Simple Priced Timed Games are not that simple

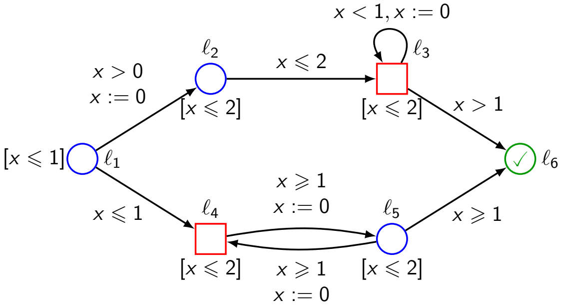
From FSTTCS 2015

Lefaucheux Engel
Inria Rennes, France

Thomas Brihaye (UMons), Gilles Geeraerts (ULB),
Axel Haddad (UMons), Benjamin Monmege (LIF Marseille)

September 7, 2016

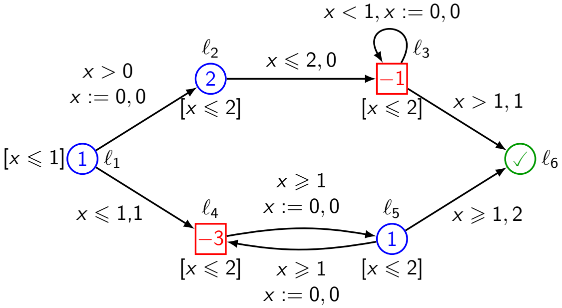
Priced Timed Games



Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

Priced Timed Games

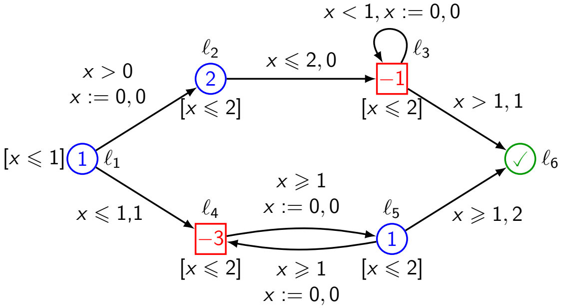


Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

$(\ell_1, 0)$

Priced Timed Games

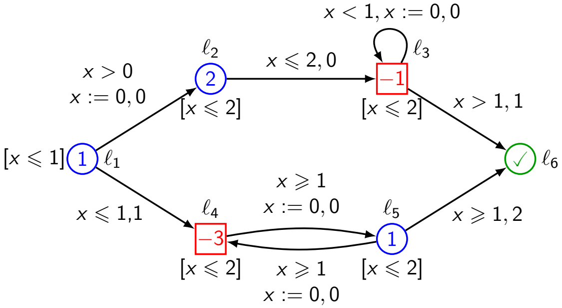


$(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4)$

Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

Priced Timed Games

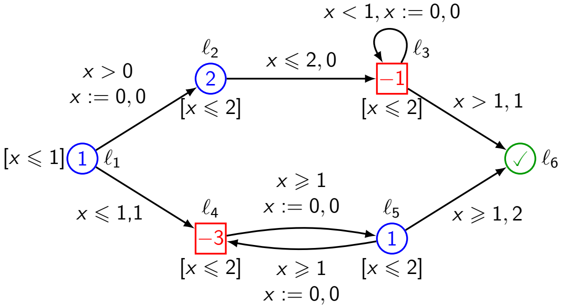


$(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4) \xrightarrow{0.6, \rightarrow} (\ell_5, 0)$

Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

Priced Timed Games

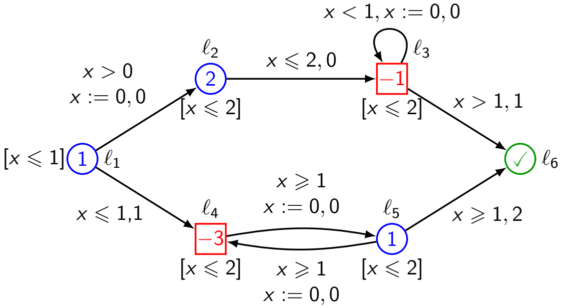


Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

$$(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4) \xrightarrow{0.6, \rightarrow} (\ell_5, 0) \xrightarrow{1.5, \leftarrow} (\ell_4, 0) \xrightarrow{1.1, \rightarrow} (\ell_5, 0) \xrightarrow{2, \nearrow} (\checkmark, 2)$$

Priced Timed Games



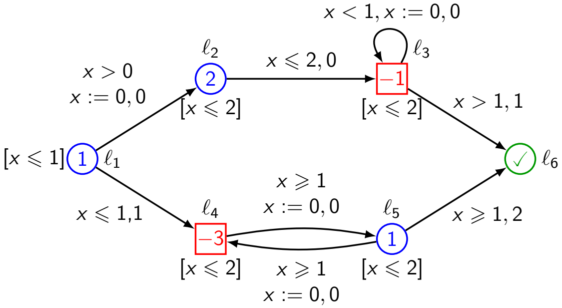
Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

$$(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4) \xrightarrow{0.6, \rightarrow} (\ell_5, 0) \xrightarrow{1.5, \leftarrow} (\ell_4, 0) \xrightarrow{1.1, \rightarrow} (\ell_5, 0) \xrightarrow{2, \nearrow} (\checkmark, 2)$$

$$0.4 + 1 \quad -3 \times 0.6 \quad +1.5 \quad -3 \times 1.1 \quad +2 \times 2 + 2 \quad = 3.8$$

Priced Timed Games



Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

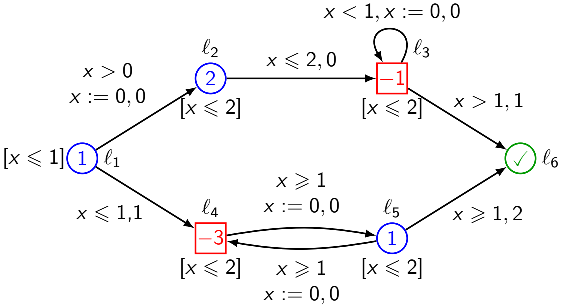
$$(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4) \xrightarrow{0.6, \rightarrow} (\ell_5, 0) \xrightarrow{1.5, \leftarrow} (\ell_4, 0) \xrightarrow{1.1, \rightarrow} (\ell_5, 0) \xrightarrow{2, \nearrow} (\checkmark, 2)$$

$$0.4 + 1 \quad -3 \times 0.6 \quad +1.5 \quad -3 \times 1.1 \quad +2 \times 2 + 2 \quad = 3.8$$

$$(\ell_1, 0) \xrightarrow{0.2, \nearrow} (\ell_2, 0) \xrightarrow{0.9, \rightarrow} (\ell_3, 0.9) \xrightarrow{0.2, \circlearrowleft} (\ell_3, 0) \xrightarrow{0.9, \circlearrowleft} (\ell_3, 0) \quad \cdots$$

$$0.2 \quad +0.9 \quad -0.2 \quad -0.9 \quad \cdots \quad = +\infty \ (\checkmark \text{ not reached})$$

Priced Timed Games



Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

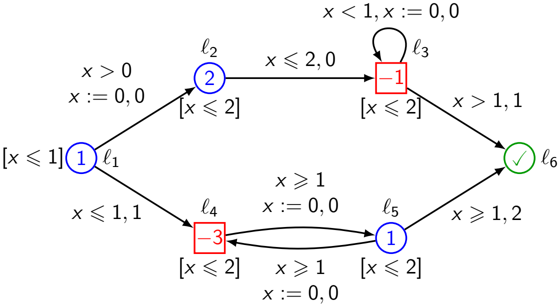
Semantics in terms of
infinite game with weights

$$(\ell_1, 0) \xrightarrow[0.4+1]{0.4, \searrow} (\ell_4, 0.4) \xrightarrow[-3 \times 0.6]{0.6, \rightarrow} (\ell_5, 0) \xrightarrow[+1.5]{1.5, \leftarrow} (\ell_4, 0) \xrightarrow[-3 \times 1.1]{1.1, \rightarrow} (\ell_5, 0) \xrightarrow[+2 \times 2 + 2]{2, \nearrow} (\checkmark, 2) \quad = 3.8$$

$$(\ell_1, 0) \xrightarrow[0.2]{0.2, \nearrow} (\ell_2, 0) \xrightarrow[+0.9]{0.9, \rightarrow} (\ell_3, 0.9) \xrightarrow[-0.2]{0.2, \circlearrowleft} (\ell_3, 0) \xrightarrow[-0.9]{0.9, \circlearrowleft} (\ell_3, 0) \quad \cdots \quad = +\infty \ (\checkmark \text{ not reached})$$

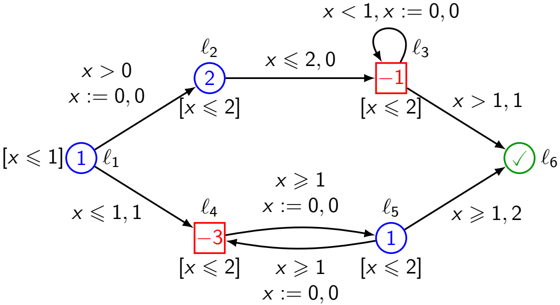
$$\text{Cost of a play: } \begin{cases} +\infty & \text{if } \checkmark \text{ not reached} \\ \text{total payoff up to } \checkmark & \text{otherwise} \end{cases}$$

Strategies and objectives



Strategy for each player: mapping of finite runs to a delay and an action

Strategies and objectives

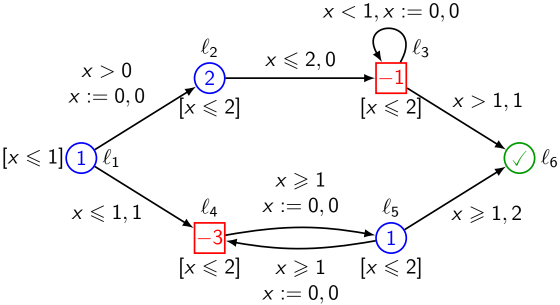


Strategy for each player: mapping of finite runs to a delay and an action

Goal of player ○: reach ✓ **and** minimize the cost

Goal of player □: avoid ✓ **or, if not possible,** maximize the cost

Strategies and objectives



Strategy for each player: mapping of finite runs to a delay and an action

Goal of player ○: reach ✓ **and** minimize the cost
Goal of player □: avoid ✓ **or, if not possible,** maximize the cost

Main object of interest:
$\overline{\mathsf{Val}}(\ell, v)$ = minimal cost player ○ can guarantee
$\underline{\mathsf{Val}}(\ell, v)$ = maximal cost player □ can guarantee
What can players guarantee as a payoff? And design *good* strategies.

State of the art

$\mathsf{F}_{\leqslant K}\checkmark$: $\exists$ a strategy in the PTG (priced timed game) for player $\bigcirc$ reaching $\checkmark$ with a cost $\leqslant K$?

State of the art

$\mathsf{F}_{\leqslant K}\checkmark$: $\exists$ a strategy in the PTG (priced timed game) for player ○ reaching ✓ with a cost $\leqslant K$?

- 2-player PTGs: undecidable [Brihaye, Bruyère, and Raskin, 2005, Bouyer, Brihaye, and Markey, 2006a], even with only non-negative costs and 3 clocks
- PTGs with non-negative costs and strictly non-Zeno cost cycles or with one clock: exponential algorithm [Bouyer, Cassez, Fleury, and Larsen, 2004, Alur, Bernadsky, and Madhusudan, 2004, Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011, Hansen, Ibsen-Jensen, and Miltersen, 2013]

State of the art

$F_{\leqslant K}\checkmark$: $\exists$ a strategy in the PTG (priced timed game) for player $\bigcirc$ reaching $\checkmark$ with a cost $\leqslant K$?

- 2-player PTGs: undecidable [Brihaye, Bruyère, and Raskin, 2005, Bouyer, Brihaye, and Markey, 2006a], even with only non-negative costs and 3 clocks
- PTGs with non-negative costs and strictly non-Zeno cost cycles or with one clock: exponential algorithm [Bouyer, Cassez, Fleury, and Larsen, 2004, Alur, Bernadsky, and Madhusudan, 2004, Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011, Hansen, Ibsen-Jensen, and Miltersen, 2013]

More complex with negative costs.

State of the art

$\mathsf{F}_{\leqslant K}$✓: $\exists$ a strategy in the PTG (priced timed game) for player ○ reaching ✓ with a cost $\leqslant K$?

- 2-player PTGs: undecidable [Brihaye, Bruyère, and Raskin, 2005, Bouyer, Brihaye, and Markey, 2006a], even with only non-negative costs and 3 clocks
- PTGs with non-negative costs and strictly non-Zeno cost cycles or with one clock: exponential algorithm [Bouyer, Cassez, Fleury, and Larsen, 2004, Alur, Bernadsky, and Madhusudan, 2004, Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011, Hansen, Ibsen-Jensen, and Miltersen, 2013]

More complex with negative costs.

- $\mathsf{F}_{\leqslant K}$✓ undecidable for 2 or more clocks and pseudo-polynomial algorithm for One-clock Bi-valued PTG [Brihaye, Geeraerts, Krishna, Manasa, Monmege, and Trivedi, 2014]

One-clock Bi-valued PTG: important restriction on the allowed rates of locations

Inspired by other previous techniques for 1-clock PTGs?

[Hansen, Ibsen-Jensen, and Miltersen, 2013]: strategy improvement algorithm

[Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]: iterative elimination of locations

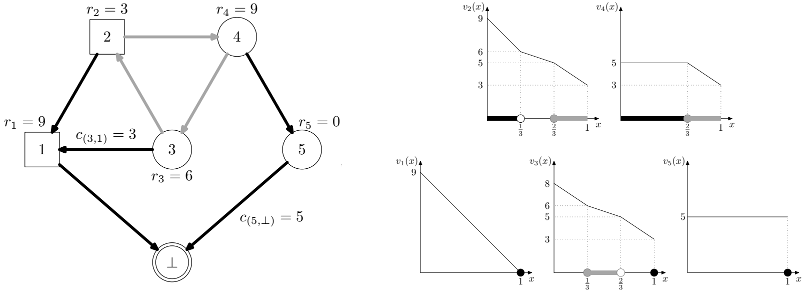
- precomputation: polynomial-time cascade of simplification of 1-clock PTGs into simple 1-clock PTGs (SPTGs)
  - clock bounded by 1, no guards/invariants, no resets

Inspired by other previous techniques for 1-clock PTGs?

[Hansen, Ibsen-Jensen, and Miltersen, 2013]: strategy improvement algorithm

[Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]: iterative elimination of locations

- precomputation: polynomial-time cascade of simplification of 1-clock PTGs into simple 1-clock PTGs (SPTGs)
  - clock bounded by 1, no guards/invariants, no resets
- for SPTGs: compute value functions $\overline{\mathsf{Val}}(\ell, x)$.



Recursive elimination of states

Intuition from [Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]:

- Player ○ prefers to stay as long as possible in locations with **minimal price**: add a final location allowing him to stay until the end, and make the location urgent

Recursive elimination of states

Intuition from [Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]:

- Player ○ prefers to stay as long as possible in locations with **minimal price**: add a final location allowing him to stay until the end, and make the location urgent
- Player □ prefers to leave as soon as possible in locations with **minimal price**: make the location urgent

Recursive elimination of states

Intuition from [Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]:

- Player ○ prefers to stay as long as possible in locations with **minimal price**: add a final location allowing him to stay until the end, and make the location urgent
- Player □ prefers to leave as soon as possible in locations with **minimal price**: make the location urgent

Problem: intuition not always true... you may have to change decision!
Recursive algorithm + construction of the value functions from right ($x = 1$) to left ($x = 0$)

Recursive elimination of states

Intuition from [Bouyer, Larsen, Markey, and Rasmussen, 2006b, Rutkowski, 2011]:

- Player ○ prefers to stay as long as possible in locations with **minimal price**: add a final location allowing him to stay until the end, and make the location urgent
- Player □ prefers to leave as soon as possible in locations with **minimal price**: make the location urgent

Problem: intuition not always true... you may have to change decision!
Recursive algorithm + construction of the value functions from right ($x = 1$) to left ($x = 0$)

Challenges with arbitrary weights:

- Proof of correctness does not generalise: initially two distinct proofs for ○ and □
- Proof of termination does not generalise: difficult because of the double recursion...

Make a symmetric treatment of ○ and □

Theorem

PTGs are determined ($\overline{\mathsf{Val}} = \underline{\mathsf{Val}}$), and value functions are continuous (over regions).

Determinacy follows from Gale-Stewart determinacy result.

Make a symmetric treatment of ○ and □

Theorem

PTGs are determined ($\overline{\mathsf{Val}} = \underline{\mathsf{Val}}$), and value functions are continuous (over regions).

Determinacy follows from Gale-Stewart determinacy result.

Advantage: both players are dual...

Make a symmetric treatment of ○ and □

Theorem

PTGs are determined ($\overline{\mathsf{Val}} = \underline{\mathsf{Val}}$), and value functions are continuous (over regions).

Determinacy follows from Gale-Stewart determinacy result.

Advantage: both players are dual...

Theorem

For every SPTG, all value functions are piecewise affine, with at most an exponential number of cutpoints (in number of locations).

Make a symmetric treatment of ○ and □

Theorem

PTGs are determined ($\overline{\mathsf{Val}} = \underline{\mathsf{Val}}$), and value functions are continuous (over regions).

Determinacy follows from Gale-Stewart determinacy result.

Advantage: both players are dual...

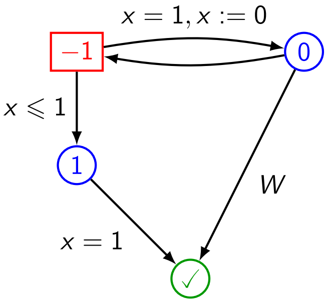
Theorem

For every SPTG, all value functions are piecewise affine, with at most an exponential number of cutpoints (in number of locations).

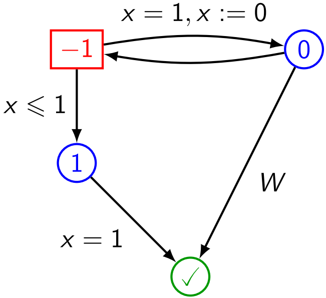
For general 1-clock PTGs?

- removing guards and invariants: previously used techniques work!
- removing resets: previously, bound the number of resets...

Bounding the number of resets needed is not possible

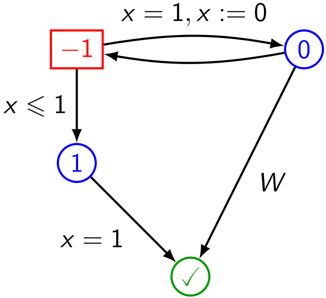


Bounding the number of resets needed is not possible



Player ○ can guarantee (i.e., ensure to be below) value ε for all $\varepsilon > 0$...

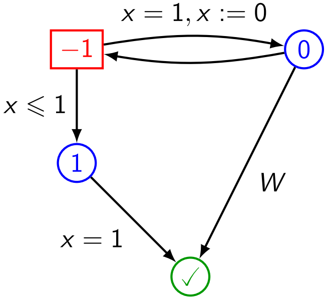
Bounding the number of resets needed is not possible



Player ○ can guarantee (i.e., ensure to be below) value ε for all $\varepsilon > 0$...

... but cannot obtain 0: hence, no optimal strategy...

Bounding the number of resets needed is not possible



Player ○ can guarantee (i.e., ensure to be below) value ε for all $\varepsilon > 0$...

... but cannot obtain 0: hence, no optimal strategy...

... moreover, to obtain ε, ○ needs to loop at least $W + \lceil 1/\log \varepsilon \rceil$ times before reaching ✓!

Current solution: Reset-acyclic 1-clock PTGs

exponential time algorithm for reset-acyclic 1-clock PTGs with arbitrary weights

Summary and Future Work

Results

- Extension of iterative elimination for reset-acyclic 1-clock PTGs with arbitrary weights
- Study of the value function: determination, upper and lower bound, number of cutpoints...

Summary and Future Work

Results

- Extension of iterative elimination for reset-acyclic 1-clock PTGs with arbitrary weights
- Study of the value function: determination, upper and lower bound, number of cutpoints. . .

- Future work: final extension of the result for all 1-clock PTGs?
- Use the result for 1-clock to approximate/compute the value of general PTGs with adequate structural properties
- Implementation and test of different algorithms on real instances

Summary and Future Work

Results

- Extension of iterative elimination for reset-acyclic 1-clock PTGs with arbitrary weights
- Study of the value function: determination, upper and lower bound, number of cutpoints...

- Future work: final extension of the result for all 1-clock PTGs?
- Use the result for 1-clock to approximate/compute the value of general PTGs with adequate structural properties
- Implementation and test of different algorithms on real instances

Thank you for your attention

References I

Rajeev Alur, Mikhail Bernadsky, and P. Madhusudan. Optimal reachability for weighted timed games. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP'04)*, volume 3142 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2004.

Patricia Bouyer, Franck Cassez, Emmanuel Fleury, and Kim G. Larsen. Optimal strategies in priced timed game automata. In *Proceedings of the 24th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'04)*, volume 3328 of *Lecture Notes in Computer Science*, pages 148–160. Springer, 2004.

Patricia Bouyer, Thomas Brihaye, and Nicolas Markey. Improved undecidability results on weighted timed automata. *Information Processing Letters*, 98(5):188–194, 2006a.

Patricia Bouyer, Kim G. Larsen, Nicolas Markey, and Jacob Illum Rasmussen. Almost optimal strategies in one-clock priced timed games. In *Proceedings of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)*, volume 4337 of *Lecture Notes in Computer Science*, pages 345–356. Springer, 2006b.

Thomas Brihaye, Véronique Bruyère, and Jean-François Raskin. On optimal timed strategies. In *Proceedings of the Third international conference on Formal Modeling and Analysis of Timed Systems (FORMATS'05)*, volume 3829 of *Lecture Notes in Computer Science*, pages 49–64. Springer, 2005.

References II

Thomas Brihaye, Gilles Geeraerts, Shankara Narayanan Krishna, Lakshmi Manasa, Benjamin Monmege, and Ashutosh Trivedi. Adding Negative Prices to Priced Timed Games. In *Proceedings of the 25th International Conference on Concurrency Theory (CONCUR'13)*, volume 8704 of *Lecture Notes in Computer Science*, pages 560–575. Springer, 2014.

Thomas Dueholm Hansen, Rasmus Ibsen-Jensen, and Peter Bro Miltersen. A faster algorithm for solving one-clock priced timed games. In *Proceedings of the 24th International Conference on Concurrency Theory (CONCUR'13)*, volume 8052 of *Lecture Notes in Computer Science*, pages 531–545. Springer, 2013.

Michał Rutkowski. Two-player reachability-price games on single-clock timed automata. In *Proceedings of the Ninth Workshop on Quantitative Aspects of Programming Languages (QAPL'11)*, volume 57 of *Electronic Proceedings in Theoretical Computer Science*, pages 31–46, 2011.