Reachability in timed and probabilistic systems

B. Srivathsan

Chennai Mathematical Institute

FM Update, Delhi

July 2013

Reachability in TA

Reachability in PTA

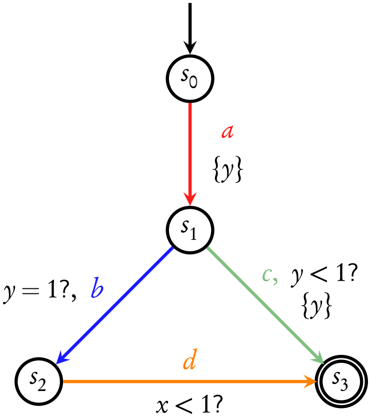
Conclusions

Reachability in TA

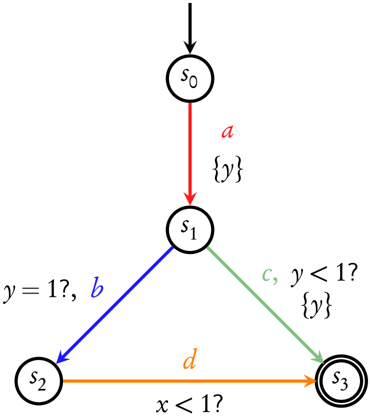
Reachability in PTA

Conclusions

Timed Automata [AD90]



Timed Automata [AD90]



Run:

$(s_0, 0, 0)$

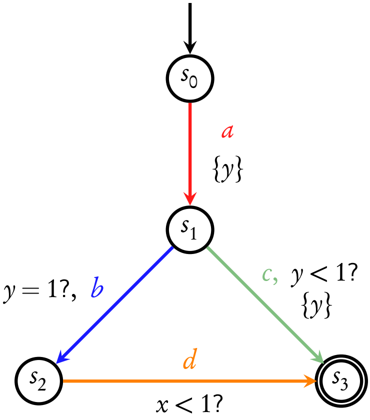
$\xrightarrow{0.4,\ a}$

$(s_1, 0.4, 0)$

$\xrightarrow{0.5,\ c}$

$(s_3, 0.9, 0.5)$

Timed Automata [AD90]



Run:

$(s_0, 0, 0)$

$\xrightarrow{0.4,\ a}$

$(s_1, 0.4, 0)$

$\xrightarrow{0.5,\ c}$

$(s_3, 0.9, 0.5)$

Reachability Problem

Does there **exist** a run to the final state?

Reachability

- PSPACE-complete [Alur, Dill '90]

Reachability

- PSPACE-complete [Alur, Dill '90]
- PSPACE-complete with **3 clocks** [Courcoubetis, Yannakakis '92]

Reachability

- PSPACE-complete [Alur, Dill '90]
- PSPACE-complete with **3 clocks** [Courcoubetis, Yannakakis '92]
- NLOGSPACE-complete with **1 clock** [Laroussinie, Markey, Schnoebelen '04]

Reachability

- PSPACE-complete [Alur, Dill '90]
- PSPACE-complete with **3 clocks** [Courcoubetis, Yannakakis '92]
- NLOGSPACE-complete with **1 clock** [Laroussinie, Markey, Schnoebelen '04]
- 2 clock TA $\leftarrow$ LOGSPACE $\rightarrow$ **Bounded 1-counter automata**

  [Hasse, Ouaknine, Worrell '12]

Reachability

- PSPACE-complete [Alur, Dill '90]
- PSPACE-complete with **3 clocks** [Courcoubetis, Yannakakis '92]
- NLOGSPACE-complete with **1 clock** [Laroussinie, Markey, Schnoebelen '04]
- 2 clock TA $\leftarrow$ LOGSPACE $\rightarrow$ **Bounded 1-counter automata** [Hasse, Ouaknine, Worrell '12]
- PSPACE-complete for bounded 1-counter automata [Fearnley, Jurdziǹski '13]

Reachability in 2-clock TA is PSPACE-complete

Tools

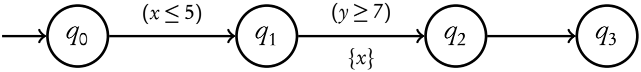
- UPPAAL [Aalborg (Denmark)]
- KRONOS [Verimag (France)]
- RED [Wang]

Tools

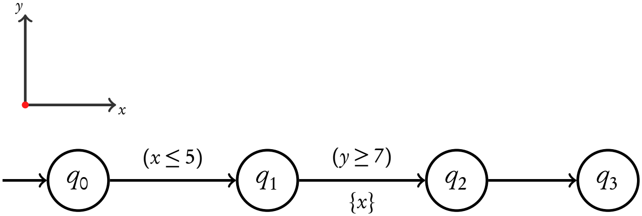
- UPPAAL [Aalborg (Denmark)]
- KRONOS [Verimag (France)]
- RED [Wang]

Coming next: zone-based approach of UPPAAL, KRONOS

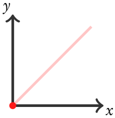
Forward analysis

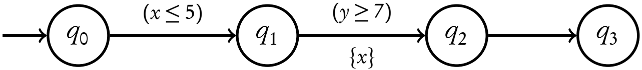


Forward analysis

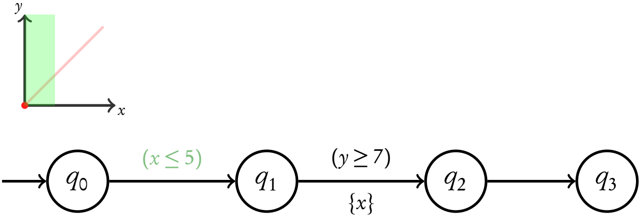


Forward analysis

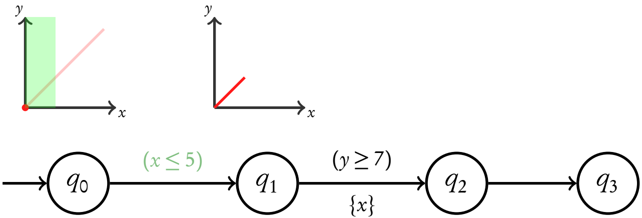




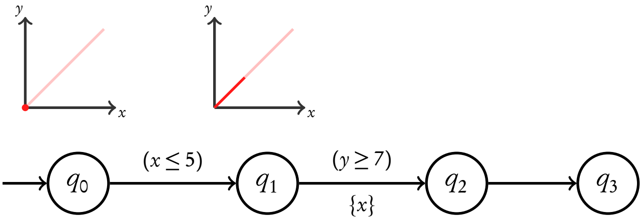
Forward analysis



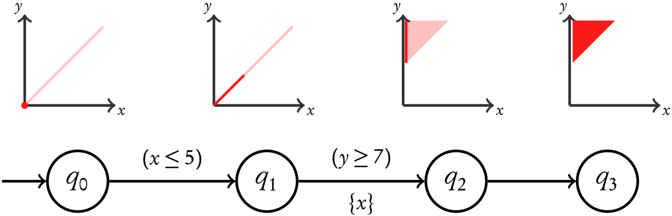
Forward analysis



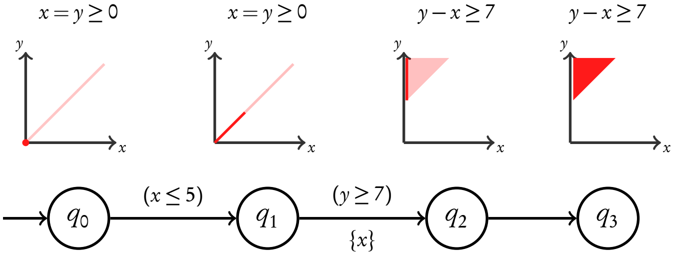
Forward analysis



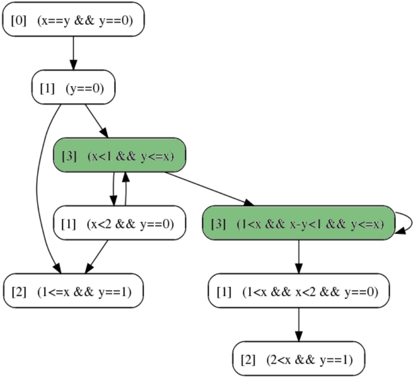
Forward analysis



Forward analysis



Zones and zone graph



- Zone: set of valuations defined by conjunctions of constraints:

$$x \sim c$$
$$x - y \sim c$$

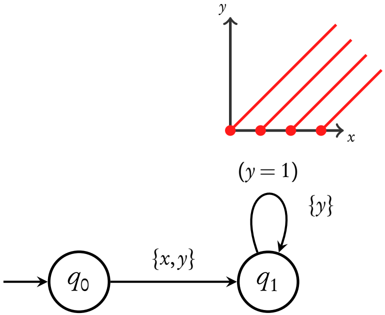
e.g. $(x - y \geq 1) \wedge (y < 2)$

- Representation: by DBM [Dil89]

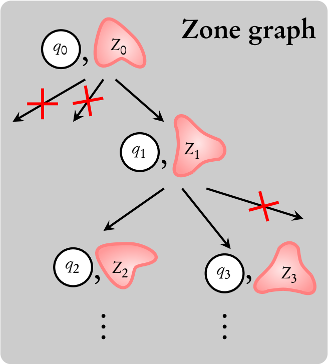
Sound and complete [DT98]

Zone graph preserves state **reachability**

Problem of non-termination

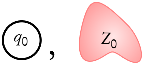


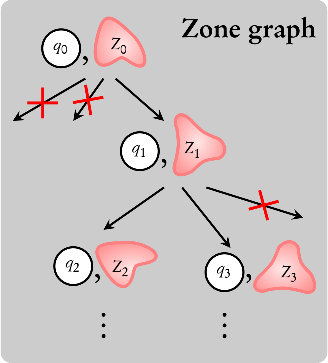
Abstractions



potentially infinite...

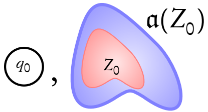
Abstractions

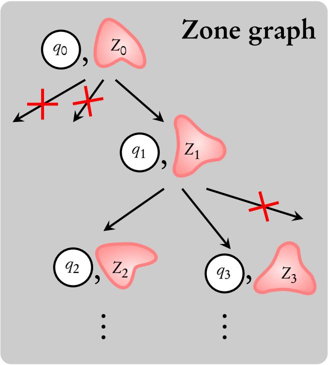




potentially infinite...

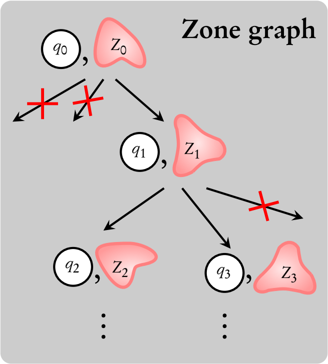
Abstractions



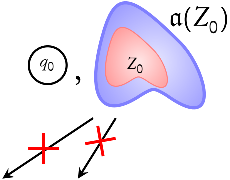


potentially infinite...

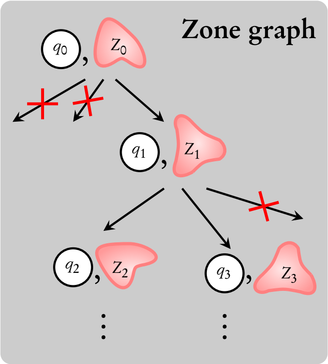
Abstractions



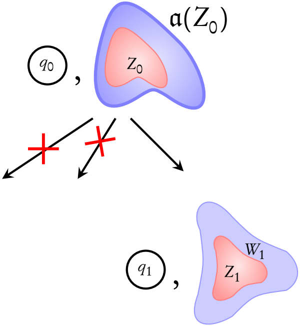
potentially infinite...



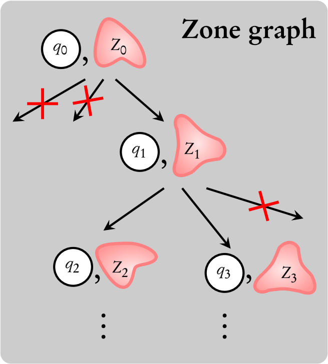
Abstractions



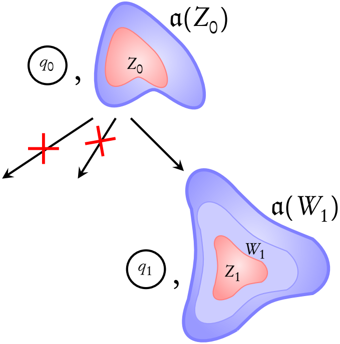
potentially infinite...



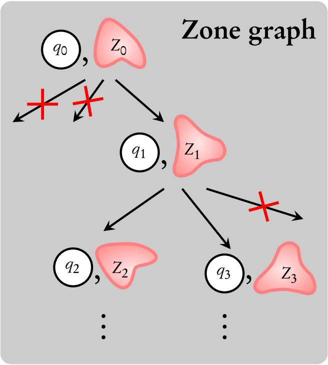
Abstractions



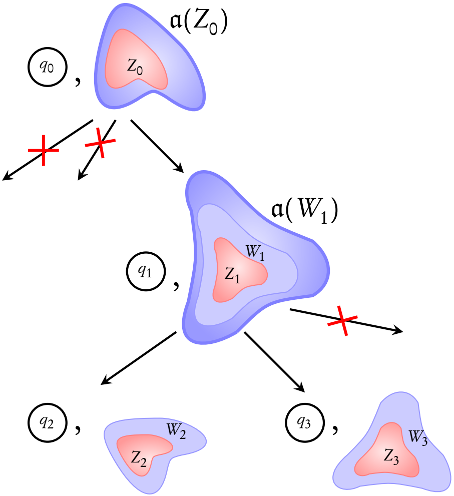
potentially infinite...



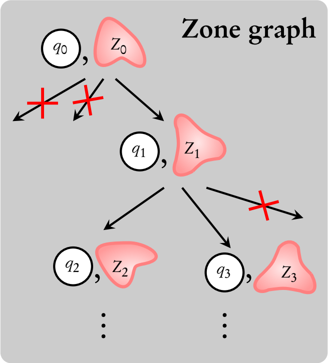
Abstractions



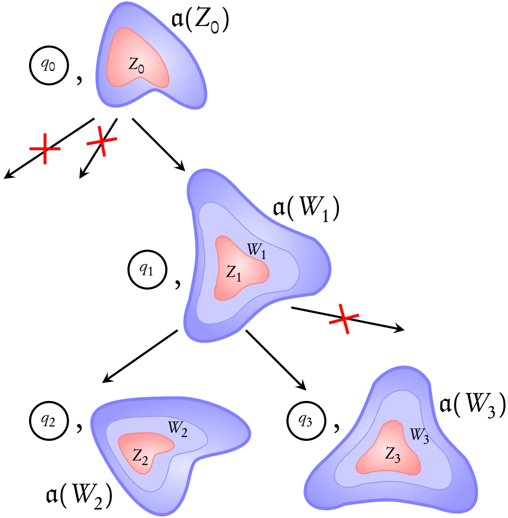
potentially infinite...



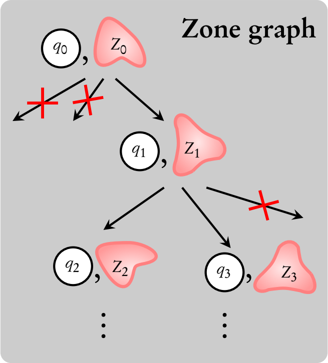
Abstractions



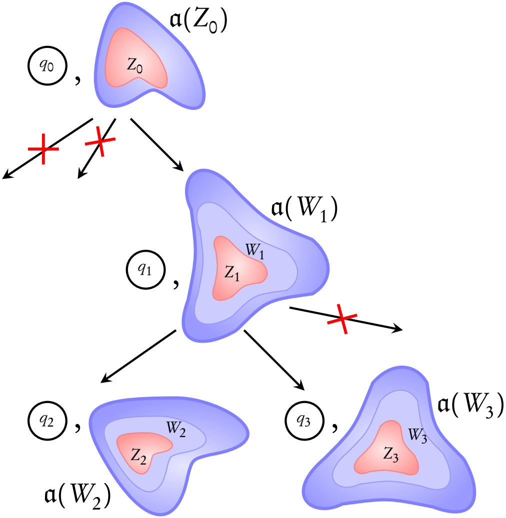
potentially infinite...



Abstractions

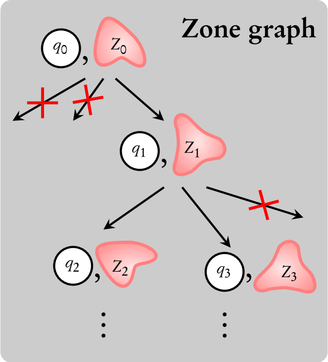


potentially infinite...

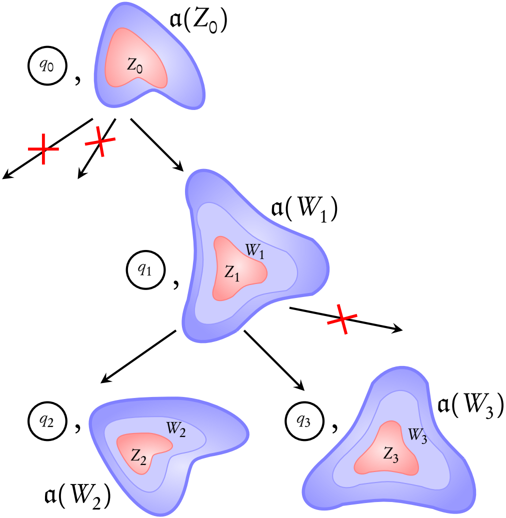


Find $\mathfrak{a}$ such that number of **abstracted** sets is **finite**

Abstractions



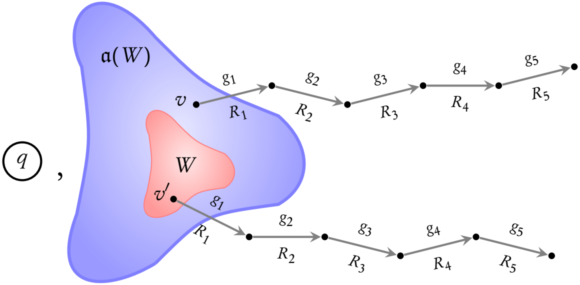
potentially infinite...



Coarser the abstraction, **smaller** the abstracted graph

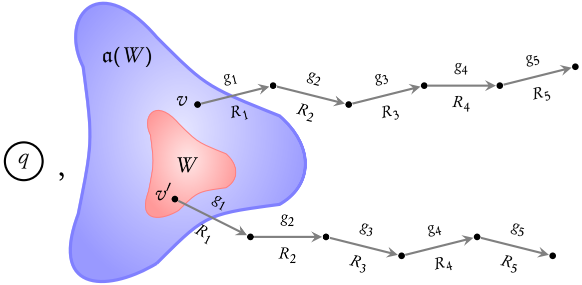
Condition 1: Abstractions should have **finite range**

Condition 2: Abstractions should be sound $\Rightarrow$ $\mathfrak{a}(W)$ can contain only valuations **simulated** by W



Condition 1: Abstractions should have **finite range**

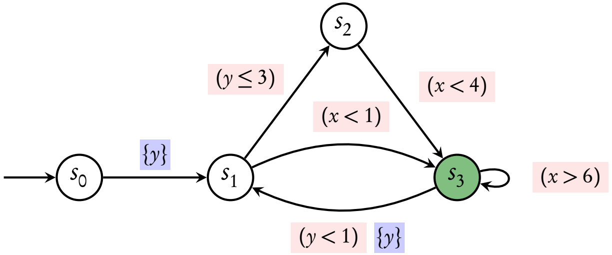
Condition 2: Abstractions should be sound $\Rightarrow \mathfrak{a}(W)$ can contain only valuations **simulated** by W



Question: Why not add **all** the valuations **simulated** by W?

Theorem [LS00]

Coarsest simulation relation is **EXPTIME-hard**



Theorem [LS00]

Coarsest simulation relation is **EXPTIME-hard**

$(y \leq 3)$

$(x < 4)$

$(x < 1)$

$(x > 6)$

$(y < 1)$

Theorem [LS00]

Coarsest simulation relation is **EXPTIME-hard**

$(y \leq 3)$

$(x < 4)$

$(x < 1)$

$(x > 6)$

$(y < 1)$

M-bounds [AD94]

$$M(x) = 6,\ M(y) = 3$$

$$v \preceq_M v'$$

Theorem [LS00]

Coarsest simulation relation is **EXPTIME-hard**

$(y \leq 3)$ $(x < 4)$

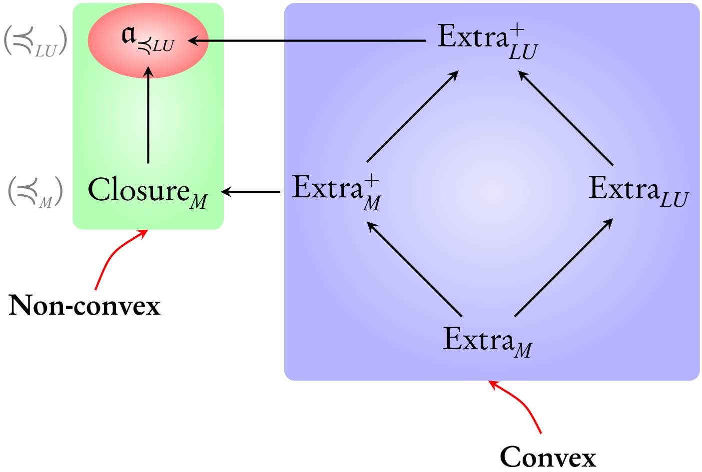
$(x < 1)$

$(x > 6)$

$(y < 1)$

M-bounds [AD94]	**LU-bounds** [BBLP06]
$M(x) = 6,\ M(y) = 3$	$L(x) = 6,\ L(y) = -\infty$ $U(x) = 4,\ U(y) = 3$
$v \preccurlyeq_M v'$	$v \preccurlyeq_{LU} v'$

Abstractions in literature [BBLP06, Bou04]



Recent results

- $\mathfrak{a}_{\preccurlyeq LU}$ can be **efficiently** used

  Given LU, $\mathfrak{a}_{\preccurlyeq LU}$ is **optimal**

  [Herbreteau, S., Walukiewicz '12]

- Better LU-bounds in a **lazy** way

  [Herbreteau, S., Walukiewicz '13]

- Multicore

  [Larsen et al. '12]

Reachability in TA

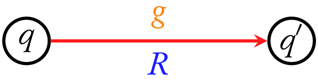
Reachability in PTA

Conclusions

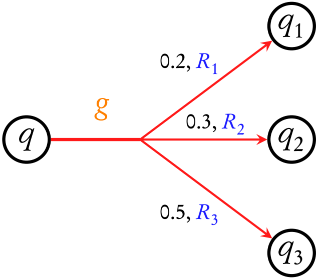
Transition:

$$q \xrightarrow[R]{g} q'$$

Transition:

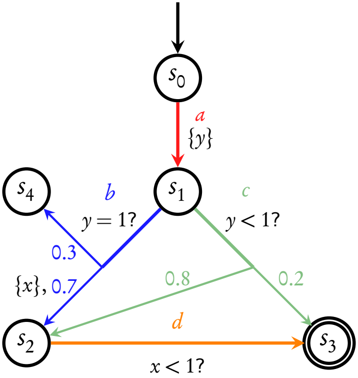


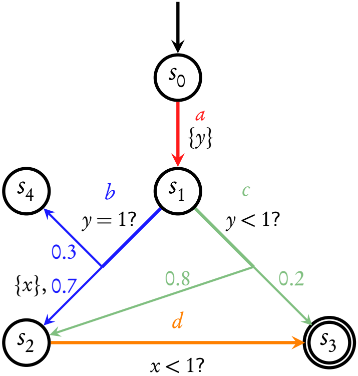
Probabilistic transition:



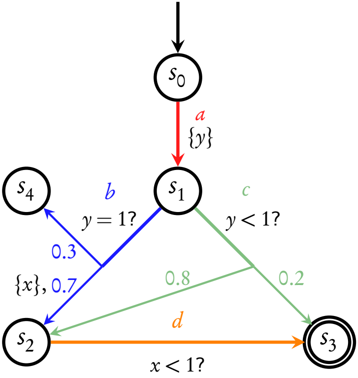
Probabilistic Timed Automata

[Jen96, KNSS02]

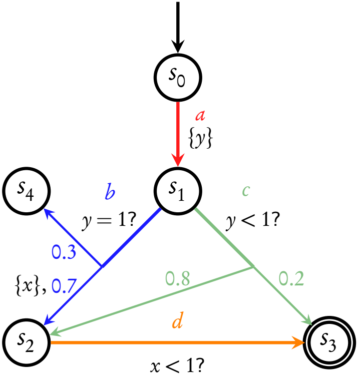




s_0
a
$\{y\}$
s_1
s_4
b
$y = 1?$
c
$y < 1?$
0.3
$\{x\}$, 0.7
0.8
0.2
d
s_2
s_3
$x < 1?$



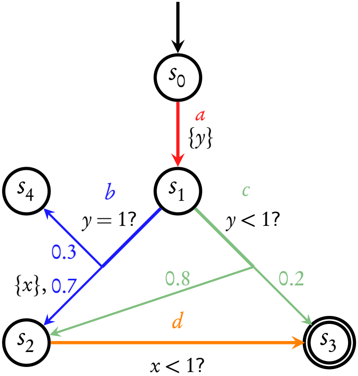
$n_0 : (s_0 , 0 , 0)$



Scheduler

$\sigma_1(n_0) = (0.4, a)$

$n_0 : (s_0 , 0 , 0)$



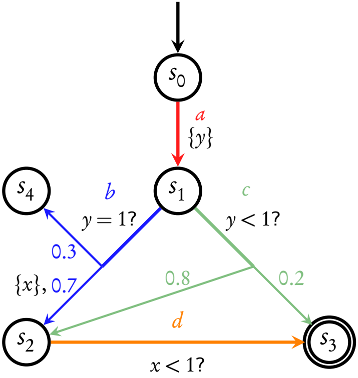
Scheduler

$\sigma_1(n_0) = (0.4, a)$

$n_0 : (\, s_0 \,,\, 0 \,,\, 0 \,)$

$\downarrow (0.4, a)$

$n_1 : (\, s_1 \,,\, 0.4 \,,\, 0 \,)$

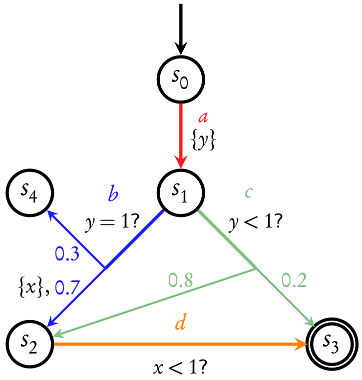


Scheduler

$$\sigma_1(n_0) = (0.4, a)$$

$$\sigma_1(n_0 n_1) = (1, b)$$

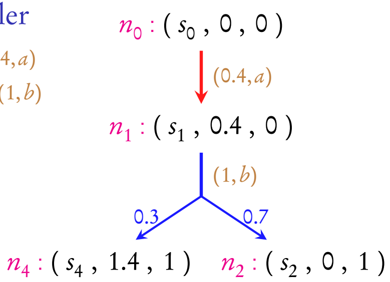
$$\begin{tikzcd} n_0 : (\, s_0 \,,\, 0 \,,\, 0 \,) \arrow[d, "{(0.4, a)}"] \\ n_1 : (\, s_1 \,,\, 0.4 \,,\, 0 \,) \end{tikzcd}$$

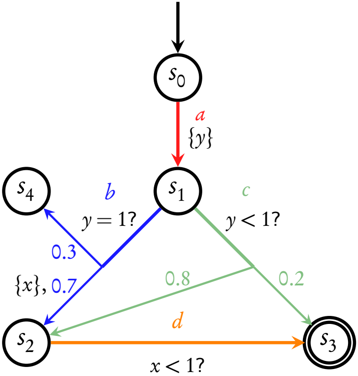


Scheduler

$\sigma_1(n_0) = (0.4, a)$

$\sigma_1(n_0 n_1) = (1, b)$



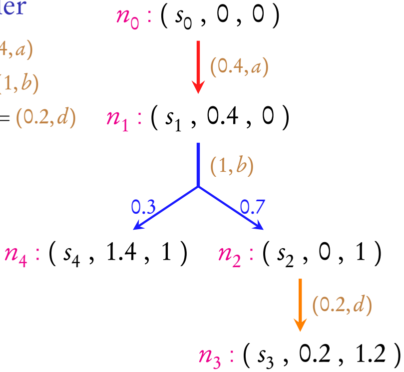


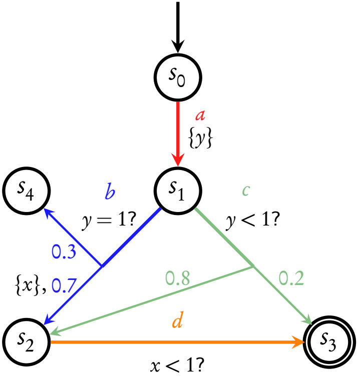
Scheduler

$\sigma_1(n_0) = (0.4, a)$

$\sigma_1(n_0 n_1) = (1, b)$

$\sigma_1(n_0 n_1 n_2) = (0.2, d)$





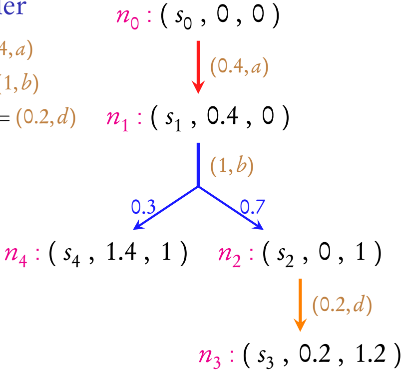
Scheduler

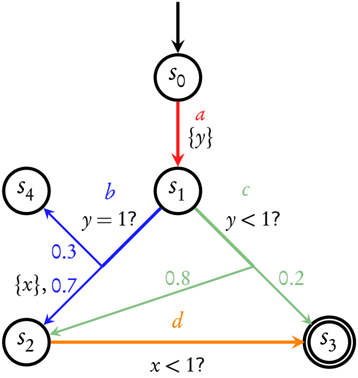
$\sigma_1(n_0) = (0.4, a)$

$\sigma_1(n_0 n_1) = (1, b)$

$\sigma_1(n_0 n_1 n_2) = (0.2, d)$

Markov chain





Scheduler

$\sigma_1(n_0) = (0.4, a)$

$\sigma_1(n_0 n_1) = (1, b)$

$\sigma_1(n_0 n_1 n_2) = (0.2, d)$

Markov chain

$n_0 : (\, s_0 \,,\, 0 \,,\, 0 \,)$

$\downarrow (0.4, a)$

$n_1 : (\, s_1 \,,\, 0.4 \,,\, 0 \,)$

$(1, b)$: with probability 0.3 to n_4, with probability 0.7 to n_2

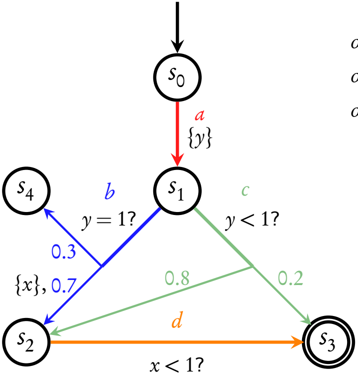
$n_4 : (\, s_4 \,,\, 1.4 \,,\, 1 \,)$ $\quad$ $n_2 : (\, s_2 \,,\, 0 \,,\, 1 \,)$

$\downarrow (0.2, d)$

$n_3 : (\, s_3 \,,\, 0.2 \,,\, 1.2 \,)$

Probability

$\mathbb{P}_{\sigma_1}(s_3) = 0.7$



$\sigma_2(n_0) = (0.4, a)$

$\sigma_2(n_0 n_1) = (0.5, c)$

$\sigma_2(n_0 n_1 n_2) = (0.05, d)$

$n_0 : (\, s_0 \,,\, 0 \,,\, 0 \,)$

$\xrightarrow{(0.4, a)}$

$n_1 : (\, s_1 \,,\, 0.4 \,,\, 0 \,)$

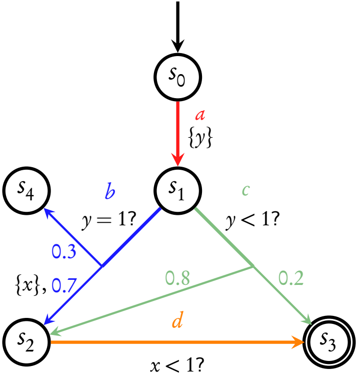
$(0.5, c)$: $0.8 \to n_2$, $0.2 \to n_{31}$

$n_2 : (\, s_2 \,,\, 0.9 \,,\, 0.5 \,)$ $\quad n_{31} : (\, s_3 \,,\, 0.9 \,,\, 0.5 \,)$

$\xrightarrow{(0.05, d)}$

$n_{32} : (\, s_3 \,,\, 0.95 \,,\, 0.55 \,)$

$$\mathbb{P}_{\sigma_2}(s_3) = 1$$



Reachability Problem for PTA

Given $\lambda \in [0, 1]$, does there **exist a scheduler** σ with

$$\mathbb{P}_\sigma(\text{final state}) \geq \lambda?$$

PTA reachability

- EXPTIME-complete

  [Kwiatkowska, Norman, Segala, Sproston '02]

  [Jurdziǹski, Sproston, Larrousinie '08]

PTA reachability

- EXPTIME-complete

  [Kwiatkowska, Norman, Segala, Sproston '02]

  [Jurdziǹski, Sproston, Larrousinie '08]

- EXPTIME-complete for **2-clocks**

  PTIME-complete for **1-clock**

  [Jurdziǹski, Sproston, Larrousinie '08]

Tools

- PRISM [Oxford]

Tools

- PRISM [Oxford]

Coming next: Reachability algorithm of PRISM

$$
\begin{tikzcd}
\text{PTA} \arrow[d] \\
\text{Zone MDP} \arrow[d] \\
\exists\ \text{an } \textbf{(untimed) scheduler} \text{ in zone MDP with probability } \geq \lambda\text{?}
\end{tikzcd}
$$

PTA

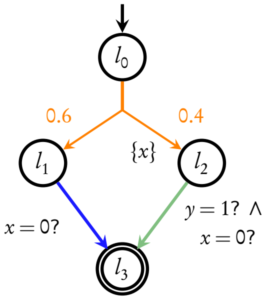
$\downarrow$

Zone MDP

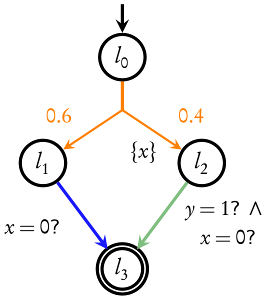
$\downarrow$

$\exists$ an **(untimed) scheduler** in zone MDP with probability $\geq \lambda$?

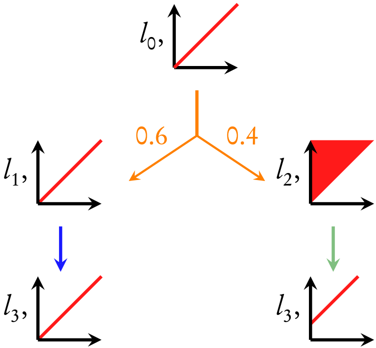
Coming next: Problem with forward analysis [KNSS02]

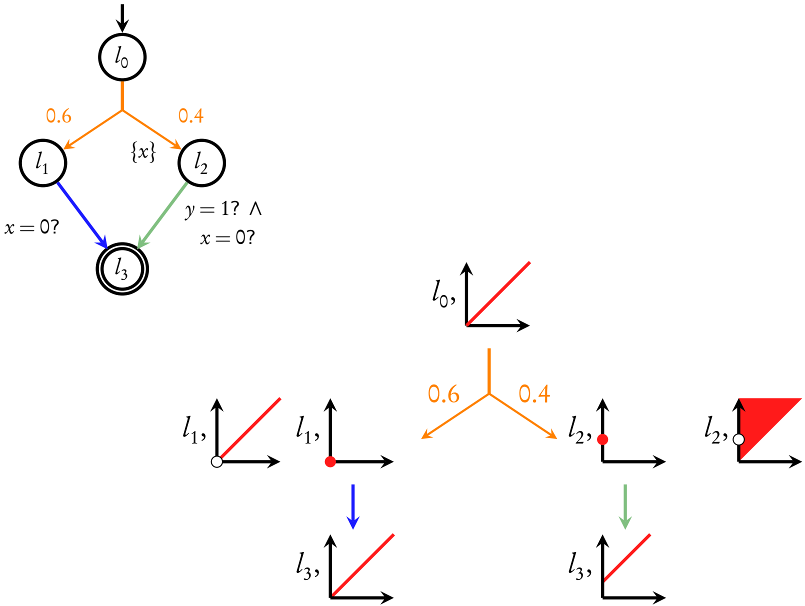


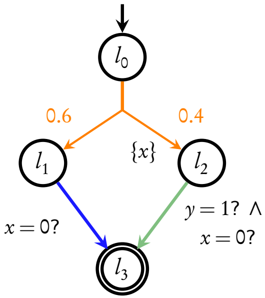
Max prob is 0.6



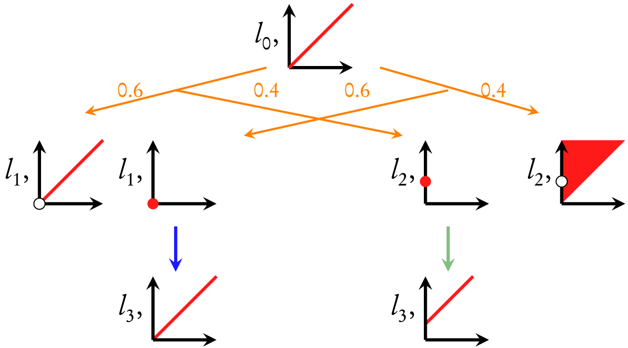
Zone MDP says 1




l_0
0.6
0.4
l_1
$\{x\}$
l_2
$y = 1? \wedge$
$x = 0?$
$x = 0?$
l_3
$l_0,$
0.6
0.4
$l_1,$
$l_1,$
$l_2,$
$l_2,$
$l_3,$
$l_3,$



Need an MDP like this



Two main approaches

- Game-based forward analysis

  [Kwiatkowska, Norman, Parker '09]

- Backward analysis

  [Kwiatkowska, Norman, Parker, Wang '07]

  [Berendsen, Jansen, Vaandrager]

Two main approaches

- Game-based forward analysis

  [Kwiatkowska, Norman, Parker '09]

- Backward analysis

  [Kwiatkowska, Norman, Parker, Wang '07]

  [Berendsen, Jansen, Vaandrager]

In many cases, **backward** performs better

Reachability in TA

Reachability in PTA

Conclusions

Summary

- Complexity:

  PSPACE-complete for TA, EXPTIME-complete for PTA

  (even for 2 clocks)

- Algorithm for TA:

  Forward analysis with abstraction

- Algorithm for PTA:

  Backward analysis, Game-based forward analysis

Perspectives

- Forward **versus** Backward
- Better abstractions for TA using **more semantics**
- Strategies for **order** of exploration
- Diagonal constraints
- Better methods/abstractions for PTA? Optimality?
- Rectangular hybrid automata

References I

Rajeev Alur and David L. Dill.
Automata for modeling real-time systems.
In *ICALP*, pages 322–335, 1990.

R. Alur and D.L. Dill.
A theory of timed automata.
Theoretical Computer Science, 126(2):183–235, 1994.

G. Behrmann, P. Bouyer, K. G. Larsen, and R. Pelanek.
Lower and upper bounds in zone-based abstractions of timed automata.
Int. Journal on Software Tools for Technology Transfer, 8(3):204–215, 2006.

P. Bouyer.
Forward analysis of updatable timed automata.
Form. Methods in Syst. Des., 24(3):281–320, 2004.

D. Dill.
Timing assumptions and verification of finite-state concurrent systems.
In *AVMFSS*, volume 407 of *LNCS*, pages 197–212. Springer, 1989.

C. Daws and S. Tripakis.
Model checking of real-time reachability properties using abstractions.
In *TACAS'98*, volume 1384 of *LNCS*, pages 313–329. Springer, 1998.

Henrik Ejersbo Jensen.
Model checking probabilistic real time systems.
In *Chalmers Institute of Technology*, pages 247–261, 1996.

References II

M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston.
Automatic verification of real-time systems with discrete probability distributions.
Theoretical Computer Science, 282(1):101–150, 2002.

François Laroussinie and Ph. Schnoebelen.
The state explosion problem from trace to bisimulation equivalence.
In *Proceedings of the Third International Conference on Foundations of Software Science and Computation Structures*, FOSSACS '00, pages 192–207. Springer-Verlag, 2000.