# Assessing the Robustness of Arrival Curves Models for Real-time Systems

Mahmoud Salem[1][0000−0002−9787−0525], Gonzalo Carvajal[2][0000−0003−1116−6180], Tong Liu[1], and Sebastian Fischmeister[1]

[1] University of Waterloo, Waterloo, ON, Canada
{m4salem, t49liu, sfischme}@uwaterloo.ca
[2] Universidad Técnica Federico Santa María, Valparaíso, Chile
gonzalo.carvajalb@usm.cl

**Abstract.** Design of real-time systems is prone to uncertainty due to software and hardware changes throughout their deployment. In this context, both industry and academia have shown interest in new trace mining approaches for diagnosis and prognosis of complex embedded systems. Trace mining techniques construct empirical models that mainly target achieving high accuracy in detecting anomalies. However, when applied to safety-critical systems, such models lack in providing theoretical bounds on the system resilience to variations from these anomalies. This paper presents the first work that derives robustness criteria on a trace mining approach that constructs arrival-curves models from dataset of traces collected from real-time systems. Through abstracting arrival-curves models to the demand-bound functions of a sporadic task under an EDF scheduler, the analysis presented in the paper enables designers to quantify the permissible change to the parameters of a given task model by relating to the variation expressed within the empirical model. The result is a methodology to evaluate a system to dynamically changing workloads. We evaluate the proposed approach on an industrial cyber-physical system that generates traces of timestamped QNX events.

**Keywords:** Arrival curves · Demand-bound functions· Trace mining

## 1 Introduction

Modern real-time systems are becoming increasingly complex, and their runtime behavior is subject to uncertainties arising from dynamic workloads and changes in their underlying software and hardware. For example, a platform executing a real-time application may suffer a degradation in processor performance if maliciously switched to a low-power mode, or it may sporadically increase its processor demand when handling an anomalous execution scenario. To model and analyze those systems, designers usually apply traditional formal methods that use worst-case analysis to bound any possible workload that can occur at runtime. Although traditional formal methods are relatively mature and have become a standard practice in the industry, they tend to be overly pessimistic and have limited applicability for modern practical systems with dynamic properties.

With the rise of Industry 4.0 and digital twin concepts [8, 23], researchers have started using runtime traces collected from non-invasive tracing tools to improve diagnostics and prognostics. Event traces provide valuable information for performing data-driven analysis when formal methods become complicated or infeasible [1]. For example, formal methods become inadequate when analyzing complex system-level timing requirements of interacting processes. Alternatively, trace mining is proving useful for characterizing real-time systems, as they construct models using traces from different processes, in addition to component-level trace events representing core switching and resource allocations [7].

One relevant open question associated to empirical models constructed from traces is how to evaluate their effectiveness. For example, surveys [4,16,26] highlight that the primary evaluation method of empirical models used for anomaly detection is by their ability to classify normal versus anomalous behavior. However, the current research work shows a lack in the methods that derive robustness bounds on the acceptable behavior of a given system using margins provided by the empirical models. In this context, authors in [9] acknowledge that, unlike traditional formal methods, empirical models for anomaly detection are generally tuned in an ad-hoc manner without guidance by well-found theoretical framework or analysis. As a result, authors claim that there are no guarantees on the effectiveness of the empirical models after deployment.

Authors in [22] show the feasibility of a trace mining approach in modeling the behavior of a real-time system using arrival curves [13] constructed from event traces. The proposed framework computes empirical arrival curves by traversing the trace with a sliding window, capturing the maximum and minimum observation counts of different system events for windows of different length. In a typical classification setting, a normal profile for the system corresponds to a model that aggregates arrival curves computed over a set of representative traces that characterize the normal system behavior. Finally, a classifier uses the model to label unseen traces with a specified accuracy for anomaly detection purposes.

This paper presents an analysis to assess the robustness of arrival-curves models used to characterize the ranges of tolerable behavioral variations of a real-time system such as hardware degradation, external attacks, etc. The presented analysis is based on the assumption that an arrival curve can be analogous to a demand bound function. We state the problem as follows: *Given an empirical arrival curve for a system that can be represented by a sporadic task-set scheduled using an EDF scheduler, and associated upper and lower bounds on allowed variations in the demand of the task set, obtain a range of allowed variations in the task parameters (period, execution time) such that the system stays operational within the allowed variations in the expected overall demand.*

The rest of the paper is organized as follows: We present the background and assumptions for the system models in Section 2. We derive the bounds on the task parameters that correspond to the deviation in the dbf of the task model in Section 3, and we perform an asymptotic analysis to these variation bounds of the task parameters corresponding to the change of the demand deviation in Section 4. In Section 5, we evaluate the robustness assessment framework of empirical arrival-curves models of an actual real-time system. Section 6 discusses the validity of our assumptions, Section 7 reviews the related work, and finally Section 8 concludes the paper.

## 2 Arrival Curves and Demand Bound Functions

This section reviews some basic definitions of arrival curves and establishes a relationship between these curves and demand bound functions for a given task model $T(p, e, d)$ with period $p$, execution time $e$, and deadline $d$. The relationship between arrival curves and dbf provides the basis for the theoretical analysis presented in the paper since our work attempts to fill the gap between the empirically constructed arrival curves and the theoretical models of demand-bound functions that are typically used for formal analysis of a given system. We evaluate and validate the assumptions presented in this section using data from a real-world application in Section 5.

### 2.1 Overview of Arrival Curves

Arrival curves are widely used abstractions for modeling temporal workloads in real-time systems. Multiple frameworks based on Network Calculus [13] rely on arrival curves to model worst-case workloads and perform exhaustive analysis of real-time systems at design-time, obtaining guaranteed performance metrics before system deployment [28]. More recently, multiple authors have shown that analyzing the properties of arrival curves constructed from execution traces collected while the system is operating opens new avenues in applications such as resource management [12, 18] and anomaly detection [22]. The ever-increasing accessibility of system-specific traces from embedded systems and the availability of tools to accelerate the construction of accurate empirical arrival curves [5] facilitate the development of new data-driven methods to complement traditional formal methods in the analysis of modern real-time systems.

Arrival curves are functions of interval time domain that provide upper and lower limits to the number of events that can occur in a system within any time interval of length $\Delta t$. Starting from a timestamped trace of events, it is possible to obtain an empirical arrival curve by sliding a window of varying length $\Delta t$, and registering the maximum and the minimum number of events enclosed within the window while traversing the trace. The resulting curves bound the lower and upper event counts versus the corresponding time interval lengths.

An empirical arrival curve representing a maximum count of events for different interval lengths is a non-decreasing function that starts at the origin [22], and it can be approximated by a line passing through the origin. In the rest of this section, we present the dbf of a sporadic task model under EDF scheduler, which can also be approximated through a line passing through the origin. This assumption allows us to relate the arrival curves with the dbf of a given task-set, enabling us to perform a mathematical analysis for the resilience of systems to the dynamically changing workloads.

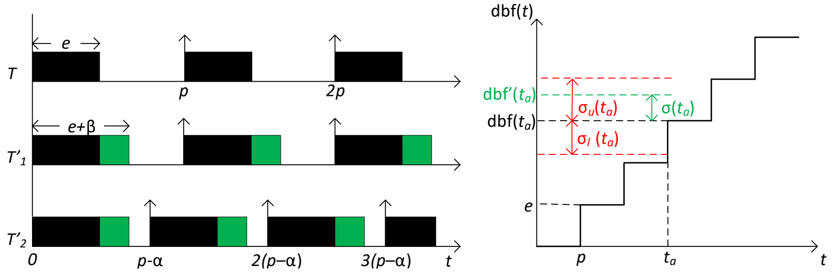


Fig. 1: Graphical representation of variations in task parameters and dbf.

### 2.2 Assumed task model and demand bound functions

**Definition 1.** *A task $T(p, e, d)$ is a dispatchable entity in the system where the period $p$ is the number of time units between successive dispatches, $e$ is the execution time (in time units) required to complete the work, and the deadline $d$ is the maximum time available to complete the work after dispatching.*

A demand-bound function (dbf) models the maximum processor demand by a task over any interval of length $t$ [2]. The dbf of a given sporadic task under EDF assumption is defined as:

$$\mathrm{dbf}(t) = \left\lfloor \frac{t + p - d}{p} \right\rfloor e \tag{1}$$

We will consider that the sporadic task has an implicit deadline $(d = p)$ and there are no overloads, restricting the possible values of $e$ to $]0, p]$, with $p \in \mathbb{R}$.

Due to the empirical nature of the target arrival-curves model, the purpose of the chosen task model is to provide a reasonable approximation to the arrival curve that describes an increasing events count versus an increasing sliding window interval [22]. Hence, we choose the specified sporadic task model with an implicit deadline under EDF scheduler, which yields an increasing function that steps $e$ units every $p$ time units. The function can be approximated by a straight line with slope $\frac{e}{p}$. We evaluate the choice of this task model and the empirical model approximation in Section 5.

Variations in the nominal task parameters can either increase or decrease the task demand. In practical settings, changes in the task parameters may arise from changing operational conditions. We formalize the range of possible values of the altered task parameters as follows:

**Definition 2.** *Decreasing the period of a task.* $\alpha$ *is defined as the reduction of the task period* $p$ *in time units, therefore* $\alpha \in (-\infty, p[$.

**Definition 3.** *Increasing the execution time of a task.* $\beta$ *is defined as the increase of the task execution time* $e$ *in time units, therefore* $\beta \in ]-e, (p-\alpha)-e[$.

We define $\alpha$ as a decrement and $\beta$ as an increment for mathematical convenience. But to generalize our analysis, we highlight that both Definitions 2 and 3 allow negative values for $\alpha$ and $\beta$.

We now introduce the general model for an altered task $T'(p-\alpha, e+\beta)$, which incorporates the variations in period and execution time while maintaining the condition of implicit deadlines but for the altered period in this case, i.e., $(d-\alpha = p-\alpha)$. We can obtain a corresponding altered dbf as follows:

$$\mathrm{dbf}'(t) = \left\lfloor \frac{t}{p-\alpha} \right\rfloor (e+\beta) \tag{2}$$

Let us now consider that for each interval length $t$, we define arbitrary bounds on allowed variations in the nominal dbf from Eq. 1 (with $\alpha = \beta = 0$), restricting the valid values of $\mathrm{dbf}'$ for a given application.

**Definition 4.** *Variation Bound on Task Demand. We denote the allowed variations of the dbf at time interval $t$ as $\sigma(t) = \mathrm{dbf}'(t) - \mathrm{dbf}(t)$, where $\sigma(t) \in [\sigma_l(t), \sigma_u(t)]$, and $\sigma_l(t), \sigma_u(t) \in \mathbb{R}$.*

The restriction in the allowed values of $\sigma(t)$ can be either set by the system designer according to some specific operational accuracy requirement or can represent some uncertainty in the specifications. Note that Definition 4 permits describing deviations above and below the nominal demand. This is a key difference of our analysis with respect to related work on sensitivity analysis from the scheduling domain [20, 27, 29–31], which focuses on verifying that the demand stays below a certain limit such that the system remains schedulable. We contrast our work with sensitivity analysis in Section 7.

Fig. 1 illustrates the previous definitions for the variations in the nominal task parameters and the corresponding dbf. The diagram on the left shows the timeline for the execution of a task $T$ with period $p$ and execution time $e$, and also the execution of tasks $T'_1$ and $T'_2$ that include variations in the nominal parameters. In specific, $T'_1$ increments the nominal execution time by $\beta$ time units (represented in the shaded green box), and $T'_2$ aggregates a reduction in the period. Both $T'_1$ and $T'_2$ generate a demand above the nominal value. The diagram to the right shows the step-wise nominal dbf together with the terms defined earlier for allowed variations at a certain point $t_a$. In this case, the demand of the altered task $\mathrm{dbf}'(t_a)$ is above the nominal value, but within the specified boundaries of allowed variations $\sigma_l(t_a)$ and $\sigma_u(t_a)$.

Considering the previous definitions, we can tackle the problem introduced in Section 1 by finding the region of allowed values of $\alpha$ and $\beta$, such that the value of $\sigma(t)$, representing the deviation in the demand of the altered task $\mathrm{dbf}'(t)$ with respect to the nominal demand $\mathrm{dbf}(t)$, stays within the predefined range $[\sigma_l(t), \sigma_u(t)]$. Traditionally for the assumed task model, a utilization-based approach is the solution to evaluate timing properties of a given real-time system; however, this work uses demand-bound functions since we hypothesize their feasible abstraction to empirical arrival curves as we demonstrate in Section 5.

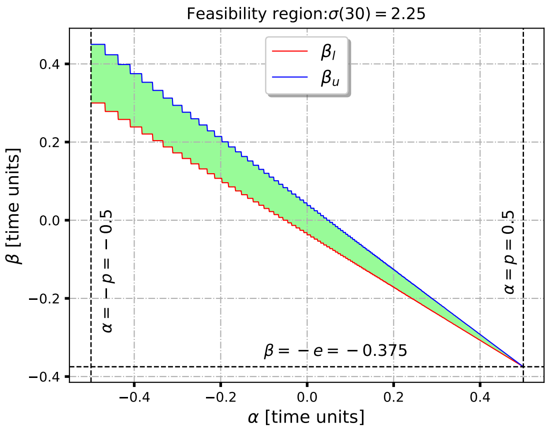


Fig. 2: Permissible task parameter alteration in Example 1

## 3 Computing Bounds on Task-Model Alteration

In this section, we relate the demand deviation bound to a feasibility region for the parameters $\alpha$ and $\beta$ of the altered task model $T'$. The mathematical foundations assume a specified demand variation bound for a given task. However, the analysis presented in this section can be directly extended to specified demand variation bounds for multiple independent tasks, i.e., a task $T_i$ has a specified demand variation bound $\sigma_i$ where $\sum_i \sigma_i = \sigma$. Such problem breaks down into multiple sub-problems that can be solved by finding the feasible region for each $\alpha_i$ and $\beta_i$ for each task $T_i$ separately.

Substituting Eq. 1 and Eq. 2 in Definition 4, we can derive a relationship between $\alpha$ and $\beta$ values that alter a nominal task model $T$ while meeting a deviation demand $\sigma(t_a)$ at a given time interval $t_a$ as follows:

$$\beta = \frac{\sigma(t_a) - \left( \left\lfloor \frac{t_a}{p - \alpha} \right\rfloor - \left\lfloor \frac{t_a}{p} \right\rfloor \right) e}{\left\lfloor \frac{t_a}{p - \alpha} \right\rfloor} \tag{3}$$

The allowed deviation from the nominal dbf is bounded by $[\sigma_l(t_a), \sigma_u(t_a)]$. By replacing $\sigma(t_a)$ by $\sigma_l(t_a)$ in Eq. 3, we can establish a relationship between a lower bound for the parameter $\beta_l$, and the possible values of $\alpha$. In a similar manner, we can replace $\sigma(t_a)$ by $\sigma_u(t_a)$ to obtain the upper bound $\beta_u$.

Fig. 2 illustrates how we can use the relationships described above to obtain a feasibility region for the values of $\alpha$ and $\beta$ given a certain $\sigma(t_a)$. The figure shows a plot of $\beta$ as a function of $\alpha$, in addition to the resulting $\beta_l$ and $\beta_u$. The dashed straight lines delimit the valid intervals for $\alpha$ and $\beta$ according to Definitions 2 and 3, respectively. The lines for $\beta_l$ and $\beta_u$ intersect at the point

$(\alpha, \beta) = (p, -e)$. We restrict the lower bound of $\alpha$ to $-p$, so the range of allowed $\alpha$ values from Definition 2 changes to $[-p,\ p]$. Considering the limits $\beta_l$ and $\beta_u$ and the restrictions over the parameters, we can obtain a feasibility region (shown in shaded green) for the valid combinations of $\alpha$ and $\beta$ that will allow to keep the altered demand within predefined boundaries.

To illustrate the theoretical foundations, we present the following example with concrete task parameters that we will use throughout the rest of the paper.

*Example 1.* Consider a sporadic task with parameters $e = 0.375$ and $p = d = 0.5$. Find the feasibility region for $\alpha$ and $\beta$ such that the demand of the altered task at $t_a = 30$ remains within a range of $\pm 10\%$ of the nominal demand.

Fig. 2 shows the computed upper bound $\beta_u$ and lower bound $\beta_l$ with respect to valid values for $\alpha$ by applying Eq. 3 to the demand of the task in Example 1. The resulting feasibility region for the variations in parameters is shaded green. When drawing a vertical straight line for a given value of $\alpha$, any value of $\beta$ within that region will ensure that the resulting demand from the altered system will remain within the specified variations.

## 4 Asymptotic Analysis for Task Alteration Parameters

This section describes how variations in $\alpha$ and $\beta$ change over increasing time intervals $t$ to meet the specified demand bounds. We use Eq. 3 to establish the relation between $\beta$ and the time interval $t$ for a given $\alpha$. Similarly, for a given $\beta$, the equation defines the relation between $\alpha$ and time interval $t$. Analyzing the change of $\beta$ and $\alpha$ as the time interval $t$ increases gives us an insight into the change of permissible system parameters alteration over different time intervals.

To compute these asymptotic bounds, we need to apply a transformation to Eq. 3 using approximations that are valid for asymptotic values of time intervals.

First, we relate the decrease in period $\alpha$ to the period $p$ using a variable $k$, where $\alpha = k \times p$ such that $k \in (-\infty, 1[$. In other words, the variable $k$ is a ratio of the decrease in period $\alpha$ with respect to the nominal period $p$. Second, we relate $t$ to both $p$ and $\alpha$ by defining $c$ where $t \approx c\ (p - \alpha)$ assuming $c$ is some factor much larger than $(p - \alpha)$. Hence, $t \approx c\ p\ (1 - k)$ as well.

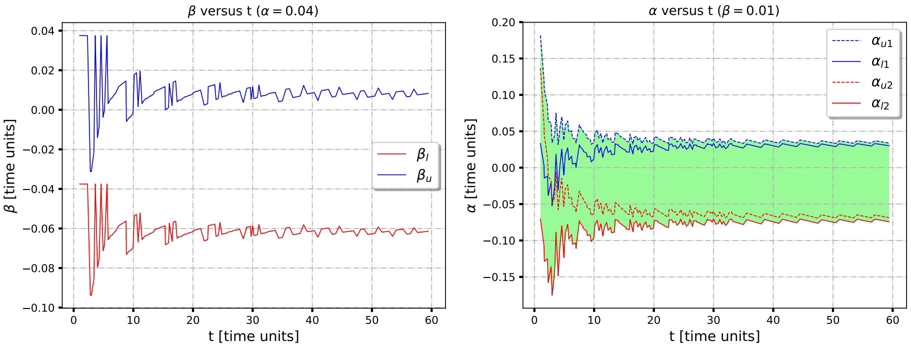
We evaluate these approximation as $t \to \infty$. We obtain the limit of the floor operator using the Squeeze Theorem of Limits [10], which allows us to find $\lim_{x\to\infty} f(x)$ where $f(x)$ is bounded by $g(x)$ and $h(x)$, $g(x) \leq f(x) \leq h(x)$ as follows:

$$\lim_{x\to\infty} g(x) \leq \lim_{x\to\infty} f(x) \leq \lim_{x\to\infty} h(x) \tag{4}$$

Applying Eq. 4 to the definition of floor function, $c - 1 \leq \lfloor c \rfloor < c$, we deduce that $\lim_{c\to+\infty} c - 1 = c$ and $\lim_{c\to+\infty} c = c$, and as a result:

$$\lim_{c\to+\infty} \lfloor c \rfloor = c \tag{5}$$

Similarly, since $(1 - k)$ is a constant. We obtain the following result in Eq. 6. Combining Eq. 5 and Eq. 6 allow for transforming Eq. 3 to obtain $\beta$.



(a) Analysis of $\boldsymbol{\beta}$ using relative $\boldsymbol{\sigma}$

(b) Analysis of $\boldsymbol{\alpha}$ using relative $\boldsymbol{\sigma}$

Fig. 3: Asymptotic analysis for the $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ using relative $\boldsymbol{\sigma}$

$$\lim_{c \to +\infty} \lfloor c\ (1-k) \rfloor = c\ (1-k) \tag{6}$$

**Asymptotic analysis for variation in execution time $\boldsymbol{\beta}$** We consider $\sigma(t)$ values that can be defined relatively to the nominal demand $\mathrm{dbf}(t)$. Let us define $\sigma$ as a fraction $f$ of the nominal demand $\mathrm{dbf}(t)$. For example, the demand variation bound can be set to be ±10% of the nominal demand at any given interval $t$. In this case, to compute the asymptotic values we use Eq. 7 as follows:

$$\sigma(t) = f\ \ \mathrm{dbf}(t) = f\ \left\lfloor \frac{t}{p} \right\rfloor\ e \tag{7}$$

Using $\sigma(t)$ from Eq. 7, the asymptotic values of a function $\beta_v(t)$ that varies with $t$ can be derived as follows:

$$\begin{aligned}
\lim_{t \to +\infty} \beta_v(t) &= \lim_{t \to +\infty} \frac{f \left\lfloor \frac{t}{p} \right\rfloor e - \left( \left\lfloor \frac{t}{p-\alpha} \right\rfloor - \left\lfloor \frac{t}{p} \right\rfloor \right) e}{\left\lfloor \frac{t}{p-\alpha} \right\rfloor} \\
&= \lim_{c \to +\infty} \frac{f \left\lfloor \frac{cp(1-k)}{p} \right\rfloor e - \left( \left\lfloor \frac{c(p-\alpha)}{p-\alpha} \right\rfloor - \left\lfloor \frac{cp(1-k)}{p} \right\rfloor \right) e}{\lfloor c \rfloor} \\
&= \lim_{c \to +\infty} \frac{f \lfloor c(1-k) \rfloor e - (\lfloor c \rfloor - \lfloor c(1-k) \rfloor) e}{\lfloor c \rfloor} \\
&= \lim_{c \to +\infty} \frac{fc(1-k)e - (c - c(1-k))e}{c} \\
&= -ke + f(1-k)e
\end{aligned} \tag{8}$$

Fig. 3a shows the boundaries $\beta_u$ and $\beta_l$ when $\sigma(t) \in [0.9 * \mathrm{dbf}(t), 1.1 * \mathrm{dbf}(t)]$ in Example 1 for an arbitrary value $\alpha = 0.04$. Using Eq. 8, we find the asymptotic values for the boundaries are $\beta_l \approx -0.064$ and $\beta_u \approx 0.0045$. The figure shows that the boundary curve smooths as $t$ increases due to the diminishing effect of the floor operator in Eq. 7.

**Asymptotic analysis for variation in period $\boldsymbol{\alpha}$** For a given value of $\beta$, we study how a varying function $\alpha_v(t)$ changes over time interval $t$ by obtaining the relation between $\alpha_v(t)$ and $t$ from Eq. 3 as follows:

$$Z = \left\lfloor \frac{t}{p - \alpha_v(t)} \right\rfloor = \frac{\sigma + \left\lfloor \frac{t}{p} \right\rfloor e}{e + \beta} \tag{9}$$

Unlike the analysis for the values of $\beta_v(t)$, defining a precise relation between $\alpha_v(t)$ and $t$ for a given $\beta$ is not a straightforward operation. Since the inverse of the floor operator is undefined, we cannot obtain a closed formula for $\alpha_v(t)$. Instead, we restrict the analysis to obtain conservative bounds for the range of $\alpha_v(t)$ values that satisfy Eq. 9. To do this, we can apply the range property of the floor operator [11], which states the following:

$$\lfloor x \rfloor = m \iff m \leq x < m + 1 \tag{10}$$

Using the property in (10), we can describe a range for the values of $\alpha$ $v(t)$ as:

$$p - \frac{t}{Z} \;\leq\; \alpha_v(t) \;<\; p - \frac{t}{Z+1} \text{ , with } Z \;=\; \frac{\sigma + \left\lfloor \frac{t}{p} \right\rfloor e}{(e + \beta)} \tag{11}$$

Substituting $\sigma$ for the specified demand variation bounds $\sigma_l$ and $\sigma_u$ in the obtained inequality, we can obtain the relation of the corresponding boundaries for $\alpha_v(t)$ versus time interval $t$ for a given $\beta$. Note that each boundary for $\sigma$ leads to a feasible range of $\alpha_v(t)$, so we define $Z_u$ and $Z_l$, which we obtain replacing $\sigma_u$ and $\sigma_l$ in the term Z defined in Eq. 9, respectively. Substituting $Z$ by $Z_u$ and $Z_l$ in Eq. 11 yields two inequalities with four boundaries which can be bounded by the $\alpha_v(t)$ in Eq. 12. Now, we show the asymptotic values for both boundaries when the demand variation bound $\sigma$ is defined as a function of nominal demand and we visualize these results in Fig. 3b.

$$p - \frac{t}{Z_l} \;\leq\; \alpha_v(t) \;<\; p - \frac{t}{Z_u + 1} \tag{12}$$

We consider $\sigma(t)$ values that are relative to the nominal demand $\mathrm{dbf}(t)$ where the variation of the demand is constrained by a given range $[\sigma_l, \sigma_u]$ that changes over time intervals $t$. Using the transformation from Eq. 7, the asymptotic values for $\alpha$ can be computed using the boundaries in Eq. 11 again as follows starting with the left-hand side in Eq. 13 then the right-hand size in Eq. 14:

$$p - \frac{t}{Z} = p - \frac{c\ p\ (1-k)}{\frac{f\ c\ (1-k)\ e + c\ (1-k)\ e}{e\ + \beta}} = p\ - \frac{c\ p\ (1-k)(e\ + \beta)}{c\ (1-k)\ e\ (1+f)} = p\ - \frac{p\ (e\ + \beta)}{e\ (1+\ f)} \tag{13}$$

$$p - \frac{t}{Z+1} = p - \frac{cp(1-k)}{\frac{c(1-k)e(1+f)+(e+\beta)}{e+\beta}} = p - \frac{cp(1-k)(e+\beta)}{c(1-k)e(1+f)+(e+\beta)} \tag{14}$$

Then taking the limit as $t$ goes to $\infty$ for Eq. 13 yields the same equation, however, for Eq. 14, we obtain the asymptotic value as follows:

$$\lim_{c \to +\infty} \frac{cp(1-k)}{\frac{c(1-k)e(1+f)+(e+\beta)}{e+\beta}} = \lim_{c \to +\infty} \frac{p(1-k)(e+\beta)}{e(1-k)(1+f) + \frac{(e+\beta)}{c}} = \frac{p(1+\beta)}{e(1+f)} \tag{15}$$

Thus, we conclude that asymptotically, both sides of the inequality will converge to the same limit. According to the Squeeze Theorem, $\alpha$ can be defined asymptotically in this case as:

$$\alpha \approx\ p - \frac{p\ (e+\beta)}{e(1+f)} \tag{16}$$

Figure 3b shows the boundaries for $\alpha_u$ and $\alpha_l$ when $\sigma(t) \in [0.9 * \mathrm{dbf}(t), 1.1 * \mathrm{dbf}(t)]$ in Example 1 for an arbitrary value $\beta = 0.01$. Using Equation 16, we find the asymptotic values for the boundaries are $\alpha_l \approx -0.07$ and $\alpha_u \approx 0.03$. The boundaries in Figure 3 shows the importance of the asymptotic analysis since we can observe that these asymptotic limits are not necessarily the tightest over the time interval $t$. As a result, this analysis provides the designer with a tool to assess the validity of the calculated robustness bounds versus increasing $t$ values.

In the following section, we apply the theoretical analysis to an application of interest by tackling the problem statement introduced in Section 1, which aims at assessing the robustness of data-driven trace mining approach that uses arrival-curves models for a deployed real-time system.

## 5 Application: Robustness Assessment for Empirical Arrival-Curves Models

In this section, we evaluate the hypothesis that an empirical arrival curve can be represented as linear demand-bound functions of the assumed task model in this paper, and as a result, we perform robustness evaluation for the arrival-curves models using the presented theoretical foundations.

The procedure of robustness assessment for the empirical arrival-curve model follows these steps: a) abstract an arrival-curves model to a sporadic task model as in Section 2, b) obtain the relation that describes the feasibility region of the allowed alteration for the task model which corresponds to a variation in the arrival behavior shown by the empirical model as in Section 3, c) evaluate the approach feasibility by quantifying the effect of approximating the curves to a linear demand-bound function of a sporadic task under an EDF scheduler which we demonstrate in this section.

### 5.1 Representing Arrival Curves as Demand-Bound Functions

A model of empirical arrival-curves is an aggregate for the curves computed over a set of multiple traces collected from the system [22]. In our application, we consider a model that is comprised of the mean of arrival curves describing the maximum counts of events within variable window sizes, in addition to two boundary curves that described a confidence interval for that mean. Fig. 4a shows the arrival-curves model of a specified QNX event computed using a set of traces that represent the normal behavior of a real-time system, we discuss the experimental setup later in this section.

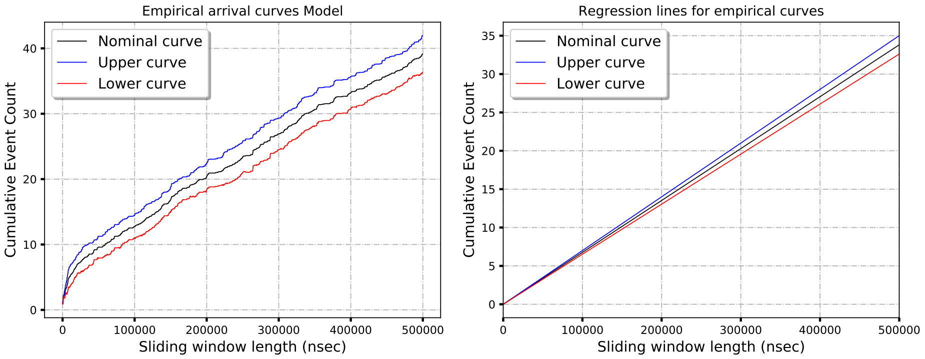
In Section 2, the mathematical foundation uses a task model of a sporadic task using the dbf under an EDF scheduler. As a result, we obtain a demand function, which we can approximate by a line passing through the origin. Similarly, an empirical arrival curve representing a maximum count is a function whose non-decreasing curve starts at the origin [22]. Now, we can introduce the methodology that relates both the arrival curve and the demand-bound function.

We apply Linear Regression [17] to obtain the line that best fits an empirical arrival curve. We offset the fitted line to pass by the origin, and as a result, it can be analogous to a demand-bound function. Later in this section, we quantify the negligible error introduced by this process. For example, Fig. 4b shows the fitted regression lines for the mean arrival curve and the two confidence interval curves after being offset to pass by the origin. The linearity of the curves makes them a good pick for our demonstration.

The regression lines in Fig. 4b are now analogous to a nominal $\mathrm{dbf}(t)$ with an upper and lower variation bound to the demand $\sigma_l$ and $\sigma_u$ respectively which are both functions of $t$. In other words, we can define a sporadic task which demands $e$ execution time units every $p$ time interval whose demand-bound function can be approximated by an empirical arrival curve counting a maximum of $e$ instances of an event in a trace every sliding window of $p$ time units.

### 5.2 Robustness Assessment using Task Alteration Parameters

In order to enable the analysis presented in the previous sections to the robustness assessment of empirical arrival curves, we use the task model assumed in Section 2 to map the task parameters and its variations to the slopes of the regression lines obtained in Fig. 4b. We denote these slopes as, $S$ for the slope of



(a) Empirical arrival-curves model

(b) Corresponding regression lines

Fig. 4: Fitting empirical arrival-curves model to demand-bound functions

the regression line for the mean curve, $S_u$, and $S_l$ for the slopes of the regression lines for both confidence interval curves. We compute these slopes as follows:

$$S = \frac{e}{p}, \quad S_u = \frac{e + \beta_\mathrm{u}}{p - \alpha_\mathrm{u}}, \quad S_l = \frac{e + \beta_\mathrm{l}}{p - \alpha_\mathrm{l}} \tag{17}$$

Eq. 17 defines the relation between task parameters $e$ and $p$ and the regression slopes. We obtain the relations between the variation of parameters $\alpha$ and $\beta$ from Eq. 3 using the definitions of $S_u$ and $S_l$ as follows:

$$\beta_u = \frac{\sigma_u - \left( \left\lfloor \frac{t_a\ S_u}{e + \beta_u} \right\rfloor - \left\lfloor \frac{t_a\ S}{e} \right\rfloor \right)\ e}{\left\lfloor \frac{t_a\ S_u}{e + \beta_u} \right\rfloor}, \quad \beta_l = \frac{\sigma_l - \left( \left\lfloor \frac{t_a\ S_l}{e + \beta_l} \right\rfloor - \left\lfloor \frac{t_a\ S}{e} \right\rfloor \right)\ e}{\left\lfloor \frac{t_a\ S_l}{e + \beta_l} \right\rfloor} \tag{18}$$

The above equations provide the relation between bounds on $\beta$ versus execution time $e$. To define the relation between $\alpha$ and the period $p$, we substitute $e$ by $S \times p$ from Eq. 17:

$$\alpha_u = p - \frac{e + \beta_u}{S_u}, \quad \alpha_l = p - \frac{e + \beta_l}{S_l} \tag{19}$$

The above set of equations provide the relations between the parameters $e$ and $p$, and the corresponding alterations $\beta$ and $\alpha$. The relations evaluate the alteration that would cause a deviation $\sigma$ to the dbfs obtained by approximating the fitted regression lines of the empirical arrival-curves model. Analogously, $\beta$

and $\alpha$ now describe the permissible variation to the arrival-curves model, i.e., the count of events of the corresponding sliding window interval of observance.

Now, we present an application to demonstrate how to use these relations to assess the robustness of a model for a real-time system. We exploit the proposed approach by obtaining the feasibility region for the permissible task parameter variations of the mapped task model through the approximation of the empirical arrival curves to demand-bound functions.

### 5.3 Evaluation on QNX Traces from UAV

The dataset traces are generated from an unmanned aerial vehicle (UAV) running the real-time operating system QNX Neutrino 6.4. The UAV was developed at the University of Waterloo, received the Special Flight Operating Certificate (SFOC), and flew real mapping and payload-drop missions in Nova Scotia and Ontario. The traces are collected using the tracing facility *tracelogger*. A trace entry is a timestamped kernel event that shows the type of an event generated while running a specific process on a specified CPU core. In this section, we represent an arrival-curves model for a specific QNX event `THREAD THRUNNING` that marks every start of a thread execution for a specified process `proc/boot/procnto-instr`. To evaluate the robustness of the example model in Fig. 4a, we perform the following steps:

**a) Compute Regression Slopes.** We obtain the slopes of the fitted regression lines for the mean arrival curve and its confidence interval. It is advisable to assess the adjusted R squared of the regression model. The metric measures the goodness of the linear fit to evaluate whether the assumption that the model is linear was valid [17]. In our example, the slopes of the lines in Fig. 4b can be obtained as $S = 6.76 \times 10^{-5}, S_u = 7.01 \times 10^{-5}$, and $S_l = 6.52 \times 10^{-5}$. The adjusted R squared is 98% indicating a good linear fit.

**b) Choose Task Parameters.** Next step is to specify the task parameters $e$ and $p$ in order to obtain the relation between $\alpha$ and $\beta$ from Eq. 18 and Eq. 19. However, the provided empirical arrival-curves model cannot be used to obtain the $e$ and $p$ values. This comes from the fact that the slopes of fitted regression lines can represent any underlying task model satisfying the relation $S = \frac{e}{p}$.

Therefore, to obtain reasonable values for $e$ and $p$, we need to choose $p$ that is a small fraction of $t_a$ to obtain the asymptotic values for $\alpha$ and $\beta$, in other words, we aim to maximize the factor $c$ defined in Section 4. Additionally, the choice of $p$ or $e$ can be arbitrarily guided by domain knowledge of the system under scrutiny. The other parameter can be estimated using the relation $S = \frac{e}{p}$ from Eq. 17 upon deciding on the value of $p$ or $e$. In our example, we choose an arbitrary value $t_a = 4 \times 10^5$ that captures a sufficient number of trace events. Then, we choose $p = 0.001$ x $t_a = 400$, and as a result, we compute $e = S$ x $p = 0.027$.

**c) Obtain Demand Variation Bound.** The last parameters needed to obtain the relation between $\alpha$ and $\beta$ are $\sigma_u$ and $\sigma_l$. The $\sigma$ parameter defines the difference between the fitted confidence interval curves after offsetting them to pass by the origin $(0, 0)$. Note that $\sigma$ optimally can be expressed as $\sigma =$

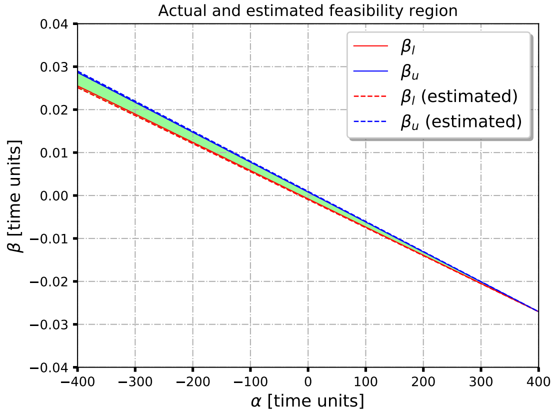


Fig. 5: Feasibility region for representative parameters $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$

$(\mathrm{dbf}' + \mathrm{intercept}') - (\mathrm{dbf} + \mathrm{intercept})$, but we discard the difference between both intercepts as the error resulting from that approximation is negligible. For the example in Fig. 4b, we measure $\sigma_u = -\sigma_l = \sigma'$ where $\sigma' = 0.957$ at $t_a = 4 \times 10^5$.

**d) Apply Feasibility Region Formulas.** To obtain the estimated feasibility region for $\alpha$ and $\beta$, we plot the values of $\beta_u$ and $\beta_l$ for a valid range of $\alpha$ values using the slopes of fitted regression lines from Eq. 18. For the actual feasibility region, we plot the values of $\beta_u$ and $\beta_l$ versus $\alpha$ using Eq. 3. Fig. 5 shows the overlay of both feasibility regions using the arrival-curves model and the actual task model which, similarly to the illustration in Fig. 2, describe the permissible values of $\alpha$ and the corresponding bounds on $\beta$. The negligible error between both the actual and estimated feasibility regions validates that the approximation of a linear empirical arrival-curves model to the assumed demand-bound function is reasonable.

The case study shows an example of an arrival-curves model that characterizes the behavior of QNX kernel event on a real-time system. The empirical model can now be represented as the demand-bound function of an equivalent task model whose parameters alteration can be bounded. The boundaries describe the robustness of the model as it quantifies the variation captured in the underlying normal behavior of the system. Such quantification provides designers a valuable tool on how robust the model is, and allows for comparing different models by assessing the feasibility regions of the task parameter variation.

## 6 Discussion

### 6.1 Linearity Assumption for Arrival Curves

We showed that having an empirical model that can be best approximated by a regression line minimizes the error between the actual and the estimated feasibility region. However, the linearity assumption might not hold for other arrival-curves models. For example, mode-switching [19] yields an increasing arrival curve but with horizontal gaps that correspond to the mode switches, because of the lack of events arrival versus the increasing sliding window size.

### 6.2 Compositionality and Empirical Arrival Curves

We presented an empirical arrival-curves model that corresponds to a single QNX event, however, our work can be extended by using compositionality [6, 24] to combine the task models describing empirical arrival curves originating from multiple events into a system-level task model. In this case, the robustness evaluation can be performed on a system-level which considers inter- and intra-event interactions in contrast to the evaluation using event-level models.

### 6.3 Handling Heterogeneous Task Parameters

Finding the feasibility region for the permissible task variation becomes a more complex problem if there exist different parameters $\alpha_i$ and $\beta_i$ of heterogeneous tasks. One reason is that the mathematical foundation presented in our work assumes that the variations of nominal parameters for multiple tasks are independent. In practice, the tasks of a given real-time system might not encounter the same alteration, and in this case, translating such complex interaction into a single demand-bound function, using compositionality for example, might be a solution that would enable extending our work to multiple dependent tasks.

### 6.4 Iterative Model Assessment for Anomaly Detection

In anomaly detection, it is essential to evaluate whether the model is good enough during the training process. Our approach can be integrated with the model training procedure, such that the model is iteratively evaluated as new traces are added. A designer can limit the model tolerance to a given specification that relates $\alpha$ and $\beta$ as represented in the feasibility region, and then a certification procedure can assess the overlap of this region and the computed one.

## 7 Related Work on Sensitivity Analysis

Our work in this paper assumes that the demand boundaries of a given task are defined and aims to find the feasible task parameters that would not exceed such demand. Contrarily, research work in the domain of schedulability analysis aims to study whether a given set of tasks can be scheduled, i.e., meet the task demand without exceeding a given deadline, using different scheduling methods [2,15,25]. However, in the domain of scheduling, the analysis in this paper can be closely related to sensitivity analysis.

Sensitivity analysis [20, 21, 29, 31] studies how much change to task parameters, i.e., execution time or task period, will not violate scheduling constraints.

The early work on sensitivity analysis [14] computed the maximum variation of all execution time for a given set of tasks that keep a system schedulable for a rate-monotonic scheduler. Further work considered parameters other than execution time, for example, the authors in [3] presents a feasibility space for task deadlines to meet the constraint of schedulability. Authors in [29–31] study the sensitivity analysis for EDF scheduling through the computation of optimal task parameters such that a given system remains schedulable. Particularly, [29] applied sensitivity analysis considering a varying task execution and [30] considered the case when the task period can be varied, while [31] assumed a fixed ratio between relative deadline and period. Our work considers a novel scope by obtaining feasibility regions for the permissible variation of task parameters, without restricting such variation to a single task variation parameter, to meet defined constraints on the increase and decrease to task demand rather than the schedulability condition.

## 8 Conclusion

This paper presents an approach to evaluate the robustness of empirical arrival-curves models that characterize the behavior of real-time systems. We derive theoretical bounds on task parameter alteration permissible by the demand variation represented in the demand-bound function of a sporadic task with an implicit deadline under an EDF scheduler. We demonstrate the feasibility of the approach through an abstraction of an empirical arrival-curves model to a demand-bound function of the assumed task model. We evaluate the approach on the arrival-curves models constructed from QNX operating system events that describe the behavior of a real-time system.

## Acknowledgments

This work was supported by grants FONDECYT 11160375 and CONICYT-Basal Project FB0008.

## References

1. Ahrendts, L., Ernst, R., Quinton, S.: Exploiting execution dynamics in timing analysis using job sequences. IEEE Design & Test **35**(4), 16–22 (2018). https://doi.org/10.1109/MDAT.2017.2746638
2. Baruah, S.K., Mok, A.K., Rosier, L.E.: Preemptively scheduling hard-real-time sporadic tasks on one processor. In: Real-Time Systems Symposium, 1990. Proceedings., 11th. pp. 182–190. IEEE (1990). https://doi.org/10.1109/REAL.1990.128746
3. Bini, E., Buttazzo, G.: The space of EDF deadlines: the exact region and a convex approximation. Real-Time Systems **41**(1), 27–51 (2009). https://doi.org/10.1007/s11241-008-9060-7

4. Cardenas, A.A., Stakhanova, N.: Analysis of metrics for classification accuracy in intrusion detection. In: Empirical Research for Software Security, pp. 173–199. CRC Press (2017). https://doi.org/10.1201/9781315154855
5. Carvajal, G., Salem, M., Benann, N., Fischmeister, S.: Enabling rapid construction of arrival curves from execution traces. IEEE Design and Test **35**(4), 23–30 (Aug 2018). https://doi.org/10.1109/MDAT.2017.2771210
6. Chakraborty, S., Künzli, S., Thiele, L.: A general framework for analysing system properties in platform-based embedded system designs. In: DATE. vol. 3, p. 10190 (2003). https://doi.org/10.1109/DATE.2003.1253607
7. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection for discrete sequences: A survey. Knowledge and Data Engineering, IEEE Transactions on **24**(5), 823–839 (2012). https://doi.org/10.1109/TKDE.2010.235
8. Jazdi, N.: Cyber physical systems in the context of industry 4.0. In: Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on. pp. 1–4. IEEE (2014). https://doi.org/10.1109/AQTR.2014.6857843
9. Juba, B., Musco, C., Long, F., Sidiroglou-Douskos, S., Rinard, M.C.: Principled sampling for anomaly detection. In: NDSS (2015). https://doi.org/10.14722/ndss.2015.23268
10. Knapp, A.W.: Basic real analysis. Springer Science & Business Media (2005). https://doi.org/10.1007/0-8176-4441-5
11. Knuth, D.E., Graham, R.L., Patashnik, O., et al.: Concrete mathematics. Adison Wesley (1989)
12. Lampka, K., Forsberg, B., Spiliopoulos, V.: Keep it cool and in time: With runtime monitoring to thermal-aware execution speeds for deadline constrained systems. Journal of Parallel and Distributed Computing **95**, 79–91 (2016). https://doi.org/10.1016/j.jpdc.2016.03.002
13. Le Boudec, J.Y., Thiran, P.: Network calculus: a theory of deterministic queuing systems for the internet, vol. 2050. Springer Science & Business Media (2001). https://doi.org/10.1007/3-540-45318-0
14. Lehoczky, J., Sha, L., Ding, Y.: The rate monotonic scheduling algorithm: Exact characterization and average case behavior. In: Real Time Systems Symposium, 1989., Proceedings. pp. 166–171. IEEE (1989). https://doi.org/10.1109/REAL.1989.63567
15. Liu, C.L., Layland, J.W.: Scheduling algorithms for multiprogramming in a hard-real-time environment. Journal of the ACM (JACM) **20**(1), 46–61 (1973). https://doi.org/10.1145/321738.321743
16. Milenkoski, A., Vieira, M., Kounev, S., Avritzer, A., Payne, B.D.: Evaluating computer intrusion detection systems: A survey of common practices. ACM Computing Surveys (CSUR) **48**(1), 12 (2015). https://doi.org/10.1145/2808691
17. Neter, J.: Applied linear regression models
18. Neukirchner, M., Axer, P., Michaels, T., Ernst, R.: Monitoring of workload arrival functions for mixed-criticality systems. In: IEEE 34th Real-Time Systems Symposium (RTSS). pp. 88–96 (Dec 2013). https://doi.org/10.1109/RTSS.2013.17
19. Neukirchner, M., Lampka, K., Quinton, S., Ernst, R.: Multi-mode monitoring for mixed-criticality real-time systems. In: Hardware/Software Codesign and System Synthesis (CODES+ ISSS), 2013 International Conference on. pp. 1–10. IEEE (2013). https://doi.org/10.1109/CODES-ISSS.2013.6659021
20. Punnekkat, S., Davis, R., Burns, A.: Sensitivity analysis of real-time task sets. Advances in Computing ScienceâĂŤASIAN'97 pp. 72–82 (1997). https://doi.org/10.1007/3-540-63875-X_44

21. Racu, R., Jersak, M., Ernst, R.: Applying sensitivity analysis in real-time distributed systems. In: Real Time and Embedded Technology and Applications Symposium, 2005. RTAS 2005. 11th IEEE. pp. 160–169. IEEE (2005). https://doi.org/10.1109/RTAS.2005.10
22. Salem, M., Crowley, M., Fischmeister, S.: Anomaly detection using inter-arrival curves for real-time systems. In: Real-Time Systems (ECRTS), 2016 28th Euromicro Conference on. pp. 97–106. IEEE (2016). https://doi.org/10.1109/ECRTS.2016.22
23. Schleich, B., Anwer, N., Mathieu, L., Wartzack, S.: Shaping the digital twin for design and production engineering. CIRP Annals **66**(1), 141–144 (2017). https://doi.org/10.1016/j.cirp.2017.04.040
24. Shin, I., Lee, I.: Compositional real-time scheduling framework. In: Real-Time Systems Symposium, 2004. Proceedings. 25th IEEE International. pp. 57–67. IEEE (2004). https://doi.org/10.1109/REAL.2004.15
25. Spuri, M.: Analysis of deadline scheduled real-time systems (1996)
26. Tavallaee, M., Stakhanova, N., Ghorbani, A.A.: Toward credible evaluation of anomaly-based intrusion-detection methods. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) **40**(5), 516–524 (2010). https://doi.org/10.1109/TSMCC.2010.2048428
27. Vestal, S.: Fixed-priority sensitivity analysis for linear compute time models. IEEE Transactions on Software Engineering **20**(4), 308–317 (1994). https://doi.org/10.1109/32.277577
28. Wandeler, E., Thiele, L., Verhoef, M., Lieverse, P.: System architecture evaluation using modular performance analysis: a case study. International Journal on Software Tools for Technology Transfer **8**(6), 649–667 (2006). https://doi.org/10.1007/s10009-006-0019-5
29. Zhang, F., Burns, A., Baruah, S.: Sensitivity analysis for edf scheduled arbitrary deadline real-time systems. In: Embedded and Real-Time Computing Systems and Applications (RTCSA), 2010 IEEE 16th International Conference on. pp. 61–70. IEEE (2010). https://doi.org/10.1109/RTCSA.2010.12
30. Zhang, F., Burns, A., Baruah, S.: Sensitivity analysis of task period for edf scheduled arbitrary deadline real-time systems. In: Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. vol. 3, pp. 23–28. IEEE (2010). https://doi.org/10.1109/ICCSIT.2010.5564885
31. Zhang, F., Burns, A., Baruah, S.: Task parameter computations for constraint deadline real-time systems with edf scheduling. In: Computer Design and Applications (ICCDA), 2010 International Conference on. vol. 3, pp. V3–553. IEEE (2010). https://doi.org/10.1109/ICCDA.2010.5541363