Response Time Analysis for Fixed-Priority Tasks with Multiple Probabilistic Parameters

Dorin Maxim * Liliana Cucu-Grosjean †

1 Introduction

We consider a system of n synchronous tasks $\{\tau_1, \tau_2, \ldots, \tau_n\}$ to be scheduled on one processor according to a preemptive fixed-priority task-level scheduling policy. Without loss of generality, we consider that τ_i has a higher priority than τ_j for $i < j$. We denote by $hp(i)$ the set of tasks' indexes with higher priority than τ_i. By synchronous tasks we understand that all tasks are released simultaneously the first time at $t = 0$.

Each task τ_i generates an infinite number of successive jobs $\tau_{i,j}$, 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 a generalized sporadic task [1] and it is represented by a probabilistic worst case execution time (pWCET) denoted by $\mathcal{C}_i$[1] and by a probabilistic minimal inter-arrival time (pMIT) denoted by $\mathcal{T}_i$.

The probabilistic execution time (pET) of a job of a task describes the probability that the execution time of the job is equal to a given value. A safe pWCET $\mathcal{C}_i$ is an upper bound on the pETs $\mathcal{C}_i^j$, $\forall j$ and it may be described by the relation $\succeq$ as $\mathcal{C}_i \succeq \mathcal{C}_i^j$, $\forall j$. Graphically this means that the CDF of $\mathcal{C}_i$ stays under the CDF of $\mathcal{C}_i^j$, $\forall j$.

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. The probabilistic inter-arrival time (pIT) of a job of a task describes the probability that the job's arrival time occurs at a given value. A safe pMIT $\mathcal{T}_i$ is a bound on the pITs $\mathcal{T}_i^j$, $\forall j$ and it may be described by the relation $\succeq$ as $\mathcal{T}_i^j \succeq \mathcal{T}_i$, $\forall j$. Graphically this means that the CDF of $\mathcal{T}_i$ stays below the CDF of $\mathcal{T}_i^j$, $\forall j$.

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's deadline may also be represented by a random variable $\mathcal{D}_i$ which has the same distribution as its pMIT, $\mathcal{T}_i$. Alternatively, we can consider the deadline described by a distribution different from the distribution of its pMIT if the system under consideration calls for such model, or the simpler case when the deadline of a task is given as one value. The latter case is probably the most frequent in practice, nevertheless we prefer to propose an analysis as

*dorin@isep.ipp.pt. CISTER Research Center Building. Rua Alfredo Alle, 535. Porto. Portugal.

†liliana.cucu@inria.fr. INRIA Paris-Rocquencourt. Domaine de Voluceau, BP 105. 78153, Le Chesnay. France.

[1]In this paper, we use calligraphic typeface to denote random variables.

general as possible and in the rest of the paper, we consider tasks with implicit deadlines, i.e., having the same distribution as the pMIT.

Our main result is the following:

Theorem 1 *We consider a task system of n tasks with τ_i described by probabilistic $\mathcal{C}_i$ and $\mathcal{T}_i, \forall i \in \{1, 2, \cdots, n\}$. The set is ordered according to the priorities of the tasks and the system is scheduled preemptively on a single processor. The response time distribution $\mathcal{R}_{i,1}$ of the first job of task τ_i (i.e. the job release at the critical instance) is greater than the response time distribution $\mathcal{R}_{i,j}$ of any j^{th} job of task τ_i, $\forall i \in \{1, 2, \cdots, n\}$.*

2 Probabilistic response time analysis

The probabilistic worst case response time (pWCRT) $\mathcal{R}_n$ of a task τ_n in the critical instance is computed by coalescing all the distributions $\mathcal{R}_n^{i,j}$ (called copies) resulted by iteratively solving the following equation:

$$\mathcal{R}_n^{i,j} = (\mathcal{R}_n^{i-1,head} \oplus (\mathcal{R}_n^{i-1,tail} \otimes \mathcal{C}_m^{pr})) \otimes \mathcal{P}_{pr} \tag{1}$$

where:

- $\mathcal{R}_n^{i,j}$ is the j^{th} copy of the response time distribution
- n is the index of the task under analysis;
- i is the current step of the iteration;
- j represents the index of the current value taken into consideration from the pMIT distribution of the preempting task;
- $\mathcal{R}_n^{i-1,head}$ is the part of the distribution that is not affected by the current preemption under consideration;
- $\mathcal{R}_n^{i-1,tail}$ is the part of the distribution that may be affected by the current preemption under consideration;
- m is the index of the higher priority task that is currently taken into account as a preempting task;
- $\mathcal{C}_m^{pr}$ is the execution time distribution of the currently preempting task;
- $\mathcal{P}_{pr}$ is a fake random variable used to scale the j^{th} copy of the response time with the probability of the current value i from the pMIT distribution of the preempting task. This variable has one unique value equal to 0 and its associated probability is equal to the i^{th} probability in the pMIT distribution of the preempting job.

The iterations end when there are no more arrival values $v_{m,i}^j$ of any job i of any higher priority task τ_m that is smaller than any value of the response time distribution at the current step. A stopping condition may be explicitly placed in order to stop the analysis after a desired response time accuracy has been reached. For example, the analysis can be terminated once an accuracy of 10^{-9} has been reached for the response time. In our case, the analysis stops when new arrivals of the preempting tasks are beyond the deadline of the task under analysis, i.e., the type of analysis required for systems where jobs are aborted once they reach their deadline.

Once the jobs' response time distribution is computed, the Deadline Miss Probability is obtained by comparing the response time distribution with that of the deadline, as follows:

$$\mathcal{B}_i = \mathcal{R}_i \ominus \mathcal{D}_i = \mathcal{R}_i \oplus (-\mathcal{D}_i), \tag{2}$$

where the $\ominus$ operator indicates that the values of the distribution are negated.

Note that the analysis can handle any combination of probabilistic and deterministic parameters, and in the case that all parameters are deterministic the returned result is the same as the one provided by the worst case response time analysis in [2]. More details about the analysis can be found in [3].

References

[1] Mok, Al.(1983). *Fundamental Design Problems of Distributed Systems for the Hard-Real-Time Environment.* Massachusetts Institute of Technology.

[2] Joseph, M. and Pandya, P. K.(1986). *Finding response times in a real-time system.* The Computer Journal 29(5):390–395.

[3] Maxim, D. and Cucu-Grosjean, L.(2013). *Response Time Analysis for Fixed-Priority Tasks with Multiple Probabilistic Parameters.* Proceedings of the 34th IEEE Real-Time Systems Symposium.