Response Time Stochastic Analysis for Fixed-Priority Stable Real-Time Systems

Kevin Zagalo, Yasmina Abdeddaïm, Avner Bar-Hen, Liliana Cucu-Grosjean

HAL Id: hal-03797980
https://hal.inria.fr/hal-03797980v2

Response Time Stochastic Analysis for Fixed-Priority Stable Real-time Systems

Kevin Zagalo[1], Yasmina Abdeddaïm[1,2], Avner Bar-Hen[1,3], and
Liliana Cucu-Grosjean[1]

[1]*Inria Paris*
[2]*Université Gustave Eiffel, LIGM, CNRS*
[3]*Cnam Paris*

Abstract

In this paper, we prove that a mean system utilization smaller than
one is a necessary condition for the feasibility of real-time systems. Such
systems are defined as *stable*. Stable systems have two distinct states: a
transient state, followed by a steady-state where the same distribution of
response times is repeated infinitely for each task. We prove that the Liu
and Layland theorem holds for stable probabilistic real-time systems with
implicit deadlines, we provide an analytical approximation of response
times for each of those two states and a bound of the instant when a
real-time system becomes steady.

1 Introduction

Probabilistic methods applied to real-time systems are motivated by the fact
that worst-case execution times and minimal inter-arrival times are often rare
events. The analysis induced by these methods consists in quantifying the prob-
ability that time requirements may not be satisfied. Their purpose is to soften
real-time guarantees leading to an overestimated use of processors, or as we say,
provide a pessimistic analysis [9]. Associating probabilities to worst-case values
and to other possible cases may reduce the pessimism of a real-time analysis.
Often, execution times have associated probabilities, but inter-arrival times may
also be considered as probabilistic. In fact, many real-time systems have arrival
times subject to noise and other interference from either operating systems *etc.*
which can be modeled with probability distributions [16]. We call *probabilis-
tic real-time systems* the real-time systems which have *probabilistic* parameters
such as execution times and/or inter-arrival times.

Utilization-based schedulability conditions for single-core fixed-priority pre-
emptive scheduling policies are well known [6]. The seminal work of Liu and
Layland [20] introduces a sufficient condition for the feasibility of a real-time

system using its *maximal utilization*. Nevertheless, a real-time system not satisfying this sufficient condition remains *schedulable* with a given probability (see (1) in Section 3). Moreover, while probabilistic methods have been focused towards fitting in this sufficient condition by providing less pessimistic analyses, their domain of feasibility needs to be defined as well. In this paper, we build necessary feasibility conditions for fixed-priority scheduling policies based on the mean utilization of the real-time system. We demonstrate that a mean utilization smaller than 1 is mandatory for *response times* to be finite, *c.f.* Propositions 4 and 5. We call systems with a mean utilization smaller than 1, *stable* real-time systems.

Response times depend on properties of real-time systems such as the scheduling policy, the preemptiveness, *etc.*. Our motivation is to exploit those properties leading response times to the domain of a certain probability distribution. In this work, we emphasize the inverse Gaussian distribution as the appropriate distribution for an approximation of response times in the context of fixed-priority scheduling policies, using asymptotic results of queueing theory, *c.f.* Propositions 8. We propose two different approximations, a worst-case approximation before the system is in its *steady-state* (*c.f.* Proposition 6, see Definition 10) and another one when the system is *steady*, *c.f.* Propositions 1 and 7. We prove in Proposition 2 that a real-time system is in its steady-state at the first idle time.

The contributions of this paper are:

- a necessary condition for the feasibility of probabilistic real-time systems with implicit deadlines under preemptive fixed-priority scheduling policies,

- an approximation of the demand of probabilistic real-time systems,

- a first proof that the Liu and Layland theorem [20] stands only for stable probabilistic real-time systems,

- a first proof that a real-time system is steady after the first idle time and an analytical formulation of its distribution,

- an analytical formulation of response time distributions of tasks before and after the first idle time,

- numerical evidence and comparison against simulated results and other existing estimations.

The paper is organized as follows. We present in Section 2 relevant existing results and in Section 3 the main definitions of the considered model. We introduce in Section 4 the *Loynes theorem* and the *heavy-traffic theorem* which are the central theorems used this work. These theorems provide analytical expressions of the *steadiness* of the system, through the stochastic analysis of *backlogs*. In Section 5 we provide the time demand analysis associated to the probabilistic real-time systems. Moreover, we demonstrate the importance of the first idle time within any feasibility analysis. We detail the experimental

results associated to the proposed contribution in Section 6 and we discuss future work opened by our current contribution in Section 7.

2   Related work

Probabilistic methods for the analysis of response times have many applications in real-time systems [7]. Two main directions have been explored: static methods for the exact computation and approximation of response time distributions [8, 17, 23, 25] *a priori*, and the measurement-based application of the extreme value theory (EVT) [21,22] approximating the distribution of the maximum values of response times *a posteriori*. Often, probabilities are considered for execution times and few papers consider probabilistic inter-arrival times and deadlines [11, 19, 25]. Moreover, the method introduced in [8, 17] requires a large amount of convolutions which have a high space and time complexity, and the analysis provided by Lehoczky [19] is not suited to express response time distributions.

The contributions of this paper are based on queueing theory results [1, 4, 31, 32]. To the best of our knowledge, no result from the queueing theory is focused on general execution times and inter-arrival times, multi-class clients (*i.e.* different tasks) and the quantization of deadline misses of such systems. The only results based on queueing theory for real-time systems have been published within the thread of papers related to [19], where the author approximates the number of simultaneously activated jobs by a *reflected at the origin Brownian motion*. However, the author makes strong hypotheses restricting the model. The exponential distribution of inter-arrivals and execution times suggested in [19] is a strong hypothesis. Furthermore, his model has another important limitation as it considers systems of jobs of only one task, which does not allow a response time analysis.

3   Model

We consider a probabilistic real-time system Γ composed of a finite number of tasks ordered in the decreasing priority order, *i.e.* a task τ_i has a higher priority than τ_j if and only if $i < j$. The tasks are scheduled on a single-core processor, using a fixed-priority preemptive policy.

3.1   Probabilistic real-time tasks

A probabilistic task τ_k is characterized by:

1. its *execution time* $C_k > 0$, with a distribution function F_k, and finite mean $m_k \triangleq \mathbf{E}[C_k]$ and non-negative variance $s_k^2 \triangleq \mathbf{E}[(C_k - m_k)^2]$,

2. its *inter-arrival time* $T_k > 0$ between two consecutive instances of τ_k, with a distribution function G_k of rate $\lambda_k \triangleq 1/\mathbf{E}[T_k]$,

3. its *relative implicit deadline* $D_k > 0$, with the same distribution function G_k as its inter-arrival time.

Without loss of generality, we consider the functions F_k and G_k discrete. Also note that all the parameters written in capital letters are probabilistic.

The l-th instance of the task τ_k is called a *job* and we denote it $\tau_{k,l}$. Its execution time is $C_{k,l}$, with distribution function F_k. $T_{k,l}$ is the *inter-arrival time* between the arrival times of the jobs $\tau_{k,l-1}$ and $\tau_{k,l}$, with distribution function G_k, *i.e.* identically distributed as T_k. We suppose all the rates of arrival λ_k distinct.

Due to the fixed-priority policy, the response times of the task τ_k depend only those of higher priority tasks. We call job of *level k* any job of a task of higher or equal priority than τ_k, *i.e.* any job $\tau_{i,j}, 1 \leq i \leq k, j \in \mathbb{N}$. Let $u_i \triangleq \mathbf{E}[C_i]/\mathbf{E}[T_i] = \lambda_i m_i$ be the *mean utilization* of the task τ_i, $\bar{u}_k \triangleq \sum_{i=1}^{k} u_i$ be the *k-level mean utilization* of Γ, $\bar{u} \triangleq \sum_{\tau_k \in \Gamma} u_k$ the *total mean utilization* of Γ, and, the *maximum utilization* of level k, $u_k^{max} \triangleq \sum_{i=1}^{k} c_i^{max}/t_i^{min}$, where c_i^{max} is the maximum possible value of C_i and t_i^{min} is the minimum possible value of T_i, and the *total maximum utilization* $u^{max} \triangleq \sum_{\tau_k \in \Gamma} c_k^{max}/t_k^{min}$.

Definition 1. *Let Γ be a real-time system with total mean utilization $\bar{u}$. Γ is said stable if and only if $\bar{u} < 1$.*

The *response time $R_{k,l}$* of a job $\tau_{k,l}$ is the elapsed time between its activation (or arrival) $A_{k,l} \triangleq \sum_{j=1}^{l} T_{k,j}$ and the end of its execution. We consider implicit deadlines, *i.e.* the absolute deadline of the job $\tau_{k,l}$ is $A_{k,l+1}$ and $A_{k,l+1} - A_{k,l} = T_{k,l+1} \sim D_k$ is its relative deadline with distribution function G_k. For the sake of comprehension, we refer to the relative deadline of τ_k with the variable D_k and to its inter-arrival time T_k, even though they have the same distribution. At each arrival of jobs of a same task, previous jobs miss their deadline and are discarded.

Remark 1. *Such discarding policy is the reason why response times of jobs of a same task are independent: response times are functions of their associated execution times and only the higher priority jobs execution times. As described in (13), discarding jobs makes their backlog disappear from the computation of the response time of the next job of the same task. However, in the response time analysis in Section 5 we build response time approximations regardless of this discarding policy which makes them upper-bounds of response times, see (14).*

Example 1 (Utilization and probabilistic deadlines)**.** *Let us consider the task set $\{\tau_1, \tau_2\}$, with $\mathbf{P}(C_1 = 1) = 1$, $\mathbf{P}(T_1 = 2) = 1$, $\mathbf{P}(C_2 = 1) = 1/2$, $\mathbf{P}(C_2 = 2) = 1/2$, $\mathbf{P}(T_2 = 3.1) = 1/2$, $\mathbf{P}(T_2 = 4) = 1/2$. Then $\bar{u} = 1/2 + ((1 + 2)/2)/((3.1 + 4/2) \approx 0.92$, $u^{max} = 1/2 + 2/3.1 \approx 1.16$. Furthermore, because deadlines are implicit, if $T_{2,2} = 3.1$, the deadline of the job $\tau_{2,1}$ is 3.1, if $T_{2,2} = 4$ its deadline is 4. The deadline of any job $\tau_{1,j}$ is 2 because $T_{1,j}$ is always 2 for all $j \in \mathbb{N}$.*

A job $\tau_{k,l}$ respects its deadline if its response time $R_{k,l}$ is smaller or equal to its deadline, *i.e.* $R_{k,l} \leq D_k$. In order to be *schedulable*, the *worst-case deadline failure probability* [29,34] of a task τ_k should be smaller than its *permitted failure rate* $\alpha_k \in (0,1)$ [12], *i.e.*

$$\mathbf{P}\left(\sup_{l \in \mathbb{N}} R_{k,l} > D_k\right) = \int \mathbf{P}\left(\sup_{l \in \mathbb{N}} R_{k,l} > t\right) dG_k(t) \leq \alpha_k \qquad (1)$$

where α_k is a parameter given *a priori*.

Definition 2. *Let Γ be a real-time system. Γ is said feasible in the fixed-priority domain if and only if there exists at least one fixed-priority preemptive scheduling policy such that all tasks $\tau_k \in \Gamma$ satisfy* (1).

Remark 2. *For all tasks $\tau_k \in \Gamma$ and all jobs $\tau_{k,l}, l \in \mathbb{N}$ the execution times $C_{k,l}$ are independent. We discuss in Section 7 how the model may be extended in the dependent case, i.e. when the execution times $C_{k,l}, \tau_k \in \Gamma, l \in \mathbb{N}$ have a dependence structure.*

The inter-arrival times $T_{k,l}, l \in \mathbb{N}$ of a given task τ_k are independent, however inter-arrival times of two different tasks do not need to be independent for the provided results.

For a given job $\tau_{k,l}$, its execution time $C_{k,l}$ and the inter-arrival time $T_{k,l}$ are not necessarily independent, meaning that the analysis we provide holds if execution times are correlated to the time separating a job to the previous instance of the same task.

In the rest of this paper, we refer to *probabilistic real-time systems* as simply *real-time systems* when there is no ambiguity.

3.2   Elements of probability

Let us define some elements of probability that we use in this work. Let X and Y be two non-negative variables.

Definition 3 (Distribution function, probability function and distribution)**.** *The distribution function of X is the function defined by $F_X(x) = \mathbf{P}(X \leq x)$. The probability function of X is the function $p_X(x) = (dF_X/dx)(x)$. We denote the distribution of X by $dF_X(x) = p_X(x)dx$.*

Definition 4 (Identically distributed)**.** *X and Y are identically distributed and we write $X \sim Y$, or $X \sim dF_Y$, if and only if $F_X(x) = F_Y(x)$ for all $x > 0$.*

Definition 5 (Independence)**.** *X and Y are independent if and only if $\mathbf{P}(X \leq x, Y \leq y) = F_X(x) \cdot F_Y(y)$ for all $x, y > 0$.*

Definition 6 (Conditional distribution)**.** *The conditional distribution function of X given $Y = y$ is written*

$$\mathbf{P}(X \leq x \mid Y = y) = \frac{1}{p_Y(y)} \cdot \frac{d}{dy}\mathbf{P}(X \leq x, Y \leq y)$$

for all $x, y > 0$, and is such that

$$\mathbf{P}(X \leq x) = \int \mathbf{P}(X \leq x|Y = y)dF_Y(y) \qquad (2)$$

for all $x > 0$.

Definition 7 (Convolutions). *Let X and Y be independent and $Z = X + Y$. Then the distribution function of Z is the convolution $F_Z = F_X * F_Y$ defined by*

$$F_Z(z) = \int F_X(z - y)dF_Y(y) = \int F_Y(z - x)dF_X(x)$$

Definition 8 (Brownian motions). *A standard Brownian motion is a process B such that $B(0) = 0$ and with independent increments $B(t) - B(s)$ of distribution function $\Phi\left(\cdot/\sqrt{t - s}\right)$ for any $0 \leq s < t$, where*

$$d\Phi(x) = \exp\left(-x^2/2\right)/\sqrt{2\pi}dx$$

is the standard normal distribution.

A Brownian motion of drift u and deviation $v > 0$ is a process W such that there exists a standard Brownian motion B such that

$$W(t) = ut + vB(t) \qquad (3)$$

Definition 9 (Inverse Gaussian distribution). *Let $\theta > 0$ and $\gamma > 0$. The inverse Gaussian distribution of parameters (θ, γ) has a probability function ψ defined by*

$$\psi(t; \theta, \gamma) = \sqrt{\frac{\gamma}{2\pi t^3}} \exp\left(-\frac{\gamma(t - \theta)^2}{2\theta^2 t}\right) \qquad (4)$$

Remark 3. *Note that deterministic[1] real-time systems form a subclass of probabilistic real-time systems, as the distributions of the parameters of Γ can describe a unique value if needed. Hence, we propose results covering also deterministic parameters, would they be execution times, inter-arrival times or deadlines. For instance, if the execution time C_k of a task $\tau_k \in \Gamma$ is deterministic and equal to 5, then its distribution function is $F_k(x) = \mathbf{1}_{[5,\infty)}(x)$, where*

$$\mathbf{1}_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases}$$

3.3 Backlog stochastic process

Let us define the following stochastic processes:

1. $N_k(t) \triangleq \sum_{l=1}^{\infty} \mathbf{1}_{[0,t]}(A_{k,l})$ as the number of jobs of τ_k released before $t > 0$, of mean $\mathbf{E}[N_k(t)] = \lambda_k t$ [1, (1.1.22)]. N_k is right-continuous.

---

[1] A real-time system is deterministic if all task parameters are described by unique values.

2. the k-level demand $W_k(t)$ as the accumulation of the execution times required by jobs of priority higher or equal than τ_k, regardless of potential deadline misses, released before the instant t to complete,

$$W_k(t) \triangleq \sum_{i=1}^{k} \sum_{j=1}^{N_i(t)} C_{i,j}$$

of mean $\mathbf{E}[W_k(t)] = \sum_{i=1}^{k} \mathbf{E}[N_i(t)]\mathbf{E}[C_i] = \bar{u}_k t$, see [14, (6.53)]. W_k is right-continuous.

3. the k-level backlog $\beta_k(t)$ as the remaining demand at $t \geq 0$, defined by $\beta_k(0) \geq 0$ and

$$\beta_k(t) \triangleq \beta_k(0) + W_k(t-) - \int_0^t \mathbf{1}_{\{\beta_k(s)>0\}} ds \qquad (5)$$

where $W_k(t-) = \lim_{\varepsilon \to 0, \varepsilon > 0} W_k(t-\varepsilon)$ is the left limit of W_k at the instant t, and $\int_0^t \mathbf{1}_{\{\beta_k(s)>0\}} ds$ is the total busy time of level k [18, p. 2] before $t > 0$.

Example 2 (Backlog). *Consider the task set of Example 1. Suppose both tasks τ_1 and τ_2 are activated at time $t = 0$ and $T_{2,2} = 3.1$, $C_{2,1} = 2$, $C_{2,2} = 1$. Then, $\beta_2(0) = 0$,*

$$
\begin{aligned}
\beta_2(2) &= \beta_2(0) + W_2(2-) - \int_0^2 \mathbf{1}_{\{\beta_2(s)>0\}} ds \\
&= 0 + (1+2) - 2 = 1 \\
\beta_2(3) &= \beta_2(0) + W_2(3-) - \int_0^3 \mathbf{1}_{\{\beta_2(s)>0\}} ds \\
&= 0 + (1+2) - 3 = 0 \\
\beta_2(3.1) &= \beta_2(0) + W_2(3.1-) - \int_0^{3.1} \mathbf{1}_{\{\beta_2(s)>0\}} ds \\
&= 0 + (1+2) - 3 = 0 \\
\beta_2(3.2) &= \beta_2(0) + W_2(3.2-) - \int_0^{3.2} \mathbf{1}_{\{\beta_2(s)>0\}} ds \\
&= 0 + (1+2+1) - 3.1 = 0.9
\end{aligned}
$$

At the instant $t = 3.2$, the backlog of level 2 is 0.9.

The process β_k describes the remaining demand without considering deadline misses, *i.e.* while jobs are discarded their demand remains in the backlog analysis. Thus, the process β_k is an upper-bound of the blocking time, see (14), *i.e.* the response time analysis that we provide in Section 5 is pessimistic as defined in [9].

In stable real-time systems, there exist two main states of the backlog [8, Section 4.2].

Definition 10 (Transient and steady states). *For a stable real-time system Γ, the steady-state backlog is defined by*

$$\tilde{\beta}_k \triangleq \lim_{t \to \infty} \beta_k(t)$$

We say that Γ is steady at the instant $t > 0$ when its backlog is in steady-state, i.e. $\beta_k(t) \sim \tilde{\beta}_k$, and transient before β_k reaches its steady-state.
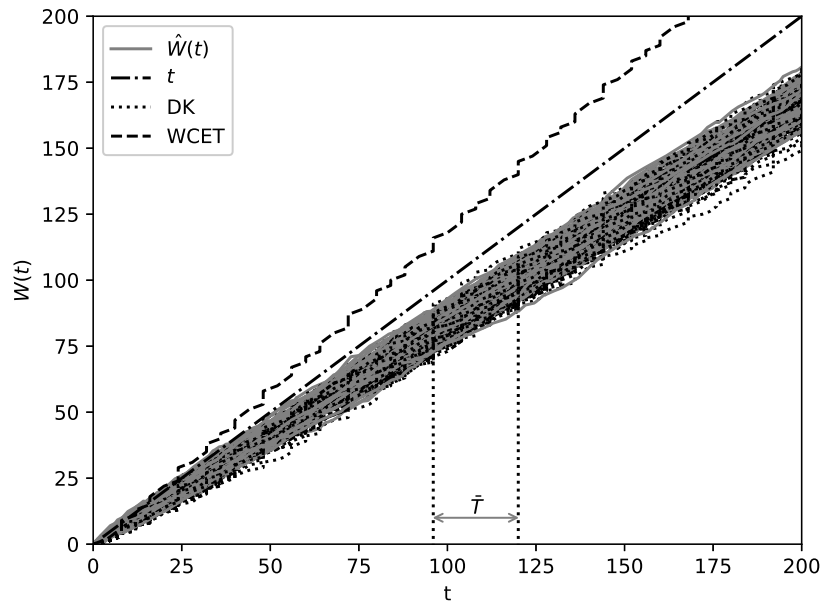
Figure 1: Level 3 demand of $1\,000$ instances of the DK model system, the heavy-traffic demand process and the classical deterministic worst-case analysis (WCET) considering only the maximal execution time for each task, for $\bar{u} = 0.838$ and $u^{max} = 1.208$ and hyper-period $\bar{T}$, for Γ defined in Table 1.

In the following sections, we define the domain of this steadiness, provide an expression of the instant when the system goes from transient to steady, and prove that this time instant exists and is finite under some conditions.

4 Heavy-traffic theory

In this section, we present an analytical approximation of the demand of probabilistic real-time systems. Let us denote $\pi_k(x) \triangleq \mathbf{P}\left(\tilde{\beta}_k \leq x\right)$ the distribution function of the steady-state backlog of level k. We use background results of queueing theory presented in 4.1 and 4.2, provide the exact formulation of the steady-state backlog distribution π_k in 4.3 and illustrate this result in the deterministic inter-arrival case already studied by Diaz *et al.* [8] in Section 4.4.

4.1 The Loynes theorem

As the demand and backlog processes W_k and β_k are well studied in queueing theory [1, 4, 15], we provide the formula of the steady-state of the system, by adapting the *Loynes theorem* [1, Theorem 2.1.1].

Theorem 1 (Loynes theorem)**.** *Let* $A_l^k = \inf\{t > 0 : \sum_{i=1}^k N_i(t) = l\}$ *be the activation time of the l-th job of level* k, *and* I_l^k *the index of the task of the l-th job of level* k. *If* $\bar{u}_k < 1$, *then the steady-state backlog* $\tilde{\beta}_k$ *exists, is finite and we obtain*

$$\tilde{\beta}_k = \sup_{n \geq 0} \left(\sum_{l=1}^n C_{I_l^k,l} - (A_{l+1}^k - A_l^k) \right)^+ \tag{6}$$

where $x^+ = \max(0, x)$, *c.f.* *[1, Property 2.2.1]. In addition, there is an infinite number of idle times, c.f. [1, Property 2.2.5]. If* $\bar{u}_k = 1$, *then the existence of a finite steady-state* $\tilde{\beta}_k$ *is uncertain. If* $\bar{u}_k > 1$, *there exists a finite number of idle times of level k and no finite steady-state, c.f. [1, Property (2.2.2)], backlogs are always transient.*

Our goal is to find the distribution of the steady-state backlog $\tilde{\beta}_k$. However, given the generality of this model, we cannot provide an exact description of the process β_k. This is why we use the *heavy-traffic assumptions* to find an exact formula for the distribution of the steady-state backlog $\tilde{\beta}_k$ when the system utilization gets close to 1, *c.f.* Figure 2, *i.e.* we build a process $\hat{\beta}_k$ such that its steady-state is also $\tilde{\beta}_k$.

4.2 The heavy-traffic theorem

A first step in the approximation of the backlog process β_k is the approximation of the demand process W_k.

(a) $\bar{u} = 0.625, u^{max} = 0.833$

(b) $\bar{u} = 0.838, u^{max} = 1.208$

(c) $\bar{u} = 0.998, u^{max} = 1.508$
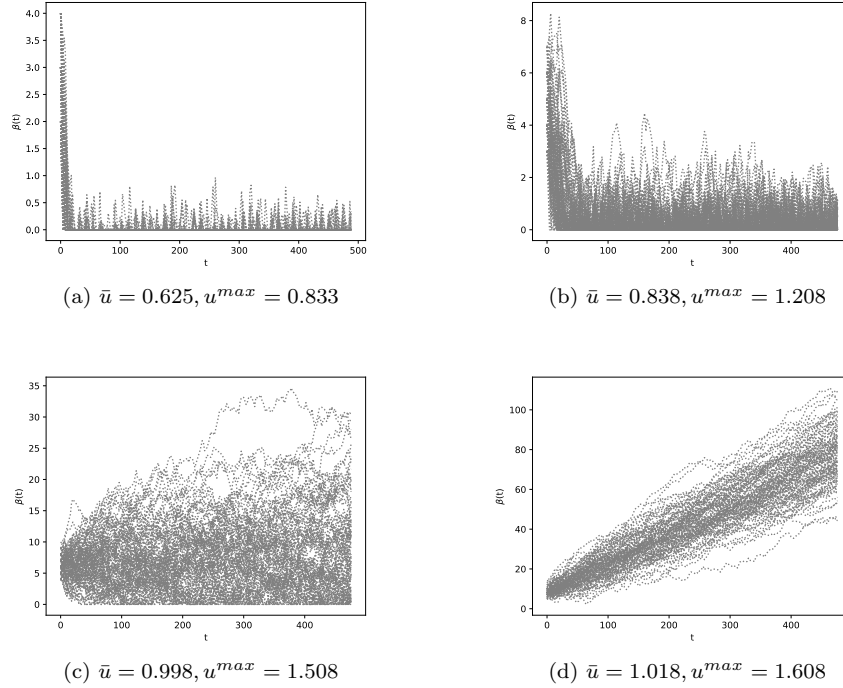
(d) $\bar{u} = 1.018, u^{max} = 1.608$

Figure 2: The heavy-traffic backlog process of systems with different mean utilization, initialized with $\beta(0) = \sum_{\tau_i \in \Gamma} C_i$.

Theorem 2 (Heavy traffic theorem, Section 6.5, [4]). *The level k heavy-traffic demand process*

$$\hat{W}_k(t) \quad \triangleq \quad \lim_{n \to \infty} \ W_k(nt)/\sqrt{n}$$

is a Brownian motion of drift $\bar{u}_k$ and deviation v_k, where $v_k^2 \ \triangleq \ \sum_{i=1}^{k} \ \lambda_i \cdot s_i^2$. See Figure 1.

In [19], the author uses the *heavy-traffic approximation* providing the distribution of the lateness of jobs in a system with exponential inter-arrival times. To illustrate the heavy-traffic approximation, one can think of water continuously flowing into a sink at a rate λ_i and the execution times as the rate $1/m_i$ the water leaves the sink. It is usually understood as true when the system is at full processor utilization, because the theorems of heavy-traffic theory are exact when $\bar{u} \to 1$. However we use it as a way to build an upper-bound of the backlog process. Indeed, the heavy-traffic assumption should be seen as a bound, or more specifically, a way to suppose that the system utilization is at its maximum (*i.e.* $\bar{u} = 1$), providing *upper-bounds* that are exact when the processor utilization at 100%. Figure 3b illustrates this upper-bound becoming exact in Figure 3c and Figure 3d.

10

In this section, we use the heavy-traffic assumption to find the steady-state backlog $\tilde{\beta}_k$ as an approximation upper-bounding the *blocking time* of the system in its steady-state, see (14). The approximation in Theorem 2 leads to the standard Brownian motions which is continuous. It means that instead of looking at the demand for a large amount of time, we consider a re-scaled version of the demand in order to build a good approximation. Theorem 2 can be written as

$$\hat{W}_k(t) = \bar{u}_k t + v_k B(t) \tag{7}$$

where B is a standard Brownian motion, as defined in (3).

4.3   Steady-state backlog distribution

In order to find the steady-state backlog $\tilde{\beta}_k$, we work with the heavy-traffic demand $\hat{W}_k$. We define the *heavy-traffic backlog process* by

$$\hat{\beta}_k(t) = \beta_k(0) + \hat{W}_k(t) - \int_0^t \mathbf{1}_{\{\hat{\beta}_k(s)>0\}} ds \tag{8}$$

This heavy-traffic backlog process $\hat{\beta}_k$ is a Brownian motion *reflected at the origin* of drift $\bar{u}_k - 1$ and deviation v_k, and its limit is also $\tilde{\beta}_k$, *c.f.* [4, Section 6.5]. Its stationary distribution π_k being the same for the initial backlog process β_k, *i.e.* $\lim_{t\to\infty}\beta_k(t) = \tilde{\beta}_k = \lim_{t\to\infty}\hat{\beta}_k(t)$ we get its explicit expression in the following proposition.

Proposition 1. *Let $\varepsilon > 0$ and $\bar{u} = 1 - \varepsilon$. Consider the parameter*

$$\eta_k \triangleq 2\left(1 - u_k\right) / \left(\lambda_k(\gamma_{a,k}^2 + \gamma_{e,k}^2)\right) \tag{9}$$

where $\gamma_{a,k}$ and $\gamma_{e,k}$ denote the coefficients of variation[2] of respectively inter-arrival and execution times of τ_k. Then, when $\varepsilon \to 0$, the distribution function π_k of the steady-state backlog $\tilde{\beta}_k$ is defined for $x > 0$ by

$$\pi_k(x) = \sum_{i=1}^{k} \left(1 - \exp\left(-\eta_i x\right)\right) \prod_{\substack{j=1 \\ j \neq i}}^{k} (1 - \eta_i/\eta_j)^{-1} \tag{10}$$

Proof. From [4, Section 6.5, p. 144], we know that when $\varepsilon \to 0$, $\tilde{\beta}_k - \tilde{\beta}_{k-1}$ is an exponential variable of parameter η_k. As execution times are independent, $\tilde{\beta}_i - \tilde{\beta}_{i-1}$ and $\tilde{\beta}_j - \tilde{\beta}_{j-1}$ are independent for any $i \neq j$. Furthermore, $\tilde{\beta}_k = \tilde{\beta}_1 + \sum_{i=2}^{k} \tilde{\beta}_i - \tilde{\beta}_{i-1}$, thus we know from [30] that the distribution of $\tilde{\beta}_k$ is

$$d\pi_k(x) = \prod_{i=1}^{k} \eta_i \sum_{i=1}^{k} \frac{\exp\left(-\eta_i x\right)}{\prod_{j=1, j\neq i}^{k}(\eta_j - \eta_i)} dx$$

for $x > 0$, which is the convolution of k exponential distributions of parameters $\eta_1, \ldots, \eta_k$. See Figure 3 for an illustration of the probability function of π_k. $\qquad\square$

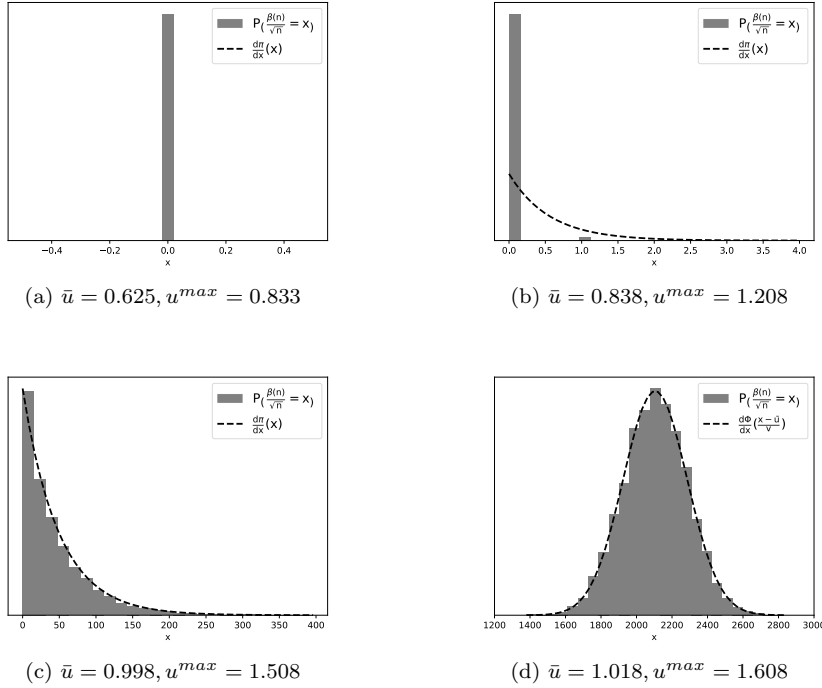[2]The ratio of the standard deviation and the mean.

Figure 3: Steady-state backlog simulations with different mean utilizations in the DK model. In grey the histogram of simulations of $\beta(n)/\sqrt{n}$ for $n = 10000$, in black the probability function of $\tilde{\beta}$.

It is shown in [4, Remark 6.17, p. 148], that the error of the estimation stated in (8) is

$$\sup_{s \in [0,t]} |\beta_k(s) - \hat{\beta}_k(s)| = \mathcal{O}\left((t \log \log t)^{1/4}(\log t)^{1/2}\right) \tag{11}$$

Note that (10) applies to the general case, hence the deterministic case, which means that the approximation of the stationary distribution in the periodic case is given in an explicit formula, without solving any system of equations neither computing a large number of convolutions.

4.4 The Diaz and Kim model: periodic workloads

For the case where the tasks of the real-time system Γ are periodic and have deterministic deadlines, *i.e.* $T_k = \lambda_k^{-1} \in \mathbb{N}$ and $G_k(x) = \mathbf{1}_{[\lambda_k^{-1}, \infty)}(x)$ for all $\tau_k \in \Gamma$ and $x > 0$, also known as the Diaz and Kim (DK) model [8,9,17], the authors approximate the distribution function π_k, resolving linear system equations and compute response times distributions with the help of convolutions.

The fastest computational complexity of convolutions is $\mathcal{O}(N \log N)$ when N is the number of values that a probabilistic variable can take. Computing the exact values of π_k quickly becomes an expensive operation when the number of tasks or the number of possible execution times gets larger, even with methods providing tools to soften those computations like Markovic *et al.* [24], Milutinovic *et al.* [27] or Maxim *et al.* [26] for example. Moreover, the computation of response times has the same problem, as the number of possible values of $\tilde{\beta}_k$ quickly becomes large. With (6) we have an explicit formula of the distribution of $\tilde{\beta}_k$, with (11) we know the error of the heavy-traffic approximation, and with Theorem 1 we have an analytical expression of the backlog in the deterministic case.

Indeed, let $\bar{T}_k = \text{lcm}\left(\lambda_1^{-1}, \ldots, \lambda_k^{-1}\right)$ be the *hyper-period* of level k of Γ. In the DK model [8,17], the authors consider the k-level backlog $\beta_k(t\bar{T}_k), t \in \mathbb{N}$, *i.e.* the remaining demand of level k at the beginning of the t-th hyper-period. Diaz *et al.* [8] have proven that the sequence $\left(\beta_k\left(t\bar{T}_k\right)\right)_{t \in \mathbb{N}}$ is a *stationary* Markov chain when $\bar{u}_k < 1$. The sequence $\left(\beta_k(t\bar{T}_k)\right)_{t \in \mathbb{N}}$ is defined by $\beta_k(0) \geq 0$ and $\beta_k\left((t+1)\bar{T}_k\right) = \left(\beta_k(t\bar{T}_k) + W_k(\bar{T}_k) - \bar{T}_k\right)^+$ for $t \in \mathbb{N}$.

In order to apply (6) to the DK model, we consider a system with only one task with an execution time equal to the total demand of one hyper-period $W_k(\bar{T}_k)$. Indeed, the demand of one hyper-period is repeated at each new hyper-period [8]. Let $W_{k,t}, t \in \mathbb{N}$ be independent and identically distributed variables with the same distribution than $W_k(\bar{T}_k)$. Then from (6) we have

$$\tilde{\beta}_k \sim \sup_{n \geq 1} \left(\sum_{t=1}^{n} W_{k,t} - n\bar{T}_k\right)^+ \tag{12}$$

The representation in (12) is an efficient method to approximate the stationary distribution π_k of the backlog process β_k. Indeed, let us take an integer $n > 0$, and generate a sample $(W_{k,t})_{t=1}^{n}$ independent and identically distributed sequence with the distribution of $W_k(\bar{T}_k)$. Eq. (12) provides the variable of distribution π_k found in Diaz *et al.* [8] and Kim *et al.* [17]. It also means that the variable $\max_{1 \leq j \leq n} \left(\sum_{t=1}^{j} \left(W_{k,t} - \bar{T}_k\right)\right)^+$ is an approximation of $\tilde{\beta}_k$ when n is large enough. This method is not expensive in complexity as it requires only to build the distribution function of $W_k(\bar{T}_k)$ once.

Remark 4. *In the* DK *model, the coefficient of variation $\gamma_{a,k}$ is 0, thus*

$$\eta_k = 2m_k^2 \frac{1 - u_k}{\lambda_k s_k^2}$$

5   Time demand analysis

In this section, we express the response times distribution of a task τ_k using the heavy-traffic analysis of Section 4. The Markovian property of Brownian motions allows to approximate the distribution of any response time $R_{k,l}$ in

terms of the backlog $\hat{\beta}_{k-1}(A_{k,l})$ and the first response time $R_{k,1}$, thus response times will be conditioned to backlogs and execution times, and represented as idle times following the inverse Gaussian distribution. In a second part, we provide an analytical expression of the worst-case heavy-traffic response time distribution, and in a third part we do the same for the steady-state heavy-traffic response time distribution. Finally, we explain how to simulate heavy-traffic response times of a task τ_k from the distribution functions F_k and G_k.

The response time $R_{k,l}$ of a job $\tau_{k,l}$ is the smallest instant after its arrival time $A_{k,l}$ smaller than the time it is given to run the level $k-1$ demand and its execution time, *i.e.* the smallest $t > 0$ such that

$$B_{k-1}(A_{k,l}) + C_{k,l} + W_{k-1}(t + A_{k,l}) - W_{k-1}(A_{k,l}) \leq t \tag{13}$$

where $B_k(t) = \sum_{i=1}^{k} \min\left(\beta_i(t) - \beta_{i-1}(t), C_i(t)\right)$ is the *blocking time of level k* at the instant $t > 0$, $\beta_0(t) = 0$, and $C_i(t)$ is the execution time of the most recent job of τ_i released before t, for all $t > 0$. This relation holds for any fixed-priority preemptive single-core model with the discarding policy discussed in Section 2. By definition, the blocking time is such that

$$B_k(t) \leq \min\left(\beta_k(t), \sum_{i=1}^{k} C_i(t)\right) \tag{14}$$

for all $t > 0$, which makes our response time analysis build upper-bounds of response times.

Furthermore, the heavy-traffic demand $\hat{W}_{k-1}$ is a Brownian motion, thus $\hat{W}_{k-1}(t + A_{k,l}) - \hat{W}_{k-1}(A_{k,l}) \sim \hat{W}_{k-1}(t)$, and it is continuous. Then we define the *heavy-traffic response time* of a job $\tau_{k,l}$ as

$$\hat{R}_{k,l} \sim \inf_{t>0}\left\{\hat{\beta}_{k-1}(A_{k,l}) + C_{k,l} + \hat{W}_{k-1}(t) = t\right\} \tag{15}$$

In the following sections we consider response times as *idle times* properly conditioned, and from this analysis provide the worst-case response time distribution and the steady-state response time distribution.

5.1 First idle time

In the following, we consider the conditional probability that the system starts with a level $k-1$ backlog $x \geq 0$ and the job $\tau_{k,1}$,

$$\mathbf{P}_k^x(\cdot) = \mathbf{P}(\cdot \mid \beta_{k-1}(0) = x, A_{k,1} = 0)$$

Whenever we need to suppose $\beta_{k-1}(0) = x, A_{k,1} = 0$, we say that we work *under the probability $\mathbf{P}_k^x$*.

As the steady state of backlogs repeats itself infinitely, the analysis of response time stands for all future response times. In the transient state though, each response time has its own distribution function. Hence, in Section 5.3 we
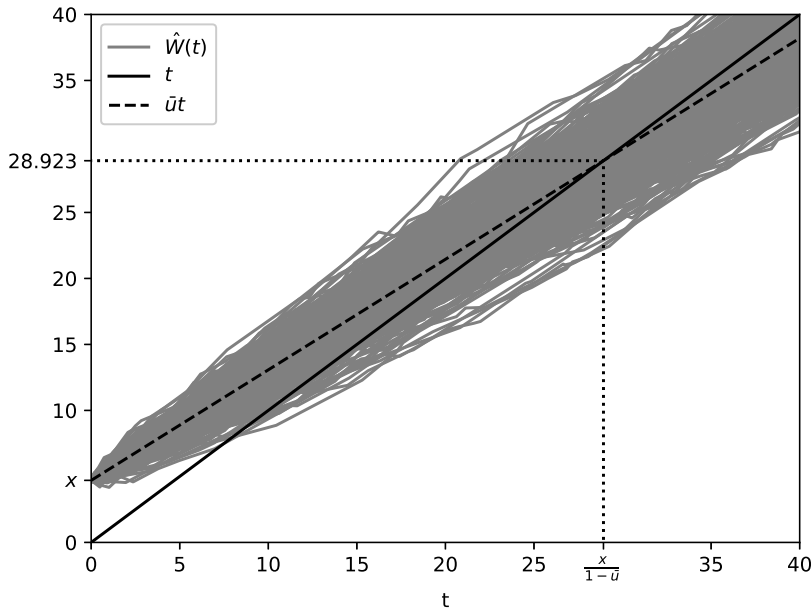
Figure 4: Trajectories of $\hat{W}$ and average of the first idle time for $\bar{u} = 0.838$.

use the worst-case reasoning to bound the *transient response times*. A question rises at this point: when does the system get steady ? Let us look closely at the dynamics of the heavy-traffic backlog process. Let $\hat{\beta}(t) \triangleq \max_{\tau_k \in \Gamma} \hat{\beta}_k(t)$, $\hat{W}(t) \triangleq \max_{\tau_k \in \Gamma} \hat{W}_k(t)$ and $v \triangleq \max_{\tau_k \in \Gamma} v_k$. As we work with fixed-priority scheduling policies, $\hat{\beta}, \hat{W}$ and v refer to the lowest priority level.

Proposition 2. *Let $\bar{u} < 1, x > 0$ and $\beta(0) = x$. The steady-state backlog is reached at the first idle time*

$$\mathcal{I}^x \triangleq \inf \left\{ t > 0 : \hat{\beta}(t) = 0 \right\}$$

Proof. For any $x > 0$, at the instant $\mathcal{I}^x$, the next idle time has the distribution of $\mathcal{I}^0$, so is the third, the fourth, *etc*. This is sufficient to say that the system is steady for all $t > \mathcal{I}^x$. Moreover, one can see in [4, Theorem 6.8] that when $\bar{u} < 1$ the steady-state backlog $\tilde{\beta}$ is 0. □

Proposition 2 is coherent with the proven results in the deterministic case [8] and the simulations shown in Figure 2. Let us now refine the analysis for each task $\tau_k \in \Gamma$. Let $\mathcal{I}_k^x$ be the *first idle time of level k*, *i.e.*

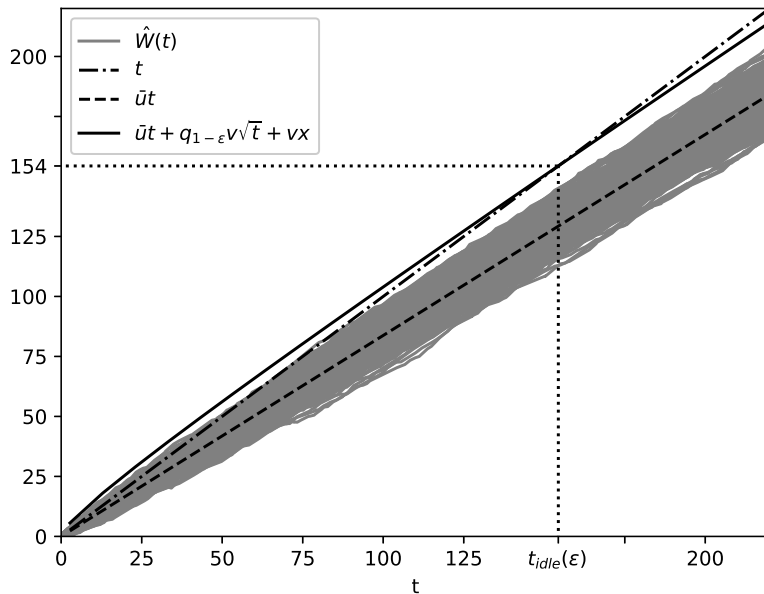$$\mathcal{I}_k^x = \inf \left\{ t > 0 : \hat{\beta}_k(t) = 0 \right\} \tag{16}$$

15

Figure 5: First maximum ε-idle time for $\varepsilon = 10^{-6}$. for $\bar{u} = 0.838$.

when the initial backlog of level k is equal to $x > 0$.

Lemma 1. *The distribution of $\mathcal{I}_k^x$ is an inverse Gaussian distribution with probability function*

$$\psi_k^x(t) \triangleq \psi\left(t; \frac{x}{1 - \bar{u}_k}, \frac{x^2}{v_k^2}\right), t > 0$$

Proof. The idle time $\mathcal{I}_k^x$ defined in (16) is a quantity called *first-passage time* of a Brownian motion [28, Eq. (28)]. Until the first idle time, we know that $\hat{\beta}_k(t) = x + \hat{W}_k(t) - t$, because $\mathbf{1}_{\beta_k(s)>0} = 1$ for $0 \le s < t \le \mathcal{I}_k^x$. Then from (7) we have $\mathcal{I}_k^x \sim \inf\{t > 0 : B(t) = t(1 - \bar{u}_k)/v_k - x/v_k\}$ where B is a standard Brownian motion. When $\beta_k(0) = x$, the distribution of $\mathcal{I}_k^x$ is an inverse Gaussian distribution of mean $x/(1 - \bar{u}_k)$ and shape x^2/v_k^2. See Figure 5.1 and [15, p. 146], for more details. □

The first idle time is also a probabilistic variable, but we can build an instant $t_{idle}(\varepsilon, x)$ such that for any $\varepsilon > 0$ and all $t > t_{idle}(\varepsilon, x)$, the probability that the first idle time of the lowest priority level is greater than t is less than ε, *i.e.* $\mathbf{P}(\mathcal{I}^x > t) \le \varepsilon$. We call *first ε-idle time* the instant $t_{idle}(\varepsilon, x)$ from which we can guarantee the system is steady with probability $1 - \varepsilon$ and initial backlog $x > 0$.

We know that $\hat{\beta}$ is a Brownian motion reflected at the origin. This means that until it reaches zero, *i.e.* an idle time, it is a simple Brownian motion. In fact, between two consecutive idle times the backlog process has the same dynamic. The following proposition bounds with a given probability the first idle time and the instant the system gets steady.

Proposition 3. *Let $\varepsilon \in (0,1)$ and $x \geq 0$. If $\bar{u} < 1$, the system of initial demand x is steady with probability at least $1 - \varepsilon$ at the instant*

$$t_{idle}(\varepsilon, x) = \left(\frac{q_{1-\varepsilon} + \sqrt{q_{1-\varepsilon}^2 + 4\left(\frac{1-\bar{u}}{v}\right)x}}{2\left(\frac{1-\bar{u}}{v}\right)} \right)^2 \tag{17}$$

where $q_{1-\varepsilon} = \Phi^{-1}(1-\varepsilon)$ is the $(1-\varepsilon)$-quantile of a standard normal distribution.

Proof. Let us consider the lowest priority first idle time $\mathcal{I}^x$ and remark that

$$\mathbf{P}(\mathcal{I}^x > t) = \mathbf{P}\left(\inf_{s \in [0,t]} \hat{W}(s) - s > -x \right) \tag{18}$$

$$\leq \mathbf{P}\left(\hat{W}(t) > t - x \right)$$

$$= \mathbf{P}\left(B(t) > t\frac{1-\bar{u}}{v} - \frac{x}{v} \right)$$

$$= 1 - \Phi\left(\frac{(1-\bar{u})t - x}{v\sqrt{t}} \right)$$

Hence, let us define $t_{idle}(\varepsilon, x)$ such that

$$1 - \Phi\left(\frac{(1-\bar{u})t_{idle}(\varepsilon, x) - x}{v\sqrt{t_{idle}(\varepsilon, x)}} \right) \leq \varepsilon$$

which implies that for $t > t_{idle}(\varepsilon, x)$ we should have

$$t > \bar{u}t + \Phi^{-1}(1-\varepsilon)v\sqrt{t} + vx \tag{19}$$

see Figure 5.1. This is a second order polynomial equation, which admits no solution when $\bar{u} > 1$, and, when $\bar{u} \leq 1$ and $\varepsilon \in (0,1)$, the smallest solution of (19) is (17). $\qquad\square$

Note that for all $x \geq 0$, $\lim_{\varepsilon \to 0} t_{idle}(\varepsilon, x) = \infty$, $\lim_{\varepsilon \to 1} t_{idle}(\varepsilon, x) = 0$, and

- $\bar{u} < 1 \implies t_{idle}(\varepsilon, x) < \infty$,

- $\bar{u} = 1 \implies t_{idle}(\varepsilon, x) = \infty$,

- $\bar{u} > 1 \implies t_{idle}(\varepsilon, x)$ is not defined.

Remark 5. *We provide the analysis for the lowest priority level, but in fact each priority level has its own first ε-idle time.*

Finally we consider the *maximum first ε-idle time* $t_{idle}^{max}(\varepsilon)$ by considering a synchronous activation for the distribution μ^{max} of the initial demand, *i.e.* $\beta(0) = \sum_{\tau_k \in \Gamma} C_k$. See Proposition 5 for a more detailed explanation. The maximum first ε-idle time is then

$$t_{idle}^{max}(\varepsilon) = \int t_{idle}(\varepsilon, x) d\mu^{max}(x) \tag{20}$$

5.2 Conditioning response times

From (15), any job $\tau_{k,l}$ has a heavy-traffic response time distribution that can be expressed from the response time $\hat{R}_{k,1}$,

$$\mathbf{P}\left(\hat{R}_{k,l} > t \mid \hat{\beta}_{k-1}(A_{k,l}) = x\right) = \mathbf{P}_k^x\left(\hat{R}_{k,1} > t\right) \tag{21}$$

Let us condition this probability for specific values of execution times.

Lemma 2. *Under $\mathbf{P}_k^x$ the heavy-traffic response time $\hat{R}_{k,1}$ given $C_{k,1} = y$ is identically distributed as $\mathcal{I}_{k-1}^{x+y}$.*

Proof. As stated in (21), the proper conditioning on backlogs provides the distribution of the response time of $\tau_{k,1}$. Furthermore, when the backlog $\hat{\beta}_{k-1}(A_{k,l}) = x$ and the execution time $C_{k,l} = y$, the response time $\hat{R}_{k,l}$ is the time it takes for all level $k-1$ jobs to finish plus the time it takes for level $k-1$ to stay idle for $x + y$ instants. It means that we can artificially set the initial backlog to $x + y$ and look at $\mathcal{I}_{k-1}^{x+y}$, the first idle time of level $k-1$, as represented in Figure 6. In other words,

$$\mathbf{P}_k^x\left(\hat{R}_{k,1} > t \mid C_{k,1} = y\right) = \mathbf{P}\left(\inf_{s \in [0,t]} \hat{W}_{k-1}(s) - s > -x - y\right) = \mathbf{P}\left(\mathcal{I}_{k-1}^{x+y} > t\right)$$

which is sufficient to conclude. $\qquad\square$

Example 3 (Response times as idle times). *Consider a task set $\{\tau_1, \tau_2\}$, and that τ_2 is activated at $t = 0$, i.e. $A_{2,1} = 0$, and $C_{2,1} = y = 3$. Let $\mathbf{P}(C_1 = 1) = 1/2, \mathbf{P}(C_1 = 2) = 1/2, \mathbf{P}(T_1 = 2) = 1/2, \mathbf{P}(T_1 = 4) = 1/2$, and suppose $\beta_1(0) = x + y = 8$, $C_{1,1} = 1$, $C_{1,2} = 2$, $C_{1,3} = 2$, and $A_{1,1} = 2, A_{1,2} = 6, A_{1,3} = 10$ and $A_{1,4} = 14$. Then the response time $R_{2,1}$ is the first instant t when $\beta_1(t) = 0$:*

$$\begin{aligned}
\beta_1(1) &= \beta_1(0) = x + y \\
\beta_1(3) &= \beta_1(0) + C_{1,1} - 3 = x + y - 2 \\
\beta_1(7) &= \beta_1(0) + C_{1,1} + C_{1,2} - 7 \\
&= x + y - 4 \\
\beta_1(11) &= \beta_1(0) + C_{1,1} + C_{1,2} + C_{1,3} - 11 \\
&= x + y - 6 \\
\beta_1(13) &= \beta_1(0) + C_{1,1} + C_{1,2} + C_{1,3} - 13 \\
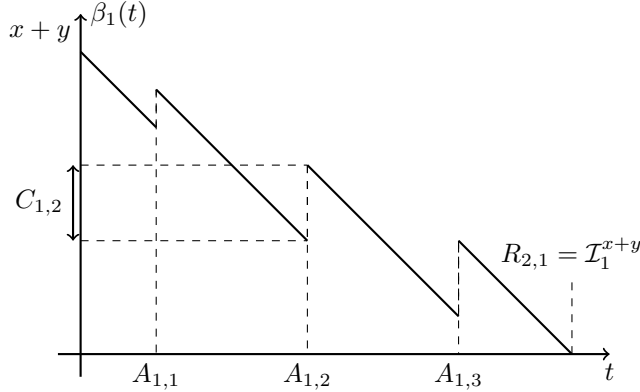&= x + y - 8 = 0
\end{aligned}$$

18

Figure 6: Representation of the response time $R_{2,1}$ as an idle time when $A_{2,1} = 0, C_{2,1} = y$ and $\beta_1(0) = x$, in Example 3.

Hence $R_{2,1} = 13$. This is illustrated in Figure 6. To build all possible values of $R_{2,1}$, one must do the same for all combinations of possible values of $(A_{1,1}, A_{1,2}, A_{1,3}, A_{1,4})$ and $(C_{1,1}, C_{1,2}, C_{1,3}, C_{1,4})$. Lemma 2 provides an analytical expression of this result for the heavy-traffic response time $\hat{R}_{2,1}$.

From (15) and Lemma 2 we establish a necessary condition of the feasibility of Γ.

Proposition 4. *A non-stable real-time system with implicit deadlines and independent execution times is not feasible under a fixed-priority scheduling policy.*

Proof. Let $\tau_k \in \Gamma$ be such that $\bar{u}_k > 1$. The Loynes theorem 1 states that there is a finite number of idle times of level k, which implies with (15) that heavy-traffic response times get infinite at some point, *i.e.* for all $t > 0$, $\mathbf{P}\left(\sup_{l \in \mathbb{N}} \hat{R}_{k,l} > t\right) = 1$. In other words, there is no permitted failure rate $\alpha_k \in (0, 1)$ such that τ_k is schedulable as defined in (1). As we consider a fixed-priority scheduling policy, then the lowest level backlog is larger than all k-level backlogs $\beta_k(t)$ at any time $t > 0$. □

With the same reasoning we get the explicit formula of the distribution function of response times.

Let us denote the conditional probability that the distribution of the initial backlog $\beta_{k-1}(0)$ is the probability $d\mu$ and the first job released is $\tau_{k,1}$,

$$\mathbf{P}_k^\mu(\cdot) \triangleq \mathbf{P}(\cdot \mid \beta_{k-1}(0) \sim d\mu, A_{k,1} = 0) = \int \mathbf{P}_k^x(\cdot) d\mu(x)$$

Lemma 3. *Under $\mathbf{P}_k^\mu$, the probability function of the heavy-traffic response time $\hat{R}_{k,1}$ is*

$$h_k^\mu(t) = \int \psi_{k-1}^x(t) d(\mu * F_k)(x)$$

19

Proof. We conclude with Lemmas 1 and 2 and the definition of conditional probabilities, *c.f.* (2). $\qquad\square$

In Sections 5.3 and 5.4, we prove that the proper initialization of the system puts the system in two specific cases: the worst-case and the steady-state.

5.3 Worst-case response time

Before the system reaches its steady-state, we say it is *transient*. In that case we cannot provide the exact distribution of the backlog in an analytical formulation. However, we can bound it by using the *worst-case backlog*.

Let $\mu_k = F_1 * \cdots * F_k$ be the distribution function of the demand of level k of a *synchronous activation* $\sum_{i=1}^{k} C_i$.

Proposition 5. *If $\bar{u}_{k-1} < 1$, the worst-case heavy-traffic response time distribution of τ_k is defined by*

$$\mathbf{P}\left(\sup_{l\in\mathbb{N}} \hat{R}_{k,l} > t\right) = \mathbf{P}_k^{\mu_{k-1}}\left(\hat{R}_{k,1} > t\right) \tag{22}$$

Proof. First, let us consider $\bar{u}_{k-1} < 1$, as stated in Theorem 1 the backlog process converges to $\tilde{\beta}_k$ which is finite. Jobs are discarded if they miss their deadlines, and as we consider implicit deadlines, there can be at most one job per task activated simultaneously. This leads into considering $\beta_{k-1}(0) = \sum_{i=1}^{k-1} C_i$ as the maximum backlog of level $k-1$, *c.f.* (14), and use the property stated in (21).

When $\bar{u}_{k-1} = 1$, idle times of level k may or may not be finite, thus we cannot conclude anything on the distribution of response times.

When $\bar{u}_{k-1} > 1$, the largest response time does not come from the synchronous activation and is in fact ∞. As we have already demonstrated in the proof of Proposition 4, response times of the task τ_k increase to ∞, due to the absence of idle times of level $k-1$. Then we conclude that the worst-case heavy-traffic response time is ∞.

See Figure 3. $\qquad\square$

The worst-case response time being identified as the synchronous activation case, we provide an explicit expression its distribution function.

Proposition 6. *Let $\tau_k \in \Gamma$ and $\bar{u}_k < 1$. The probability function of the worst-case heavy-traffic response time of τ_k is*

$$h_k^{max}(t) \triangleq \int_0^\infty \psi_{k-1}^x(t)d\mu_k(x) \tag{23}$$

Proof. From Proposition 5 we know that when $\bar{u}_k < 1$, the worst-case response time of $\tau_k \in \Gamma$ is $\hat{R}_{k,1}$ under $\mathbf{P}_k^{\mu_{k-1}}$. Furthermore, from Lemma 3 we have $h_k^{max}(t) = \int \psi_{k-1}^z(t)\, d(\mu_{k-1} * F_k)(z)$. In addition, we have the relation $\mu_{k-1} * F_k = \mu_k$. $\qquad\square$

5.4 Steady-state response time

The backlog process of level k being stationary and with a stationary distribution π_k, in the steady-state (21) becomes

$$\mathbf{P}_k^{\pi_{k-1}}\left(\hat{R}_{k,l} > t\right) = \mathbf{P}_k^{\pi_{k-1}}\left(\hat{R}_{k,1} > t\right) \tag{24}$$

for any $l \in \mathbb{N}$. Then if it holds for any $l \in \mathbb{N}$, it holds for all the response times after the convergence of the backlog process. This is why in the steady-state, the distribution of heavy-traffic response times is unique. Consider the steady-state response time

$$\tilde{R}_k \triangleq \inf\left\{t > 0 : \tilde{\beta}_{k-1} + C_k + \hat{W}_{k-1}(t) = t\right\} \tag{25}$$

As in Section 5.3, we get the distribution function of steady-state response times.

Proposition 7. *Let $\tau_k \in \Gamma$, $\bar{u}_k < 1$, $\tilde{h}_k$ be the probability function of the steady-state response time of τ_k, and for $i \in \{1, \ldots, k\}$ let η_i be as defined in Proposition 1 and $\xi_{i,k} = \prod_{j=1, j \neq i}^{k}(1 - \eta_i/\eta_j)^{-1}$. Then for all $t > 0$,*

$$\tilde{h}_k(t) = \sum_{i=1}^{k-1} \xi_{i,k-1} \int_0^\infty \psi_{k-1}^z(t) d\nu_{i,k}(z)$$

where $d\nu_{i,k}(z) = \eta_i \int_0^z e^{-\eta_i(z-x)} dF_k(x) dz$ is the distribution of $C_k + \tilde{\beta}_i - \tilde{\beta}_{i-1}$.

Proof. We know from Proposition 1 that the steady-state level $(k-1)$ backlog distribution is π_{k-1}. We know from Lemma 2 that the stead-state response time $\tilde{R}_k$ is the first idle time with an initial backlog of $\tilde{\beta}_{k-1} + C_k$. Finally from Lemma 3, we know that the probability function of $\tilde{R}_k$ is $\tilde{h}_k \triangleq h_k^{\pi_{k-1}}$. Then, let us compute the convolution $\pi_{k-1} * F_k$ which is the distribution function of the sum $\tilde{\beta}_{k-1} + C_k$. For any $z > 0$, we have $d(\pi_{k-1} * F_k)(z) = \frac{d\pi_{k-1}}{dz} * F_k(z) dz$ and

$$\frac{d\pi_{k-1}}{dz} * F_k(z) dz = dz \int_0^\infty \frac{d\pi_{k-1}}{dz}(z-x) dF_k(x)$$

$$= dz \int_0^z \eta_1 \ldots \eta_{k-1} \sum_{i=1}^{k-1} \frac{\exp(-\eta_i(z-x))}{\prod_{j=1, j\neq i}^{k-1}(\eta_j - \eta_i)} dF_k(x)$$

$$= dz \int_0^z \sum_{i=1}^{k-1} \eta_i e^{-\eta_i(z-x)} \xi_{i,k-1} dF_k(x)$$

$$= \sum_{i=1}^{k-1} \xi_{i,k-1} d\nu_{i,k}(z)$$

Finally by definition of conditional probabilities, *c.f.* (2), we have $\tilde{h}_k(t) = \int \psi_{k-1}^z(t) d(\pi_{k-1} * F_k)(z)$. $\square$

Remark 6. *In the periodic case, Diaz [8] has proven that the system is always steady when $u_k^{max} \leq 1$, e.g. Figure 3a, i.e. the steady-state backlog is $\tilde{\beta}_k = 0$, and that is it reached at $t = 0$.*

$$\tilde{h}_k(t) = \int \psi_{k-1}^x(t) dF_k(x) \tag{26}$$

This steadiness is illustrated in Figure 2.

6 Simulations

In Lemma 2 we prove that response times $R_{k,l}$ can be simulated from a sample of idle times of level $k-1$, a sample of execution times and the proper initialization backlog sample, the steady-case or the worst-case.

6.1 Generating heavy-traffic response times

The procedure is as follows:

- **Generate** a backlog b with the distribution function μ_{k-1} (resp. π_{k-1}),

- **Generate** an execution time c with the distribution function F_k,

- **Generate** the response time $R_k = \mathcal{I}_{k-1}^{b+c}$, with the probability function ψ_{k-1}^{b+c}.

See Figure 6 and Figure 7a.

Proposition 8. *The probability function of R_k is h_k^{max} (resp. $\tilde{h}_k$).*

Proof. It is a consequence of Lemma 3. Alternatively, we can see that by construction (23) and (7) come from conditional probabilities. The representation in Lemma 2 indicates that the distribution of response times is conditioned by the values of the backlog and the execution time. □

6.2 Experimental results

The purpose of this section is to illustrate that the response times generated from Proposition 8 provide a good approximation, by comparing distribution functions of simulations and its associated EVT estimation, and generated heavy-traffic response times distributions. Closer are the curves, better is the estimation. Those results are not exhaustive and are an illustration, we do not cover in this work the sensitivity of the model for different values of $\bar{u}$. In order to do so, and to compare the distribution of the generated response times with the ones provided by SimSo, we modified the SimSo tool [5] to simulate probabilistic systems and compare the scheduled (SimSo) simulations and heavy-traffic response times simulations. We use the data generated by SimSo to apply EVT

on response times (using the Scipy framework[3] on Python). Finally we compare our results with SimSo simulations and EVT estimations. For simplicity the periods are deterministic in these simulations.

In order to illustrate the stability described in this paper, we use the task set Γ provided in Table 1, with $\mathcal{C}_k$ being the set of possible values for the execution time C_k and $p_k = dF_k/dx$ its probability function. The level 2 maximum utilization is smaller than 1, hence backlogs of level 2 converge quickly to 0, see Figure 3a. The level 3 maximum utilization is greater than 1 and the level 3 mean utilization is smaller than 1. Thus, τ_3 is the task of interest, see Figure 2b and Figure 3b. The level 4 mean utilization is close to 1, hence Figure 2c and Figure 3c illustrate the behavior of backlogs when the utilization approaches its phase transition. The level 5 mean utilization is greater than 1, which illustrates the *explosion* (infinite response times) of the system, see Figure 2d and Figure 3d.

In Figure 2 we observe that the demand $W(t)$ follows the line $\bar{u}t$ (its mean). In Figure 2 and 3 we see what happens when the system mean utilization is smaller, close and greater than 1 : for smaller values of $\bar{u}$ the system stays with zero backlog at some point, see Figure 2b, but for values greater than 1 the system explodes, see Figure 2d and 3d. In Figure 2c and 3c we see that even for $\bar{u}$ close to 1, the system always admits finite idle times. Most importantly, we see in Figure 2b and 3b that when $\bar{u} < 1$ and $u^{max} > 1$, the analysis holds and provides quantifiable response times.

In Figure 5.1, we see to what corresponds idle times and ε-idle times graphically. In Figure 5.1 the average idle time corresponds to the point where the line t and $\bar{u}t$ meet, and the distribution of the idle times correspond to the frequency the demand process meets the line t for each instant $t > 0$.

Finally, in Figure 7a, we can see the simulations presented in Section 6.1 and a comparison with response times simulated via SimSo [5]. The two *ground truth* samples are the two subsets *SimSo-transient* and *SimSo-steady*, which are composed respectively of simulated response time released before and after the maximum ε-idle time $t_{idle}^{max}(\varepsilon) = 154$ where $\varepsilon = 10^{-6}$. The maximum idle time $t_{idle}^{max}(\varepsilon)$ is computed via Monte-Carlo approximations with the representation in (20).

The EVT estimation (green curve) from these SimSo simulations and the steady-state and worst-case response times suggested in this paper are compared via their distribution functions in Figure 7b. In this case, where $\bar{u}$ is not too close to 1 and u^{max} greater than 1, we can see that the *Worst-case* response times, the *EVT* estimation and the steady-state response times are greater (in the stochastic sense [9]) than the *true* response times simulated with SimSo. The heavy-traffic *Worst-case* response time seems to be an upper-bound in practice, and the *Steady-state* response time is quite accurate.

Closer is the *Steady-state* to the *SimSo-steady* curve, more accurate the approximation is. We can see a big difference between the *Worst-case* curve and the *Steady-state* curve. However the proposed analysis does not permit to

---

[3]https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gereme.html

Table 1: Task set used in the simulations of the experimental results.

task τ_k	T_k	$\mathcal{C}_k$	p_k	$\bar{u}_k$	u_k^{max}
τ_1	4	$(1,2)$	$(0.5, 0.5)$	0.375	0.5
τ_2	6	$(1,2)$	$(0.5, 0.5)$	0.625	0.833
τ_3	8	$(1,2,3)$	$(0.5, 0.3, 0.2)$	0.838	1.208
τ_4	10	$(1,2,3)$	$(0.6, 0.2, 0.2)$	0.998	1.508
τ_5	12	$(1,2,3,4)$	$(0.5, 0.3, 0.1, 0.1)$	1.148	1.841

quantify analytically this difference.

7 Future work

The results of this paper can be generalized if some assumptions of the model are relaxed. Indeed, we have seen that the steady-state only depends on inter-arrivals only through their mean value in the heavy-traffic analysis. This means that one could suppose inter-arrival independent, also known as *Poisson arrivals*, and use the wide list of results on Poisson arrivals to extend the results of this paper.

7.1 Extension to dynamic-priority policies

The first step into dynamic scheduling, as for example Earliest Deadline First (EDF) is to study the processes $\hat{\beta}$ and $\hat{W}$ introduced in Section 5.1. Indeed, for fixed-priority policies, those variables are simply the backlog and demand of the lowest priority level. However, for EDF, levels of priority need to be defined not only for tasks but for jobs. In [8, Section 4.4], authors use the concept of *ground jobs* which are jobs released at an instant where the system is idle, and as shown in the Loynes theorem, when $\bar{u} < 1$, idle times are finite. This means that an analysis mixing the concept of ground jobs and idle times as defined in this paper can provide an extension of our results for dynamic-priority scheduling.

7.2 Extension to dependent execution times

Considering time-driven execution times, *i.e.* execution times with distribution functions changing over time, is possible. We used drifted Brownian motions to describe the demand. However, we used a model with constant coefficients. A more general class of drifted Brownian motion involve drift and deviation function, depending not only on the values of the demand but also on the the time interval they are being considered. This is a natural extension of the model presented in our work and leads to a more general type of real-time system that needs to be defined.

Considering dependence between tasks is also possible. It would lead the analysis to multivariate Brownian motion which requires some new concepts but uses the same arguments we used in this work. The demand should describe

not only priority levels but task levels, and cannot be superposed as we have done. Indeed, as a some properties are quite instinctive when independence is assumed (Wald's lemma and the law of large numbers for example), they need some workaround when this assumption is relaxed.

7.3 Extension to multi-core processors

The Loynes theorem also exists for the multi-core case [1], which means that there is a way to apply the results we provide in this paper to a multi-core model, for a global fixed-priority scheduling policy, and a global dynamic-scheduling policy. Furthermore, for independent inter-arrivals, optimal algorithm already exist for priority assignment and processor assignment, *c.f.* [1, 2, 4, 13, 33].

8 Conclusion

We have seen that real-time systems can reach steadiness over a finite and quantifiable amount of time, and that a necessary condition to assure this stability is that the mean utilization is smaller than 1. In practice, systems with a mean utilization $\bar{u} > 0.9$ have many deadline misses, which requires from system designers to quantify deadline miss probabilities carefully by using the distributions of response times provided in this paper. No schedulability tests considering the steadiness of response times exist. This is a natural step in our opinion for the use of the analysis provided in this work. For example, the analysis we have presented in this paper is well suited for an application of a Monte-Carlo response time analysis [3] which has recently been proven efficient.

We have expressed the probability function of response times in a specific family of distributions, the inverse Gaussian distribution, and provided a method to generate them. However, the distribution functions of execution time need to be known, but they are usually unknown. The methods built in this paper could be used in empirical and measurement-based methods, for example using clustering methods [10, 35].

References

[1] Francois Baccelli and Pierre Brémaud. *Elements of queueing theory: Palm Martingale calculus and stochastic recurrences*, volume 26. Springer Science & Business Media, 2013.

[2] Onno J Boxma. Workloads and waiting times in single-server systems with multiple customer classes. *Queueing Systems*, 5(1):185–214, 1989.

[3] Sergey Bozhko, Georg von der Brüggen, and Björn B Brandenburg. Monte carlo response-time analysis. In *Real-Time Systems Symposium*, 2021.

[4] Hong Chen and David D Yao. *Fundamentals of queueing networks: Performance, asymptotics, and optimization*, volume 4. Springer, 2001.

[5] Maxime Chéramy, Pierre-Emmanuel Hladik, and Anne-Marie Déplanche. Simso: A simulation tool to evaluate real-time multiprocessor scheduling algorithms. In *Proc. of the 5th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems*, WATERS, 2014.

[6] Robert I Davis, Liliana Cucu-Grosjean, Marko Bertogna, and Alan Burns. A review of priority assignment in real-time systems. *Journal of systems architecture*, 65:64–82, 2016.

[7] Robert Ian Davis and Liliana Cucu-Grosjean. A survey of probabilistic timing analysis techniques for real-time systems. *eibniz Transactions on Embedded Systems*, pages 1–60, 2019.

[8] José Luis Díaz, Daniel F García, Kanghee Kim, Chang-Gun Lee, L Lo Bello, José María López, Sang Lyul Min, and Orazio Mirabella. Stochastic analysis of periodic real-time systems. In *23rd IEEE Real-Time Systems Symposium*, pages 289–300. IEEE, 2002.

[9] José Luis Díaz, José María López, Manuel Garcia, Antonio M Campos, Kanghee Kim, and Lucia Lo Bello. Pessimism in the stochastic analysis of real-time systems: Concept and applications. In *25th IEEE International Real-Time Systems Symposium*, pages 197–207. IEEE, 2004.

[10] Anna Friebe, Alessandro V Papadopoulos, and Thomas Nolte. Identification and validation of markov models with continuous emission distributions for execution times. In *IEEE 26th International Conference on Embedded and Real-Time Computing Systems and Applications*, pages 1–10. IEEE, 2020.

[11] Bruno Gaujal, Alain Girault, and Stéphan Plassart. Dynamic speed scaling minimizing expected energy consumption for real-time tasks. *Journal of Scheduling*, 23(5):555–574, 2020.

[12] Zhishan Guo, Luca Santinelli, and Kecheng Yang. Edf schedulability analysis on mixed-criticality systems with permitted failure probability. In *IEEE 21st International Conference on Embedded and Real-Time Computing Systems and Applications*, pages 187–196. IEEE, 2015.

[13] Junfei Huang, Boaz Carmeli, and Avishai Mandelbaum. Control of patient flow in emergency departments, or multiclass queues with deadlines and feedback. *Operations Research*, 63(4):892–908, 2015.

[14] Jacques Janssen and Raimondo Manca. Renewal theory. *Applied Semi-Markov Processes*, pages 45–104, 2006.

[15] Monique Jeanblanc, Marc Yor, and Marc Chesney. *Mathematical methods for financial markets*. Springer Science & Business Media, 2009.

[16] Marwan Wehaiba El Khazen, Kevin Zagalo, Hadrien Clarke, Mehdi Mezouak, Yasmina Abdeddaïm, Avner Bar-Hen, Slim Ben Amor, Rihab Bennour, Adriana Gogonel, Kossivi Kougblenou, Yves Sorel, and Liliana Cucu-Grosjean. Work in progress: Kdbench – towards open source benchmarks for measurement-based multicore wcet estimators. In *IEEE RTAS*, 2022.

[17] Kanghee Kim, Jose Luis Diaz, Lucia Lo Bello, José María López, Chang-Gun Lee, and Sang Lyul Min. An exact stochastic analysis of priority-driven periodic real-time systems and its approximations. *IEEE Transactions on Computers*, 54(11):1460–1466, 2005.

[18] John P Lehoczky. Fixed priority scheduling of periodic task sets with arbitrary deadlines. In *Proceedings 11th Real-Time Systems Symposium*, pages 201–209. IEEE, 1990.

[19] John P Lehoczky. Real-time queueing theory. In *17th IEEE Real-Time Systems Symposium*, pages 186–195. IEEE, 1996.

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

[21] Meng Liu, Moris Behnam, and Thomas Nolte. An evt-based worst-case response time analysis of complex real-time systems. In *8th IEEE International Symposium on Industrial Embedded Systems (SIES)*, pages 249–258. IEEE, 2013.

[22] Yue Lu, Thomas Nolte, Iain Bate, and Liliana Cucu-Grosjean. A statistical response-time analysis of real-time embedded systems. In *IEEE 33rd Real-Time Systems Symposium*, pages 351–362. IEEE, 2012.

[23] Sorin Manolache, Petru Eles, and Zebo Peng. Memory and time-efficient schedulability analysis of task sets with stochastic execution time. In *Proceedings 13th EUROMICRO conference on Real-time Systems*, pages 19–26. IEEE, 2001.

[24] Filip Marković, Alessandro Vittorio Papadopoulos, and Thomas Nolte. On the Convolution Efficiency for Probabilistic Analysis of Real-Time Systems. In *33rd Euromicro Conference on Real-Time Systems*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

[25] Dorin Maxim and Liliana Cucu-Grosjean. Response time analysis for fixed-priority tasks with multiple probabilistic parameters. In *IEEE 34th Real-Time Systems Symposium*, pages 224–235. IEEE, 2013.

[26] Dorin Maxim, Mike Houston, Luca Santinelli, Guillem Bernat, Robert I Davis, and Liliana Cucu-Grosjean. Re-sampling for statistical timing analysis of real-time systems. In *Proceedings of the 20th International Conference on Real-Time and Network Systems*, pages 111–120, 2012.

[27] Suzana Milutinovic, Jaume Abella, Damien Hardy, Eduardo Quinones, Isabelle Puaut, and Francisco J Cazorla. Speeding up static probabilistic timing analysis. In *International Conference on Architecture of Computing Systems*, pages 236–247. Springer, 2015.

[28] Annalisa Molini, Peter Talkner, Gabriel G Katul, and Amilcare Porporato. First passage time statistics of brownian motion with purely time dependent drift and diffusion. *Physica A: Statistical Mechanics and its Applications*, 390(11):1841–1852, 2011.

[29] Nicolas Navet, Y-Q Song, and Françoise Simonot. Worst-case deadline failure probability in real-time applications distributed over controller area network. *Journal of systems Architecture*, 46(7):607–617, 2000.

[30] Sheldon M Ross. *Probability models for computer science*. Harcourt Academic Press San Diego, 2002.

[31] Jay Sethuraman and Mark S Squillante. Optimal stochastic scheduling in multiclass parallel queues. In *ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 93–102, 1999.

[32] Panayotis D Sparaggis and Don Towsley. Optimal routing and scheduling of customers with deadlines. *Probability in the Engineering and Informational Sciences*, 8(1):33–49, 1994.

[33] Jan A Van Mieghem. Dynamic scheduling with convex delay costs: The generalized c— mu rule. *The Annals of Applied Probability*, pages 809–833, 1995.

[34] Georg von der Brüggen, Nico Piatkowski, Kuan-Hsun Chen, Jian-Jia Chen, Katharina Morik, and Björn B Brandenburg. Efficiently approximating the worst-case deadline failure probability under edf. In *IEEE Real-Time Systems Symposium*, pages 214–226, 2021.

[35] Kevin Zagalo, Liliana Cucu-Grosjean, and Avner Bar-Hen. Identification of execution modes for real-time systems using cluster analysis. In *25th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, pages 1173–1176, 2020.
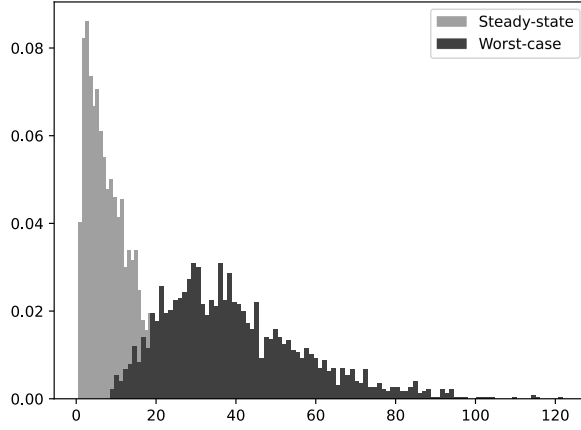
(a) Histogram of the generated response times of τ_3 in Table 1, as defined in Section 6.1.
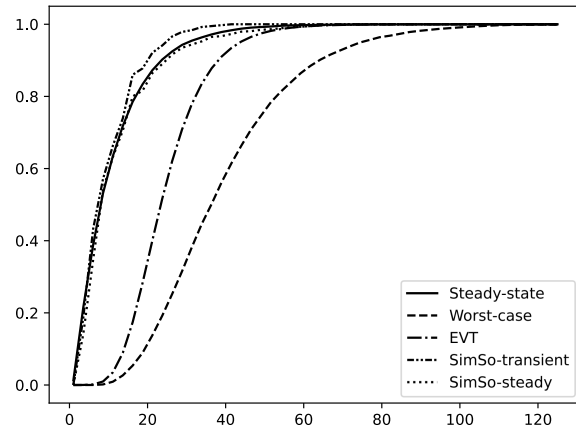
(b) Response time distribution functions of τ_3 in Table 1.

Figure 7: Simulations of a 10 000 instances for the SimSo simulations, steady and transient, EVT estimation of the worst-case response time of the SimSo simulations and simulations of heavy-traffic worst-case response time and steady-state response-time when $\bar{u} = 0.838$.