Safe and Optimal Scheduling of Hard and Soft Tasks (Extended Abstract)

Gilles Geeraerts, Shibashis Guha, and Jean-Francois Raskin

Université Libre de Bruxelles

In modern real-time systems, we usually need to distinguish between two types of tasks: *hard tasks* that ought to be scheduled so that they meet their deadline with *absolute certainty* and *soft tasks* for which missing a deadline is tolerated. Typically, hard tasks are vital for the correct execution of the system and missing a deadline for such tasks may have catastrophic consequences while missing a deadline of a soft task only degrades the overall performances of the system. It is also usual to distinguish between tasks for which the inter-arrival time is fixed and which are often called *periodic* tasks, and tasks for which the inter-arrival time is subject to uncertainty and specified by an interval constraints and which are often called *sporadic* tasks. Periodic tasks are suitable for applications where it is required to sample regularly a given physical entity (e.g. a temperature, pressure, torque, speed, ...), or actuate regularly on the system via an actuator. Sporadic tasks are suitable for modelling scenarios where the activation instants cannot be forecast with precision, like in human-machine interfaces, or external asynchronous interrupts. Most real systems contain naturally both periodic and aperiodic events/tasks.

In this work, we consider a rich formal model of ∞-duration scheduling that is applicable to systems with both periodic/aperiodic and hard/soft tasks. Additionally, we assume our schedulers to be *non-clairvoyant* in the sense that the execution and inter-arrival times of tasks are subject to uncertainty modelled by stochastic distributions. More precisely, each task in our system is identified as either hard or soft. Each hard and soft task is specified by a fixed deadline, and two discrete finite support distributions: one for their computation times and one for their inter-arrival times. When a job (i.e. an instance of a task) associated to a task is created, its computation time is not known but only the probability distribution among the possible computation times is known. Similarly, the arrival time of the next job associated to a task is not known but only the distribution among the possible arrival times is known. In addition, each soft task comes with a cost that is incurred each time a deadline for a job of this task is missed. The objective of the scheduler in this model is twofold:

- the deadline of all jobs corresponding to hard tasks must be met with certainty, and
- the expected mean cost of missing deadlines of jobs associated to soft tasks must be minimised.

Contributions: First, we define formally the non-clairvoyant scheduling problem with both periodic/aperiodic and hard/soft tasks and show how this problem can be reduced to a non-standard optimisation problem for finite Markov decision processes (MDP). More precisely, we consider MDPs with two objectives that need to be satisfied *simultaneously*: one safety objective and one expected mean-cost minimisation objective. The safety objective is used to model the constraints on hard tasks: the deadline of each job associated to such task must be met with absolute certainty. While the expected mean-cost minimisation objective is used to model the preference for schedules that minimise the expected mean-cost of missing job deadlines associated to soft tasks.

Second, we provide a worst-case exponential time algorithm that decides the existence of a safe and optimal schedule, and we study the computational complexity of our scheduling problem. While our scheduling problem generalises scheduling problems that are known to be NP-Complete and coNP-Complete (see related works below), we show that our problem is harder than those problems by proving hardness of our problem for the class PP, i.e. the class of problems that can be solved by a *probabilistic* time Turing machine that operates in polynomial time [13]. This complexity class contains both NP and coNP, and is closed under complement, see e.g. [6].

Third, we have implemented a prototype of tool on top of the probabilistic model-checker Storm [11]. Given a description of the set of hard and soft tasks, our prototype first computes the set of safe schedules, i.e., all the schedules in which the deadline of all the hard tasks are met with certainty. Then our prototype produces a Storm model for the soft tasks that only allows those schedules that have been shown safe by the previous step of the algorithm. A strategy that minimises the expected mean-cost for missing the deadlines of the soft tasks in this model is guaranteed to be an optimal strategy among all the safe strategies for the hard tasks. We are looking for such an optimal strategy. Currently the prototype tool can handle a total of 4-5 hard and soft tasks that can lead to almost a million states in the constructed MDP. We plan to develop a more efficient implementation with a notion of maximality over states that forms an antichain. We think that the use of antichain will allow us to handle systems with more states.

Finally, we compare the optimal solutions obtained with our procedure against an adaptation of the *earliest deadline first* (EDF) algorithm to account for the soft tasks. We show that this EDF-like strategy can be arbitrarily worse in terms of the expected mean cost for missing the deadlines of the soft tasks when compared to the optimal strategy that our procedure constructs. This is shown with a family of examples where the gap of performances can be made as large as desired. We also provide examples and experimental results using the probabilistic model-checker Storm in which the solutions computed by our algorithm have been compared to solutions provided by EDF-like strategies. Note that the safe and optimal scheduler computed by our algorithm consists of a simple control table that can be directly used at execution time.

Related work: The schedulability of (hard) periodic tasks is a classical problem that have been studied in details in the literature, see e.g. [20, 17, 18, 12, 5]. This classical scheduling problem can be seen as a special case of our problem with only hard tasks for which the inter-arrival times of tasks and the computation times are fixed, which can both be modelled as Dirac distributions in our model. The schedulability of (hard) periodic tasks has been shown coNP-complete in [17, 5]. In the literature, there are also scheduling problems that consider tasks that are not (strictly) periodic but whose inter-arrival time is specified by an interval [23, 16, 14, 4]. If the interval of inter-arrival times is finite, then this scheduling problem can also be seen as a special case of our problem with only hard tasks. Indeed, if I is the finite set of inter-arrival times for a task, then we can model this by a probability distribution of inter-arrival times with a support equal to I. For the schedulability of the hard tasks, only the support of the distributions matters rather than the actual distribution.

The *clairvoyant* scheduling of (only) soft tasks is also a classical problem that has attracted ample attention in the scheduling literature, see e.g. [25, 19, 3]. A common setting that is found in the literature considers only one period and a set of tasks that have both mandatory and optional sub-parts. The mandatory part can be considered as a hard task for which the execution needs to be completed while the optional part is similar to a soft task in our setting. Each task has a fixed computation time (Dirac distribution) and a fixed deadline. There is a cost equal to the computation time of the optional part unless the optional part executes entirely. The problem of minimisation of total cost is NP-complete when the optional tasks have arbitrary processing times [21]. We can see that the NP-completeness already holds when we consider tasks without the mandatory sub-part which is a proper subset of the setting that we consider here.

Finally, there are works in the literature that consider scheduling problems with both hard and soft tasks, see e.g. [8, 24, 9, 1]. But to the best of our knowledge, none of those works consider the non-clairvoyant scheduling of both hard and soft tasks with stochastic uncertainty.

In [15], *duration probabilistic automata* which is a class of acyclic timed automata has been considered to synthesize expected time optimal schedulers in a system of tasks where the task durations are uniformly distributed. In [22], the following problem has been studied: synthesis of schedule for production systems where resources can fail probabilistically and the costs incurred are of two types: storage costs that are incurred when an order is finished before the due date, and delay costs, which are incurred if an order is finished after the due date. The non-standard optimisation problem that we consider on MDP and which simultaneously asks for satisfying a safety and an expected mean-cost constraint is related to a recent line of works that mixes two-player zero sum games and MDPs, see e.g. [7, 2, 10].

References

[1] L. Abeni and G. Buttazzo. Integrating multimedia applications in hard real-time systems. In *Proceedings of the IEEE Real-Time Systems Symposium*, RTSS '98, pages 4–13, Washington, DC, USA, 1998. IEEE Computer Society.

[2] S. Almagor, O. Kupferman, and Y. Velner. Minimizing expected cost under hard boolean constraints, with applications to quantitative synthesis. In *Proc. 27th Int. Conf. on Concurrency Theory*, volume 59 of *LIPIcs*, pages 9:1–9:15, 2016.

[3] H. Aydin, R. G. Melhem, D. Mossé, and P. Mejía-Alvarez. Optimal reward-based scheduling of periodic real-time tasks. In *Proceedings of the 20th IEEE Real-Time Systems Symposium, Phoenix, AZ, USA, December 1-3, 1999*, pages 79–89, 1999.

[4] S. K. Baruah, A. K. Mok, and L. E. Rosier. Preemptively scheduling hard-real-time sporadic tasks on one processor. In *Proceedings of the Real-Time Systems Symposium - 1990, Lake Buena Vista, Florida, USA, December 1990*, pages 182–190, 1990.

[5] S. K. Baruah, L. E. Rosier, and R. R. Howell. Algorithms and complexity concerning the preemptive scheduling of periodic, real-time tasks on one processor. *Real-Time Systems*, 2(4):301–324, 1990.

[6] R. Beigel, N. Reingold, and D. A. Spielman. PP is closed under intersection. *Journal of Computer and Systems Science*, 50(2):191–202, 1995.

[7] V. Bruyère, E. Filiot, M. Randour, and J-F. Raskin. Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games. In *Proc. 31th Symp. on Theoretical Aspects of Computer Science*, volume 25 of *LIPIcs*, pages 199–213, 2014.

[8] G. Buttazzo. *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications.* Springer Publishing Company, Incorporated, 3rd edition, 2011.

[9] H. Chetto and M. Chetto. Some results of the earliest deadline scheduling algorithm. *IEEE Trans. Software Eng.*, 15(10):1261–1269, 1989.

[10] L. Clemente and J-F. Raskin. Multidimensional beyond worst-case and almost-sure problems for mean-payoff objectives. In *Proc. 30th IEEE Symp. on Logic in Computer Science*, pages 257–268, 2015.

[11] C. Dehnert, S. Junges, J-P. Katoen, and M. Volk. A storm is coming: A modern probabilistic model checker. In *Computer Aided Verification - 29th International Conference, CAV 2017, Heidelberg, Germany, Proceedings, Part II*, pages 592–600, 2017.

[12] S. K. Dhall and C. L. Liu. On a real-time scheduling problem. *Oper. Res.*, 26(1):127–140, February 1978.

[13] John Gill. Computational complexity of probabilistic turing machines. *SIAM Journal on Computing*, 6(4):675–695, 1977.

[14] K. S. Hong and J. Y.-T. Leung. On-line scheduling of real-time tasks. In *Proceedings of the 9th IEEE Real-Time Systems Symposium (RTSS '88), December 6-8, 1988, Huntsville, Alabama, USA*, pages 244–250, 1988.

[15] J-F Kempf, M. Bozga, and O. Maler. As soon as probable: Optimal scheduling under stochastic uncertainty. In *Proc. 19th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, pages 385–400, 2013.

[16] J. P. Lehoczky, L. Sha, and I. Strosnider. Enhanced aperiodic responsiveness in a hard real-time environment. In *Real-Time Systems Symposium*, pages 261–270, 1987.

[17] J. Y.-T. Leung and M. L. Merrill. A note on preemptive scheduling of periodic, real-time tasks. *Information Processing Letters*, 11(3):115–118, 1980.

[18] J. Y.-T. Leung and J. Whitehead. On the complexity of fixed-priority scheduling of periodic, real-time tasks. *Perform. Eval.*, 2(4):237–250, 1982.

[19] K-J Lin, S. Natarajan, and J. W.-S. Liu. Imprecise results: Utilizing partial comptuations in real-time systems. In *Proceedings of the 8th IEEE Real-Time Systems Symposium (RTSS '87), December 1-3, 1987, San Jose, Califorinia, USA*, pages 210–217, 1987.

[20] C. L. Liu and J. W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the ACM*, 20(1):46–61, January 1973.

[21] J. W. S. Liu, K. J. Lin, W. K. Shih, A. C. Yu, J. Y. Chung, and W. Zhao. *Algorithms for Scheduling Imprecise Computations*, pages 203–249. Springer US, Boston, MA, 1991.

[22] A. Mader, H. Bohnenkamp, Y. S. Usenko, D. N. Jansen, J. Hurink, and H. Hermanns. Synthesis and stochastic assessment of cost-optimal schedules. *International Journal on Software Tools for Technology Transfer*, 12(5):305–318, Sep 2010.

[23] A. K. Mok. *Fundamental design problems of distributed systems for the hard-real-time environment.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1983.

[24] M. Spuri and G. Buttazzo. Scheduling aperiodic tasks in dynamic priority systems. *Real-Time Systems*, 10(2):179–210, 1996.

[25] A. Srinivasan and J. H. Anderson. Efficient scheduling of soft real-time applications on multiprocessors. *J. Embedded Comput.*, 1(2):285–302, April 2005.