UNIVERSITY OF TWENTE.

Formal Methods & Tools.

Confluence Reduction versus Partial-Order Reduction in Probabilistic and Non-Probabilistic Branching Time

Mark Timmer

October 6, 2011

ROCKS 2011

Joint work with Henri Hansen

The context – probabilistic model checking

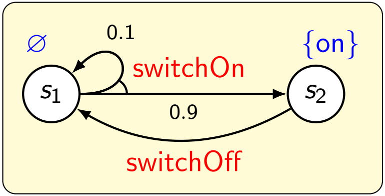
Probabilistic model checking:

- Verifying quantitative properties,
- Using a probabilistic model (e.g., an MDP)

The context – probabilistic model checking

Probabilistic model checking:

- Verifying quantitative properties,
- Using a probabilistic model (e.g., an MDP)

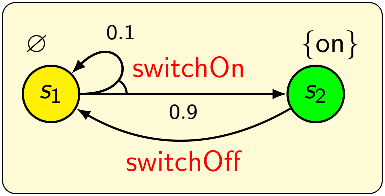


- Non-deterministically choose a transition
- Probabilistically choose the next state

The context – probabilistic model checking

Probabilistic model checking:

- Verifying quantitative properties,
- Using a probabilistic model (e.g., an MDP)

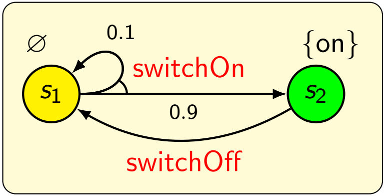


- Non-deterministically choose a transition
- Probabilistically choose the next state

The context – probabilistic model checking

Probabilistic model checking:

- Verifying quantitative properties,
- Using a probabilistic model (e.g., an MDP)

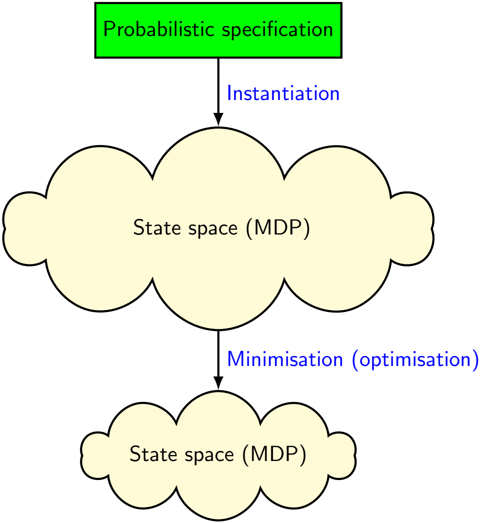


- Non-deterministically choose a transition
- Probabilistically choose the next state

Main limitation (as for non-probabilistic model checking):

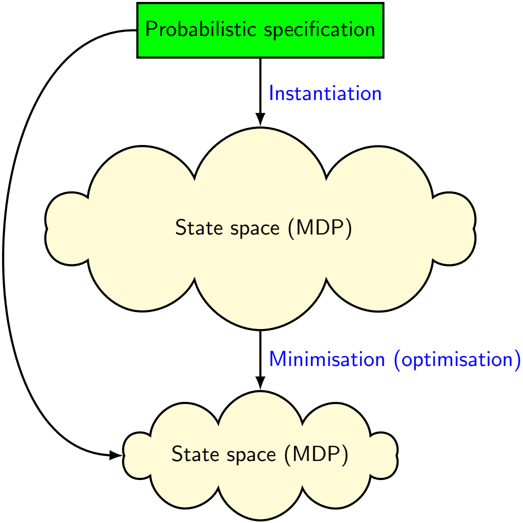
- Susceptible to the state space explosion problem

Combating the state space explosion

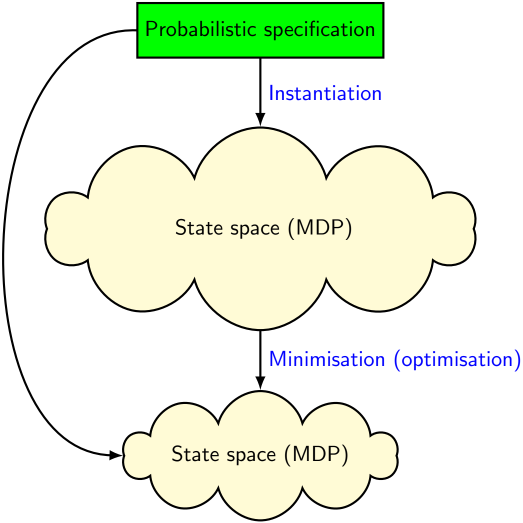


Combating the state space explosion

Optimised instantiation



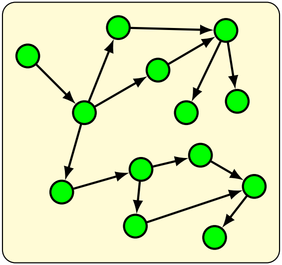
Combating the state space explosion



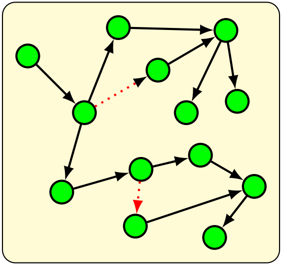
Optimised instantiation

- Partial-order reduction
- Confluence reduction (initially for PAs)

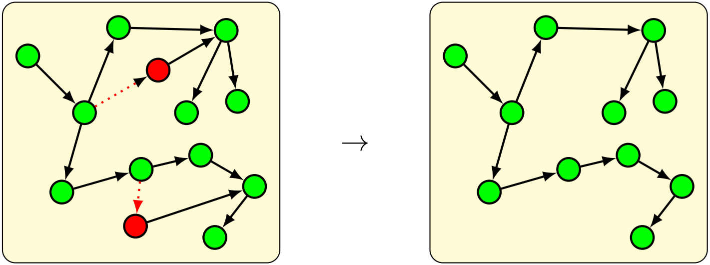
Reductions – an overview



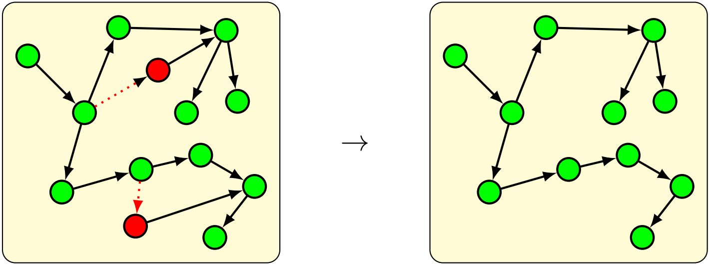
Reductions – an overview



Reductions – an overview



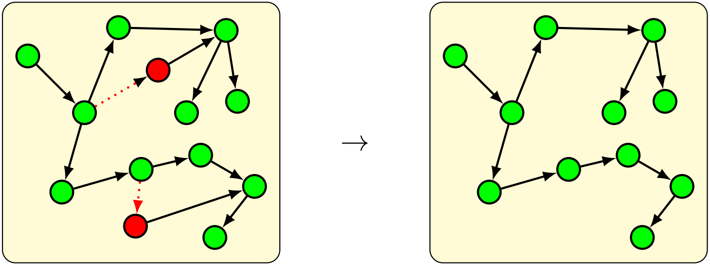
Reductions – an overview



Reduction function:

$R\colon S \to 2^{\Sigma}$

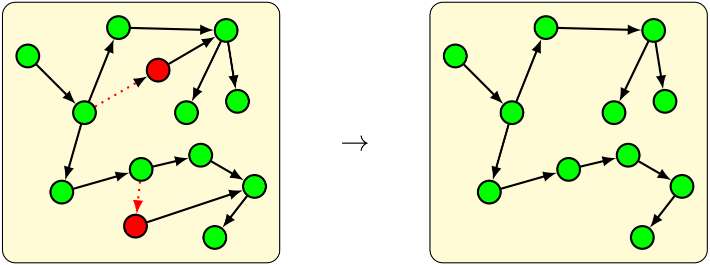
Reductions – an overview



Reduction function:

$R\colon S \to 2^{\Sigma}$ $\quad (R(s) \subseteq \text{enabled}(s))$

Reductions – an overview

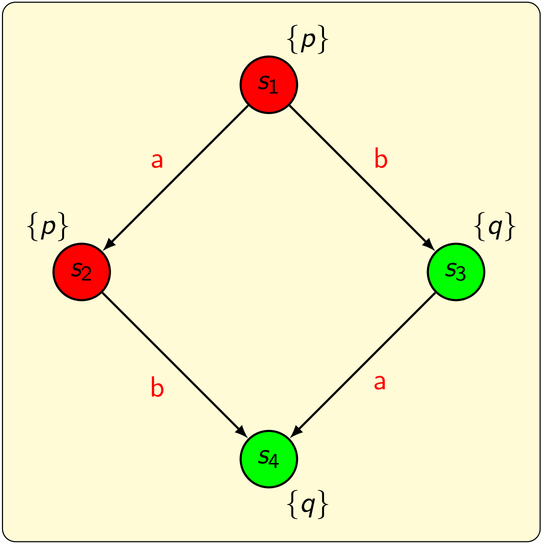


Reduction function:

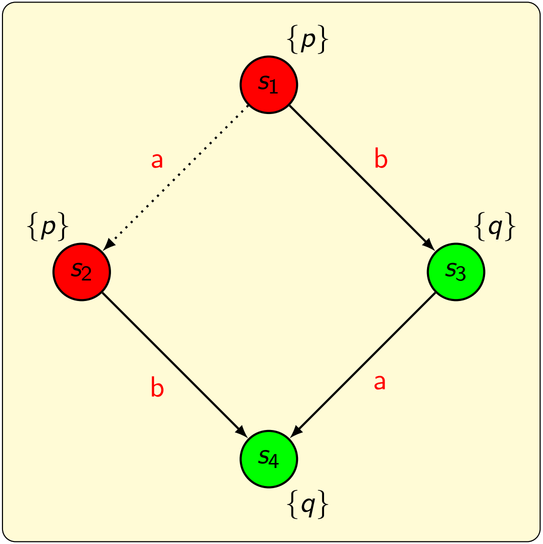
$R\colon S \to 2^{\Sigma}$ ($R(s) \subseteq \text{enabled}(s)$)

If $R(s) \neq \text{enabled}(s)$, then $R(s)$ consists of reduction transitions.

Basic concepts



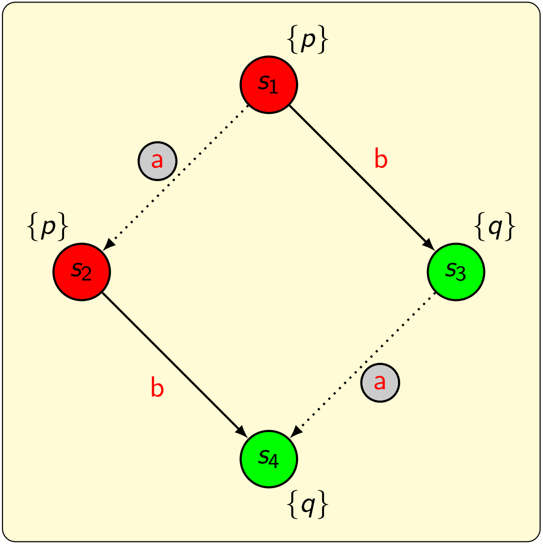
Basic concepts



Stuttering transition:

- No observable change

Basic concepts



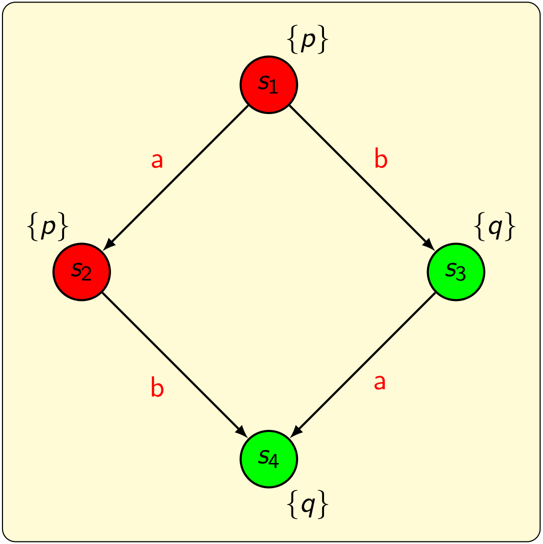
Stuttering transition:

- No observable change

Stuttering action:

- Yields only stuttering transitions

Basic concepts



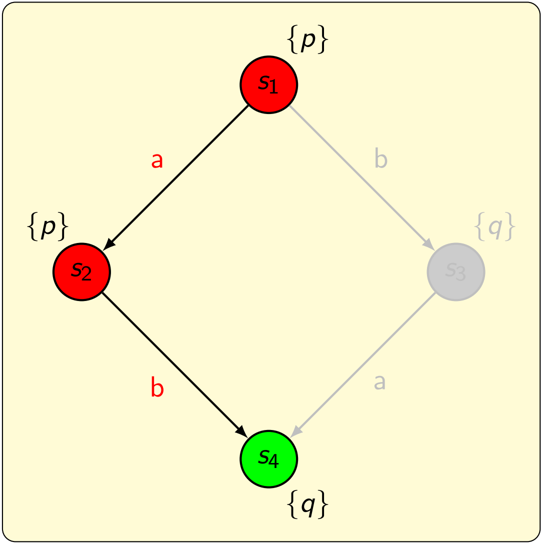
Stuttering transition:

- No observable change

Stuttering action:

- Yields only stuttering transitions

Basic concepts



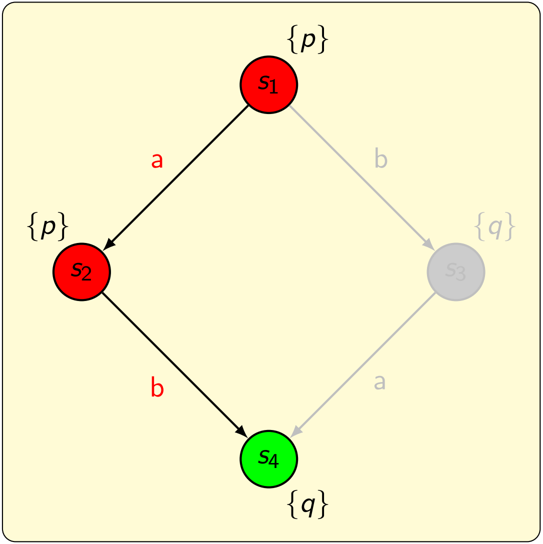
Stuttering transition:

- No observable change

Stuttering action:

- Yields only stuttering transitions

Basic concepts



Stuttering transition:

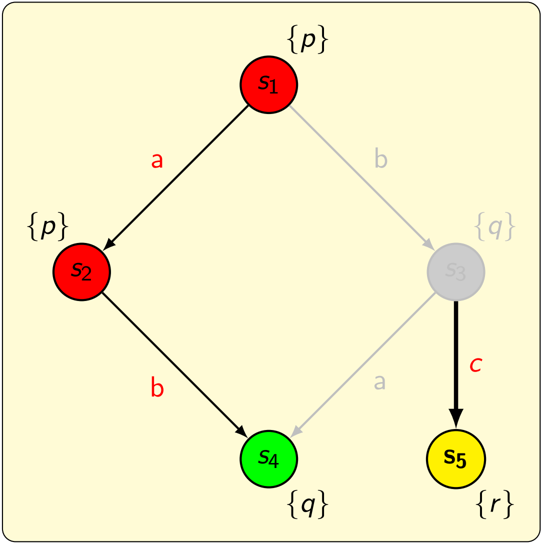
- No observable change

Stuttering action:

- Yields only stuttering transitions

$\{p\}\{p\}\{q\} =_{st} \{p\}\{q\}\{q\}$

Basic concepts



Stuttering transition:

- No observable change

Stuttering action:

- Yields only stuttering transitions

$\{\mathbf{p}\}\{\mathbf{p}\}\{\mathbf{q}\} =_{st} \{\mathbf{p}\}\{\mathbf{q}\}\{\mathbf{q}\}$

Correctness criteria

Correctness criteria for reductions:

- Preservation of $\text{LTL}_{\setminus X}$ (linear time)
- Preservation of $\text{CTL}^*_{\setminus X}$ (branching time)

Correctness criteria

Correctness criteria for reductions:

- Preservation of (quantitative) $\mathrm{LTL}_{\setminus X}$ (linear time)
- Preservation of (P)$\mathrm{CTL}^*_{\setminus X}$ (branching time)

Correctness criteria

Correctness criteria for reductions:

- Preservation of (quantitative) $LTL_{\setminus X}$ (linear time)
- Preservation of (P)$CTL^{*}_{\setminus X}$ (branching time)

	Partial-order reduction	Confluence reduction
Linear time	[BGC'04, AN'04]	–
Branching time	[BAG'06]	[TSP'11]

Correctness criteria

Correctness criteria for reductions:

- Preservation of (quantitative) $LTL_{\setminus X}$ (linear time)
- Preservation of (P)$CTL^*_{\setminus X}$ (branching time)

	Partial-order reduction		Confluence reduction
Linear time	[BGC'04, AN'04]		–
Branching time	[BAG'06]	$\overset{?}{\Longleftrightarrow}$	[TSP'11]

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

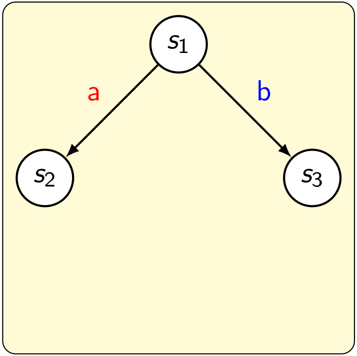
- Based on independent actions and ample sets

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of *a* and *b*:

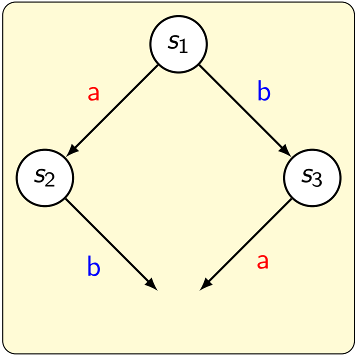


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of a and b:

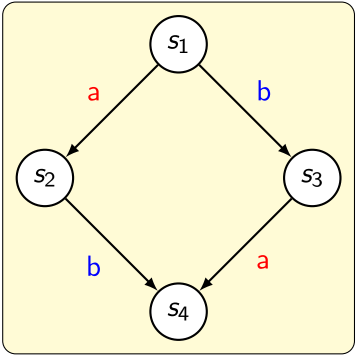


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of a and b:

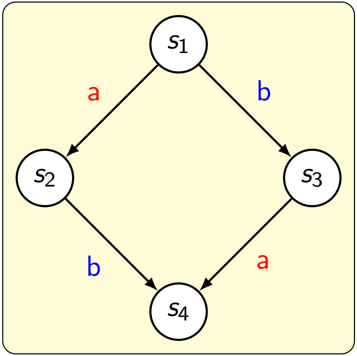


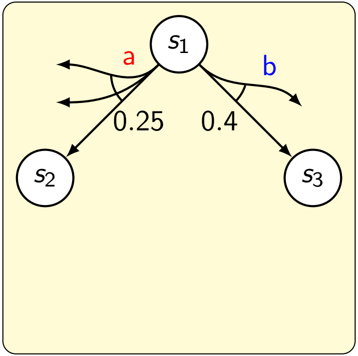
Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of *a* and *b*:



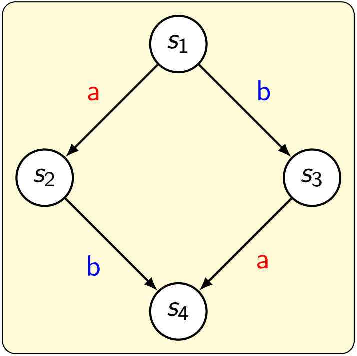


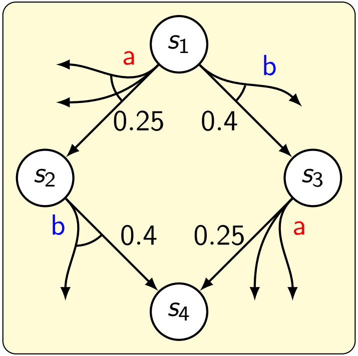
Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of *a* and *b*:



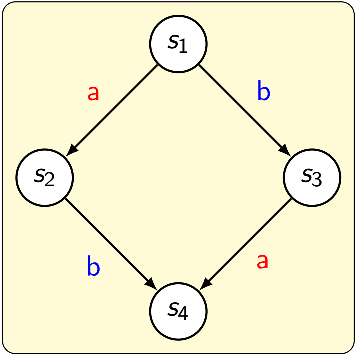


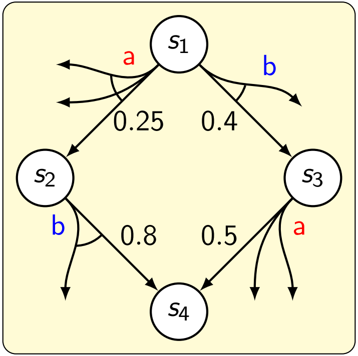
Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of *a* and *b*:



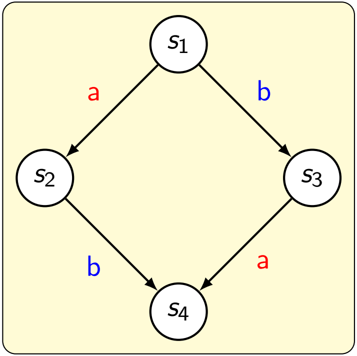


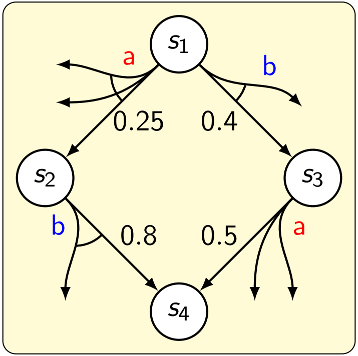
Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Independence of a and b:





$$\mathbb{P}[s_1 \xrightarrow{ab} s] = \mathbb{P}[s_1 \xrightarrow{ba} s], \forall s$$

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R \colon S \to 2^{\Sigma}$, for every $s \in S$

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R \colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1

A2

A3

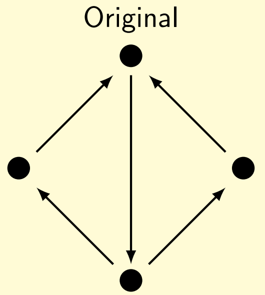
A4

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

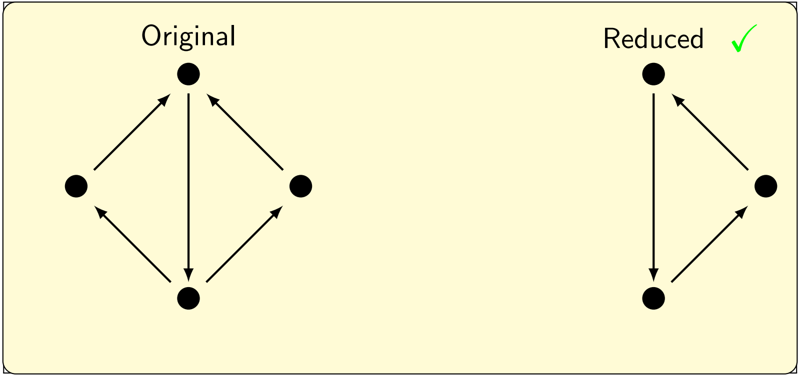


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

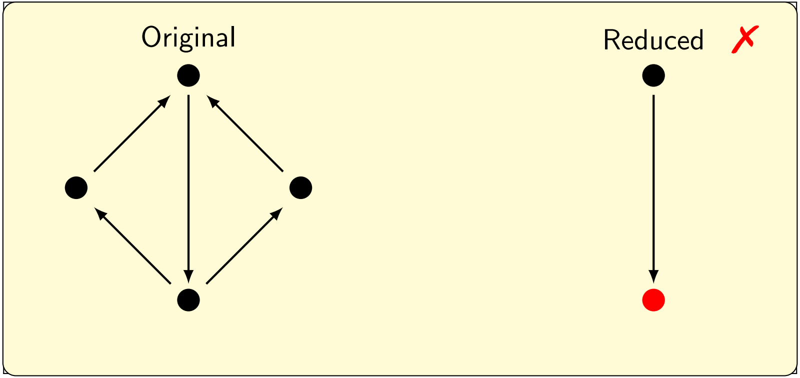


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:



Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R\colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2

A3

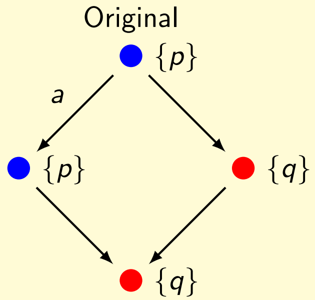
A4

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

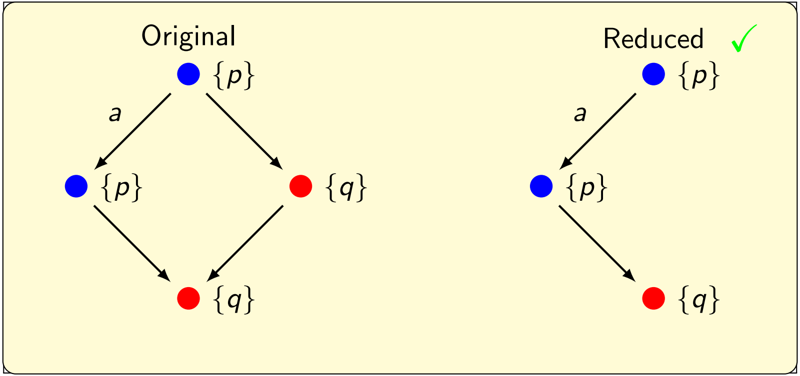


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

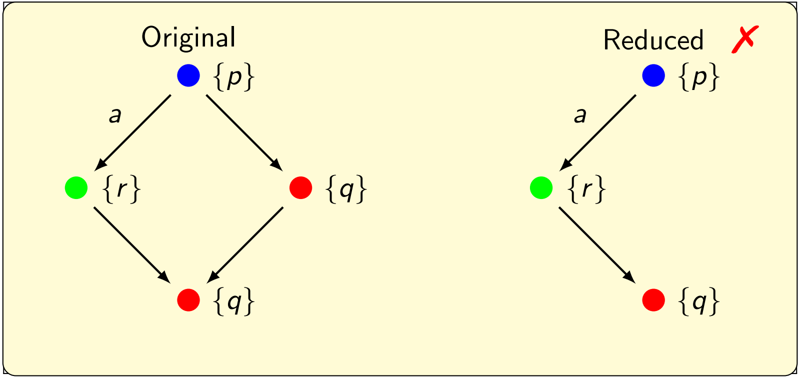


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:



Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R\colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2 For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

A3

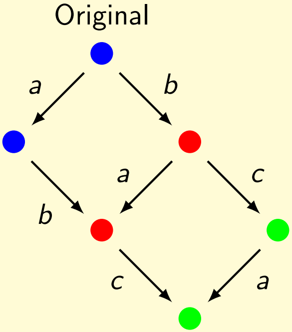
A4

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

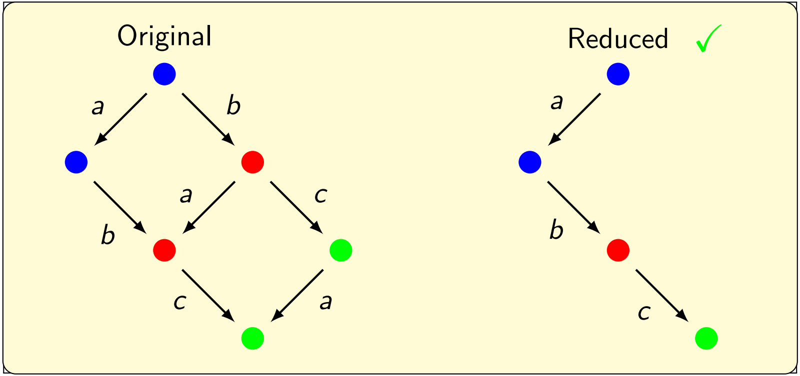


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

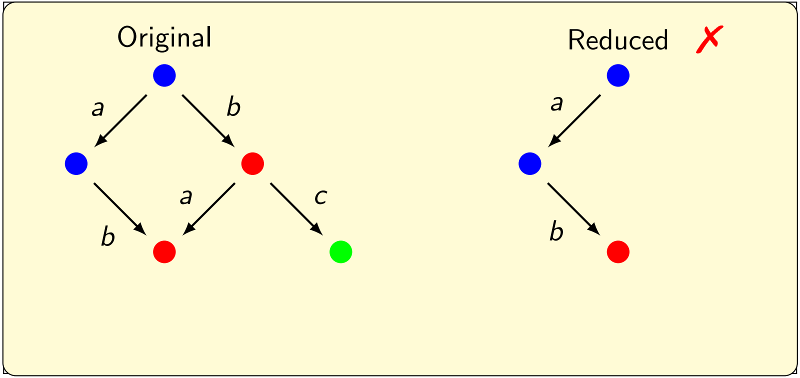


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:



Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R\colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2 For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

A3 Every cycle in the reduced MDP contains a fully-expanded state (if $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n = s$, then $\exists s_i \,.\, R(s_i) = \text{enabled}(s_i)$)

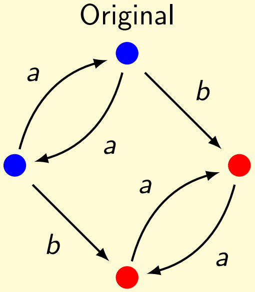
A4

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

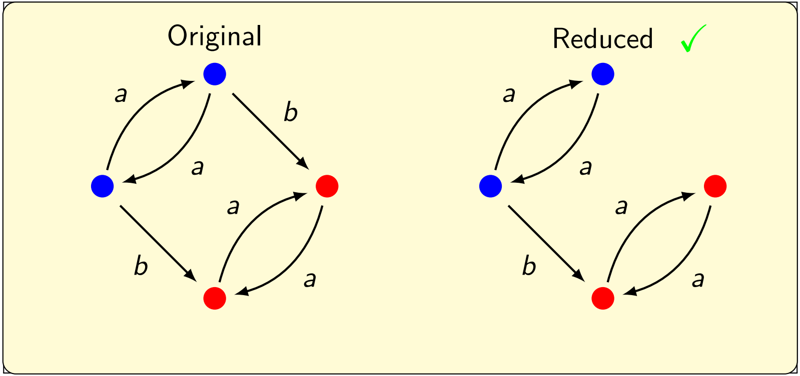


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

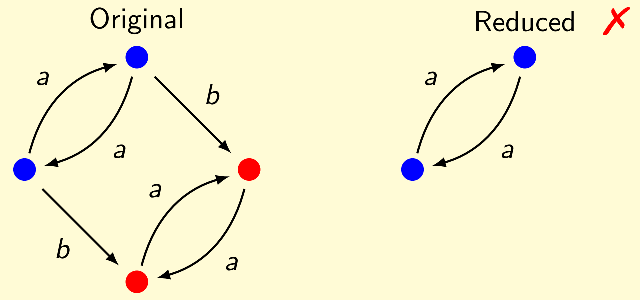


Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:



Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R\colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2 For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

A3 Every cycle in the reduced MDP contains a fully-expanded state (if $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n = s$, then $\exists s_i \,.\, R(s_i) = \text{enabled}(s_i)$)

A4 if $R(s) \neq \text{enabled}(s)$, then $|R(s)| = 1$ and the chosen action is deterministic

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R\colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2 For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

A3 Every cycle in the reduced MDP contains a fully-expanded state (if $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n = s$, then $\exists s_i \,.\, R(s_i) = \text{enabled}(s_i)$)

A4 if $R(s) \neq \text{enabled}(s)$, then $|R(s)| = 1$ and the chosen action is deterministic

Partial-order reduction: ample sets

Partial-order reduction [Baier, D'Argenio, Größer, 2006]

- Based on independent actions and ample sets

Ample set conditions:

Given a reduction function $R \colon S \to 2^{\Sigma}$, for every $s \in S$

A0 $\varnothing \neq R(s)$

A1 if $R(s) \neq \text{enabled}(s)$, then $R(s)$ contains only stuttering actions

A2 For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

A3 Every cycle in the reduced MDP contains a fully-expanded state (if $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n = s$, then $\exists s_i \, . \, R(s_i) = \text{enabled}(s_i)$)

A4 if $R(s) \neq \text{enabled}(s)$, then $|R(s)| = 1$ and the chosen action is deterministic and stuttering

Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

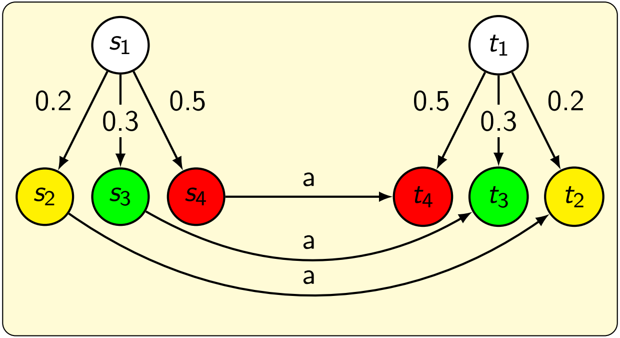
- Based on equivalent distributions and confluent transitions

Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

- Based on equivalent distributions and confluent transitions

T-equivalent distributions

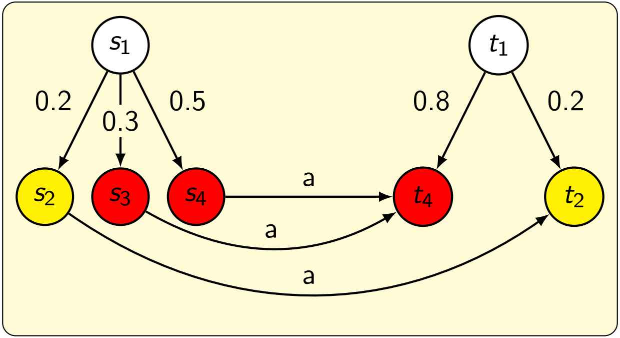


Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

- Based on equivalent distributions and confluent transitions

T-equivalent distributions

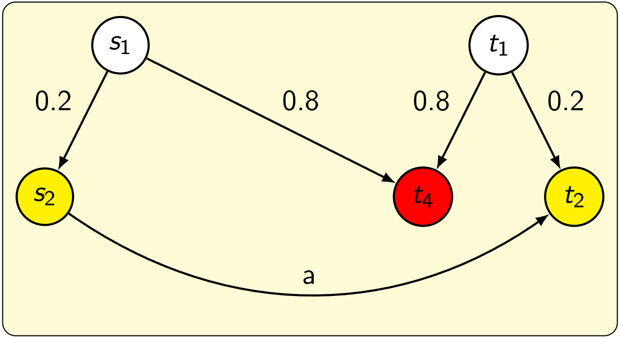


Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

- Based on equivalent distributions and confluent transitions

T-equivalent distributions



Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

- Based on equivalent distributions and confluent transitions

The main idea:

- Choose a set T of transitions
- Make sure all of them are confluent
- $R(s) = \text{enabled}(s)$ or $R(s) = \{a\}$ such that $(s \xrightarrow{a} t) \in T$

Confluence

Confluence reduction [Timmer, Stoelinga, van de Pol, 2011]

- Based on equivalent distributions and confluent transitions

The main idea:

- Choose a set T of transitions
- Make sure all of them are confluent
- $R(s) = \text{enabled}(s)$ or $R(s) = \{a\}$ such that $(s \xrightarrow{a} t) \in T$
- Make sure T is acyclic to prevent infinite postponing

Confluence

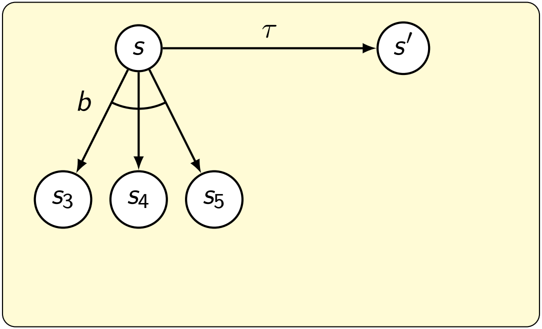
A set of transitions T is confluent if

- Every transition is labelled by a deterministic stuttering action
- If $s \xrightarrow{\tau} s' \in T$ and $s \xrightarrow{b} \mu$, then
  1. either $s' \xrightarrow{b} \nu$ and μ is T-equivalent to ν
  2. or $\mu(s') = 1$ (b deterministically goes to s')

Confluence

A set of transitions T is confluent if

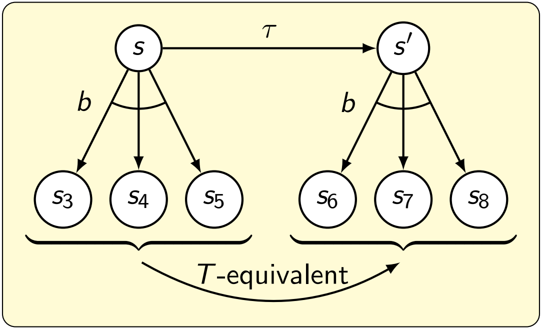
- Every transition is labelled by a deterministic stuttering action
- If $s \xrightarrow{\tau} s' \in T$ and $s \xrightarrow{b} \mu$, then
  1. either $s' \xrightarrow{b} \nu$ and μ is T-equivalent to ν
  2. or $\mu(s') = 1$ (b deterministically goes to s')



Confluence

A set of transitions T is confluent if

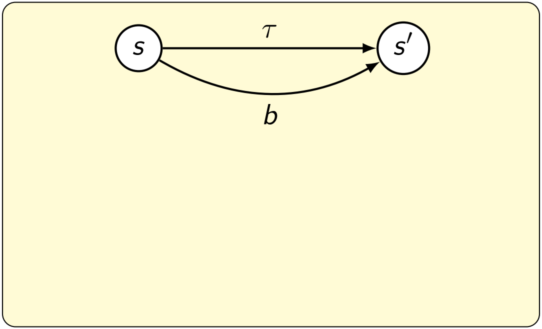
- Every transition is labelled by a deterministic stuttering action
- If $s \xrightarrow{\tau} s' \in T$ and $s \xrightarrow{b} \mu$, then
  1. either $s' \xrightarrow{b} \nu$ and μ is T-equivalent to ν
  2. or $\mu(s') = 1$ (b deterministically goes to s')



Confluence

A set of transitions T is confluent if

- Every transition is labelled by a deterministic stuttering action
- If $s \xrightarrow{\tau} s' \in T$ and $s \xrightarrow{b} \mu$, then
  1. either $s' \xrightarrow{b} \nu$ and μ is T-equivalent to ν
  2. or $\mu(s') = 1$ (b deterministically goes to s')



Comparison

Similarities among ample sets and confluence:

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $\lvert R(s) \rvert = 1$

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $\lvert R(s)\rvert = 1$
Reduction transitions	Deterministic and stuttering

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $|R(s)| = 1$
Reduction transitions	Deterministic and stuttering
Acyclicity	No cycle of reduction transitions allowed

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $\lvert R(s) \rvert = 1$
Reduction transitions	Deterministic and stuttering
Acyclicity	No cycle of reduction transitions allowed
Preservation	Branching time properties

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $\lvert R(s)\rvert = 1$
Reduction transitions	Deterministic and stuttering
Acyclicity	No cycle of reduction transitions allowed
Preservation	Branching time properties

Differences between ample sets and confluence:

POR For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

Comparison

Similarities among ample sets and confluence:

	Requirement
Size of $R(s)$	$R(s) = \text{enabled}(s)$ or $\lvert R(s) \rvert = 1$
Reduction transitions	Deterministic and stuttering
Acyclicity	No cycle of reduction transitions allowed
Preservation	Branching time properties

Differences between ample sets and confluence:

POR For every original path $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} s_n \xrightarrow{b} t$ such that $b \notin R(s)$ and b depends on $R(s)$, there exists an i such that $a_i \in R(s)$

Conf If $s \xrightarrow{\tau} t$ and $s \xrightarrow{b} \mu$, then $\mu = \text{dirac}(t)$ or $t \xrightarrow{b} \nu$ and μ is equivalent to ν.

Comparison – POR implies Confluence

Theorem

Let R be a reduction function satisfying the ample set conditions. Then, all reduction transitions are confluent.

Comparison – POR implies Confluence

Theorem

Let R be a reduction function satisfying the ample set conditions. Then, all reduction transitions are confluent.

Or:

> *Any reduction allowed by partial-order reduction is also allowed by confluence reduction.*

Comparison – POR implies Confluence

Theorem

Let R be a reduction function satisfying the ample set conditions. Then, all reduction transitions are confluent.

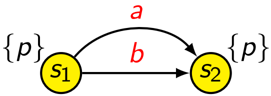
Or:

> *Any reduction allowed by partial-order reduction is also allowed by confluence reduction.*

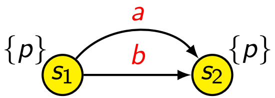
Proof (sketch).

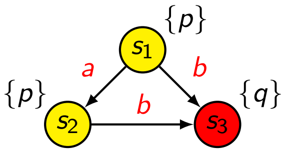
1. Take the set of all reduction transitions of the partial-order reduction.
2. Recursively add transitions needed to complete the confluence diamonds.
3. Proof that the resulting set is indeed confluent. □

Comparison – Confluence does not imply POR

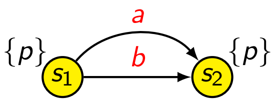


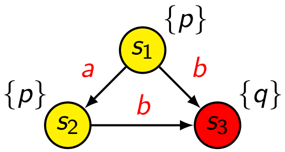
Comparison – Confluence does not imply POR

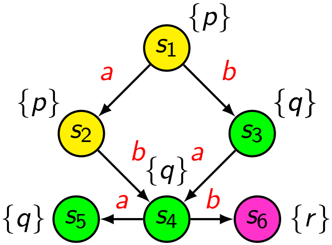




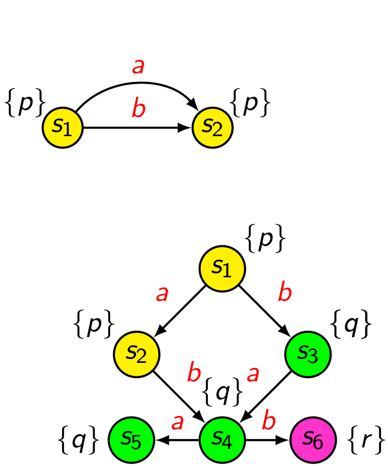
Comparison – Confluence does not imply POR



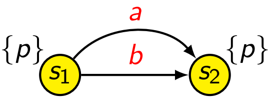


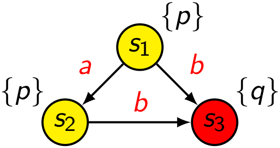


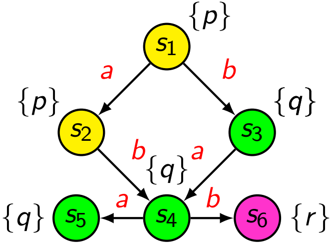
Comparison – Confluence does not imply POR

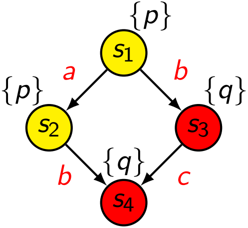


Comparison – Confluence does not imply POR









POR's notion of independence is stronger than necessary.

Strengthening of confluence

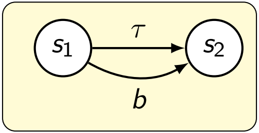
We can change confluence in the following way:

- Do not allow shortcuts

Strengthening of confluence

We can change confluence in the following way:

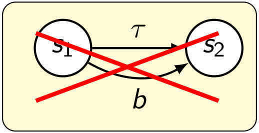
- Do not allow shortcuts



Strengthening of confluence

We can change confluence in the following way:

- Do not allow shortcuts



Strengthening of confluence

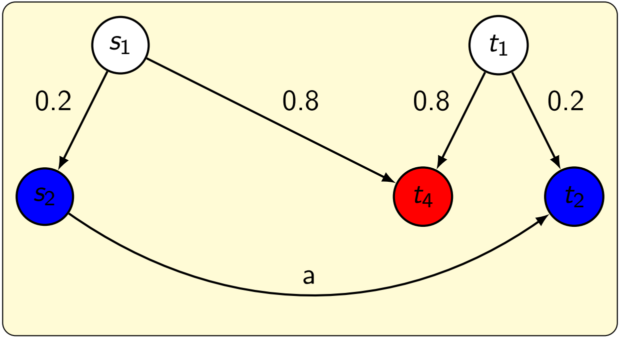
We can change confluence in the following way:

- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent

Strengthening of confluence

We can change confluence in the following way:

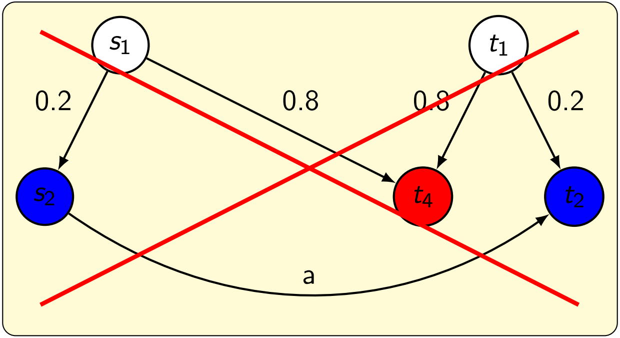
- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent



Strengthening of confluence

We can change confluence in the following way:

- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent



Strengthening of confluence

We can change confluence in the following way:

- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent
- Require confluent actions to be globally independent of other actions enabled in the same states (or weaken ample sets)

Strengthening of confluence

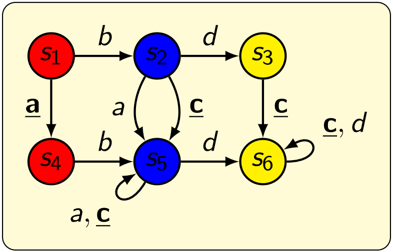
We can change confluence in the following way:

- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent
- Require confluent actions to be globally independent of other actions enabled in the same states (or weaken ample sets)
- Require action-separability

Strengthening of confluence

We can change confluence in the following way:

- Do not allow shortcuts
- Do not allow overlapping distributions to be equivalent
- Require confluent actions to be globally independent of other actions enabled in the same states (or weaken ample sets)
- Require action-separability



Strengthening of confluence

Theorem

Under the strengthened notion of confluence, every acyclic confluence reduction is an ample set reduction.

Strengthening of confluence

Theorem

Under the strengthened notion of confluence, every acyclic confluence reduction is an ample set reduction.

Corollary

Under the above circumstances, confluence reduction and ample set reduction coincide.

Strengthening of confluence

Theorem

Under the strengthened notion of confluence, every acyclic confluence reduction is an ample set reduction.

Corollary

Under the above circumstances, confluence reduction and ample set reduction coincide.

Corollary

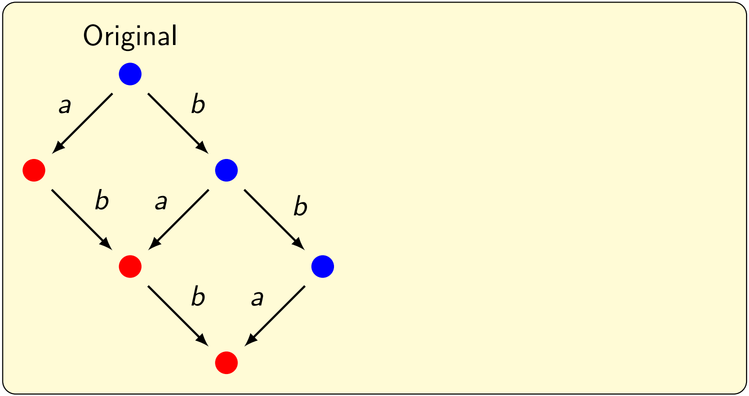
In the non-probabilistic setting, the same statements hold: confluence is stronger than partial-order reduction, and the notions are equivalent for the strengthened variant of confluence.

Implications

State space generation using representatives:

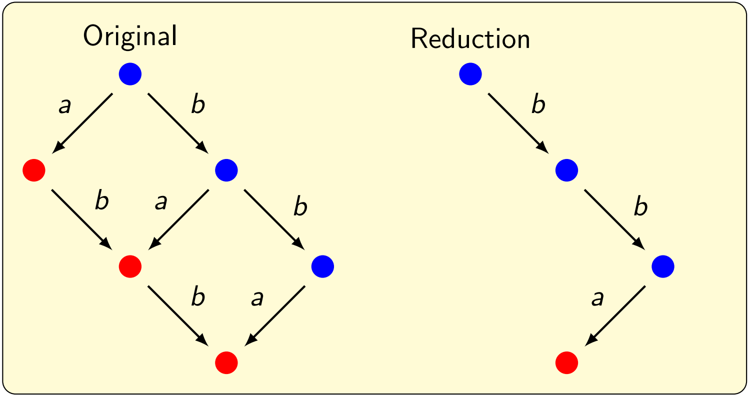
Implications

State space generation using representatives:



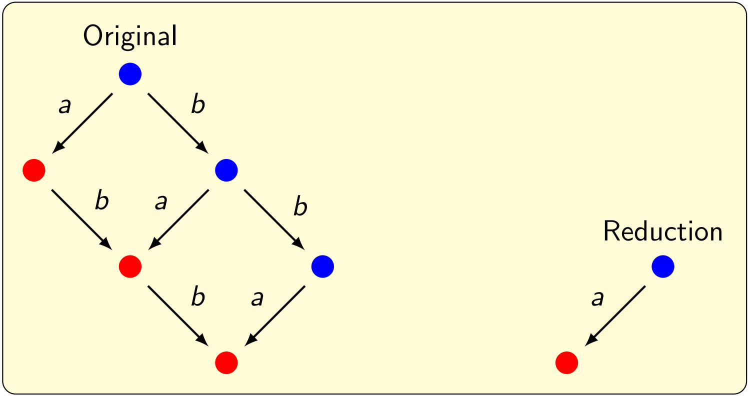
Implications

State space generation using representatives:



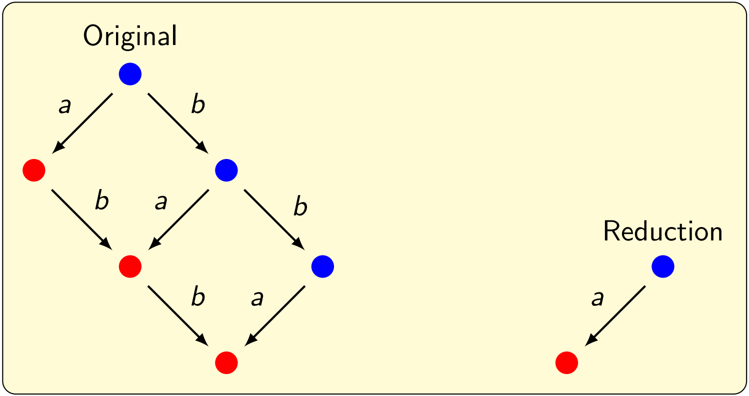
Implications

State space generation using representatives:



Implications

State space generation using representatives:



- Representative in bottom strongly connected component
- Additional reduction of states and transitions
- No need for the cycle condition anymore!

Conclusions

What to take home from this...

- We adapted the existing notion of confluence reduction to work in a state-based setting with MDPs.
- We proved that every ample set can be mimicked by a confluent set, but the the converse doesn't always hold.
- We showed how to make ample set reduction and confluence reduction equivalent
- We demonstrated one implication of our results, applying a technique from confluence reduction to POR
- The results are independent of specific heuristics, and also hold non-probabilistically

Questions

Questions?

A paper, containing all details and proofs, can be found at

`http://wwwhome.cs.utwente.nl/~timmer/research.php`