Priced Timed Automata: Theory & Tools

Kim G. Larsen*
Department of Computer Science, Aalborg University
Denmark
kgl@cs.aau.dk

ABSTRACT. Priced timed automata are emerging as useful formalisms for modeling and analysing a broad range of resource allocation problems. In this extended abstract, we highlight recent (un)decidability results related to priced timed automata as well as point to a number of open problems.

1 Introduction

The model of timed automata, introduced by Alur and Dill [2, 3], has by now established itself as a classical formalism for describing the behaviour of real-time systems. A number of important properties has been shown decidable, including reachability, model checking and several behavioural equivalences and preorders.

By now, real-time model checking tools such as UPPAAL [11, 39] and KRONOS [30] are based on the timed automata formalism and on the substantial body of research on this model that has been targeted towards transforming the early results into practically efficient algorithms — *e.g.* [9, 15, 8, 13] — and data structures — *e.g.*[38, 37, 14, 14].

More recently, model-checking tools in general and UPPAAL in particular have been applied to solve realistic scheduling problems by a reformulation as reachability problems — *e.g.* [34, 35, 1, 41]. Aiming at *optimal* scheduling, *priced timed automata* [12, 5] are emerging as a useful formalism for formulating and solving a broad range of resource allocation problems of importance in applications areas such as, *e.g.*, embedded systems.

2 Optimal Reachability and Optimal Safety

Within the model of priced timed automata, the cost variables serve purely as *evaluation functions* or *observers*, *i.e.*, the behaviour of the underlying timed automatoa may in no way depend on the cost variables. As a consequence of this restriction – and in contrast to the related models of constant slope and linear hybrid automata – a number of optimization problems have been shown decidable for priced timed automata including minimun-cost reachability [12, 4, 20], optimal (minimum and maximum cost) reachability in multi-priced settings [40].

EXAMPLE 1. *Consider the timed automaton of Fig. 1(a) with two clocks x and y, and label set $\{a, b, c, d, e\}$. Note that no time can elapse in the middle location due to the invariant $(y = 0)$. The a, d and e transitions have guards $x \leq 2$ and $x = 2$ respectively. It is clear that no matter which*

*Partially funded by the VKR Center of Excellence, MT-LAB.

Foundations of Software Technology and Theoretical Computer Science (Kanpur) 2009.
Editors: Ravi Kannan and K. Narayan Kumar; pp 417–425
Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.
Digital Object Identifier: 10.4230/LIPIcs.FSTTCS.2009.2337

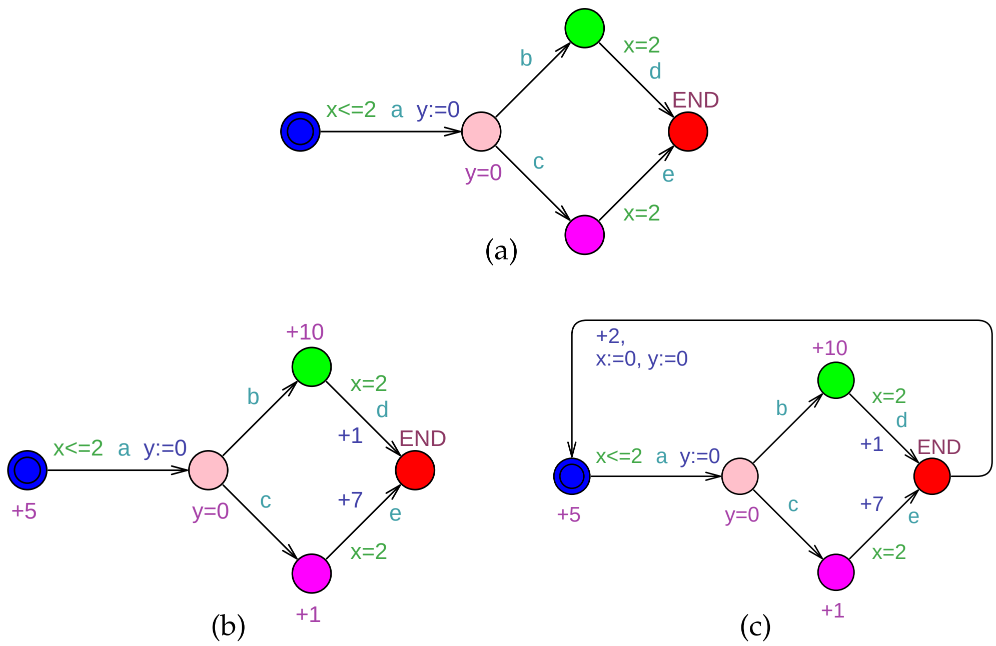


Figure 1: (a) A Small Timed Automaton. (b) A Small Priced Timed Automaton. (c) A Small Cyclic Priced Timed Automaton.

execution – differing in the delay in initial location and whether to choose the b or c transition – the minimal time for reaching the END *location is* 2.

Now consider the priced timed automaton of Fig. 1(b). Here the decoration of $+10$ *on a location indicates that cost increases by* 10 *per time unit in the location; a decoration* -7 *on a transition indicates that taking the transition increases overall cost by* 7*. Let us compute the minimum cost that is required for reacing location* END*. There are two families of executions: those that follow the b edge and those that follow the c edge. Furthermore, in each family, there is a single parameter* $t \leq 2$ *being the time elapsed in the initial location before the a edge is taken. Hence the minimum cost is:*

$$\inf_{0 \leq t \leq 2} \min \begin{pmatrix} 5t + 10(2-t) + 1 \\ 5t + (2-t) + 7 \end{pmatrix} = 9$$

where $5t + 10(2-t) + 1$ *and* $5t + (2-t) + 7$ *give the cost of executions following the b respectively the c edge.*

Dually, computability of cost-optimal infinite schedules have been established covering optimal infinite schedules in terms of minimal (or maximal) *cost per time ratio* in the limit have been obtained in [21, 22] and optimal infinite schedules in terms of minimal (or maximal) *discounted total cost* [33].

EXAMPLE 2. *Now reconsider the cyclic priced timed automaton of Fig. 1(c). Due to the simplicity of the cycle (both x and y are reset at the looping transition), the optimal schedule in terms of minimal cost per time unit has value* $(9+2)/2 = 5.5$.

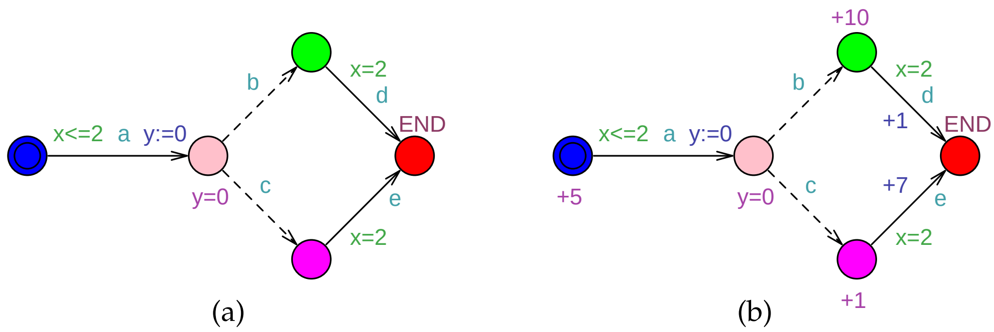


Figure 2: (a) A Small Timed Game. (b) A Small Priced Timed Game.

In terms of tool support UPPAAL Cora [36, 16, 17, 42] provides an efficient method for computing cost-optimal or near-optimal solutions to reachability questions, implementing a symbolic A^* algorithm based on a new data strucutre (so-called priced zones) allowing for efficient symbolic state-representation with additional cost-information.

3 Model-Checking

Cost-extended versions of temporal logics such as CTL (branching-time) and LTL (linear-time) appear as a natural "generalizations" of the above optimization problems. Just as TCTL and MTL provide extensions of CTL and LTL with time-constrained modalities, WCTL and WMTL are extensions with cost-constrained modalities interpreted with respect to priced timed automata. Unfortunately, the addition of cost now turns out to come with a price: whereas the model-checking problems for timed automata with respect to TCTL and MTL are decidable, it has been shown in [31] that model-checking with respect to WCTL is undecidable for priced timed automata with three clocks or more. In contrast [27, 28] shows that model checking with respect to WCTL is decidable under the single clock assumption. Decidability of WCTL for priced timed automata with two clocks is still an open (and hard) problem.

EXAMPLE 3. *Reconsider the priced timed automaton of Fig. 1(b) (with $x \leq 2$ added as an invariant to the initial location). Then the properties that i) there is a run leading to the location* END *with cost no more than* 9 *and that ii) all runs will lead to* END *within cost* 17 *may be expressed as the WCTL formula* $\mathsf{EF}_{c\leq 9}\mathsf{END}$ *and* $\mathsf{AF}_{c\leq 17}\mathsf{END}$*, respectively.*

4 Priced Timed Games

The models we have considered so far are *closed* in the sense that all transitions under control of the the user of the system. This is not sufficient to model embedded systems, where interaction with an uncontrollable environment is crucial, or systems with some imprecisions. These can be modelled using (two-player) *timed games* [7], in which some actions are triggered by the environment. The aim is to *control*, or *guide*, the system so that it is guaranteed to be safe or correct regardless of the way the environment interferes. An example of a timed game is depicted in Fig. 2. Here, a *winning strategy* clearly exists achieving the objective of reaching location END. Such winning reachability strategies, as well as time-optimal strategies [6], may be computed using efficient zone-based algorithms. Tool support is now available in UPPAAL Tiga [10] applying a symbolic on-the-fly algorithm.

It is natural to extend the timed game framework with cost information, hence making it possible to model uncertainty as well as consumption of resource, and to ask for strategies which obtain a stated objective in an optimal manner given the particular cost decoration. The model of *priced timed games* is the obvious combination of timed games and priced timed automata.

EXAMPLE 4. *Consider the example of the priced timed game of Fig. 2. Now we may want to compute the minimal cost for reaching the final location* END *regardless of whether the environment chooses to take the b or the c edge. As the system cannot control this choice, the minimum cost is given by the formula:*

$$\inf_{0 \leq t \leq 2} \max \begin{pmatrix} 5t + 10(2-t) + 1 \\ 5t + (2-t) + 7 \end{pmatrix} = 14.33$$

As for model checking priced timed automata, optimal winning strategies for priced timed games have proved much more difficult than simple optimal reachability and safety. In particular in [32] it has been shown that the problem of determining cost-optimal winning reachability strategies for priced timed games is not computable. In [19] it has been shown that these negative results hold even for priced timed (game) automata with no more than three clocks.

Decidability has been shown for classes of priced timed games with strong zone-like conditions on the evolution of cost [23, 24] and for one-clock priced timed games [29]. Again the case of two clocks is as yet unsettled.

5 Energy-Games

In [26] we began the study of a new class of resource scheduling problems, namely that of constructing infinite schedules or strategies subject to boundary constraints on the accumulation of resources, so-called *energy-games* or *energy-schedules*.

More specifically, we consider priced timed automata with *positive* as well as *negative* price-rates. This extension allows for the modelling of systems where resources are not only consumed but also occasionally produced or regained. In [26] three infinite scheduling problems was considered: *lower-bound* where the energy level never must go below zero, *interval-bound* where energy level must be maintained within a given interval, and *weak* upper bound, which does not prevent energy-increasing behaviour from proceeding once the upper bound is reached but merely maintains the energy level at the upper bound.

For one-clock priced timed automata both the lower-bound and the lower-weak-upper-bound problems are shown decidable (in polynomial time) [26], whereas the interval-bound problem is proved to be undecidable in a game setting. Decidability of the interval-bound

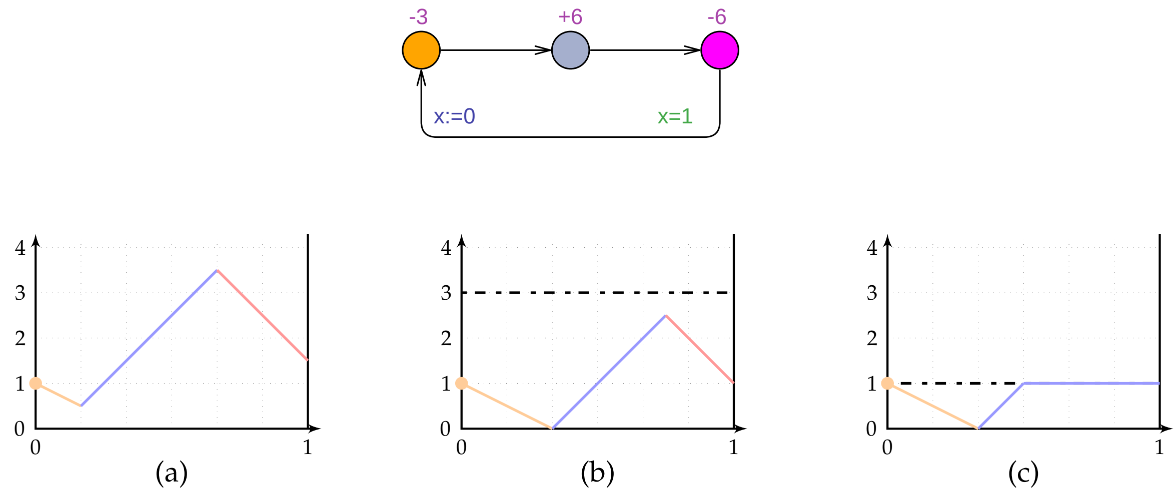


Figure 3: One-clock priced timed automaton and three types of infinite schedules: lower-bound (a), lower-upper-bound (b) and lower-weak-upper-bound (c).

problem for one-clock priced timed automata as well as decidability of all of the considered scheduling problems for priced timed automata with two or more clocks are still unsettled.

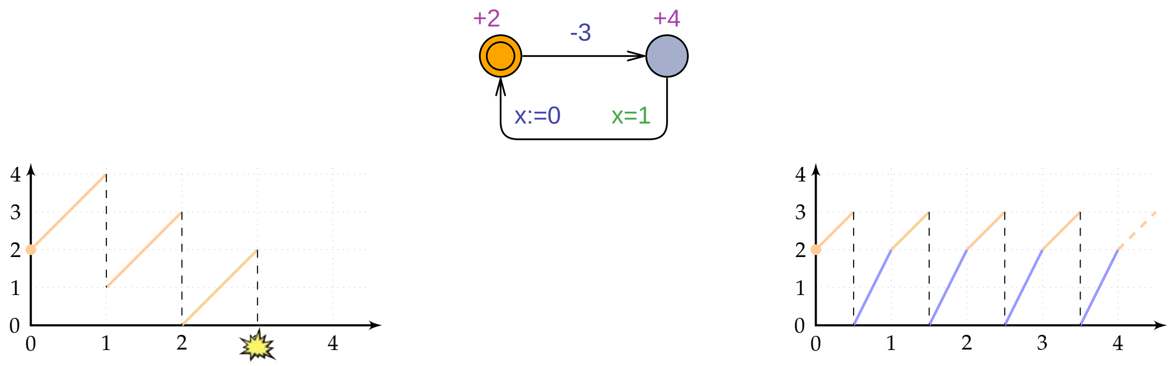
EXAMPLE 5. *Consider the priced timed automaton in Fig. 3 with infinite behaviours repeatedly delaying in the three locations for a total duration of one time unit. The negative weights (-3 and -6) indicate rates by which energy will be consumed, and the positive rate ($+6$) indicates the rate by which energy will be gained. Thus, for a given iteration the effect on the energy remaining will highly depend on the distribution of the one time unit over the three locations. The three types of schedules given an inital energy level of one are illustrated.*

Most recently, the decidability of [26] for the lower-bound problem has been extended to the setting of "$1\frac{1}{2}$" priced timed automata and with prices growing either linearly (*i.e.* $\dot{p} = k$) or exponentially (*i.e.* $\dot{p} = kp$) [25]. By "$1\frac{1}{2}$-clock" priced timed automata we refer to one-clock priced timed automata augmented with discontinuous (discrete) updates (*i.e.*, $p := p + c$) of the price on edges: discrete updates can easily be encoded using a second clock but do not provide the full expressive power of two clocks.

Surprisingly, the presence of discrete updates makes the lower-bound problem significantly more intricate. In particular, whereas region-stable strategies suffice in the search for infinite lower-bound schedules for one-clock priced timed automata, this is no longer the case when discrete updates are permitted as can be seen from Fig. 4. Not being able to rely on the classical region construction, the key to our decidability result is the notion of an *energy function* providing an abstraction of a path in the priced timed automaton (Fig. 5).

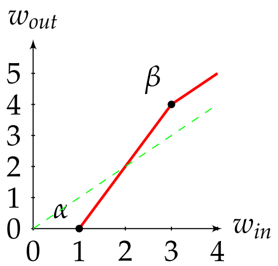
Acknowledgement

The author would like to thank Patricia Bouyer, Uli Fahrenberg, Nicolas Markey, Jiri Srba and Claus Thrane for numerous fruitful discussions on priced timed automata.



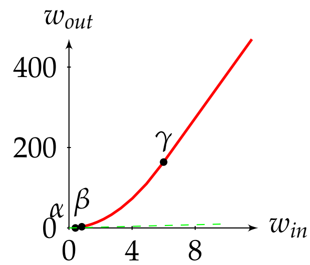
(a) (b)

Figure 4: One-clock priced timed automaton with discrete updates. Infeasibility of region-stable lower-bound schedule (a) and optimal lower-bound schedule (b).



point	w_{in}	w_{out}
α	1	0
β	3	4

(a)



pt	w_{in}	w_{out}
α	≈ 0.406006	0
β	≈ 0.812012	3
γ	6	≈ 163.794

(b)

Figure 5: Energy functions for the two-location path of the priced timed automaton of Figure 4 with linear rates (a) and exponential rates (b).

References

[1] Y. Abdeddaïm, A. Kerbaa, and O. Maler. Task graph scheduling using timed automata. In *IPDPS*, page 237. IEEE Computer Society, 2003.

[2] R. Alur and D. L. Dill. Automata for modeling real-time systems. In M. Paterson, editor, *ICALP*, volume 443 of *Lecture Notes in Computer Science*, pages 322–335. Springer-Verlag, 1990.

[3] R. Alur and D. L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.

[4] R. Alur, S. La Torre, and G. J. Pappas. Optimal paths in weighted timed automata. In M. D. Di Benedetto and A. L. Sangiovani-Vincentelli, editors, *Proceedings of the 4th International Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 49–62. Springer-Verlag, Mar. 2001.

[5] R. Alur, S. L. Torre, and G. J. Pappas. Optimal paths in weighted timed automata. In Benedetto and Sangiovanni-Vincentelli [18], pages 49–62.

[6] E. Asarin and O. Maler. As soon as possible: Time optimal control for timed automata.

In F. W. Vaandrager and J. H. van Schuppen, editors, *HSCC*, volume 1569 of *Lecture Notes in Computer Science*, pages 19–30. Springer, 1999.

[7] E. Asarina, O. Maler, A. Pnueli, and J. Sifakis. Controller synthesis for timed automata. In *IFAC Symposium on System Structure and Control*, pages 469–474. Elsevier, 1998.

[8] G. Behrmann, J. Bengtsson, A. David, K. G. Larsen, P. Pettersson, and W. Yi. Uppaal implementation secrets. In W. Damm and E.-R. Olderog, editors, *FTRTFT*, volume 2469 of *Lecture Notes in Computer Science*, pages 3–22. Springer-Verlag, 2002.

[9] G. Behrmann, P. Bouyer, K. G. Larsen, and R. Pelánek. Lower and upper bounds in zone based abstractions of timed automata. In K. Jensen and A. Podelski, editors, *TACAS*, volume 2988 of *Lecture Notes in Computer Science*, pages 312–326. Springer-Verlag, 2004.

[10] G. Behrmann, A. Cougnard, A. David, E. Fleury, K. G. Larsen, and D. Lime. Uppaal-tiga: Time for playing games! In W. Damm and H. Hermanns, editors, *CAV*, volume 4590 of *Lecture Notes in Computer Science*, pages 121–125. Springer-Verlag, 2007.

[11] G. Behrmann, A. David, and K. G. Larsen. A tutorial on UPPAAL. In M. Bernardo and F. Corradini, editors, *SFM*, volume 3185 of *Lecture Notes in Computer Science*, pages 200–236. Springer-Verlag, 2004.

[12] G. Behrmann, A. Fehnker, T. Hune, K. G. Larsen, P. Pettersson, J. Romijn, and F. W. Vaandrager. Minimum-cost reachability for priced timed automata. In Benedetto and Sangiovanni-Vincentelli [18], pages 147–161.

[13] G. Behrmann, T. Hune, and F. W. Vaandrager. Distributing timed model checking - how the search order matters. In E. A. Emerson and A. P. Sistla, editors, *CAV*, volume 1855 of *Lecture Notes in Computer Science*, pages 216–231. Springer-Verlag, 2000.

[14] G. Behrmann, K. G. Larsen, J. Pearson, C. Weise, and W. Yi. Efficient timed reachability analysis using clock difference diagrams. In N. Halbwachs and D. Peled, editors, *CAV*, volume 1633 of *Lecture Notes in Computer Science*, pages 341–353. Springer-Verlag, 1999.

[15] G. Behrmann, K. G. Larsen, and R. Pelánek. To store or not to store. In W. A. H. Jr. and F. Somenzi, editors, *CAV*, volume 2725 of *Lecture Notes in Computer Science*, pages 433–445. Springer-Verlag, 2003.

[16] G. Behrmann, K. G. Larsen, and J. I. Rasmussen. Priced timed automata: Algorithms and applications. In F. S. de Boer, M. M. Bonsangue, S. Graf, and W. P. de Roever, editors, *FMCO*, volume 3657 of *Lecture Notes in Computer Science*, pages 162–182. Springer-Verlag, 2004.

[17] G. Behrmann, K. G. Larsen, and J. I. Rasmussen. Optimal scheduling using priced timed automata. *SIGMETRICS Performance Evaluation Review*, 32(4):34–40, 2005.

[18] M. D. D. Benedetto and A. L. Sangiovanni-Vincentelli, editors. *Hybrid Systems: Computation and Control, 4th International Workshop, HSCC 2001, Rome, Italy, March 28-30, 2001, Proceedings*, volume 2034 of *Lecture Notes in Computer Science*. Springer-Verlag, 2001.

[19] P. Bouyer, T. Brihaye, and N. Markey. Improved undecidability results on weighted timed automata. *Inf. Process. Lett.*, 98(5):188–194, 2006.

[20] P. Bouyer, Th. Brihaye, V. Bruyère, and J.-F. Raskin. On the optimal reachability problem on weighted timed automata. *Formal Methods in System Design*, 31(2):135–175, Oct. 2007.

[21] P. Bouyer, E. Brinksma, and K. G. Larsen. Staying alive as cheaply as possible. In R. Alur and G. J. Pappas, editors, *HSCC*, volume 2993 of *Lecture Notes in Computer*

Science, pages 203–218. Springer-Verlag, 2004.

[22] P. Bouyer, E. Brinksma, and K. G. Larsen. Optimal infinite scheduling for multi-priced timed automata. *Formal Methods in System Design*, 32(1):2–23, Feb. 2008.

[23] P. Bouyer, F. Cassez, E. Fleury, and K. G. Larsen. Optimal strategies in priced timed game automata. In K. Lodaya and M. Mahajan, editors, *FSTTCS*, volume 3328 of *Lecture Notes in Computer Science*, pages 148–160. Springer-Verlag, 2004.

[24] P. Bouyer, F. Cassez, E. Fleury, and K. G. Larsen. Synthesis of optimal strategies using hytech. *Electr. Notes Theor. Comput. Sci.*, 119(1):11–31, 2005.

[25] P. Bouyer, U. Fahrenberg, K. G. Larsen, and N. Markey. Timed automata with observers under energy constraints. Under submission, 2009.

[26] P. Bouyer, U. Fahrenberg, K. G. Larsen, N. Markey, and J. Srba. Infinite runs in weighted timed automata with energy constraints. In F. Cassez and C. Jard, editors, *FORMATS*, volume 5215 of *Lecture Notes in Computer Science*, pages 33–47. Springer, 2008.

[27] P. Bouyer, K. G. Larsen, and N. Markey. Model-checking one-clock priced timed automata. In H. Seidl, editor, *Proceedings of the 10th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'07)*, volume 4423 of *Lecture Notes in Computer Science*, pages 108–122, Braga, Portugal, Mar. 2007. Springer.

[28] P. Bouyer, K. G. Larsen, and N. Markey. Model checking one-clock priced timed automata. *Logical Methods in Computer Science*, 4(2:9), June 2008.

[29] P. Bouyer, K. G. Larsen, N. Markey, and J. I. Rasmussen. Almost optimal strategies in one clock priced timed games. In S. Arun-Kumar and N. Garg, editors, *FSTTCS*, volume 4337 of *Lecture Notes in Computer Science*, pages 345–356. Springer-Verlag, 2006.

[30] M. Bozga, C. Daws, O. Maler, A. Olivero, S. Tripakis, and S. Yovine. Kronos: A model-checking tool for real-time systems. In A. J. Hu and M. Y. Vardi, editors, *CAV*, volume 1427 of *Lecture Notes in Computer Science*, pages 546–550. Springer-Verlag, 1998.

[31] T. Brihaye, V. Bruyère, and J.-F. Raskin. Model-checking for weighted timed automata. In Y. Lakhnech and S. Yovine, editors, *FORMATS/FTRTFT*, volume 3253 of *Lecture Notes in Computer Science*, pages 277–292. Springer, 2004.

[32] T. Brihaye, V. Bruyère, and J.-F. Raskin. On optimal timed strategies. In P. Pettersson and W. Yi, editors, *FORMATS*, volume 3829 of *Lecture Notes in Computer Science*, pages 49–64. Springer-Verlag, 2005.

[33] U. Fahrenberg and K. G. Larsen. Discount-optimal infinite runs in priced timed automata. *Electr. Notes Theor. Comput. Sci.*, 2008. To be published.

[34] A. Fehnker. Scheduling a steel plant with timed automata. In *RTCSA*, pages 280–286. IEEE Computer Society, 1999.

[35] T. Hune, K. G. Larsen, and P. Pettersson. Guided synthesis of control programs using uppaal. *Nord. J. Comput.*, 8(1):43–64, 2001.

[36] K. G. Larsen, G. Behrmann, E. Brinksma, A. Fehnker, T. Hune, P. Pettersson, and J. Romijn. As cheap as possible: Efficient cost-optimal reachability for priced timed automata. In G. Berry, H. Comon, and A. Finkel, editors, *CAV*, volume 2102 of *Lecture Notes in Computer Science*, pages 493–505. Springer-Verlag, 2001.

[37] K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi. Efficient verification of real-time systems: compact data structure and state-space reduction. In *IEEE Real-Time Systems Symposium*, pages 14–24. IEEE Computer Society, 1997.

[38] K. G. Larsen, J. Pearson, C. Weise, and W. Yi. Clock difference diagrams. *Nord. J. Comput.*, 6(3):271–298, 1999.

[39] K. G. Larsen, P. Pettersson, and W. Yi. Uppaal in a nutshell. *STTT*, 1(1–2):134–152, 1997.

[40] K. G. Larsen and J. I. Rasmussen. Optimal reachability for multi-priced timed automata. *Theor. Comput. Sci.*, 390(2-3):197–213, 2008.

[41] O. Maler. Timed automata as an underlying model for planning and scheduling. In M. Fox and A. M. Coddington, editors, *AIPS Workshop on Planning for Temporal Domains*, pages 67–70, 2002.

[42] J. I. Rasmussen, G. Behrmann, and K. G. Larsen. Complexity in simplicity: Flexible agent-based state space exploration. In O. Grumberg and M. Huth, editors, *TACAS*, volume 4424 of *Lecture Notes in Computer Science*, pages 231–245. Springer, 2007.