Towards an analysis framework for tasks with probabilistic execution times and probabilistic inter-arrival times

Dorin Maxim
INRIA Nancy Grand Est
615 rue du Jardin Botanique
54600 Villers les Nancy
dorin.maxim@inria.fr

Liliana Cucu-Grosjean
INRIA Nancy Grand Est
615 rue du Jardin Botanique
54600 Villers les Nancy
liliana.cucu@inria.fr

***Abstract*—In this paper we investigate the problem of calculating the response time distribution for real-time tasks with probabilistic worst-case execution times, probabilistic inter-arrival times and probabilistic deadlines. We propose a definition for the probabilistic deadlines and a first discussion on the response time calculation.**

***Index Terms*—probabilistic real time, probabilistic execution time, probabilistic inter-arrival times, probabilistic deadlines**

I. Introduction

In embedded real-time systems there is a strong demand for new functionality that can only be met by using advanced high performance microprocessors. Building real-time systems with reliable timing behavior on such platforms represents a considerable challenge. Deterministic analysis for these platforms may lead to significant over-provision in the system architecture, effectively placing an unnecessary low limit on the amount of new functionality that can be included in a given system. An alternative approach is to use probabilistic analysis. Probabilistic analysis techniques rather than attempting to provide an absolute guarantee of meeting the deadlines, provide the probability of meeting the deadlines.

In this paper we investigate the problem of calculating the response time distribution for real-time tasks with probabilistic worst-case execution times, probabilistic inter-arrival times and probabilistic deadlines. The scheduling policy is a preemptive fixed-priority one and it is considered as given. The tasks are scheduled on one processor.

In this paper we propose a definition for the probabilistic deadlines and a first discussion on the response time calculation for a task. As future work we leave the proposition of a general formulation for n tasks and the associated proof.

II. Model and notations

A. Model

In this paper, we consider a task set of n synchronous tasks $\{\tau_1, \tau_2, \ldots, \tau_n\}$. Each task τ_i is characterized by three parameters $(\mathcal{C}_i, \mathcal{T}_i, \mathcal{D}_i)$ where $\mathcal{T}_i$ is the minimal inter-arrival time (commonly known as period), $\mathcal{D}_i$ the relative deadline (to be defined in Section II-B), and $\mathcal{C}_i$ the worst-case execution time. The parameters are described by random variables[1].

A random variable $\mathcal{X}_i$ describing a parameter of τ_i is assumed to have a known probability function (PF) $f_{\mathcal{X}_i}(\cdot)$ with $f_{\mathcal{X}_i}(x) = P(\mathcal{X}_i = x)$ giving the probability that the respective parameter of τ_i is equal to x. The values of $\mathcal{X}_i$ are assumed to belong to the interval $[x_i^{\min}, x_i^{\max}]$.

For instance the worst-case execution time $\mathcal{C}_i$ can be written as follows:

$$\mathcal{C}_i = \left(\begin{array}{cccc} C_i^0 = C_i^{\min} & C_i^1 & \cdots & C_i^{k_i} = C_i^{\max} \\ f_{\mathcal{C}_i}(C_i^{\min}) & f_{\mathcal{C}_i}(C_i^1) & \cdots & f_{\mathcal{C}_i}(C_i^{\max}) \end{array} \right), \quad (1)$$

where $\sum_{j=0}^{k_i} f_{\mathcal{C}_i}(C_i^j) = 1$.

For example for a task τ_i we might have a worst-case execution time $\mathcal{C}_i = \left(\begin{array}{ccc} 2 & 3 & 25 \\ 0.5 & 0.45 & 0.05 \end{array} \right)$; thus $f_{C_i}(2) = 0.5$, $f_{C_i}(3) = 0.45$ and $f_{C_i}(25) = 0.05$.

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, hence the execution time of a job does not depend on, and is not correlated with, the execution time of any previous job.

The set of tasks is scheduled according to a preemptive fixed-priority policy, i.e., all jobs of the same task have the same priority.

B. Deadline

In this section we provide an answer to the question "how do we define the deadline of a task with probabilistic periods?".

Given a task set with one task, τ, with probabilistic period described by the random variable $\mathcal{T} = \left(\begin{array}{cc} 3 & 2 \\ 0.7 & 0.3 \end{array} \right)$, we show that its probabilistic deadline should have the same distribution as the period of the same task.

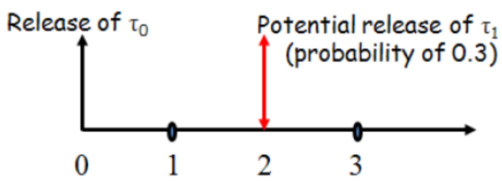
We analyze the possible scenarios and extract the corresponding probabilities.

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

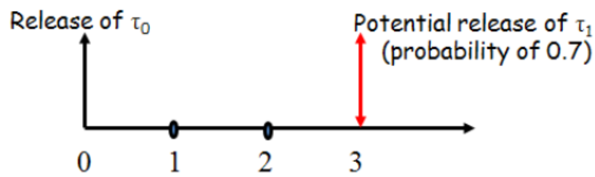
For the first job τ_0, released at $t = 0$, we have two possible scenarios:

Scenario 1: a new job τ_1 will be released at $t = 2$. This moment becomes the deadline of τ_0. The probability associated to this scenario is 0.3. This scenario is depicted in Figure 1a.

Scenario 2: if τ_1 does not arrive at $t = 2$ but it arrives at $t = 3$, then this is considered to be the deadline of τ_0. The probability associated to this scenario is 0.7. This scenario is depicted in Figure 1b.



(a) Scenario 1



(b) Scenario 2

Fig. 1: The two possible release scenarios of τ_1

Combining the two scenarios we obtain a distribution of the deadline of the first job equal to $\mathcal{D}_0 = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} = \mathcal{T}$, which was expected and somewhat obvious.

Let us now analyze what happens in the case of the second job of τ.

For the second job there are four release scenarios, two for each scenario of the previous job.

Scenario 1, i.e. τ_1 arrived at $t = 2$, has two possible continuations: τ_2 can arrive either at $t = 4$ or at $t = 5$, i.e. 2, respectively 3 units of time after the release of τ_1. Subtracting the two already passed units of time we obtain a relative deadline of $\begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$. *

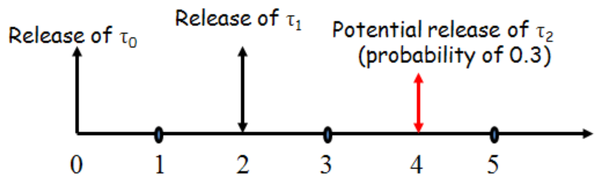
These two possibilities are depicted in Figure 2.

Scenario 2, i.e. τ_1 arrived at $t = 3$, has two possible continuations: τ_2 can arrive either at $t = 5$ or at $t = 6$, i.e. 2, respectively 3 units of after the release of τ_1. Subtracting the three already passed units of time we obtain a relative deadline of $\begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$. **

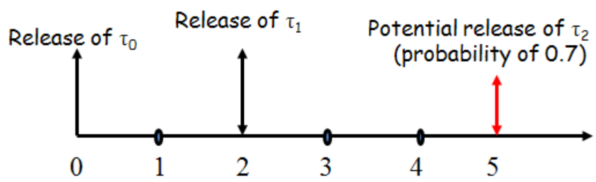
These two possibilities are depicted in Figure 3.

From * and ** we conclude that the relative deadline of τ_1 is $\mathcal{D}_1 = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} = \mathcal{T}$.

Continuing this reasoning, we obtain that the relative deadlines of all jobs of a task with probabilistic period have the same probability distribution as the period.

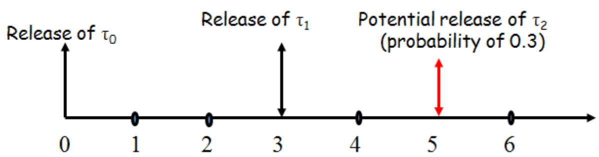


(a) τ_2 has a probability of 0.3 to be released at $t = 4$

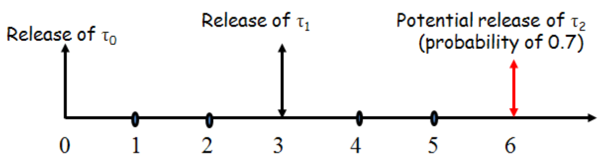


(b) τ_2 has a probability of 0.7 to be released at $t = 5$

Fig. 2: Scenario 1 continued with its two sub-scenarios



(a) τ_2 has a probability of 0.3 to be released at $t = 5$



(b) τ_2 has a probability of 0.7 to be released at $t = 6$

Fig. 3: Scenario 2 continued with its two sub-scenarios

III. Solution for a single task

A. *The case of a single task - the step by step approach*

Given a task $\tau = \left(\begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}, \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} \right)$, where $\begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}$ is the distribution of its probabilistic execution time and $\begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$ is the distribution of its probabilistic period, one has to compute the response time distributions of its jobs.

We also know about the task that its first job τ_1 is released at $t = 0$ and that its deadline is implicit, i.e., the release of a job determines the deadline of the previous job.

1) Response time and deadline miss probabilities: **For the first job** τ_0, which is released at $t = 0$, its response time has a distribution equal to the distribution of the execution time.

In order to obtain the probability of τ_0 missing its deadline, we analyse the two scenarios given by the deadline.

In the first scenario, when τ_1 is released at $t = 2$ we have two possibilities:

If its execution time is $C = 2$, then τ_0 reaches its deadline. The probability of this happening is $0.3 \times 0.8 = 0.24$, i.e., the probability of the job having an execution time of 2 and a deadline of 2.

If its execution time is $C = 3$, then τ_0 misses its deadline. The probability of this happening is $0.3 \times 0.2 = 0.06$, i.e., the probability of the job having an execution time of 3 and a deadline of 2.

In the second scenario, when the deadline is equal to 3, τ_0 reaches its deadline regardless if it has an execution time of 2 or of 3. This two sub-scenarios summed have a probability of $0.7 \times 0.8 + 0.7 \times 0.2 = 0.7$.

Combining the two scenarios, we obtain that τ_0 has a 0.06 probability of missing its deadline and a 0.94 probability of finishing before the next release, i.e. reaching its deadline.

For the second job there are the following possible scenarios:

Scenario 1: τ_1 **arrives at** $t = 2$**,** has a probability of 0.3 of happening. There are two possibilities:

a) τ_0 finishes execution at $t = 2$. This has a 0.8 probability of happening. In this case there is no backlog and there are once more two possible outcomes: τ_4 arrives at $t = 4$, in which case τ_1 reaches its deadline if it has an execution time of 2 or **it misses its deadline** if it has an execution time of 3. The probability of having an execution time of 3 is 0.2 which gives a probability of τ_1 missing its deadline equal to $0.3 \times 0.8 \times 0.3 \times 0.2 = 0.0144$, i.e., the multiplication of, respectively, the probability that τ_1 arrives at $t = 2$, the probability that τ_0 has an execution time of 2, the probability that τ_2 arrives at $t = 4$ and the probability that τ_1 has an execution time of 3.

b) τ_0 finishes execution at $t = 3$. There are here multiple possibilities:

If τ_1 has an execution time of 2 (0.8 probability) and τ_2 arrives at $t = 5$ (probability 0.7) then τ_1 finishes its execution before its deadline. The probability of this happening is $0.3 \times 0.8 \times 0.2 \times 0.7 = 0.0294$.

If τ_1 has an execution time of 2 (0.8 probability) and τ_2 arrives at $t = 4$ (probability 0.3) then τ_1 **misses its deadline**. The probability of this happening is $0.3 \times 0.8 \times 0.2 \times 0.3 = 0.0144$.

If τ_1 has an execution time of 3, then it misses its deadline no mater if τ_2 arrives at $t = 4$ or at $t = 5$. This has a probability of happening of $0.06 \times 0.2 = 0.012$

Summing up all the deadline miss probabilities obtained for Scenario 1, we get a partial probability of $0.0144 + 0.0144 + 0.012 = 0.0408$ that τ_1 misses its deadline.

Scenario 2: τ_1 **arrives at** $t = 3$. In this scenario there is no backlog, which means that everything happens as for the first release (at $t = 0$) just that the probabilities are multiplied by 0.7.

We get that τ_1 has a $0.06 \times 0.7 = 0.042$ probability of missing its deadline and a $0.7 \times 0.94 = 0.658$ probability of reaching it.

Combining the two scenarios we obtain that τ_1 has a total probability of $0.0408 + 0.042 = 0.0828$ of missing its deadline.

2) Discussion: The value obtained for the deadline miss probability of a job is an upper bound and not and exact value, since it is the summation of two probabilities, the ones resulted in the two scenarios.

One could argue towards not summing the two scenarios since this gives pessimistic results. Another option would be to do the average of the resulting scenarios. this is not a valid option though, since it can produce optimistic results. For example, by doing the average of the two scenarios obtained in the previous example, we would get a value of $0,0414$ which is less than the probability computed for the first scenario (0.0444). One could argue that this second value is the real and the only value one should take into account.

Keeping the probabilities for each possible scenario separated has its own drawbacks. The first and obvious one is the fact that the number of scenarios could be very big, and not to mention storing them, it can be very complex to work with so many values.

Another drawback is, one could argue, that each scenario in itself can be optimistic. For example, for the second job (τ_1), second scenario, one could argue that the respective probabilities are exactly the ones obtained for the first job (τ_0), and not multiplied with 0.7 as we did in computing the probabilities of the scenario. This would mean that its deadline miss probability in the scenario would be 0.06, which is greater than 0.042. Fortunately, the upper bound (summation of all the scenarios) covers the value 0.06, so it is a safe (pessimistic) bound.

B. The case of a single task - The analytical approach

We recall here the task system that is under analysis. This task system has only one task characterized by:
$\tau = \left(\begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}, \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} \right)$,
where $\begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}$ is the distribution of its probabilistic execution time and $\begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$ is the distribution of its probabilistic period.

We also know about this task that its first release occurs at $t = 0$, i.e., $r_0 = 0$. This, in turn, implies that there is no backlog at the moment of its arrival, which we denote with $\mathcal{B}_0 = 0$, backlog at the arrival of job τ_0.

The release distributions of subsequent jobs can be computed using the formula $r_i = r_{i-1} \otimes \mathcal{T}$. We would have $r_1 = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} = r_0 \otimes \mathcal{T}$ and $r_2 = \begin{pmatrix} 6 & 5 & 4 \\ 0.39 & 0.42 & 0.09 \end{pmatrix} = r_1 \otimes \mathcal{T}$.

The response time of the first job is $\mathcal{R}_0 = \begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}$.

Knowing that the deadline of τ_0 is $\mathcal{D} = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$, we can compute the backlog at the release of the next instance of τ and also the deadline miss probability of τ_0, by using an operation similar to the convolution, which, instead of doing addition of the values, does subtraction. The probabilities are multiplied the same as in a convolution. We call this operation a "subtracting convolution".

For example, the backlog at the release of τ_2 can be computed as:

$\mathcal{B}_1 = \mathcal{D} \ominus \mathcal{R}_0 = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} \ominus \begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0.56 & 0.24 & 0.14 & 0.06 \end{pmatrix}$.

By gathering all the non-negative values in zero, we obtain $\mathcal{B}_1 = \begin{pmatrix} 0 & -1 \\ 0.94 & 0.06 \end{pmatrix}$. This random variable implies that there is a 94% chance that there will be 0 backlog at the arrival of τ_1 and a 6% chance that 1 more unit of time is necessary for τ_0 to finish its execution.

In the distribution of the backlog, the deadline miss probability of τ_0 is the (summed) probability corresponding to the negative values of the distribution, in this case 0.06, the probability corresponding to -1.

By using the backlog and the execution time distributions, the response time distribution of the next instance of the task can be computed using the formula
$\mathcal{R}_i = |\mathcal{B}_i| \otimes \mathcal{C}$.

The response time of τ_0 is, indeed, $\mathcal{R}_0 = |\mathcal{B}_0| \otimes \mathcal{C} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix} = \begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix}$.

For τ_1 we have that $\mathcal{R}_1 = |\mathcal{B}_1| \otimes \mathcal{C} = |\begin{pmatrix} 0 & -1 \\ 0.94 & 0.06 \end{pmatrix}| \otimes \begin{pmatrix} 2 & 3 \\ 0.8 & 0.2 \end{pmatrix} = \begin{pmatrix} 2 & 3 & 4 \\ 0.752 & 0.236 & 0.012 \end{pmatrix}$.

The response time distribution can be compared with the relative deadline, $\mathcal{D} = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix}$, in order to see which are the combinations that would lead to a deadline miss, i.e., the situations when the response time is (strictly) larger than the deadline. Those situations are given by the following combinations:

- when the response time is 3 and the deadline is 2, which has a probability of $0.3 \times 0.236 = 0.0708$;
- when the response time is 4 and the deadline is 2, which has a probability of $0.7 \times 0.012 = 0.0084$;
- and also when the response time is 4 and the deadline is 3, which has a probability of $0.012 \times 0.3 = 0.0036$.

Summing up the above probabilities, we obtain that τ_1 has a probability to miss its deadline equal to $0.0708 + 0.0084 + 0.0036 = 0.0828$, which is exactly what we got from the step-by-step verification in the previous section.

By computing the backlog with the formula $\mathcal{B}_i = \mathcal{D} \ominus \mathcal{R}_{i-1}$ we obtain that $\mathcal{B}_2 = \mathcal{D} \ominus \mathcal{R}_1 = \begin{pmatrix} 3 & 2 \\ 0.7 & 0.3 \end{pmatrix} \ominus \begin{pmatrix} 2 & 3 & 4 \\ 0.752 & 0.236 & 0.012 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & -1 & -1 & -2 \\ 0.5264 & 0.2256 & 0.1652 & 0.0708 & 0.0084 & 0.0036 \end{pmatrix}$

After gathering all the non-negative values in zero, we obtain that the backlog at the release of τ_2 is $\mathcal{B}_2 = \begin{pmatrix} 0 & -1 & -2 \\ 0.9172 & 0.0792 & 0.0036 \end{pmatrix}$, where the probabilities of the negative values give the deadline miss probability DMP of τ_1, i.e., $DMP_1 = 0.0792 + 0.0036 = 0.0828$, which is, once more, exactly the value obtained trough the step-by-step verification in the previous section.

In conclusion, we have that the backlog at the release of a job can be computed using the formula

$$\mathcal{B}_i = \mathcal{D} \ominus \mathcal{R}_{i-1}, \text{ with } \mathcal{B}_0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

and the response time can be computed using the formula

$\mathcal{R}_i = |\mathcal{B}_i| \otimes \mathcal{C}$, where $|\mathcal{B}_i|$ is the modulo of the backlog, meaning that each value of the backlog is taken in the positive.

This way we may compute for example that the response time of τ_2 is
$\mathcal{R}_2 = |\mathcal{B}_2| \otimes \mathcal{C} = \begin{pmatrix} 2 & 3 & 4 & 5 \\ 0.73376 & 0.2468 & 0.01872 & 0.00072 \end{pmatrix}$

We can compute also the backlog at the release of τ_3, which is

$$\mathcal{B}_3 = \mathcal{D} \ominus \mathcal{R}_2 = \begin{pmatrix} 0 & -1 & -2 & -3 \\ 0.90652 & 0.087144 & 0.00612 & 0.000216 \end{pmatrix},$$

which means that τ_2 has a probability of missing its deadline of $0.087144 + 0.00612 + 0.000216 = 0.09348$ and a probability of 0.90652 to finish execution in time.

IV. Related work

There has been a significant work devoted to probabilistic real-time analysis the last years. However, and to our best knowledge, there are no comparable results so far. In fact, there are a few related works which consider special scheduling models providing isolation between tasks [1], or assuming a known (a priori) maximum number of arrivals, thus introducing an unnecessary level of pessimism in the analysis [2], [3].

For the problem where the worst-case execution times are probabilistic the approach in [4] is the most general. However, to our best knowledge their approach is not extended to the case of random inter-arrival times.

The closest work to our contribution is described in [5], but the results are only valid for particular cases of random variables.

V. Conclusion

In this paper we propose a definition for the probabilistic deadlines and a first discussion on the response time calculation for a task. As future work we leave the proposition of a general formulation for n tasks and the associated proof.

References

[1] L. Abeni and Buttazzo, "QoS guarantee using probabilistic deadlines," in *IEEE Euromicro Conference on Real-Time Systems (ECRTS99)*, 1999.

[2] A. Burns, G. Bernat, and I. Broster, "A probabilistic framework for schedulability analysis," in *Third International Embedded Software Conference (EMSOFT03)*, 2003, pp. 1–15.

[3] I. Broster and A. Burns, "Applying random arrival models to fixed priority analysis," in *Proceedings of the Work-In-Progress of the 25th of the IEEE Real-Time Systems Symposium (RTSS04)*, 2004.

[4] J. Díaz, D. Garcia, K. Kim, C. Lee, L. Bello, L. J.M., and O. Mirabella, "Stochastic analysis of periodic real-time systems," in *23rd of the IEEE Real-Time Systems Symposium (RTSS02)*, 2002, pp. 289–300.

[5] J. Lehoczky, "Real-time queueing theory," in *10th of the IEEE Real-Time Systems Symposium (RTSS96)*, 1996, pp. 186–195.