Behavioural Preorders on Stochastic Systems - Logical, Topological, and Computational Aspects

Thesis defence

January 29, 2019

Mathias Ruggaard Pedersen

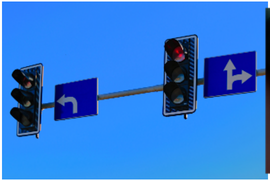
Department of Computer Science, Aalborg University, Denmark

AALBORG UNIVERSITY
DENMARK

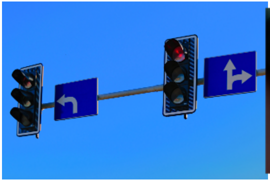
Agenda

Introduction

Introduction
Background



Introduction
Background



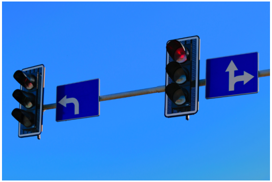




Introduction
Background

Introduction
Time is important

- Airbag must deploy within a precise time window.
- Light must not be red for more than a minute.
- A pacemaker must take over quickly and produce a precisely timed pattern.



Introduction

Time is important

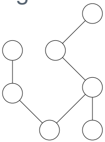
We want to be able to analyse timing aspects of systems.

Introduction
Model checking

Introduction

Model checking

Modelling/formalising



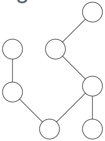
Introduction

Model checking

Requirements

"Must complete within two minutes."

Modelling/formalising



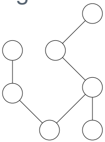
Introduction
Model checking

Requirements

"Must complete within two minutes."

Translating to formal specification language

Modelling/formalising



φ

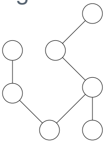
Introduction
Model checking

Requirements

"Must complete within two minutes."

Translating to formal specification language

Modelling/formalising



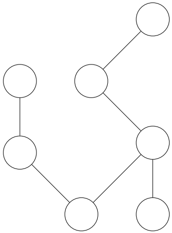
$\models \quad \varphi$

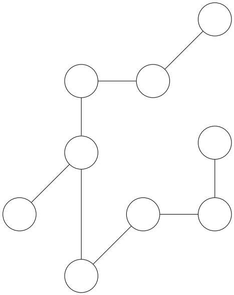
Introduction
Model checking

$$M \models \varphi$$

The model M satisfies the requirements given by φ.

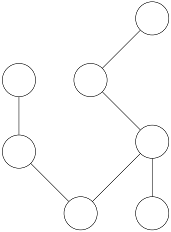
Introduction
Relations between models



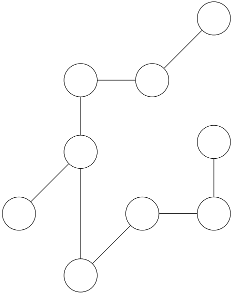


Introduction
Relations between models

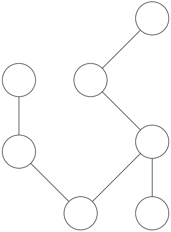
Bisimulation



$\sim$



Introduction
Relations between models

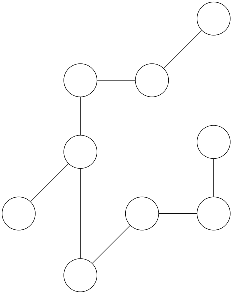


Bisimulation

$\sim$

Simulation

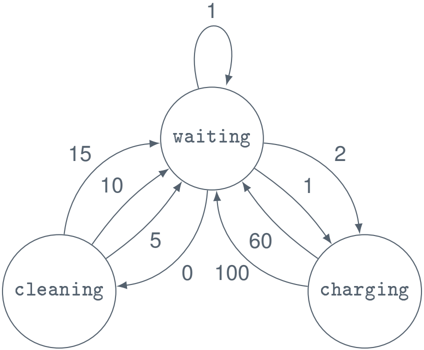
$\precsim$



Models

Models
Weighted transition systems

Robot vacuum cleaner



Models

Weighted transition systems

Definition 2.6.1

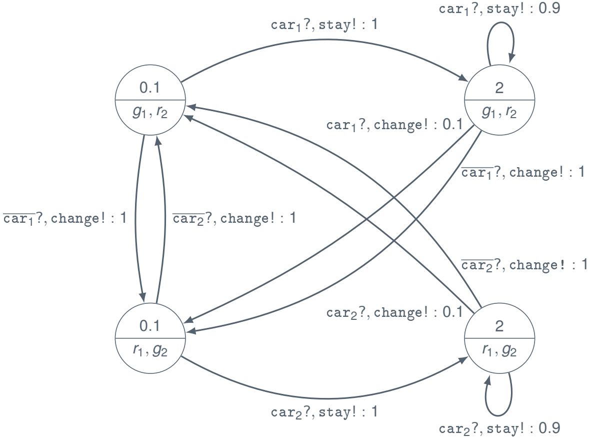
A *weighted transition system (WTS)* is a tuple $\mathcal{M} = (S, \rightarrow, \ell)$, where

- S is a set of *states*,
- $\rightarrow \subseteq S \times \mathbb{R}_{\geq 0} \times S$ is the *transition relation*, and
- $\ell : S \rightarrow 2^{\mathcal{AP}}$ is the *labelling function*.

Models

Semi-Markov processes

Intelligent traffic light



Models

Semi-Markov processes

Definition 2.6.4

A *semi-Markov process (SMP)* is a tuple $\mathcal{M} = (S, \tau, \rho, \ell)$, where

- S is a countable set of *states*,
- $\tau : S \times \mathtt{In} \to \mathcal{D}(S \times \mathtt{Out})$ is the *transition function*,
- $\rho : S \to \mathcal{D}(\mathbb{R}_{\geq 0})$ is the *time-residence function*, and
- $\ell : S \to 2^{\mathcal{AP}}$ is the *labelling* function.

Models

Semi-Markov processes

Reactive semi-Markov processes:

$$\tau : S \times \texttt{In} \to \mathcal{D}(S) \quad \textit{input}$$

Generative semi-Markov processes:

$$\tau : S \to \mathcal{D}(S \times \texttt{Out}) \quad \textit{output}$$

Contributions

Contributions
Papers

- Paper A: *Reasoning About Bounds in Weighted Transition Systems*, published in LMCS.
  Co-authors: Mikkel Hansen, Kim Guldstrand Larsen, and Radu Mardare.
- Paper B: *Timed Comparisons of Semi-Markov Processes*, published in LATA '18.
  Co-authors: Nathanaël Fijalkow, Giorgio Bacci, Kim Guldstrand Larsen, and Radu Mardare.
- Paper C: *A Faster-Than Relation for Semi-Markov Decision Processes*, unpublished.
  Co-authors: Giorgio Bacci and Kim Guldstrand Larsen.
- Paper D: *A Hemimetric Extension of Simulation for Semi-Markov Decision Processes*, published in QEST '18.
  Co-authors: Giorgio Bacci, Kim Guldstrand Larsen, and Radu Mardare.

Paper A

Contributions

Paper A

Contribution 1

We present a language for reasoning about lower and upper bounds in weighted transition systems and we show that this language characterises exactly those systems that have the same kind of behaviour.

Contributions
Paper A

Weighted logic with bounds (WLWB):

$$\varphi, \psi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid L_r\varphi \mid M_r\varphi$$

Contributions

Paper A

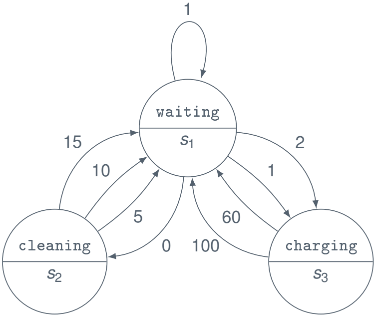
Weighted logic with bounds (WLWB):

$$\varphi, \psi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid L_r\varphi \mid M_r\varphi$$

$L_r\varphi$: a transition with *at least* weight r can be taken to where φ holds.

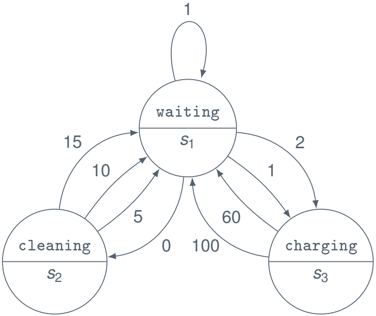
$M_r\varphi$: a transition with *at most* weight r can be taken to where φ holds.

Contributions
Paper A



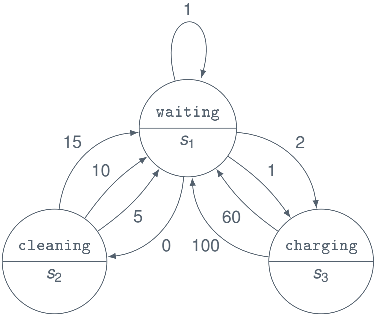
$s_1 \models M_2 \texttt{charging}$

Contributions
Paper A



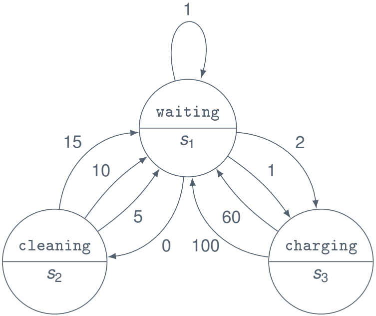
$s_1 \models M_2\,\texttt{charging}$
$s_1 \not\models M_1\,\texttt{charging}$

Contributions
Paper A



$s_1 \models M_2\texttt{charging}$
$s_1 \not\models M_1\texttt{charging}$
$s_2 \models L_2\texttt{waiting}$

Contributions
Paper A



$$s_1 \models M_2\,\texttt{charging}$$
$$s_1 \not\models M_1\,\texttt{charging}$$
$$s_2 \models L_2\,\texttt{waiting}$$
$$s_2 \not\models L_7\,\texttt{waiting}$$

Contributions
Paper A

Theorem A.2.5
For image-finite WTSs, we have

$$s \sim t \quad \text{if and only if} \quad \text{for all } \varphi, s \models \varphi \text{ if and only if } t \models \varphi.$$

Contributions

Paper A

Contribution 2

We provide a complete axiomatisation of the logical specification language, and give an algorithm for deciding the model checking problem and an algorithm for deciding satisfiability of a formula.

Contributions
Paper A

(A1):	$\vdash \neg L_0 \bot$	
(A2):	$\vdash L_{r+q}\varphi \to L_r\varphi$	if $q > 0$
(A2′):	$\vdash M_r\varphi \to M_{r+q}\varphi$	if $q > 0$
(A3):	$\vdash L_r\varphi \wedge L_q\psi \to L_{\min\{r,q\}}(\varphi \vee \psi)$	
(A3′):	$\vdash M_r\varphi \wedge M_q\psi \to M_{\max\{r,q\}}(\varphi \vee \psi)$	
(A4):	$\vdash L_r(\varphi \vee \psi) \to L_r\varphi \vee L_r\psi$	
(A5):	$\vdash \neg L_0\psi \to (L_r\varphi \to L_r(\varphi \vee \psi))$	
(A5′):	$\vdash \neg L_0\psi \to (M_r\varphi \to M_r(\varphi \vee \psi))$	
(A6):	$\vdash L_{r+q}\varphi \to \neg M_r\varphi$	if $q > 0$
(A7):	$\vdash M_r\varphi \to L_0\varphi$	
(R1):	$\vdash \varphi \to \psi \implies \vdash (L_r\psi \wedge L_0\varphi) \to L_r\varphi$	
(R1′):	$\vdash \varphi \to \psi \implies \vdash (M_r\psi \wedge L_0\varphi) \to M_r\varphi$	
(R2):	$\vdash \varphi \to \psi \implies \vdash L_0\varphi \to L_0\psi$	

+ axioms for propositional logic.

Contributions

Paper A

Soundness and completeness

Theorem A.4.2 and A.4.10

$$\vdash \varphi \quad \text{if and only if} \quad \models \varphi$$

Contributions
Paper A

Model checking: Does a given model M satisfy a given formula φ?

Theorem A.5.4

The model checking problem for WLWB is decidable.

Contributions

Paper A

Model checking: Does a given model *M* satisfy a given formula φ?

Theorem A.5.4

The model checking problem for WLWB is decidable.

Satisfiability: Does there exist a model which satisfies a given formula φ?

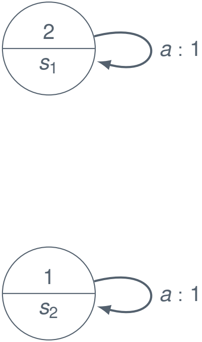
Theorem A.5.11

The satisfiability problem for WLWB is decidable.

Paper B and Paper C

Contributions

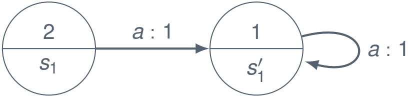
Paper B and Paper C

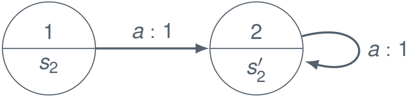


$s_1 \preceq s_2$

Contributions

Paper B and Paper C

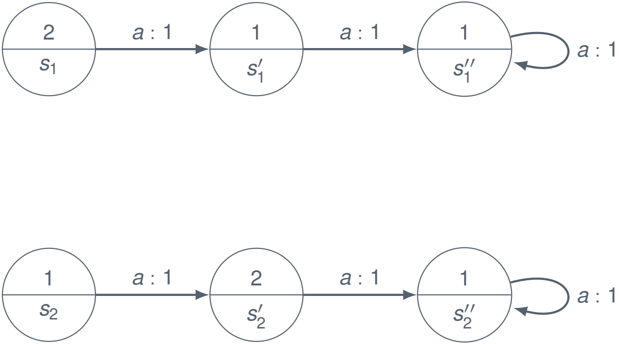




$s_1 \not\preceq s_2$

Contributions

Paper B and Paper C



$s_1 \preceq s_2$

Contributions

Paper B and Paper C

Generative:

Definition B.2.3

s_1 is *faster than* s_2 ($s_1 \preceq s_2$) if for all $a_1 \dots a_n$ and t we have

$$\mathbb{P}(s_1)(a_1 \dots a_n, t) \geq \mathbb{P}(s_2)(a_1 \dots a_n, t).$$

Contributions

Paper B and Paper C

Generative:

Definition B.2.3

s_1 is *faster than* s_2 ($s_1 \preceq s_2$) if for all $a_1 \dots a_n$ and t we have

$$\mathbb{P}(s_1)(a_1 \dots a_n, t) \geq \mathbb{P}(s_2)(a_1 \dots a_n, t).$$

Reactive:

Definition C.4.3

s_1 is *faster than* s_2 ($s_1 \preceq s_2$) if for all schedulers σ, $a_1 \dots a_n$, and t there exists a scheduler σ' such that

$$\mathbb{P}^{\sigma'}(s_1)(a_1 \dots a_n, t) \geq \mathbb{P}^{\sigma}(s_2)(a_1 \dots a_n, t).$$

Contributions
Paper B and Paper C

Contribution 3
We show that deciding the faster-than relation is a difficult problem. In particular, the relation is undecidable and approximating it up to a multiplicative constant is impossible.

Contributions

Paper B and Paper C

Contribution 4

We give an algorithm for approximating a time-bounded version of the faster-than relation up to an additive constant for slow processes.

Contributions

Paper B and Paper C

Assumptions:

- Time-bounded: We only look at behaviours up to a given time bound.
- Slow residence-time functions: all transitions take *some* time to fire.

Contributions
Paper B and Paper C

Assumptions:

- Time-bounded: We only look at behaviours up to a given time bound.
- Slow residence-time functions: all transitions take *some* time to fire.

Theorem B.4.3 and C.5.6

The time-bounded approximation problem is decidable.

Contributions

Paper B and Paper C

Contribution 5

We give an algorithm for unambiguous processes which can decide whether one process is faster than another.

Contributions

Paper B and Paper C

A SMP is *unambiguous* if every output label leads to a unique successor state.

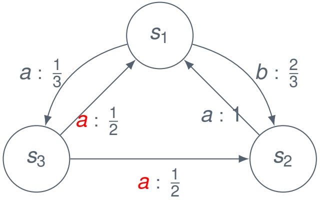


Figure 1: Ambiguous

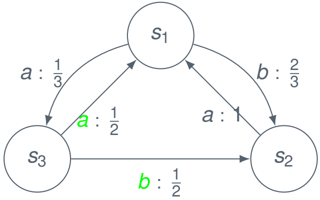


Figure 2: Unambiguous

Contributions

Paper B and Paper C

A SMP is *unambiguous* if every output label leads to a unique successor state.

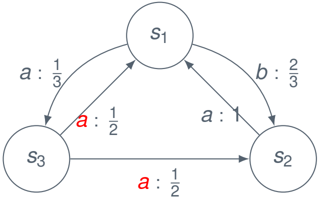


Figure 1: Ambiguous

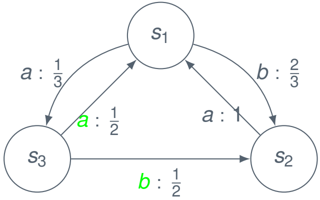


Figure 2: Unambiguous

Theorem B.5.2

For unambiguous SMPs, the faster-than problem is decidable.

Contributions

Paper B and Paper C

Contribution 6

We introduce a logical language which characterises the faster-than relation and we show that both the satisfiability problem and the model checking problem for this language are decidable.

Contributions

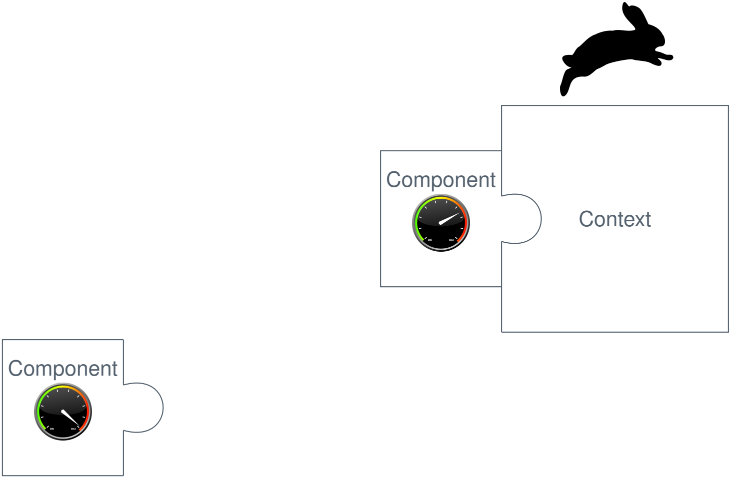
Paper B and Paper C

Contribution 7

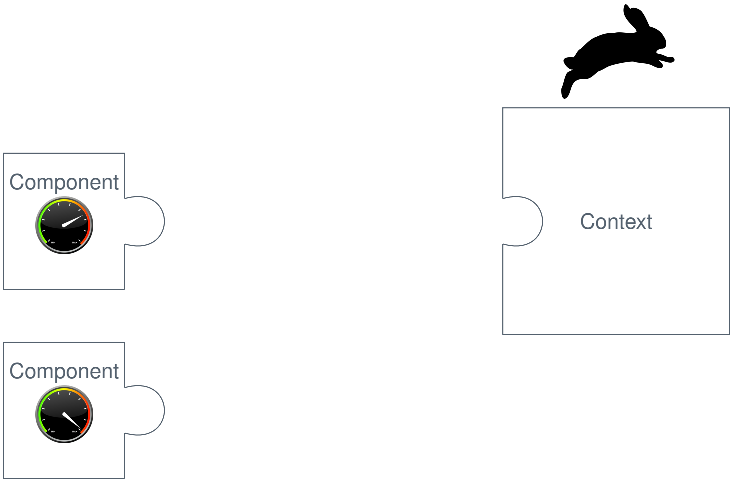
We give examples of parallel timing anomalies occuring for the faster-than relation. However, we also describe some conditions under which parallel timing anomalies can not occur, and we develop an algorithm for checking whether these conditions are met.

Contributions

Paper B and Paper C

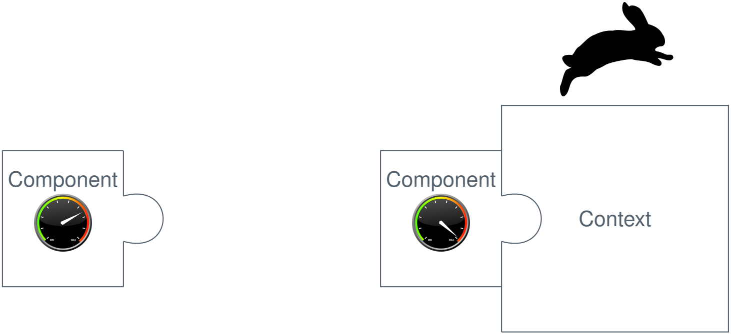


Contributions
Paper B and Paper C



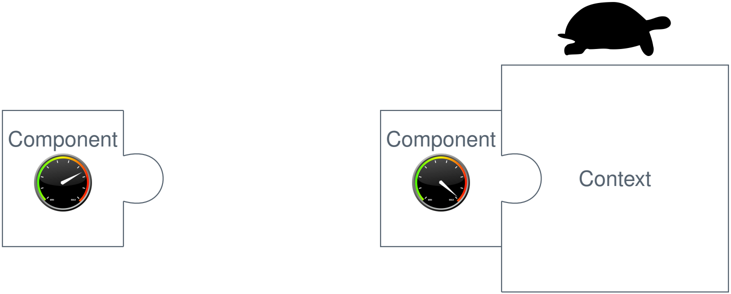
Contributions

Paper B and Paper C



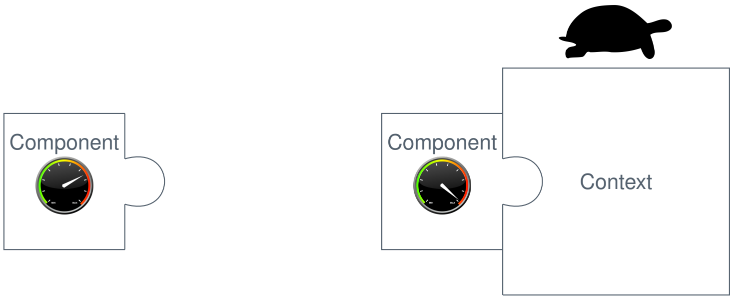
Contributions

Paper B and Paper C



Contributions

Paper B and Paper C



Timing anomaly

Contributions

Paper B and Paper C

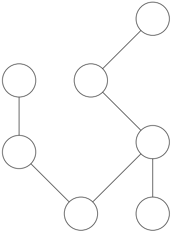
Theorem C.6.15

There exist decidable conditions that guarantee the absence of timing anomalies.

Paper D

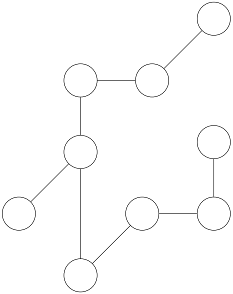
Contributions
Paper D

Reactive processes



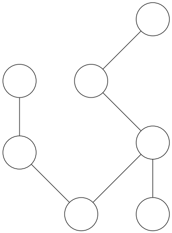
Simulation

$\precsim$



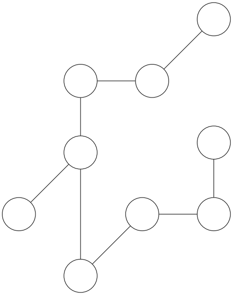
Contributions
Paper D

Reactive processes



Simulation

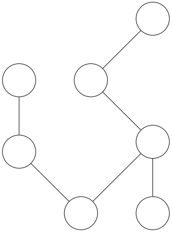
$\not\precsim$



Contributions

Paper D

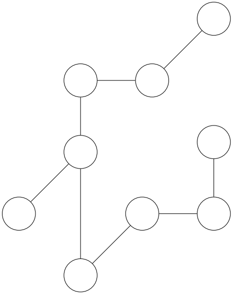
Reactive processes



Simulation

$\not\precsim$

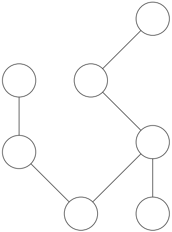
But how *close* is the process to simulating the other process?



Contributions

Paper D

Reactive processes

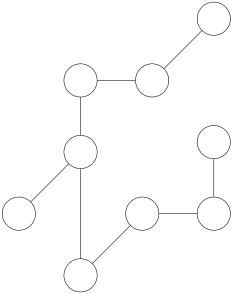


Simulation

$\not\lesssim$

But how *close* is the process to simulating the other process?

Quantitative measure of distance



Contributions
Paper D

Definition D.2.2
s_2 simulates s_1, written $s_1 \precsim s_2$, if

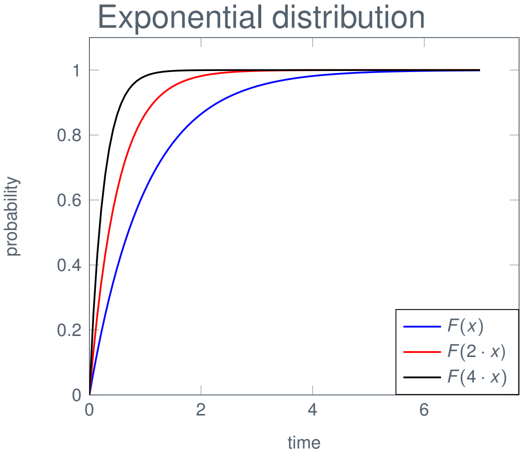
$\vdots$

- $F_{s_1}(t) \leq F_{s_2}(t)$ for all $t \in \mathbb{R}_{\geq 0}$

$\vdots$

Contributions

Paper D



Contributions

Paper D

Definition D.2.2

s_2 simulates s_1, written $s_1 \precsim s_2$, if

$\vdots$

- $F_{s_1}(t) \leq F_{s_2}(t)$ for all $t \in \mathbb{R}_{\geq 0}$

$\vdots$

Contributions
Paper D

Definition D.2.2

s_2 ε-simulates s_1, written $s_1 \precsim_{\varepsilon} s_2$, if

$\vdots$

- $F_{s_1}(t) \leq F_{s_2}(\varepsilon \cdot t)$ for all $t \in \mathbb{R}_{\geq 0}$

$\vdots$

Contributions

Paper D

Definition D.2.2

s_2 ε-simulates s_1, written $s_1 \precsim_\varepsilon s_2$, if

$\vdots$

- $F_{s_1}(t) \leq F_{s_2}(\varepsilon \cdot t)$ for all $t \in \mathbb{R}_{\geq 0}$

$\vdots$

Definition D.4.5

$$d(s_1, s_2) = \inf\{\varepsilon \geq 1 \mid s_1 \precsim_\varepsilon s_2\}$$

Contributions

Paper D

Contribution 8

We describe an algorithm for computing the distance from one process to another. This algorithm runs in polynomial time using known techniques, making it relevant for use and implementation in practice.

Contributions

Paper D

Contribution 9

We show that, under mild assumptions, composition is non-expansive with respect to the distance between semi-Markov processes.

Contributions

Paper D

Contribution 10

We introduce a logical specification language called timed Markovian logic and show that this language characterises both the ε-simulation relation and the distance between semi-Markov processes.

Contributions
Paper D

Timed Markovian logic

$$\mathrm{TML}: \quad \varphi, \varphi' ::= \alpha \mid \neg\alpha \mid \ell_p t \mid m_p t \mid L_p^a \varphi \mid M_p^a \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

Contributions
Paper D

Timed Markovian logic

$$\mathrm{TML}: \quad \varphi, \varphi' ::= \alpha \mid \neg\alpha \mid \ell_p t \mid m_p t \mid L_p^a \varphi \mid M_p^a \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

$L_p^a\varphi$: probability of going with an a to where φ holds is *at least* p.

$M_p^a\varphi$: probability of going with an a to where φ holds is *at most* p.

Contributions
Paper D

Timed Markovian logic

$$\mathrm{TML}: \quad \varphi, \varphi' ::= \alpha \mid \neg\alpha \mid \ell_p t \mid m_p t \mid L^a_p \varphi \mid M^a_p \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

$L^a_p\varphi$: probability of going with an a to where φ holds is *at least* p.

$M^a_p\varphi$: probability of going with an a to where φ holds is *at most* p.

$\ell_p t$: probability of leaving state before time t is *at least* p.

$m_p t$: probability of leaving state before time t is *at most* p.

Contributions

Paper D

Timed Markovian logic

$$\mathrm{TML}: \quad \varphi, \varphi' ::= \alpha \mid \neg\alpha \mid \ell_p t \mid m_p t \mid L_p^a \varphi \mid M_p^a \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

$L_p^a \varphi$: probability of going with an a to where φ holds is *at least* p.

$M_p^a \varphi$: probability of going with an a to where φ holds is *at most* p.

$\ell_p t$: probability of leaving state before time t is *at least* p.

$m_p t$: probability of leaving state before time t is *at most* p.

$$\mathrm{TML}^{\geq}: \quad \varphi ::= \alpha \mid \neg\alpha \mid \ell_p t \mid L_p^a \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

$$\mathrm{TML}^{\leq}: \quad \varphi ::= \alpha \mid \neg\alpha \mid m_p t \mid M_p^a \varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi'$$

Contributions

Paper D

Perturbation $(\varphi)_\varepsilon$:

- $(\ell_p t)_\varepsilon = \ell_{p\varepsilon} \cdot t$
- $(m_p t)_\varepsilon = m_{p\varepsilon} \cdot t$

Contributions

Paper D

Perturbation $(\varphi)_\varepsilon$:

- $(\ell_p t)_\varepsilon = \ell_{p\varepsilon} \cdot t$
- $(m_p t)_\varepsilon = m_{p\varepsilon} \cdot t$

Theorem D.7.2

For finite SMPs we have

- $d(s_1, s_2) \leq \varepsilon$ if and only if

  $$\text{for all } \varphi \in \mathrm{TML}^{\geq}, s_1 \models \varphi \text{ implies } s_2 \models (\varphi)_\varepsilon$$

- $d(s_2, s_1) \leq \varepsilon$ if and only if

  $$\text{for all } \varphi \in \mathrm{TML}^{\leq}, s_2 \models (\varphi)_\varepsilon \text{ implies } s_1 \models \varphi$$

Conclusion

Conclusion
Summary

- *Formalisms* for specifying, comparing, and reasoning about properties involving time.
- *Algorithms* enabling use of these formalisms in practice.

Conclusion
Summary

- *Formalisms* for specifying, comparing, and reasoning about properties involving time.
- *Algorithms* enabling use of these formalisms in practice.

- *Weighted logic with bounds* allows reasoning about upper and lower bounds on time.

Conclusion

Summary

- *Formalisms* for specifying, comparing, and reasoning about properties involving time.
- *Algorithms* enabling use of these formalisms in practice.

- *Weighted logic with bounds* allows reasoning about upper and lower bounds on time.
- *Faster-than relation* allows qualitative comparison of time behaviour of different systems.

Conclusion

Summary

- *Formalisms* for specifying, comparing, and reasoning about properties involving time.
- *Algorithms* enabling use of these formalisms in practice.

- *Weighted logic with bounds* allows reasoning about upper and lower bounds on time.
- *Faster-than relation* allows qualitative comparison of time behaviour of different systems.
- *ε-simulation* allows quantitative comparison of time behaviour of different systems.

Future work

Future work

Strong completeness

Weak completeness

$\models \varphi$ implies $\vdash \varphi$

Future work

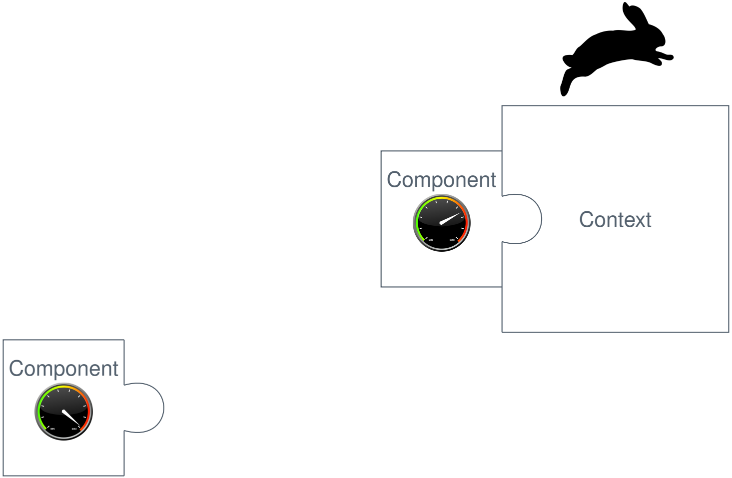
Strong completeness

Strong completeness

$\Phi \models \varphi$ implies $\Phi \vdash \varphi$

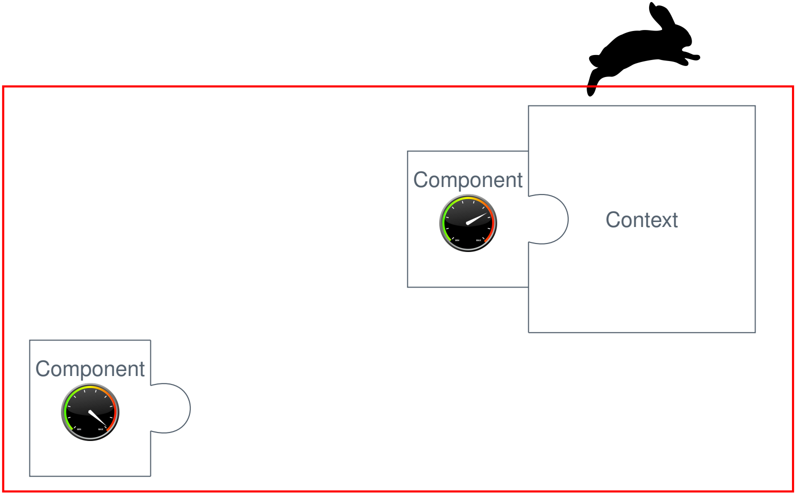
Future work

Timing anomalies

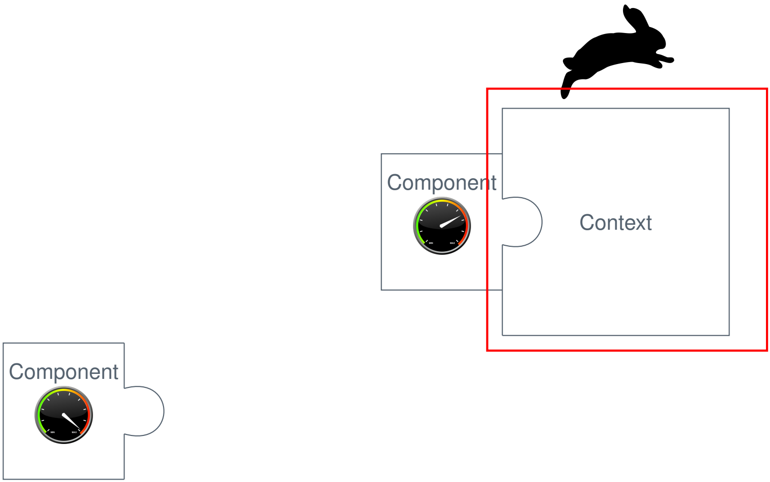


Future work

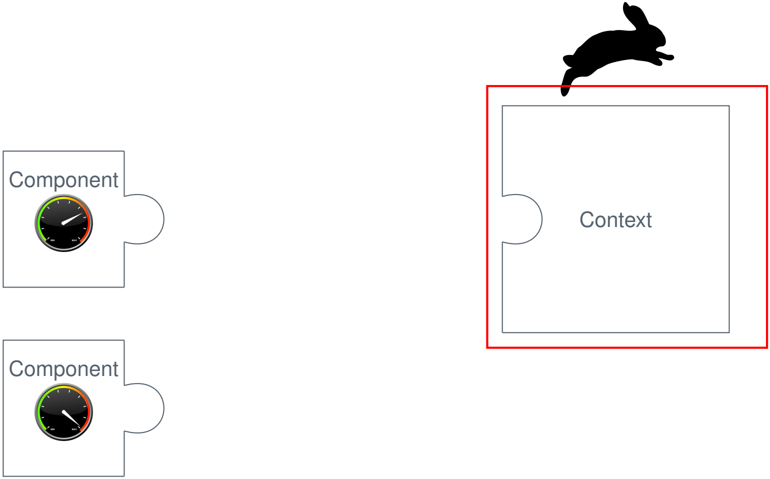
Timing anomalies



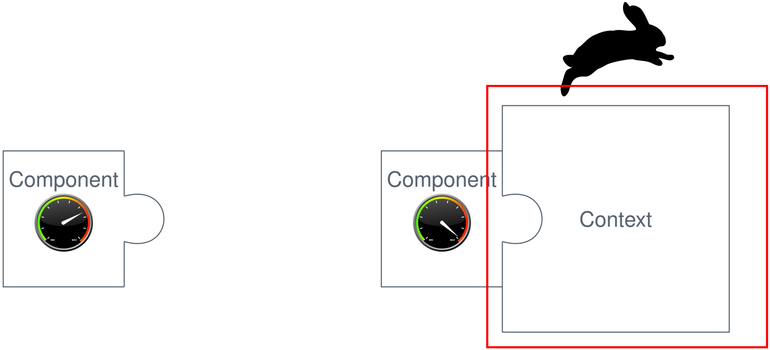
Future work
Timing anomalies



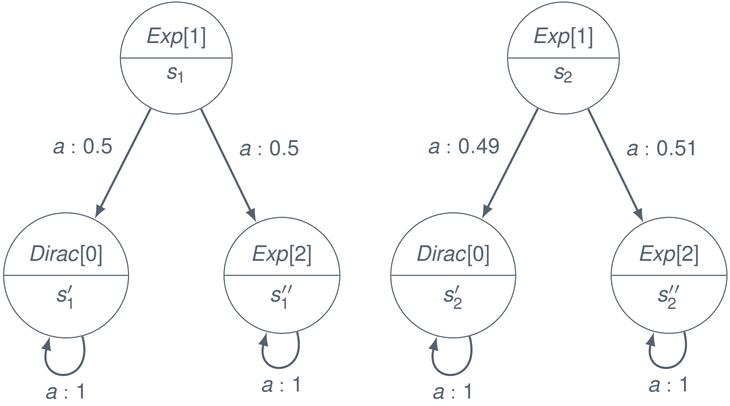
Future work
Timing anomalies



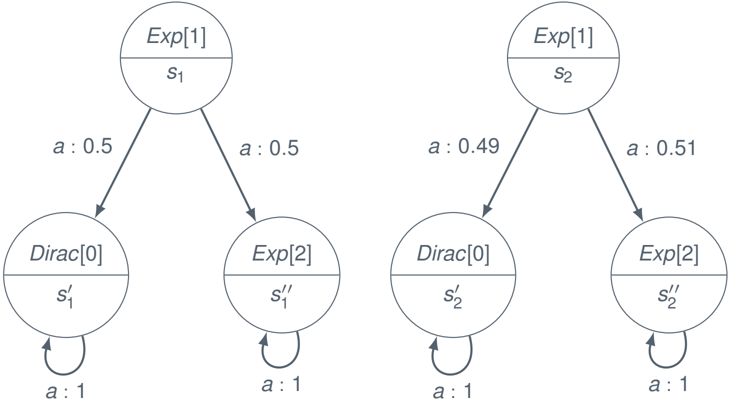
Future work
Timing anomalies



Future work
Branching in simulation distance



Future work
Branching in simulation distance



$$d(s_1, s_2) = \infty$$

Thank you!

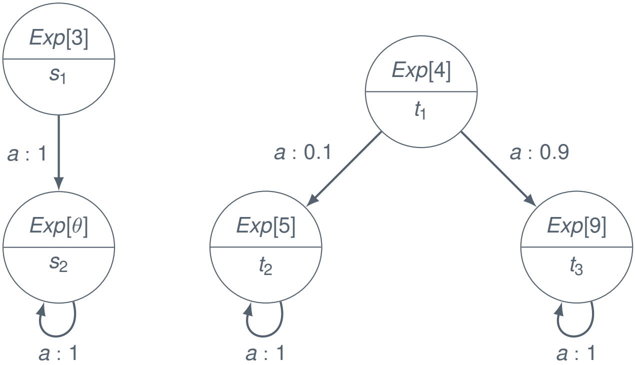
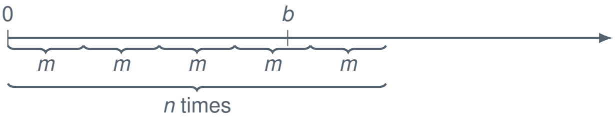


Figure 3: A semi-Markov process where $s_1 \precsim t_1$ if $\theta \leq 5$ and $s_1 \not\precsim t_1$ if $\theta > 5$.

Time-bounded approximation



- $\mathbb{P}(s, a^n, b) \to 0$ as $n \to \infty$.
- Hence we can find N such that $\mathbb{P}(s, a^n, b) \leq \varepsilon$ for all $n \geq N$.
- We only need to consider words of length $\leq N$.

Tableau

$$
(\neg\wedge)\;\dfrac{\langle\{\neg(\neg(L_2p_1 \wedge M_5L_1p_1) \wedge M_2p_2)\}, [0,0], [0,0]\rangle}{
(\neg\neg)\;\dfrac{\langle\{\neg\neg(L_2p_1 \wedge M_5L_1p_1)\}, [0,0], [0,0]\rangle}{
(\wedge)\;\dfrac{\langle\{L_2p_1 \wedge M_5L_1p_1\}, [0,0], [0,0]\rangle}{
(\mathrm{mod})\;\dfrac{\langle\{L_2p_1, M_5L_1p_1\}, [0,0], [0,0]\rangle}{
(\mathrm{mod})\;\dfrac{\langle\{p_1, L_1p_1\}, [2,\infty), [5,\infty)\rangle}{\langle\{p_1\}, [1,\infty), [0,\infty)\rangle}}}}
\qquad
(\neg\neg)\;\dfrac{\langle\{\neg\neg M_2p_2\}, [0,0], [0,0]\rangle}{
(\mathrm{mod})\;\dfrac{\langle\{M_2p_2\}, [0,0], [0,0]\rangle}{\langle\{p_2\}, [0,\infty), [0,2]\rangle}}}
$$

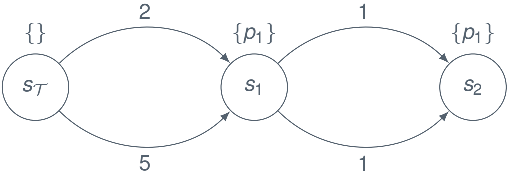


Image-finite counterexample

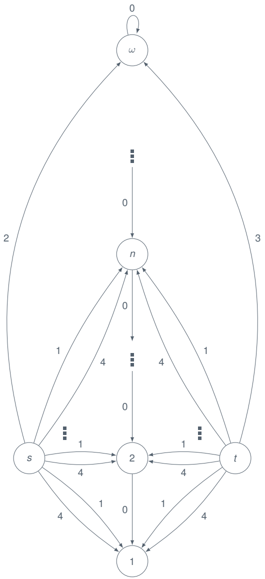


Figure 4: s and t satisfy the same logical formulas, but $s \not\sim t$.

Kantorovich counterexample

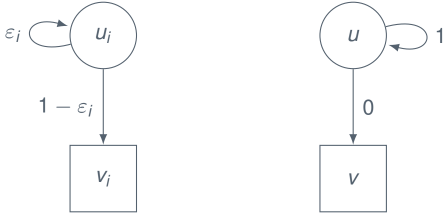
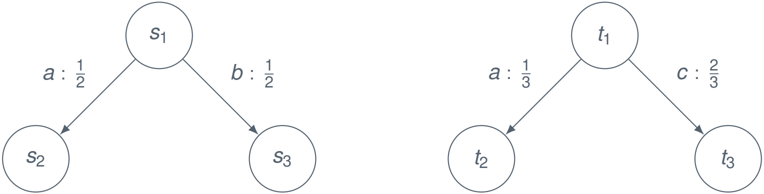


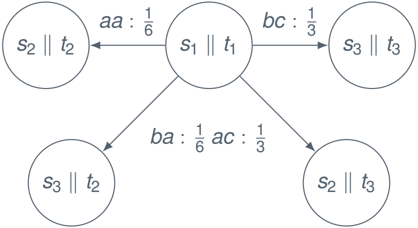
Figure 5: A Markov process with states u_i and v_i for each $i \in \mathbb{N}$.

New axioms

$$\{L_q\varphi \mid q < r\} \vdash L_r\varphi \quad \text{and} \quad \{M_q\varphi \mid q < r\} \vdash M_r\varphi$$

Generative composition – synchronous





Example from Ana Sokolova and Erik P. de Vink, *Probabilistic Automata: System Types, Parallel Composition and Comparison*, in Validation of Stochastic Systems - A Guide to Current Research, Lecture Notes in Computer Science volume 2925, pp. 1–43, 2004