Poster Abstract: Average Probabilistic Response Time Analysis of Tasks with Multiple Probabilistic Parameters

Antoine Bertout
Inria de Paris
antoine.bertout@inria.fr

Dorin Maxim
Inria de Paris
dorin.maxim@inria.fr

Liliana Cucu-Grosjean
Inria de Paris
liliana.cucu@inria.fr

***Abstract*—Probabilistic timing analyses are used to decrease the pessimism introduced by deterministic methods based on unique worst-case values. In particular, the Probabilistic Response Time Analysis computes the worst-case response time distribution of a task. Nevertheless, it is based on a worst-case pattern of execution that may rarely occurs. Thus, in order to reduce the pessimism introduced, we consider in this paper the problem of computing an average response time distribution of a task.**

I. Introduction

Different methods have been developed to assess the ability of hard-real-time systems to respect their timing constraints. In the one hand, certain of those are based on simulation or intensive testing. These are generally counterbalanced by an accepted (empirical) marge of error, introducing in this way a large pessimism to compensate the optimism of their initial results. On the other hand, the field of real-time scheduling proposes numerous theoretical analyses(e.g [1], [2]). Due to the complexity of modeling such critical systems and to the difficulty to represent their behaviour precisely, these analysis are often based on very pessimistic assumptions.

In the two cases, the pessimism leads to an over-provisioning of the systems which is costly in terms of resources. Also, it reduces the possibility to increase their performance due to the limited available resources.

Probabilistic timing analyses offer an alternative in giving a theoretical, safe, probability to miss some timing constraints instead of having only a yes or no answer. This information spares designers from rejecting systems that have actually a very low probability of failure. For instance, in the aerospace industry, one may compare the maximum allowed probability of failure of 10^{-9} per hour of operation (required by the certification authorities [3]) and the probability of failure of 10^{-15} that the system might experience per hour of operation [4].

Probabilistic timing analyses provide useful tools to reduce the pessimism of deterministic methods. Maxim and Cucu-Grosjean [5] proposed a Probabilistic Response Time Analysis (PRTA) with multiple probabilistic parameters (worst-case execution time, deadline and minimum inter-arrival time) for fixed-task priority scheduling. This analysis provides a worst-case response time (WCRT) distribution from which we can deduce the probability of a system to miss a deadline. They focused on the synchronous probabilistic analysis, where they consider that the task under analysis arriving simultaneously with all other tasks, what corresponds to the worst-case situation. However, in reality, this situation may seldom occur and sometimes never happened. This also introduces overly pessimistic estimations and gives a distorted view of the average behaviour of the system.

In this paper, we consider the problem of computing an average response time distribution on a given interval of time.

Summary: We propose to study the average response time distribution of tasks owing to the pessimism introduced by synchronous analysis and unsafe response time obtained by simulation. On this regard, the problem we address is twofold. First, we need to determine a relevant interval of study to compute a representative and safe behaviour of the response time distribution. Second, it requires to compute the response time of any job of a task with multiple probabilistic parameters. Both questions relate to open problems.

References

[1] M. Joseph and P. Pandya, "Finding response times in a real-time system," *The Computer Journal*, vol. 29, no. 5, pp. 390–395, 1986.
[2] S. K. Baruah, A. K. Mok, and L. E. Rosier, "Preemptively scheduling hard-real-time sporadic tasks on one processor," in *Real-Time Systems Symposium, 1990. Proceedings., 11th*, Dec 1990, pp. 182–190.
[3] W. River, "Arinc 653–an avionics standard for safe, partitioned systems," in *IEEE Seminar*, 2008.
[4] L. Cucu-Grosjean, L. Santinelli, M. Houston, C. Lo, T. Vardanega, L. Kosmidis, J. Abella, E. Mezzetti, E. Quinones, and F. J. Cazorla, "Measurement-based probabilistic timing analysis for multi-path programs," in *24th Euromicro Conference on Real-Time Systems*. IEEE, 2012, pp. 91–101.
[5] D. Maxim and L. Cucu-Grosjean, "Response time analysis for fixed-priority tasks with multiple probabilistic parameters," in *Real-Time Systems Symposium (RTSS), 2013 IEEE 34th*. IEEE, 2013, pp. 224–235.

Average Probabilistic Response Time Analysis of Tasks with Multiple Probabilistic Parameters

Antoine Bertout
Inria de Paris
antoine.bertout@inria.fr

Dorin Maxim
Inria de Paris
dorin.maxim@inria.fr

Liliana Cucu-Grosjean
Inria de Paris
liliana.cucu@inria.fr

***Abstract*—Probabilistic timing analyses are used to decrease the pessimism introduced by deterministic methods based on unique worst-case values. In particular, the Probabilistic Response Time Analysis computes the worst-case response time distribution of a task. Nevertheless, it is based on a worst-case pattern of execution that may rarely occurs. Thus, in order to reduce the pessimism introduced, we consider in this paper the problem of computing an average response time distribution of a task.**

I. Introduction

Different methods have been developed to assess the ability of hard-real-time systems to respect their timing constraints. In the one hand, certain of those are based on simulation or intensive testing. These are generally counterbalanced by an accepted (empirical) marge of error, introducing in this way a large pessimism to compensate the optimism of their initial results. On the other hand, the field of real-time scheduling proposes numerous theoretical analyses(e.g [1], [2]). Due to the complexity of modeling such critical systems and to the difficulty to represent their behaviour precisely, these analysis are often based on very pessimistic assumptions.

In the two cases, the pessimism leads to an over-provisioning of the systems which is costly in terms of resources. Also, it reduces the possibility to increase their performance due to the limited available resources.

Probabilistic timing analyses offer an alternative in giving a theoretical, safe, probability to miss some timing constraints instead of having only a yes or no answer. This information spares designers from rejecting systems that have actually a very low probability of failure. For instance, in the aerospace industry, one may compare the maximum allowed probability of failure of 10^{-9} per hour of operation (required by the certification authorities [3]) and the probability of failure of 10^{-15} that the system might experience per hour of operation [4].

Probabilistic timing analyses provide useful tools to reduce the pessimism of deterministic methods. Maxim and Cucu-Grosjean [5] proposed a Probabilistic Response Time Analysis (PRTA) with multiple probabilistic parameters (worst-case execution time, deadline and minimum inter-arrival time) for fixed-task priority scheduling. This analysis provides a worst-case response time (WCRT) distribution from which we can deduce the probability of a system to miss a deadline. They focused on the synchronous probabilistic analysis, where they consider that the task under analysis arriving simultaneously with all other tasks, what corresponds to the worst-case situation. However, in reality, this situation may seldom occur and sometimes never happened. This also introduces overly pessimistic estimations and gives a distorted view of the average behaviour of the system.

In this paper, we consider the problem of computing an average response time distribution on a given interval of time.

II. Related work

Real-time software designers need to verify the functional and non-functional behaviours of their systems. Timing constraints are part of the latter and can be checked by theoretical analysis and simulation.

A. Analysis

The real-time literature provides a considerable number of deterministic schedulability analyses for different platforms and model. Most of them are based on the processor demand analysis (PDA) [2] or on the response time analysis (RTA) [1]. While the first only assesses if a system is schedulable or not, the second gives the worst-case response time of each task in the system. Thus, if all the response times of the tasks are inferior or equal to their deadline, the system is considered to be schedulable. These deterministic schedulability tests consider worst-case values and often produce pessimistic results. To cope with this pessimism, probabilistic or stochastic counterparts have been proposed. For instance, Diaz et. al [6] proposed a probabilistic response time analysis (PRTA) for tasks having their execution time characterised by a random variable. Later, Maxim and Cucu-Grosjean [5] added the support of probabilistic worst-case execution times (pWCET) and probabilistic minimum-inter-arrival time (pMIT). Besides theoretical analysis, a common practice in the industry is to rely on testing or on simulation in order to decrease the pessimism of analysis.

B. Simulation

A simulator models and reproduces the temporal behaviour of a system. If compliant with the true system operating, it provides an empirical way to verify the timing constraints instead of running the original system which is not always possible. To counterbalance the optimism possibly introduced, a widespread technique is to add a margin of error empirically chosen by domain experts. Among the academic (deterministic) simulators, one may cite SimSo [7] which is compatible with a lot of multiprocessor scheduling policies or Mast [8]. In the industry, RTaW-Sim [9] from RTaW is focused on the analysis of real-time networks while RapiTime developed by Rapita Systems [10] targets general real-time embedded applications. As far as we know, there is only one probabilistic analysis tool, specifically prototyped for the work of Maxim et al. [11].

In our work, we mean to make use of simulators to obtain empirical (average) response time distributions. This, in order to guide our efforts in providing a safe analysis to determine the average response time distribution of a task described by probabilistic WCET and probabilistic MIT.

III. Problem description

A. System model

We model a system as a set of n tasks $\{\tau_1, \tau_2, ..., \tau_n\}$ scheduled by fixed-task priority algorithm (as Deadline Monotonic (RM)) on a uniprocessor platform. Without loss of generality, we consider that τ_i has a higher priority than τ_j for $i < j$.

Each task τ_i generates an infinite number of successive jobs τ_{ij}, with $j = 1, \ldots, \infty$. All jobs are assumed to be independent of other jobs of the same task and those of other tasks.

Each task τ_i is characterised by two probabilistic parameters: a pWCET denoted by $\mathcal{C}_i$ and a probabilistic minimal inter-arrival time (pMIT) denoted by $\mathcal{T}_i$. Note that in this paper, we use calligraphic typeface to denote random variables. Each parameter is represented by a random variable $\mathcal{X}$ having a probability function (PF) $f_{\mathcal{X}}(\cdot)$ with $f_{\mathcal{X}}(x) = P(\mathcal{X} = x)$. The possible values of $\mathcal{X}_i$ belong to the interval $[X^{\min}, X^{\max}]$. In this paper we associate the probabilities with the possible values of a random variable using the following notation:

$$\mathcal{X} = \begin{pmatrix} X^0 = X^{min} & X^1 & \cdots & X^k = X^{max} \\ f_{\mathcal{X}}(X^{min}) & f_{\mathcal{X}}(X^1) & \cdots & f_{\mathcal{X}}(X^{max}) \end{pmatrix} \quad (1)$$

where $\sum_{j=0}^{k} f_{\mathcal{X}}(X^j) = 1$.

The notions of pWCET and pMIT are defined as follows. For more details, the reader can refer to [5].

Definition 1. The probabilistic worst case execution time (pWCET) of a task describes the probability that the worst case execution time of that task is equal to a given value.

Following the same reasoning the probabilistic minimal inter-arrival time (pMIT) denoted by $\mathcal{T}_i$ describes the probabilistic minimal inter-arrival times of all jobs.

Definition 2. The probabilistic minimal inter-arrival time (pMIT) of a task describes the probability that the minimal inter-arrival time of that task is equal to a given value.

Hence, a task τ_i is represented by a tuple $(\mathcal{C}_i, \mathcal{T}_i)$. A job of a task must finish its execution before the arrival of the next job of the same task, i.e., the arrival of a new job represents the deadline of the current job. Thus, the task deadline may also be represented by a random variable $\mathcal{D}_i$ which has the same distribution as its pMIT, $\mathcal{T}_i$. We consider tasks with implicit deadlines, i.e., having the same distribution as the pMIT.

B. Motivating example

Let us consider the task-set presented in Table I, which has five tasks, and each task is represented by a pWCET distribution and a pMIT distribution, with ten values per distribution. For the sake of simplicity we have made all probabilities equal to 0.1 and so we omit them from the table. We are interested in the response time of the task on the lowest priority level, i.e τ_5.

Figure 1 depicts the response times of τ_5 obtained in three ways:

- the dashed blue curve represents, in the form of 1-CDF, the probabilistic worst case response times of the task in the synchronous case (i.e. worst case conditions) computed using the the probabilistic response time analysis of Maxim and Cucu-Grosjean [5].
- the solid red line is an empirical 1-CDF distribution formed by simulating the probabilistic task-set and recording the observed response times for $5 \cdot 10^5$ consecutive jobs of τ_5. This simulation was performed using an in-house probabilistic extension of the SimSo simulator[1].
- for completeness we have also added the deterministic worst case response time obtained using the analysis in [1]. Note that this is the largest value in the probabilistic response time distribution - the dotted blue curve - and is equal to 16341 and is represented as a dotted vertical line.

In Figure 1 we may note that the response times observed when simulating the system are far smaller than the deterministic worst case response time, but also considerably smaller than the probabilistic response times obtained through analysis. We may also note that the curve obtained through analysis is an upper-bound on the curve obtained through simulation i.e. graphically, the simulated curve is to the left of the analytical curve. While the analysis provides a safe upper-bound an all possible behaviours of the task it may, in some cases, be overly pessimistic. This difference between analysis and observation is given by the fact that the analysis concentrates on the worst case conditions, when the analysed task is activated at the same time as all higher priority tasks. But this worst

[1] The probabilistic simulator is available upon request to the authors.

Table I
PROBABILISTIC TASK SET

task											
τ_1	pWCET	134	137	140	143	146	149	152	155	158	161
	pMIT	3565	3637	3709	3781	3853	3925	3997	4069	4141	4213
τ_2	pWCET	311	318	325	332	339	346	353	360	367	374
	pMIT	7784	7940	8096	8252	8408	8564	8720	8876	9032	9188
τ_3	pWCET	2879	2949	3019	3089	3159	3229	3299	3369	3439	3509
	pMIT	26226	26751	27276	27801	28326	28851	29376	29901	30426	30951
τ_4	pWCET	5540	5675	5810	5945	6080	6215	6350	6485	6620	6755
	pMIT	19617	20010	20403	20796	21189	21582	21975	22368	22761	23154
τ_5	pWCET	3403	3486	3569	3652	3735	3818	3901	3984	4067	4150
	pMIT	32313	32960	33607	34254	34901	35548	36195	36842	37489	38136

case condition may rarely occur during the lifetime of the system as task' periods are variable, with a great impact on the probability of synchronous releases.

We observe that the largest response time recorded during simulation is equal to 15025 and it only appears once, hence an empirical probability of $2{\cdot}10^{-5}$. On the other hand, the probability of appearance of the WCRT=16341 is computed to be in the range of 10^{-15} in the synchronous case, meaning that it is highly unlike to actually see this value (or other large values) during the lifetime of the system. This means that there is still a large amount of pessimism in the existing probabilistic response time analysis and the gap between analysis and actual behaviour of the system can be further narrowed. We believe that there is a noteworthy interest in theoretically analysing the average response time probability distribution of the tasks in the system.

This led us to two underlying problems. First, it raises the question of the interval of study. How many jobs should we analyse to get a safe average response time distribution we can rely on? Second, we need a probabilistic response time analysis for any instance of the task. To our knowledge, there is no such analysis with multiple probabilistic parameters and the PRTA of Maxim and Cucu-Grosjean is restricted to the synchronous case. We will attempt to provide some intuition to these two problems, respectively in section IV and in section V.

We also note that in some cases it may be more probable that tasks synchronise their releases, for example in systems where tasks have periods that are very similar (e.g. harmonic) or even following the same (simple) distribution. In these cases the existing probabilistic analysis is sufficient as it would have a low degree of pessimism with respect to the actual behaviour of the system. For this work, we focus on the case when synchronisation is improbable and there is a need to find a more precise analysis to represent the entire lifetime of the system.

IV. Study interval

A first problem that we need to solve in order to be able to compute the average response time distribution of a task is determining the study interval in order to know how many of its jobs need to be analysed and combined into an average.

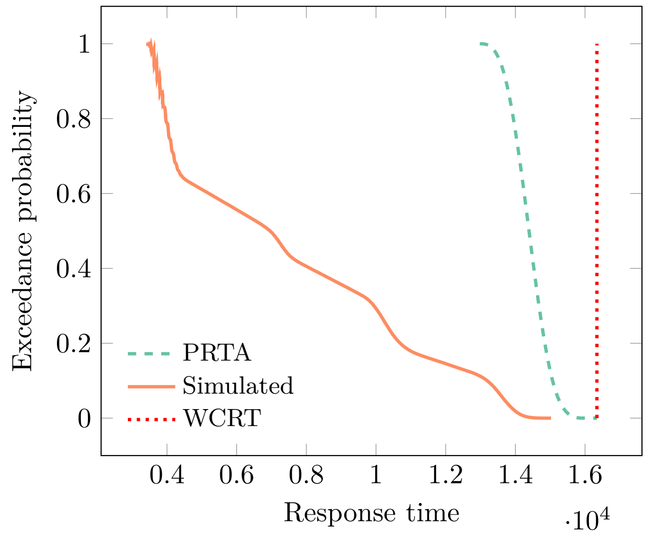

Figure 1. Response time distributions of τ_5 obtained in three different ways

As task' periods are described by random variables, it is not possible to determine a study interval in the form of a hyper-period as this implies computing the least common multiple of all periods - and the least common multiple function does not apply on distributions, but on single values.

One way of going around this problem is to compute a worst case hyper-period, using the worst case (i.e. smallest) values in each pMIT distributions. This approach would not necessarily result in a study interval, as it would be very unlikely that tasks would re-synchronise at the hyper-period and even more unlikely that the scheduling would repeat itself past this point. A deterministic hyper-period computed in this way would be, at best, a very pessimistic study interval.

We propose instead a more natural solution to this problem in the form of a probabilistic hyper-period, which we define a follows:

Definition 3 (Probabilistic Hyper-Period)**.** The Probabilistic Hyper-Period of probability p, pHP(p), is equal to the smallest interval, starting in the current time point, in which the cumulated probability of possible synchronous releases is no smaller than p.

Intuitively, starting with the current time $t = 0$ in which we presume that tasks have been synchronously activated, then, from this point onwards, there may be other potential synchronisation points, each with a certain probability of occurring. By summing up the probabilities of several such (consecutive) points, we obtain a cumulative probability of synchronisation. Once this cumulative probability reaches a given threshold, e.g. $p = 0.9$, then the last synchronisation point, $t = t_n$, in this sequence gives us the length of the probabilistic hyper-period $pHP(p) = t_n$. We can be

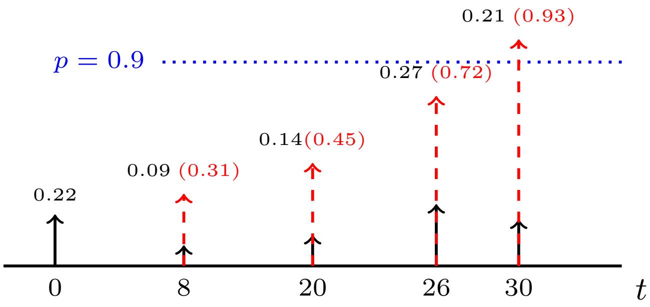

Figure 2. Probabilistic synchronisation points of task with probabilistic periods

certain with a probability p that there will be at least on synchronisation between $t = 0$ and $t = t_n$ and that the response time distributions of the jobs in this interval will be representative for all jobs that may appear during the entire life-time of the system. This concept is graphically presented in Figure 2, with the black arrows (and text) representing the various synchronisation points (and their probabilities), while the red arrows (and text) represents the cumulative probabilities of synchronisations occurring - a higher arrow implies a higher probability of occurrence.

A larger probability p, up to at most 1, implies a larger confidence in the study interval and a more precise average response time distribution, as it contains more jobs that may occur during the entire lifetime of the system. A probability of $p = 1$ means that all representative jobs of the task (and their response times) are taken into consideration when computing the average response time distribution.

Alternatively we may opt for a smaller probability p and a smaller study interval pHP which will result in a more pessimistic average response time distribution of the task. In some cases we may be forced to choose a smaller study interval, as synchronisations may be highly improbably. When synchronisations are unlikely, the resulting study interval tends to be prohibitively large and so the number of jobs that would need to be analysed would be too high and the effort of analysing them would be either too large or unjustified by the decrease in pessimism.

V. Job Response Time Analysis

In section III-B, we stressed the need for an average response time analysis of tasks in order to decrease the pessimism of existing state of the art analyses. This requires the computation of response time distributions of each job in a study interval in order to combine them into an average. This problem has already been address in the deterministic case, in papers such as [12], [13]. Audsley [12] proposed an analysis to compute the response time of any job in the study interval in the case of asynchronous tasks by determining the interference caused to the job by higher priority jobs. Later, Coutinho et al. [13] introduced an analysis based on idle instants that also supports sporadic tasks. To our knowledge, the work of Diaz et al. [6] is the only existing average probabilistic response analysis but it is restricted to only probabilistic WCET (i.e. a single source of variability in the task), which also has the advantage that a study interval can be easily defined. We are currently working on extending such analyses to task with multiple probabilistic parameters.

VI. Conclusion

In this paper, we emphasised the need for an average probabilistic response time analysis for real-time tasks, in order to cope with the pessimism of existing analyses and to bridge the gap between analysis and experiment/simulation/actual execution. To this end we identified two underlying problems: defining an appropriate study interval and being able to analyse the response time of any job released in this interval. In future work, we plan on providing such analysis for tasks with multiple probabilistic parameters.

References

[1] M. Joseph and P. Pandya, "Finding response times in a real-time system," *The Computer Journal*, vol. 29, no. 5, pp. 390–395, 1986.

[2] S. K. Baruah, A. K. Mok, and L. E. Rosier, "Preemptively scheduling hard-real-time sporadic tasks on one processor," in *Real-Time Systems Symposium, 1990. Proceedings., 11th*, Dec 1990, pp. 182–190.

[3] W. River, "Arinc 653–an avionics standard for safe, partitioned systems," in *IEEE Seminar*, 2008.

[4] L. Cucu-Grosjean, L. Santinelli, M. Houston, C. Lo, T. Vardanega, L. Kosmidis, J. Abella, E. Mezzetti, E. Quinones, and F. J. Cazorla, "Measurement-based probabilistic timing analysis for multi-path programs," in *24th Euromicro Conference on Real-Time Systems*. IEEE, 2012, pp. 91–101.

[5] D. Maxim and L. Cucu-Grosjean, "Response time analysis for fixed-priority tasks with multiple probabilistic parameters," in *Real-Time Systems Symposium (RTSS), 2013 IEEE 34th*. IEEE, 2013, pp. 224–235.

[6] J. L. Diaz, D. F. Garcia, K. Kim, C.-G. Lee, L. L. Bello, J. M. Lopez, S. L. Min, and O. Mirabella, "Stochastic analysis of periodic real-time systems," in *23rd IEEE Real-Time Systems Symposium, RTSS 2002*, 2002, pp. 289–300.

[7] M. Chéramy, P.-E. Hladik, and A.-M. Déplanche, "SimSo: A simulation tool to evaluate real-time multiprocessor scheduling algorithms," in *5th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*, 2016, p. 6.

[8] M. G. Harbour, J. G. García, J. P. Gutiérrez, and J. D. Moyano, "Mast: Modeling and analysis suite for real time applications," in *Real-Time Systems, 13th Euromicro Conference on, 2001*. IEEE, 2001, pp. 125–134.

[9] "RealTime-at-Work - RTAW-Sim: Fine-grained simulation of controller area network with fault injection capabilities," http://www.realtimeatwork.com/software/rtaw-sim/.

[10] "Rapita Systems - RapiTime," https://www.rapitasystems.com/products/rapitime.

[11] D. Maxim, F. Soboczenski, I. Bate, and E. Tovar, "Study of the reliability of statistical timing analysis for real-time systems," in *Proceedings of the 23rd International Conference on Real Time and Networks Systems*. ACM, 2015, pp. 55–64.

[12] N. C. Audsley, *Optimal priority assignment and feasibility of static priority tasks with arbitrary start times*, 1991.

[13] M. Coutinho, J. Rufino, and C. Almeida, "Response Time Analysis of Asynchronous Periodic and Sporadic Tasks Scheduled by a Fixed Priority Preemptive Algorithm," in *Euromicro Conference on Real-Time Systems, 2008. ECRTS '08*, Jul. 2008, pp. 156–167.