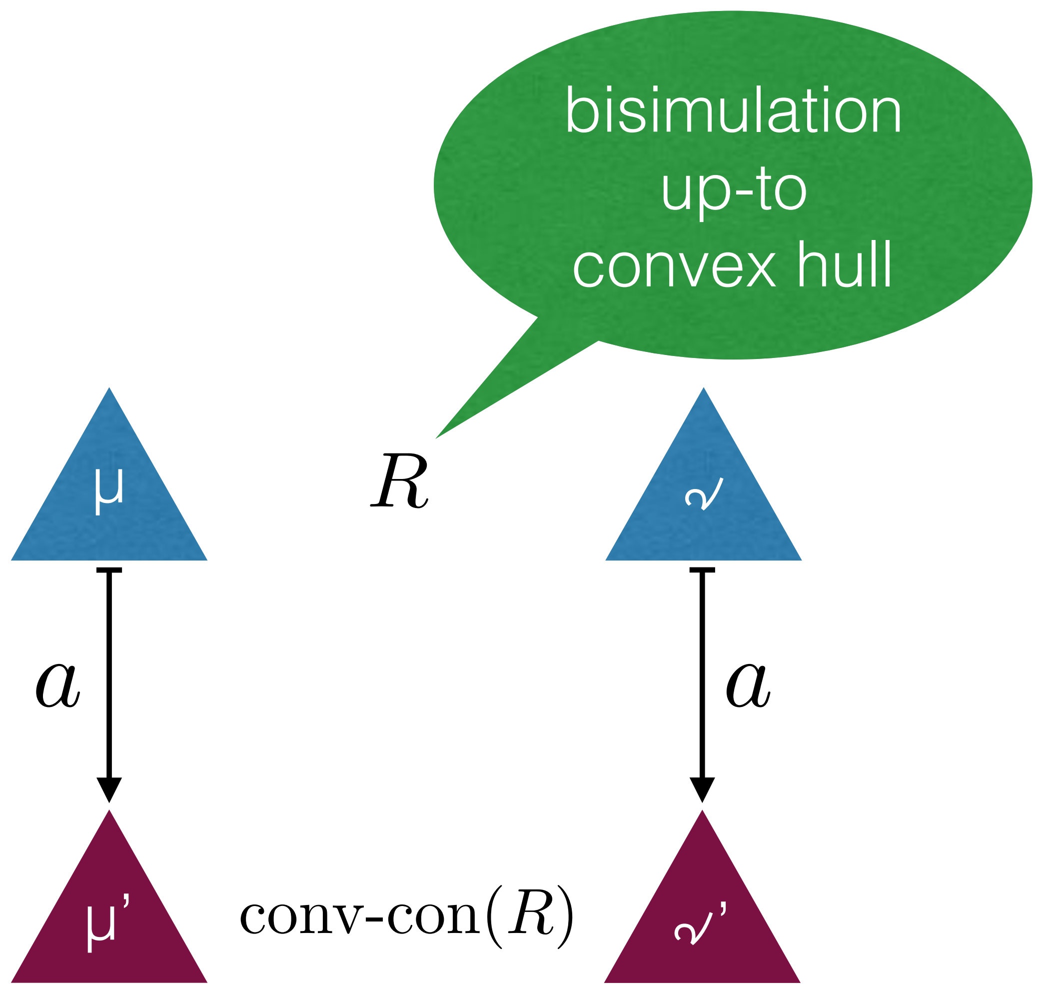
# Coinductive proof method for distribution bisimilarity



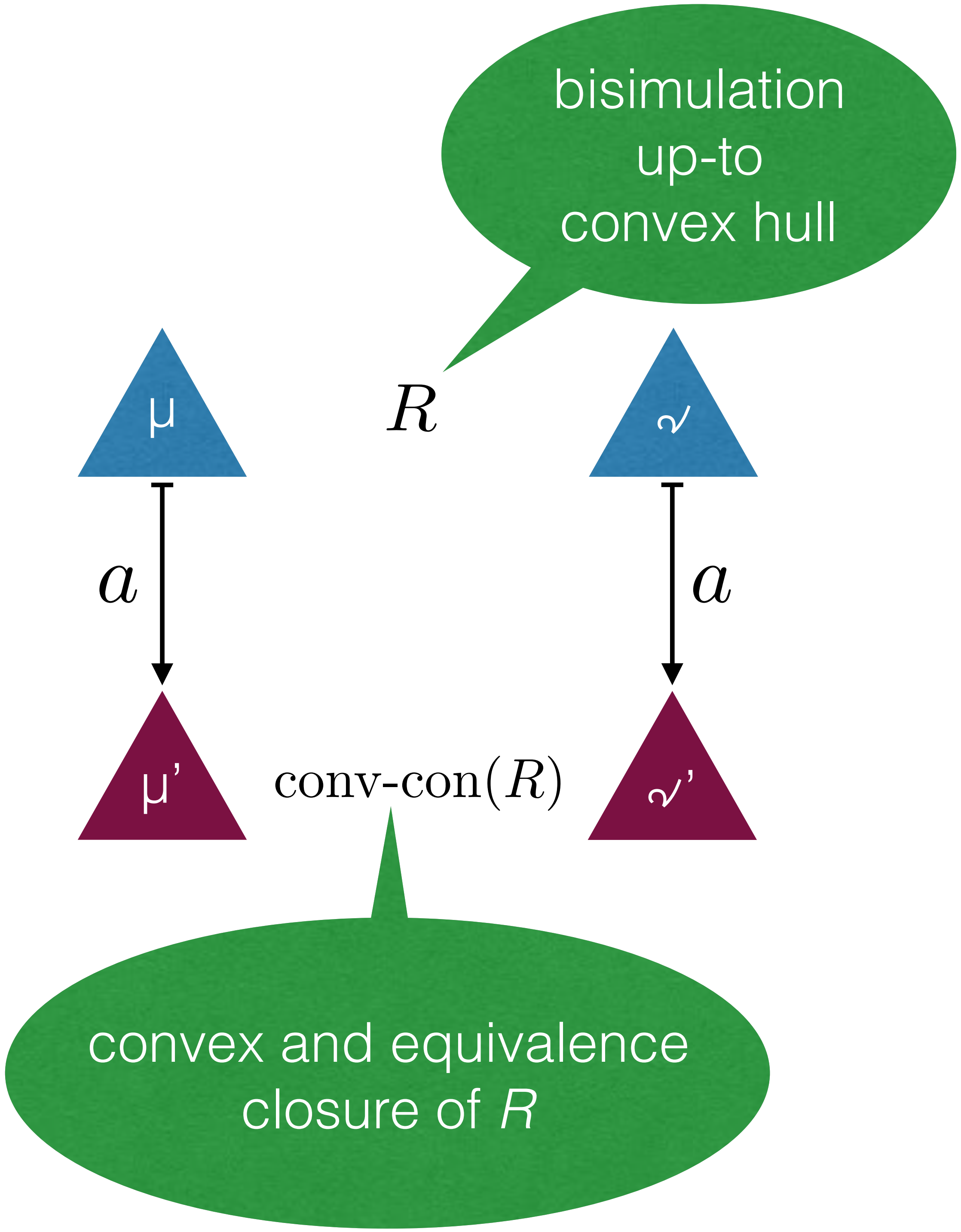
# Coinductive proof method for distribution bisimilarity



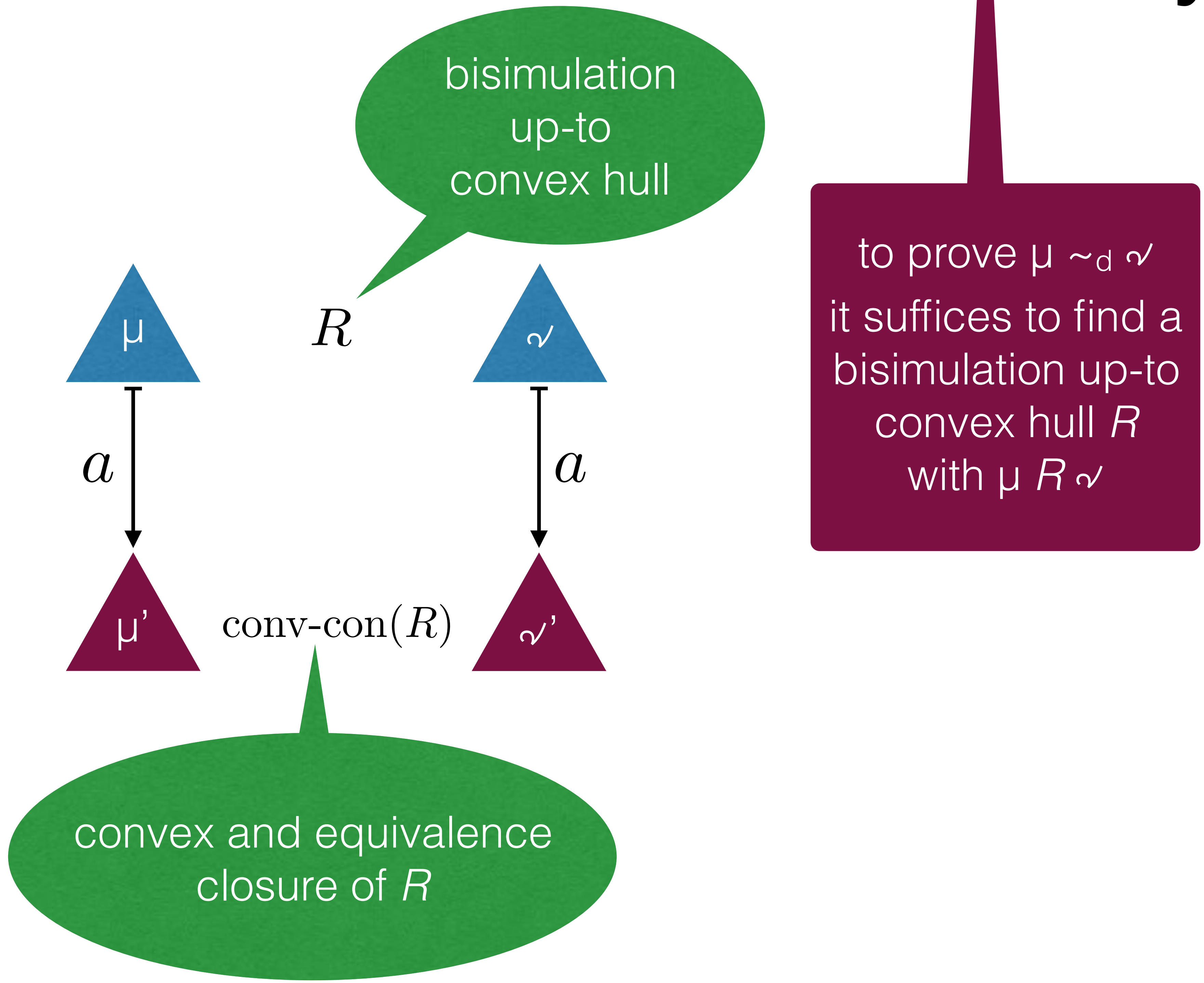
# Coinductive proof method for distribution bisimilarity



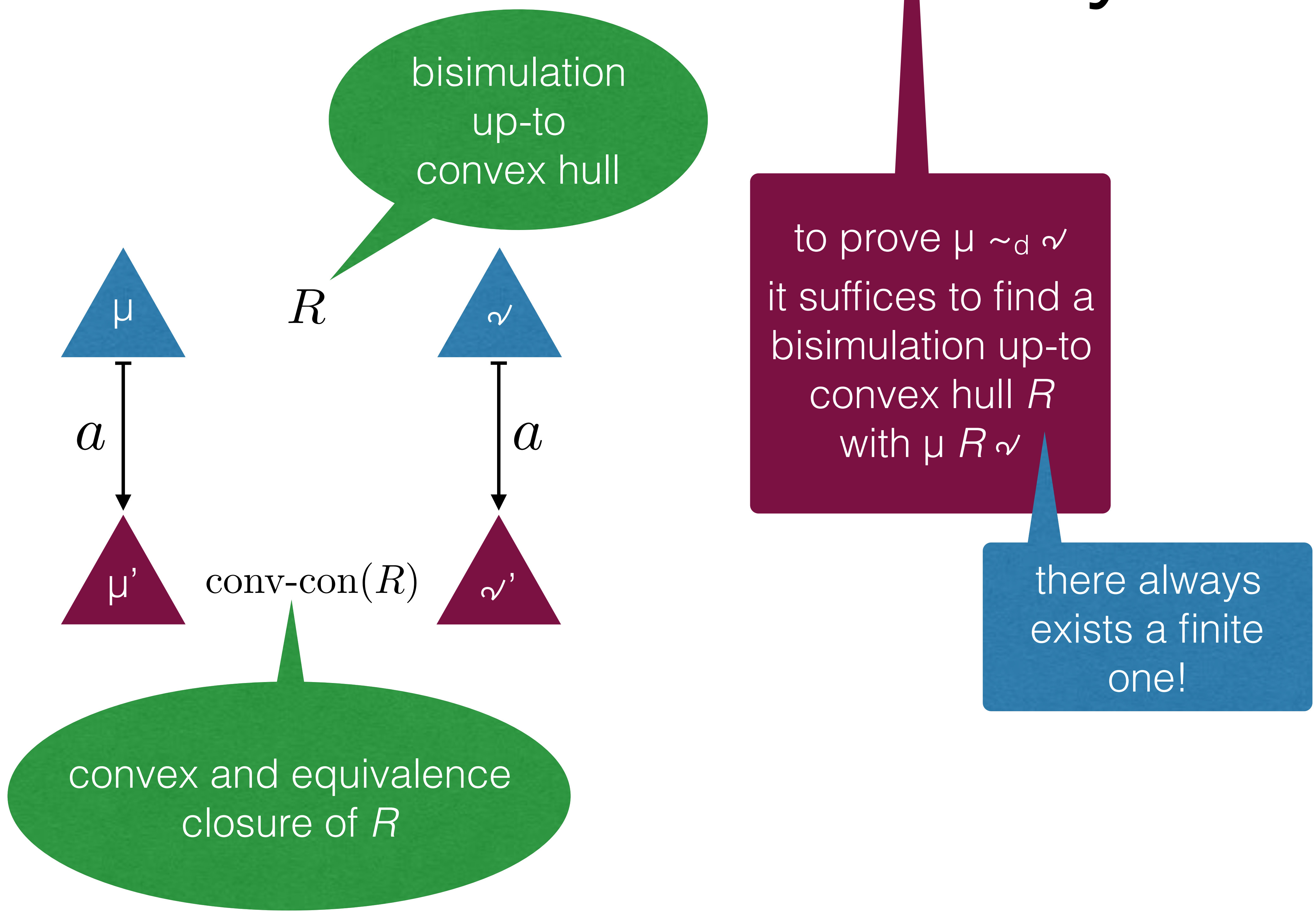
# Coinductive proof method for distribution bisimilarity



# Coinductive proof method for distribution bisimilarity

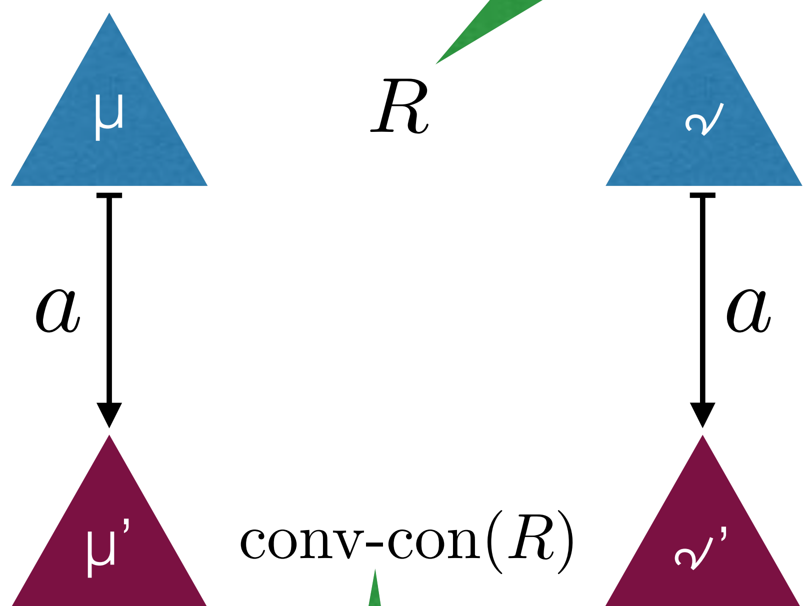


# Coinductive proof method for distribution bisimilarity



# Coinductive proof method for distribution bisimilarity

bisimulation up-to convex hull




convex and equivalence closure of *R*

to prove μ $\sim_d$ ν it suffices to find a bisimulation up-to convex hull *R* with μ *R* ν

there always exists a finite one!

[S., Woracek JPAA'15]

We looked at:

**Part 1.** Modelling probabilistic systems for branching-time semantics

bisimilarity

**Part 2.** Traces, linear-time semantics

trace equivalence

**Part 3.** Belief-state-transformer semantics via convexity

distribution bisimilarity

all with help of coalgebra

Thank You!

We looked at:

**Part 1.** Modelling probabilistic systems for branching-time semantics

bisimilarity

**Part 2.** Traces, linear-time semantics

trace equivalence

**Part 3.** Belief-state-transformer semantics via convexity

distribution bisimilarity

all with help of coalgebra