A Distribution-Agnostic and Correlation-Aware Analysis of Periodic Tasks

Filip Marković[1] Georg von der Brüggen[2] Mario Günzel[2] Jian-Jia Chen[2] Björn B. Brandenburg[1]
[1]*Max Planck Institute for Software Systems, Germany*
[2]*TU Dortmund, Department of Computer Science, Germany*

***Abstract*—Real-time tasks often exhibit correlated execution-time distributions due to common factors such as shared caches, resources, and inputs. Yet state-of-the-art probabilistic analysis still overlooks the impact of correlation, a gap that has been highlighted as a major open problem in the field.**

This paper responds to the open problem with the first *correlation-aware analysis* (CAA) of periodic tasks with stochastic execution times. The proposed analysis, which derives response-time distributions to infer upper bounds on deadline-failure probabilities, applies to a novel task model that incorporates information about both *intra-* and *inter-task* dependencies.

In addition, the paper shows how to statistically infer the two model parameters using confidence intervals obtained via nonparametric bootstrapping. Notably, the inference method described is *distribution-agnostic*, meaning that it does not assume any particular probability distribution *a priori*, thereby eliminating a major risk of misclassifying the ground-truth execution behavior.

By design, CAA dominates state-of-the-art correlation-tolerant analysis (CTA). The significantly better accuracy of CAA is demonstrated via experiments with synthetically generated workloads, while a case study based on the WATERS'17 industrial challenge provides a proof-of-concept of the statistical inference method.

I. Introduction

A major challenge in the analysis of modern real-time systems is that many state-of-the-art methods, including *worst-case execution time* (WCET) analysis, prove ineffective when applied to complex software and hardware stacks. This is mainly due to unpredictable components and effects such as hardware accelerators [50], thermal noise [35], extrinsic nondeterminism in network communication protocols [30], and complex hardware and software stacks designed with *developer productivity* and *code reuse* as the primary goals, as opposed to *performance predictability* [52]. New approaches to analyze and mitigate the timing uncertainties inherent in modern real-time systems are thus urgently needed. In this context, probabilistic analysis has emerged as the most promising direction [16].

The two overarching (and often conflicting) goals in probabilistic analysis for real-time systems are *soundness* and *accuracy*. On the one hand, to ensure soundness, an analysis must not underestimate the probability of adverse events (*e.g.*, a missed deadline). On the other hand, if the estimated probability of failure is much greater than a system's actual, ground-truth risk of failure, the analysis's excessive pessimism will result in overallocation of resources, reduced system efficiency, and ultimately increased costs and environmental impact.

A key and still largely unresolved issue at the core of the tension between analytical soundness and accuracy is the challenge of correlated execution times [16]. While it has long been recognized that ignoring potential correlations among execution-time distributions (*i.e.*, incorrectly assuming that all tasks are independent) can lead to optimistic (*i.e.*, unsound) results [54], it is only recently that research has developed techniques that address the issue in a provably sound manner [5, 46].

The conventional method for circumventing correlation issues uses the notion of a task's *probabilistic worst-case execution time* (pWCET) [2, 5, 16]. This approach assumes that each task's pWCET distribution includes sufficient *padding* to account for and mask any potentially harmful dependencies on the behavior of other tasks. Although correctly padded pWCETs in principle allow the use of independence-assuming analysis methods, it has recently been observed that even just defining the concept of a pWCET is not trivial [5], let alone determining the correct amount of padding. In addition, even correctly padded pWCETs can be a challenge to use properly [12].

Recently, *correlation-tolerant analysis* (CTA) [46] has emerged as a more direct solution to the challenge of analyzing dependent tasks without resorting to false independence assumptions or relying on pWCET-based models. Notably, CTA accommodates *arbitrary* dependencies among tasks while requiring only upper bounds on the expectation and standard deviation of otherwise unknown execution-time distributions.

However, while both CTA and the careful use of padded pWCETs can ensure soundness in the presence of correlated execution times [5, 46], neither is ideal when it comes to accuracy. As we illustrate with an example in Sec. II, CTA, and even more so analyses built on the pWCET abstraction, can suffer from significant inherent pessimism because they only *tolerate* or *mask* correlation, rather than treating it as a first-order feature of the task model being analyzed.

Thus, recent advances [5, 46] notwithstanding, the problems of **(i)** statistically inferring dependencies among execution times and **(ii)** using this data in sound analysis remain largely open. As Davis and Cucu-Grosjean [16] highlight in their list of open issues and key challenges in probabilistic analysis:

- "*How to handle issues relating to dependences between the execution times of jobs of (i) the same task, and (ii) jobs of different tasks? The impact of these dependences may vary based on how strong they are.*" [16]
- "*Appropriate statistical studies are needed to investigate the types of dependences and their impact on probabilistic schedulability analysis. Analyses are needed that can address dependencies.*" [16]

We propose the first solutions to both problems in this paper.

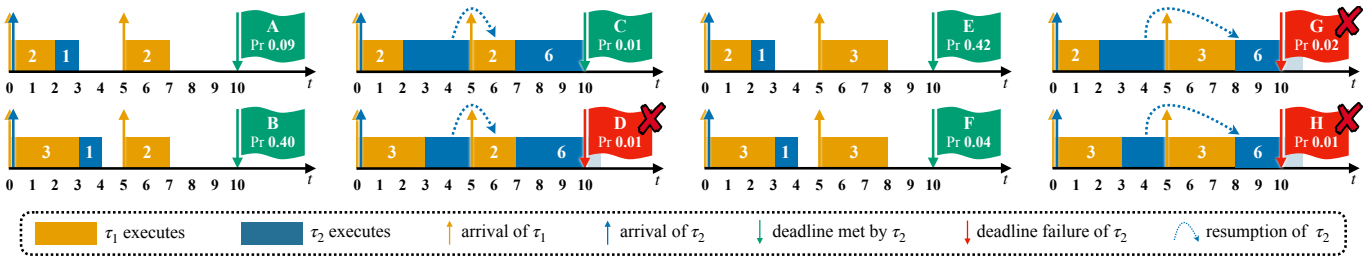

Fig. 1: Ground-truth execution scenarios of three jobs with correlated execution times.

Contribution: We propose a *correlation-aware analysis* (CAA) that estimates the response-time distribution of a given task to derive an upper bound on its deadline-failure probability (Sec. VI). CAA employs a stochastic periodic task model with parameters that capture *intra-task* and *inter-task* correlations, providing insight into execution-time dependence both within and between tasks (as illustrated in Sec. II). The *statistical inference* of these parameters (Sec. VII) uses a *distribution-agnostic* method based on *nonparametric bootstrapping*, which does *not* presuppose any specific probability distribution. A comparison of CAA and CTA using synthetically generated workloads confirms a substantial improvement in analysis accuracy (Sec. VIII). Finally, as a proof-of-concept of the proposed statistical inference method, we report on a case study that evaluated its impact on analysis accuracy (Sec. IX).

II. Motivating Example

We begin with an example that illustrates the importance of inferring and accounting for execution-time correlations. Fig. 1 shows eight possible schedules of two periodic tasks τ_1 and τ_2 with periods 5 and 10, respectively, executing on a uniprocessor under a preemptive scheduling algorithm. Task τ_1 has higher priority; both tasks release a job at time 0.

Ground truth. Suppose there are 8 possible ground-truth execution scenarios (**A**–**H**) in the first 10 time units, as shown in Fig. 1. Each scenario shows two *intra-dependent* jobs of τ_1, denoted $J_{1,1}$ and $J_{1,2}$ and one *inter-dependent* job of τ_2, denoted $J_{2,1}$. For simplicity, we assume the scenarios with the depicted related probabilities periodically repeat every 10 time units and that unfinished jobs are aborted at their deadlines. Hence, considering these 8 execution scenarios is sufficient to determine the ground-truth *deadline-failure probability* (DFP) of τ_2, which is $DFP_2 = 0.04 = 0.02 + 0.01 + 0.01$ (*i.e.*, outcomes **D**, **G** and **H** in Fig. 1). Furthermore, we can infer the following marginal execution-time distributions from Fig. 1:

- for $J_{1,1}$ the execution time (ET) is 2 with probability (Pr) 0.54 and 3 with Pr 0.46;
- for $J_{1,2}$ the ET is 2 with Pr 0.51 and 3 with Pr 0.49;
- for $J_{2,1}$ the ET is 1 with Pr 0.95 and 6 with Pr 0.05.

Assuming independence. It would be convenient to simply ignore any potential correlations and apply efficient independence-assuming methods such as convolution [44] or analytical bounds [11, 45, 56] directly to the marginal execution-time distributions. Alas, this risks underestimating the DFP.

For example, let us determine the cumulative execution time of the jobs based on their marginal execution-time distributions while *falsely assuming independence*. Aggregating scenarios with a total execution demand exceeding 10, we obtain:

- $11 = 2{+}3{+}6$, with probability $0.01323 = 0.54{\cdot}0.49{\cdot}0.05$;
- $11 = 3{+}2{+}6$, with probability $0.01173 = 0.46{\cdot}0.51{\cdot}0.05$;
- $12 = 3{+}3{+}6$, with probability $0.01127 = 0.46{\cdot}0.49{\cdot}0.05$.

Thus, in total, we obtain a DFP estimate of only $0.03623 = 0.01323{+}0.01173{+}0.01127$, which incorrectly under-estimates the ground-truth failure risk of $DFP_2 = 0.04$.

Clearly, assuming independence in systems exhibiting dependent behavior is not an option, as has long been known [54].

Prior work: over-approximation with pWCET. As already discussed in Sec. I, the main idea underlying pWCET-based approaches is to pad execution-time distributions with sufficient pessimism such that efficient independence-assuming methods [*e.g.*, 11, 44, 45, 56] can be applied.

Returning to the example in Fig. 1, we used a Python Jupyter notebook from prior work [47] to compute the *least pessimistic* pWCET distributions possible that comply with Bozhko et al.'s recent rigorous pWCET definition [5]. Due to space constraints, we omit a detailed discussion on the pWCET derivation, and report the finally computed DFP estimate of $0.873313 > 0.04$.

The pessimistic nature of the pWCET is intrinsic to its definition, as it hides dependencies by introducing pessimism through redistribution, *i.e.*, by increasing the likelihood of longer execution times. The reported DFP estimate thus represents the optimal outcome for *any* pWCET-based analysis method (applied to this workload), before factoring in any additional method-specific pessimism (which can be considerable).

Prior work: tolerating correlation with CTA. Recall that CTA [46] requires, for each task, bounds on the expected value and standard deviation of the ground-truth execution-time distribution of any of its jobs. With this information, CTA bounds DFP directly, *i.e.*, without using the pWCET abstraction, such that the derived bound is sound irrespective of any potential correlations.

In our example, the tightest possible bounds for τ_1 are its exact execution-time expectation $\widehat{e}_1 = 2.49 = 2{\cdot}0.51{+}3{\cdot}0.49$ and standard deviation $\widehat{s}_1 = 0.5$. Similarly, for τ_2, $\widehat{e}_2 = 1.25 = 1 \cdot 0.95 + 6 \cdot 0.05$ and $\widehat{s}_2 = 1.09$. Given these inputs, CTA yields a DFP estimate of $0.235 \approx \frac{(2{\cdot}\widehat{s}_1{+}\widehat{s}_2)^2}{(2{\cdot}\widehat{s}_1{+}\widehat{s}_2)^2{+}(D_2{-}(2{\cdot}\widehat{e}_1{+}\widehat{e}_2))^2}$ for task τ_2. Although this result is considerably tighter than what any pWCET analysis can achieve, CTA still significantly overestimates the ground-truth DFP. This is because it only implicitly *tolerates* correlation through standard deviation bounds (*e.g.*, $\widehat{s}_1$ and $\widehat{s}_2$), rather than accounting for it explicitly.

This paper: analyzing correlation with CAA. We propose a novel analysis inspired by CTA that uses explicit bounds on the correlation between job executions. Specifically, our approach leverages bounds on *covariance*, which measures how job-execution-time distributions vary together.

Returning to the example in Fig. 1, the observed covariance among the three jobs is as follows:

- $c_1 = -0.1754$ covariance between $J_{1,1}$ and $J_{1,2}$,
- $c_2 = -0.015$ covariance between $J_{1,1}$ and $J_{2,1}$, and
- $c_3 = 0.0275$ covariance between $J_{1,2}$ and $J_{2,1}$.

CAA uses the *maximum* observed covariance among jobs within the same task (*intra-task covariance*) and between jobs of different tasks (*inter-task covariance*), which in our case is

- $\widehat{v}_{1,1} = c_1 = -0.1754$, and
- $\widehat{v}_{1,2} = \max(c_2, c_3) = \max(-0.015, 0.0275) = 0.0275$.

Using these two values in addition to expectation ($\widehat{e}_1$ and $\widehat{e}_2$) and standard deviation ($\widehat{s}_1$ and $\widehat{s}_2$) bounds, CAA derives a DFP bound that is twice as tight compared to CTA (0.235) as $DFP_2 \leq \widehat{\Upsilon} / \left(\widehat{\Upsilon} + (D_2 - (2 \cdot \widehat{e}_1 + \widehat{e}_2))^2\right) \approx 0.1$, where $\widehat{\Upsilon} = 2 \cdot \widehat{s}_1^2 + \widehat{s}_2^2 + 2 \cdot (\widehat{v}_{1,1} + 2 \cdot \widehat{v}_{1,2})$. In the rest of the paper, we prove the soundness of this bound (Theorem 6) and describe a distribution-agnostic statistical method for inferring all the bounds used above (Sec. VII).

III. Probability Theory Background

We briefly review the probability theory concepts used in this paper and introduce our notation. We consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω is the set of all possible outcomes, $\mathcal{F} \subseteq 2^\Omega$ is the event space, and $\mathbb{P} : \mathcal{F} \to [0, 1]$ is a probability function. Let $\mathbb{R}$ denote the set of real numbers and $\overline{\mathbb{R}} \triangleq \mathbb{R} \cup \{\pm\infty\}$ the extended set of real numbers.

Def. 1 (Random variable)**.** A random variable X on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is a measurable function $X : \Omega \to \mathbb{R}$ such that $\{\omega \in \Omega : X(\omega) = x\} \in \mathcal{F}$ for all $x \in \mathbb{R}$.

We use $\mathbb{P}[\omega \in \Omega \,|\, X(\omega) = x]$ to denote the probability of random variable X taking the value x, or briefly, $\mathbb{P}[X = x]$. We define the following functions on random variables.

Def. 2 (Expected value)**.** The *expected value* $\mathbb{E}[X] \in \overline{\mathbb{R}}$ of a random variable X is a measure of the central tendency (*i.e.*, average value) of the possible outcomes of X

$$\mathbb{E}[X] \triangleq \int_{\omega \in \Omega} X(\omega)\, d\mathbb{P}$$

The expectation operator $\mathbb{E}[\cdot]$ acts linearly on sums of random variables, a property known as *linearity of expectation*.

Fact 1 (Linearity of expectation [*e.g.*, 25, p. 40])**.** *If X and Y are two (possibly dependent) random variables such that $\mathbb{E}[X]$ and $\mathbb{E}[Y]$ are finite, then*

$$\forall a, b \in \mathbb{R},\ \ \mathbb{E}[a \cdot X + b \cdot Y] = a \cdot \mathbb{E}[X] + b \cdot \mathbb{E}[Y].$$

Def. 3 (Covariance)**.** The *covariance* of two random variables X and Y, denoted by $\mathrm{Cov}[X, Y]$, is a measure of the degree to which X and Y fluctuate together.

$$\mathrm{Cov}[X, Y] \triangleq \mathbb{E}[(X - \mathbb{E}[X]) \cdot (Y - \mathbb{E}[Y])]$$

Intuitively, when $\mathrm{Cov}[X, Y] > 0$, increases in X are generally associated with increases in Y. Conversely, when $\mathrm{Cov}[X, Y] < 0$, increases in X are generally associated with decreases in Y. Note that $\mathrm{Cov}[X, Y] = \mathrm{Cov}[Y, X]$.

Def. 4 (Variance)**.** The *variance* of a random variable X, denoted by $\mathbb{V}[X]$, is a measure of the dispersion (or spread) of the possible outcomes of X.

$$\mathbb{V}[X] \triangleq \mathrm{Cov}[X, X]$$

In the following, we write $\mathbb{V}[X] < \infty$ and $\mathbb{E}[X] < \infty$ to denote that a random variable X has finite variance and mean.

Def. 5 (Standard deviation)**.** The *standard deviation* (SD) of a random variable X, denoted $\sigma[X]$, is given by

$$\sigma[X] \triangleq \sqrt{\mathbb{V}[X]}.$$

We often rely on Bienaymé's identity when calculating the variance of a sum of random variables.

Fact 2 (Bienaymé's identity [*e.g.*, 31, Eq. 5.2])**.** *Consider n possibly dependent random variables $X_1, X_2, \ldots, X_n$ such that $\mathrm{Cov}[X_i, X_j] < \infty$ for $1 \leq i \leq j \leq n$, then*

$$\mathbb{V}\left[\sum_{i=1}^{n} X_i\right] = \sum_{i,j=1}^{n} \mathrm{Cov}[X_i, X_j].$$

We also use the following covariance bound.

Fact 3 (Covariance bounds [*e.g.*, 27, Inequality 1])**.** *Let X and Y be possibly dependent random variables such that $\mathbb{V}[X] < \infty$ and $\mathbb{V}[Y] < \infty$, then covariance is bounded by*

$$-\sqrt{\mathbb{V}[X] \cdot \mathbb{V}[Y]} \leq \mathrm{Cov}[X, Y] \leq \sqrt{\mathbb{V}[X] \cdot \mathbb{V}[Y]}.$$

Furthermore, Cantelli's inequality upper-bounds the probability of a random variable deviating from its mean.

Theorem 1 (Cantelli's inequality [6])**.** *For an arbitrary random variable X such that $\mathbb{V}[X] < \infty, \mathbb{E}[X] < \infty$, and any $t > \mathbb{E}[X]$,*

$$\mathbb{P}[X \geq t] \leq \frac{\mathbb{V}[X]}{\mathbb{V}[X] + (t - \mathbb{E}[X])^2}.$$

Cantelli's bound remains valid if used with upper bounds on $\mathbb{V}[X]$ and $\mathbb{E}[X]$, which prior work observed as follows.

Fact 4 ([46, Lemma 1])**.** *Let $f(e, v) = \frac{v}{v + (t-e)^2}$. If $e_1 \leq e_2 < t$ and $0 \leq v_1 \leq v_2$, then $f(e_1, v_1) \leq f(e_2, v_2)$.*

A key mathematical building block underlying CTA [46] is the following theorem, which restates Cantelli's inequality for sums of random variables with suitable upper bounds on the sums of their expectations and standard deviations.

Theorem 2 (Correlation-tolerant inequality [46, Corollary 2])**.** *Let $X_1, \ldots, X_n$ be n possibly dependent random variables such that $\mathrm{Cov}[X_i, X_j] < \infty$ for all $i, j \in \{1, \ldots, n\}$, with upper bounds $\widehat{e} \geq \sum_{i=1}^{n} \mathbb{E}[X_i]$ and $\widehat{s} \geq \sum_{i=1}^{n} \sigma[X_i]$. Then, for any $t > \widehat{e}$,*

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq t\right] \leq \frac{\widehat{s}^{\,2}}{\widehat{s}^{\,2} + (t - \widehat{e})^2}.$$

Compared to the original Corollary 2 in [46], our restated Theorem 2 slightly generalizes the required upper bounds $\widehat{e}$ and $\widehat{s}$.

Remarkably, Theorem 2 provides an upper bound on $\mathbb{P}[\sum_{i=1}^{n} X_i \geq t]$ robust to arbitrary and *unknown* dependencies among the variables $X_1, \ldots, X_n$. However, the bound can be significantly improved if covariance information is available, as discussed in Sec. II. We therefore next derive an analogue to Theorem 2 that explicitly incorporates covariance bounds.

IV. Correlation-Aware Inequality

As before in Theorem 2, suppose we are given n random variables $X_1, \ldots, X_n$ and that an upper bound $\widehat{e}$ on the sum of their expectations is known. However, rather than a bound $\widehat{s}$ on the sum of their n (individual) standard deviations, suppose we have a bound $\widehat{c}$ on the sum of their n^2 *pairwise* covariances. It is then possible to simply replace $\widehat{s}^2$ with $\widehat{c}$ in Theorem 2.

Theorem 3 (Correlation-aware inequality)**.** *Let $X_1, \ldots, X_n$ be n possibly dependent random variables with covariance $\mathrm{Cov}[X_i, X_j] < \infty$ for all $i, j \in \{1, \ldots, n\}$. If $\widehat{e} \geq \sum_{i=1}^{n} \mathbb{E}[X_i]$ and $\widehat{c} \geq \sum_{i,j=1}^{n} \mathrm{Cov}[X_i, X_j]$, then, for any $t > \widehat{e}$,*

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq t\right] \leq \frac{\widehat{c}}{\widehat{c} + (t - \widehat{e})^2}.$$

Proof.

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i \geq t\right] \overset{\text{(i)}}{\leq} \frac{\mathbb{V}[\sum_{i=1}^{n} X_i]}{\mathbb{V}[\sum_{i=1}^{n} X_i] + (t - \mathbb{E}[\sum_{i=1}^{n} X_i])^2}$$

$$\overset{\text{(ii)}}{=} \frac{\mathbb{V}[\sum_{i=1}^{n} X_i]}{\mathbb{V}[\sum_{i=1}^{n} X_i] + (t - \sum_{i=1}^{n} \mathbb{E}[X_i])^2}$$

$$\overset{\text{(iii)}}{=} \frac{\sum_{i,j=1}^{n} \mathrm{Cov}[X_i, X_j]}{\sum_{i,j=1}^{n} \mathrm{Cov}[X_i, X_j] + (t - \sum_{i=1}^{n} \mathbb{E}[X_i])^2}$$

$$\overset{\text{(iv)}}{\leq} \frac{\widehat{c}}{\widehat{c} + (t - \widehat{e})^2}$$

Step (i) follows from Theorem 1 for $X = \sum_{i=1}^{n} X_i$. Steps (ii) and (iii) follow from Facts 1 and 2, respectively. Finally, Step (iv) follows from Fact 4 applied with the antecedents $\widehat{e}$ and $\widehat{c}$. The prerequisites of Fact 4 are met since any random variable's variance is necessarily non-negative and, by Fact 2, $\sum_{i,j=1}^{n} \mathrm{Cov}[X_i, X_j] = \mathbb{V}[\sum_{i=1}^{n} X_i] \geq 0$. □

While Theorem 3 closely resembles Theorem 2, the superficial similarity is misleading: the bound $\widehat{c}$ used in Theorem 3 can be much smaller than the term $\widehat{s}^2$ used in Theorem 2, resulting in a substantial accuracy advantage. Theorem 3 is the foundation for the probabilistic analysis developed in Sec. VI. In preparation, we next define the underlying system model, its ground-truth stochastic behavior, and recall parts of CTA [46].

V. Ground-Truth System Model and CTA

We consider a set $\tau \triangleq \{\tau_1, \tau_2, \ldots, \tau_n\}$ of n periodic tasks running on a uniprocessor or within a partitioned multiprocessor platform under fixed-priority preemptive scheduling. Tasks are indexed by decreasing priority; τ_1 holds the highest priority, and each task has a unique priority level.

TABLE I: Overview of Notation

Symbol	Explanation
τ	The task set.
τ_i	The task from τ with index i.
T_i	Period of τ_i.
D_i	Relative deadline of τ_i.
$\alpha_i(\Delta)$	Upper bound on no. of τ_i arrivals within Δ time units.
Ω	Sample space of system evolutions.
$\omega \in \Omega$	A sample (particular system evolution) from Ω.
$J_{i,j}$	The j-th job of τ_i after system startup.
$a_{i,j}$	Arrival time of $J_{i,j}$.
$d_{i,j}$	Absolute deadline of $J_{i,j}$.
$\mathcal{C}_{i,j}(\omega)$	Execution time of $J_{i,j}$ in ω (specific to each $\omega \in \Omega$).
$\mathcal{R}_{i,j}(\omega)$	Response time of $J_{i,j}$ in evolution ω.
DFP_i	The ground-truth DFP of τ_i.
$\widehat{f(\cdot)}$	An upper bound on the given function $f(\cdot)$.

Each task τ_i has a *period* T_i that defines the difference in time between any two consecutive job releases and a *constrained relative deadline* $D_i \leq T_i$. We use $\alpha_i(\Delta) \triangleq \left\lceil \frac{\Delta}{T_i} \right\rceil$ to bound the number of jobs of τ_i released within any Δ-long time interval. We assume discrete time, modeled with the set $\mathbb{N}$ of natural numbers (*e.g.*, representing processor cycles).

We let $J_{i,j}$ denote the j-th job of task τ_i after system startup, where $j \in \mathbb{N}$. We indicate by $a_{i,j}$ the *arrival time* of $J_{i,j}$ (or, interchangeably, its *release time*), which is the same across all system evolutions. The *absolute deadline* of job $J_{i,j}$ is $d_{i,j} = a_{i,j} + D_i$. When a job misses its deadline, it is immediately cut off from service (*i.e.*, aborted and discarded).

All model parameters defined so far are constants. In contrast, job execution times are modeled as *potentially dependent* random variables. Recall from Sec. III that Ω represents the set of all possible outcomes (*i.e.*, system evolutions), with each $\omega \in \Omega$ representing a single evolution. For each $\omega \in \Omega$, $\mathcal{C}_{i,j}(\omega)$ is the *execution-time requirement* of $J_{i,j}$ in evolution ω, *i.e.*, the amount of processor service $J_{i,j}$ must receive to finish. We make no assumptions on the distribution of any $\mathcal{C}_{i,j}$, nor do we require all of a task's jobs to follow the same distribution.

Finally, in a system evolution $\omega \in \Omega$ in which a job $J_{i,j}$ is not aborted, we let $\mathcal{R}_{i,j}(\omega)$ denote $J_{i,j}$'s *ground-truth response time*, *i.e.*, the time between $J_{i,j}$'s release at $a_{i,j}$ and its completion time. If $J_{i,j}$ is aborted in ω, then by definition $\mathcal{R}_{i,j}(\omega) \triangleq D_i + 1$. Table I summarizes our notation.

A. Bounding the Ground-Truth Deadline-Failure Probability

Our ultimate goal is to safely characterize the ground-truth response-time distribution $\mathcal{R}_{i,j}$ of any job of a given task τ_i. This in turn will provide us with an upper bound on the task's ground-truth *deadline-failure probability*, as defined below.

$$DFP_i \triangleq \max_{j \in \mathbb{N}} \mathbb{P}[\mathcal{R}_{i,j} > D_i]$$

In our analysis, we build on definitions and lemmas originally developed in the context of CTA [46]. We briefly recall the elements that we reuse in the following and refer interested readers to Marković et al. [46] for a detailed derivation.

In the interval from the release until the completion or abortion of an arbitrary job $J_{i,j}$, jobs of higher-priority tasks and $J_{i,j}$ itself are continuosly executed at all times. Thus, given a duration $\Delta \in (0, D_i]$, we can assess whether the response time $\mathcal{R}_{i,j}$ exceeds Δ by accumulating the execution time of all higher-or-equal priority jobs that can possibly execute in the interval $[a_{i,j}, a_{i,j} + \Delta)$. Since every incomplete job is aborted when reaching its deadline, the set

$$\Lambda_{i,j}(\Delta) \triangleq \{\, \mathcal{C}_{k,\ell} : 1 \leq k \leq i,\ a_{k,\ell} \in [m(i,j,k), a_{i,j}+\Delta) \,\},$$

where $m(i,j,k) \triangleq \max(0, a_{i,j} - D_k + 1)$, captures all relevant execution-time random variables. From $\Lambda_{i,j}(\Delta)$, we obtain a parametric upper bound $\widehat{\mathcal{R}}_{i,j}(\Delta)$ on total demand [46, Eq. 4].

$$\widehat{\mathcal{R}}_{i,j}(\Delta) \triangleq \textstyle\sum_{\mathcal{C}_{k,\ell} \in \Lambda_{i,j}(\Delta)} \mathcal{C}_{k,\ell} \tag{1}$$

The significance of $\widehat{\mathcal{R}}_{i,j}(\Delta)$ is that it is at least as likely to exceed any Δ as the ground-truth response time is to exceed D_i.

Fact 5 ([46, Lemma 6]). *For all $\tau_i \in \tau$, $j \in \mathbb{N}$, $\Delta \in (0, D_i]$,*

$$\mathbb{P}[\mathcal{R}_{i,j} > D_i] \ \leq\ \mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big].$$

Fact 5 yields the following upper bound on task τ_i's DFP.

Theorem 4 ([46, Theorem 3]). *For all $\tau_i \in \tau$,*

$$DFP_i \leq \max_{j \in \mathbb{N}} \min_{\Delta \in (0, D_i]} \left\{ \mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big] \right\}.$$

Theorem 4 is the starting point of our analysis. In preparation, we note two useful properties of $\Lambda_{i,j}(\Delta)$. For notational brevity, we define $\lambda_{i,j,k}(\Delta) \triangleq \{\, \mathcal{C}_{k,\ell} : \ell \in \mathbb{N} \,\} \cap \Lambda_{i,j}(\Delta)$ to denote the subset of random variables corresponding to task τ_k in $\Lambda_{i,j}(\Delta)$.

Lemma 1. *For $\tau_h, \tau_i \in \tau$, $h < i$, $j \in \mathbb{N}$ and $\Delta \in (0, D_i]$,*

$$|\lambda_{i,j,h}(\Delta)| \leq \alpha_h(\Delta) + 1.$$

Proof. The LHS of the inequality counts the number of random variables in $\Lambda_{i,j}(\Delta)$ corresponding to jobs of a given higher-priority task τ_h. Recall that $\Lambda_{i,j}(\Delta)$ includes a random variable for each job of τ_h released in an interval with a length of at most $\Delta + D_h - 1$. There are at most $\alpha_h(\Delta + D_h - 1) = \left\lceil \frac{\Delta + D_h - 1}{T_h} \right\rceil \leq \left\lceil \frac{\Delta}{T_h} \right\rceil + \left\lceil \frac{D_h - 1}{T_h} \right\rceil = \alpha_h(\Delta) + 1$ such jobs. □

Lemma 2. *For $\tau_i \in \tau$, $j \in \mathbb{N}$, and $\Delta \in (0, D_i]$, $|\lambda_{i,j,i}(\Delta)| = 1$.*

Proof. Exactly one job of task τ_i contributes a random variable to $\Lambda_{i,j}(\Delta)$ since $\Lambda_{i,j}(\Delta)$ includes random variables for all jobs of τ_i released during $[\max(0, a_{i,j} - D_i + 1), a_{i,j} + \Delta)$, and as $D_i \leq T_i$, there is only one such job ($J_{i,j}$ itself). □

B. Review of Correlation-Tolerant Analysis

Since we will later compare our analysis with CTA [46], we will briefly review how it applies to periodic tasks. Crucially, CTA requires two bounds $\widehat{e}_k$ and $\widehat{s}_k$ to be given for each task τ_k.

Analysis Input 1. $\widehat{e}_k$ is an upper bound on the mean execution time of any job of τ_k: $\forall j \in \mathbb{N}, \widehat{e}_k \geq \mathbb{E}[\mathcal{C}_{k,j}]$.

Analysis Input 2. $\widehat{s}_k$ is an upper bound on the standard deviation of the execution time of any τ_k's job: $\forall j \in \mathbb{N}, \widehat{s}_k \geq \sigma[\mathcal{C}_{k,j}]$.

Taking into account Lemmas 1 and 2, CTA uses the following cumulative bounds to account for each job in $\Lambda_{i,j}(\Delta)$.

$$\widehat{E}(i, \Delta) \triangleq \widehat{e}_i + \textstyle\sum_{h=1}^{i-1} (\alpha_h(\Delta) + 1) \cdot \widehat{e}_h \tag{2}$$

$$\widehat{S}(i, \Delta) \triangleq \widehat{s}_i + \textstyle\sum_{h=1}^{i-1} (\alpha_h(\Delta) + 1) \cdot \widehat{s}_h \tag{3}$$

In broad strokes, CTA then starts from Theorem 4 and, using Theorem 2 with Eqs. (2) and (3) as respectively $\widehat{e}$ and $\widehat{s}$, ultimately arrives at the following correlation-tolerant DFP bound.

Theorem 5 (CTA DFP bound [46, Theorem 4]). *For all $\tau_i \in \tau$ and $\Delta \in (0, D_i]$, if $0 < \widehat{E}(i, \Delta) < \Delta$, then:*

$$DFP_i \leq \left(\widehat{S}(i,\Delta)\right)^2 \cdot \left(\left(\widehat{S}(i,\Delta)\right)^2 + \left(\Delta - \widehat{E}(i,\Delta)\right)^2 \right)^{-1}.$$

As discussed, Theorem 5 holds in the face of unknown dependencies among execution times. We next present a novel analysis that improves upon Theorem 5 in terms of accuracy by explicitly accounting for covariance, using Theorem 3.

VI. Correlation-Aware Analysis

The main idea is to incorporate information about the correlation between random variables along with their expected values and standard deviations. Hence, in addition to Analysis Inputs 1 and 2, suppose we are given bounds on *intra-* and *inter-task* correlation via the following two task parameters.

Analysis Input 3. $\widehat{v}_{k,k}$ is an upper bound on the *intra-task covariance* of the execution time of any two different jobs of τ_k: $\widehat{v}_{k,k} \geq \mathrm{Cov}[\mathcal{C}_{k,j}, \mathcal{C}_{k,\ell}]$ for all $j, \ell \in \mathbb{N}$ and $j \neq \ell$.

Analysis Input 4. $\widehat{v}_{k,q}$ is an upper bound on the *inter-task covariance* of the execution time of any two jobs of distinct tasks τ_k and τ_q: $\widehat{v}_{k,q} \geq \mathrm{Cov}[\mathcal{C}_{k,j}, \mathcal{C}_{q,\ell}]$ for all $j, \ell \in \mathbb{N}$.

These bounds may be omitted for some tasks, as it is possible to derive safe (but pessimistic) upper bounds from Analysis Input 2 for any task for which the input lacks explicit bounds $\widehat{v}_{k,k}$ and $\widehat{v}_{k,q}$ (via Fact 3, $\widehat{v}_{k,q} \leq \widehat{s}_k \cdot \widehat{s}_q$ and $\widehat{v}_{k,k} \leq \widehat{s}_k \cdot \widehat{s}_k$).

As a first step towards exploiting Analysis Inputs 3 and 4, we connect the conservative characterization of a given job's response time $\widehat{\mathcal{R}}_{i,j}(\Delta)$ from Eq. (1) with Theorem 3. For brevity, given a set S, we write $(S)^2$ to denote the Cartesian product $S \times S$, *e.g.*, $(\Lambda_{i,j}(\Delta))^2$ denotes $\Lambda_{i,j}(\Delta) \times \Lambda_{i,j}(\Delta)$.

Lemma 3. *For any $\tau_i \in \tau$, $j \in \mathbb{N}$, and $\Delta \in (0, D_i]$, given $\widehat{e} \geq \sum_{\mathcal{C} \in \Lambda_{i,j}(\Delta)} \mathbb{E}[\mathcal{C}]$ and $\widehat{c} \geq \sum_{(\mathcal{C}, \mathcal{C}') \in (\Lambda_{i,j}(\Delta))^2} \mathrm{Cov}[\mathcal{C}, \mathcal{C}']$, if $0 < \widehat{e} < \Delta$, then*

$$\mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big] \leq \widehat{c} \cdot \left(\widehat{c} + (\Delta - \widehat{e})^2\right)^{-1}.$$

Proof. $\mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big] \overset{\text{(i)}}{=} \mathbb{P}\Big[\sum_{\mathcal{C} \in \Lambda_{i,j}(\Delta)} \mathcal{C} > \Delta\Big] \overset{\text{(ii)}}{\leq} \widehat{c} \cdot \left(\widehat{c} + (\Delta - \widehat{e})^2\right)^{-1}$

where Step (i) expands Eq. (1) and Step (ii) applies Theorem 3. □

As before, we can easily obtain a suitable bound $\widehat{e}$ from Analysis Input 1 via Eq. (2). Deriving a bound $\widehat{c}$ from Analysis Inputs 2–4 is more challenging. To this end, we define:

$$\widehat{\Upsilon}(i,\Delta) \triangleq \left(\sum_{k=1}^{i} \widehat{V}_{i,k}(\Delta) + \widehat{\mathrm{IC}}_{i,k}(\Delta)\right) + \sum_{h=1}^{i-1}\sum_{q=h+1}^{i} \widehat{\mathrm{XC}}_{i,h,q}(\Delta), \tag{4}$$

where

$$\widehat{V}_{i,k}(\Delta) \triangleq \begin{cases} (\alpha_k(\Delta)+1)\cdot \widehat{s}_k^2 & i \neq k \\ \widehat{s}_i^2 & i = k, \end{cases}$$

$$\widehat{\mathrm{IC}}_{i,k}(\Delta) \triangleq \begin{cases} (\alpha_k(\Delta)+1)\cdot \alpha_k(\Delta)\cdot \widehat{v}_{k,k} & i \neq k \\ 0 & i = k, \text{ and} \end{cases}$$

$$\widehat{\mathrm{XC}}_{i,h,q} \triangleq \begin{cases} 2\cdot(\alpha_h(\Delta)+1)\cdot(\alpha_q(\Delta)+1)\cdot \widehat{v}_{h,q} & i \neq q \\ 2\cdot(\alpha_h(\Delta)+1)\cdot \widehat{v}_{h,q} & i = q. \end{cases}$$

We establish in the following that $\widehat{\Upsilon}(i,\Delta)$ upper-bounds the *total covariance sum* $\sum_{\mathcal{C},\mathcal{C}' \in (\Lambda_{i,j}(\Delta))^2} \mathrm{Cov}[\mathcal{C},\mathcal{C}']$, as required to use Lemma 3 with $\widehat{c} = \widehat{\Upsilon}(i,\Delta)$. To start, we split the total covariance sum into three summands capturing **(I)** variance, **(II)** intra-task covariance, and **(III)** inter-task covariance.

Lemma 4.
$$\overbrace{\sum_{\substack{(\mathcal{C},\mathcal{C}')\in \\ (\Lambda_{i,j}(\Delta))^2}} \mathrm{Cov}[\mathcal{C},\mathcal{C}']}^{\text{total covariance sum}} = \overbrace{\sum_{k=1}^{i} \sum_{\substack{(\mathcal{C},\mathcal{C})\in \\ (\lambda_{i,j,k}(\Delta))^2}} \mathrm{Cov}[\mathcal{C},\mathcal{C}]}^{\text{I variance sum}} +$$
$$\underbrace{\sum_{k=1}^{i} \sum_{\substack{(\mathcal{C},\mathcal{C}')\in \\ (\lambda_{i,j,k}(\Delta))^2, \\ \mathcal{C}\neq\mathcal{C}'}} \mathrm{Cov}[\mathcal{C},\mathcal{C}']}_{\text{II intra-task covariance sum}} + \underbrace{\sum_{h=1}^{i-1}\sum_{q=h+1}^{i} \sum_{\substack{(\mathcal{C},\mathcal{C}')\in \\ (\lambda_{i,j,h}(\Delta)\times \\ \lambda_{i,j,q}(\Delta))}} 2\cdot\mathrm{Cov}[\mathcal{C},\mathcal{C}']}_{\text{III inter-task covariance sum}}$$

Proof. Consider any pair of random variables $(\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}) \in (\Lambda_{i,j}(\Delta))^2$, *i.e.*, any covariance term $\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ contributing to the LHS of the equation. If $k = q$ and $\ell = r$, then $\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ is accounted for by summand I. If $k = q$ but $\ell \neq r$, then $\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ is accounted for by summand II. If $k < q$, then $\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ is accounted for by a term $2\cdot\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ in summand III. Finally, for $k > q$, recall that $2\cdot\mathrm{Cov}[\mathcal{C},\mathcal{C}'] = \mathrm{Cov}[\mathcal{C},\mathcal{C}'] + \mathrm{Cov}[\mathcal{C}',\mathcal{C}]$. Thus, if $k > q$, then $\mathrm{Cov}[\mathcal{C}_{k,\ell},\mathcal{C}_{q,r}]$ is implicitly accounted for by the term $2\cdot\mathrm{Cov}[\mathcal{C}_{q,r},\mathcal{C}_{k,\ell}]$ in summand III. Conversely, any covariance term on the RHS is also part of the LHS sum. □

We now bound each of the three summands in turn.

Lemma 5. $\forall \tau_i,\tau_k \in \tau,\ k \leq i,\ j \in \mathbb{N},\ \Delta \in (0, D_i]$,

$$\widehat{V}_{i,k}(\Delta) \geq \textstyle\sum_{(\mathcal{C},\mathcal{C}) \,\in\, (\lambda_{i,j,k}(\Delta))^2} \mathrm{Cov}[\mathcal{C},\mathcal{C}]$$

Proof. Observe that, by Defs. 4 and 5 and the definition of Analysis Input 2, $\widehat{s}_k^2 \geq \mathbb{V}[\mathcal{C}] = \mathrm{Cov}[\mathcal{C},\mathcal{C}]$ for any $\mathcal{C} \in \lambda_{i,j,k}(\Delta)$. Further note that $\sum_{(\mathcal{C},\mathcal{C})\in(\lambda_{i,j,k}(\Delta))^2} \mathrm{Cov}[\mathcal{C},\mathcal{C}] = \sum_{\mathcal{C}\in\lambda_{i,j,k}(\Delta)} \mathrm{Cov}[\mathcal{C},\mathcal{C}] \leq |\lambda_{i,j,k}(\Delta)|\cdot\widehat{s}_k^2$. If $k \neq i$, then there are at most $\alpha_k(\Delta)+1$ random variables in $\lambda_{i,j,k}(\Delta)$ by Lemma 1. Otherwise, if $k = i$, there is exactly one random variable in $\lambda_{i,j,k}(\Delta)$ by Lemma 2. The bound follows. □

Next, we bound the intra-task covariance sum (II).

Lemma 6. $\forall \tau_i,\tau_k \in \tau,\ k \leq i,\ j \in \mathbb{N},\ \Delta \in (0, D_i]$,

$$\widehat{\mathrm{IC}}_{i,k}(\Delta) \geq \textstyle\sum_{(\mathcal{C},\mathcal{C}') \,\in\, (\lambda_{i,j,k}(\Delta))^2,\ \mathcal{C}\neq\mathcal{C}'} \mathrm{Cov}[\mathcal{C},\mathcal{C}']$$

Proof. By the definition of Analysis Input 3, $\mathrm{Cov}[\mathcal{C},\mathcal{C}'] \leq \widehat{v}_{k,k}$ for any two distinct $\mathcal{C},\mathcal{C}' \in \lambda_{i,j,k}(\Delta)$. If $i \neq k$, then, by Lemma 1, there are at most $(\alpha_k(\Delta)+1)$ random variables in $\lambda_{i,j,k}(\Delta)$, and thus at most $(\alpha_k(\Delta)+1)\cdot\alpha_k(\Delta)$ pairs of non-identical random variables in $(\lambda_{i,j,k}(\Delta))^2$. If $i = k$, then by Lemma 2 there is exactly one random variable in $\lambda_{i,j,k}(\Delta)$, and hence there are no pairs of distinct random variables. □

Finally, we bound the inter-task covariance sum (III).

Lemma 7. $\forall \tau_h,\tau_q,\tau_i \in \tau, h < q \leq i, \forall j \in \mathbb{N}, \forall \Delta \in (0, D_i]$,

$$\widehat{\mathrm{XC}}_{i,h,q}(\Delta) \geq \textstyle\sum_{(\mathcal{C},\mathcal{C}') \,\in\, (\lambda_{i,j,h}(\Delta)\times\lambda_{i,j,q}(\Delta))} 2\cdot\mathrm{Cov}[\mathcal{C},\mathcal{C}']$$

Proof. By the definition of Analysis Input 4, $\mathrm{Cov}[\mathcal{C},\mathcal{C}'] \leq \widehat{v}_{h,q}$ for any two variables $(\mathcal{C},\mathcal{C}') \in (\lambda_{i,j,h}(\Delta)\times\lambda_{i,j,q}(\Delta))$. By Lemma 1, there are at most $\alpha_h(\Delta)+1$ random variables in $\lambda_{i,j,h}(\Delta)$. If $i \neq q$, then there are, also by Lemma 1, at most $\alpha_q(\Delta)+1$ random variables in $\lambda_{i,j,q}(\Delta)$, and thus at most $(\alpha_h(\Delta)+1)\cdot(\alpha_q(\Delta)+1)$ ordered pairs in the Cartesian product $\lambda_{i,j,h}(\Delta)\times\lambda_{i,j,q}(\Delta)$. Otherwise, if $i = q$, then by Lemma 2 there is exactly one random variable in $\lambda_{i,j,q}(\Delta)$, and hence there are at most $\alpha_h(\Delta)+1$ such ordered pairs. □

Taken together, we observe that $\widehat{\Upsilon}(i,\Delta)$ as defined in Eq. (4) indeed bounds the total covariance of jobs in $\Lambda_{i,j}(\Delta)$.

Lemma 8. $\forall \tau_i \in \tau,\ j \in \mathbb{N},\ \Delta \in (0, D_i]$,

$$\widehat{\Upsilon}(i,\Delta) \geq \textstyle\sum_{(\mathcal{C},\mathcal{C}')\in(\Lambda_{i,j}(\Delta))^2} \mathrm{Cov}[\mathcal{C},\mathcal{C}']$$

Proof. Follows from Lemma 4 by applying Lemmas 5–7 to Summands I-III, respectively. □

With Lemma 8 in place, we can finally instantiate Lemma 3 with the upper bounds $\widehat{e} = \widehat{E}(i,\Delta)$ and $\widehat{c} = \widehat{\Upsilon}(i,\Delta)$.

Lemma 9. $\forall \tau_i \in \tau,\ j \in \mathbb{N},\ \Delta \in (0, D_i]$, *if* $0 < \widehat{E}(i,\Delta) < \Delta$,

$$\mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big] \;\leq\; \widehat{\Upsilon}(i,\Delta)\cdot\left(\widehat{\Upsilon}(i,\Delta) + \Big(\Delta - \widehat{E}(i,\Delta)\Big)^2\right)^{-1}$$

Proof. The inequality follows from Lemma 3 applied to Eqs. (2) and (4), as respectively justified by Fact 1 and Lemma 8. □

Lastly, using Theorem 4 as the starting point, Lemma 9 yields the final DFP bound.

Theorem 6 (Correlation-aware DFP analysis)**.** *For all* $\tau_i \in \tau$ *and any* $\Delta \in (0, D_i]$, *if* $0 < \widehat{E}(i,\Delta) < \Delta$, *then*

$$DFP_i \;\leq\; \widehat{\Upsilon}(i,\Delta)\cdot\left(\widehat{\Upsilon}(i,\Delta) + \Big(\Delta - \widehat{E}(i,\Delta)\Big)^2\right)^{-1}. \tag{5}$$

Proof. $DFP_i \overset{\text{(i)}}{\leq} \max_{j\in\mathbb{N}} \min_{\Delta^\star\in(0,D_i]} \Big\{\mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta^\star) > \Delta^\star\Big]\Big\}$

$$\overset{\text{(ii)}}{\leq} \max_{j\in\mathbb{N}} \Big\{\mathbb{P}\Big[\widehat{\mathcal{R}}_{i,j}(\Delta) > \Delta\Big]\Big\}$$

$$\overset{\text{(iii)}}{\leq} \max_{j\in\mathbb{N}}\left\{\widehat{\Upsilon}(i,\Delta)\cdot\left(\widehat{\Upsilon}(i,\Delta)+\left(\Delta-\widehat{E}(i,\Delta)\right)^2\right)^{-1}\right\}$$
$$\overset{\text{(iv)}}{\leq} \widehat{\Upsilon}(i,\Delta)\cdot\left(\widehat{\Upsilon}(i,\Delta)+\left(\Delta-\widehat{E}(i,\Delta)\right)^2\right)^{-1}$$

Step (i) is Theorem 4. Step (ii) follows since $\forall\Delta \in (0, D_i], \min_{\Delta^*\in(0,D_i]} F(\Delta^*) \leq F(\Delta)$ for any F. Step (iii) follows from Lemma 9. Finally, Step (iv) follows trivially since j no longer appears in the term being maximized. □

If Analysis Inputs 3 and 4 are substituted for all tasks via Fact 3, then CAA reduces to CTA. If, however, covariance bounds are available, then CAA naturally extends CTA.

Theorem 7 (Dominance)**.** *If $\widehat{\upsilon}_{k,q} \leq \widehat{s}_k \cdot \widehat{s}_q$ for all $\tau_k, \tau_q \in \tau$, then the upper bound on DFP_i in Theorem 6 (CAA) is lower than or equal to the upper bound on DFP_i in Theorem 5 (CTA).*

Proof. We show that $\widehat{\Upsilon}(i,\Delta) \leq \left(\widehat{S}(i,\Delta)\right)^2$, which implies that Theorem 6 dominates Theorem 5 due to the monotonicity of both bounds (Fact 4). For brevity, let $A_x \triangleq \alpha_x(\Delta)$ since Δ is constant throughout the proof.

$$\widehat{\Upsilon}(i,\Delta) \overset{\text{(i)}}{=} \left(\sum_{k=1}^{i}\widehat{V}_{i,k}(\Delta)+\widehat{\mathrm{IC}}_{i,k}(\Delta)\right)+\sum_{h=1}^{i-1}\sum_{q=h+1}^{i}\widehat{\mathrm{XC}}_{i,h,q}(\Delta)$$
$$\overset{\text{(ii)}}{=} \left(\left(\widehat{s}_i^2+0\right)+\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h^2+(A_h+1)\cdot A_h\cdot\widehat{\upsilon}_{h,h}\right)+$$
$$\left(\sum_{h=1}^{i-1}2\cdot(A_h+1)\cdot\widehat{\upsilon}_{h,i}+\sum_{q=h+1}^{i-1}2\cdot(A_h+1)\cdot(A_q+1)\cdot\widehat{\upsilon}_{h,q}\right)$$
$$\overset{\text{(iii)}}{\leq} \widehat{s}_i^2+\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h^2+(A_h+1)\cdot A_h\cdot\widehat{s}_h\cdot\widehat{s}_h\right)+$$
$$2\cdot\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\cdot\widehat{s}_i+\sum_{q=h+1}^{i-1}(A_h+1)\cdot(A_q+1)\cdot\widehat{s}_h\cdot\widehat{s}_q\right)$$
$$\overset{\text{(iv)}}{=} \widehat{s}_i^2+\left(\sum_{h=1}^{i-1}(A_h+1)^2\cdot\widehat{s}_h^2\right)+$$
$$2\cdot\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\cdot\widehat{s}_i+\sum_{q=h+1}^{i-1}(A_h+1)\cdot(A_q+1)\cdot\widehat{s}_h\cdot\widehat{s}_q\right)$$
$$\overset{\text{(v)}}{=} \widehat{s}_i^2+2\cdot\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\cdot\widehat{s}_i\right)+\left(\sum_{h=1}^{i-1}(A_h+1)^2\cdot\widehat{s}_h^2\right)+$$
$$2\cdot\left(\sum_{h=1}^{i-1}\sum_{q=h+1}^{i-1}(A_h+1)\cdot(A_q+1)\cdot\widehat{s}_h\cdot\widehat{s}_q\right)$$
$$\overset{\text{(vi)}}{=} \widehat{s}_i^2+2\cdot\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\cdot\widehat{s}_i\right)+\left(\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\right)^2$$
$$\overset{\text{(vii)}}{=} \left(\widehat{s}_i+\sum_{h=1}^{i-1}(A_h+1)\cdot\widehat{s}_h\right)^2 \overset{\text{(viii)}}{=} \left(\widehat{s}_i+\sum_{h=1}^{i-1}(\alpha_h(\Delta)+1)\cdot\widehat{s}_h\right)^2$$
$$\overset{\text{(ix)}}{=} \left(\widehat{S}(i,\Delta)\right)^2$$

Step (i) is Eq. (4). Step (ii) renames the index variable of the first sum from k to h, breaks out the case of $h = i$ in the first sum as well as the case of $q = i$ in the nested sum, and finally replaces $\widehat{V}_{i,k}(\Delta)$, $\widehat{\mathrm{IC}}_{i,k}(\Delta)$, and $\widehat{\mathrm{XC}}_{i,h,q}(\Delta)$ with their respective definitions. Step (iii) follows from the antecedent $\widehat{\upsilon}_{k,q} \leq \widehat{s}_k \cdot \widehat{s}_q$. Step (iv) rewrites the first sum using the distributive law $(A_h+1)^2\cdot\widehat{s}_h^2 = (A_h+1)\cdot\widehat{s}_h^2\cdot 1+(A_h+1)\cdot\widehat{s}_h^2\cdot A_h$. Step (v) simply splits the second sum and rearranges the resulting summands. Step (vi) rewrites the latter two sums using the expansion of a squared summation $(\sum_h^{i-1} x_h)^2 = \sum_{h=1}^{i-1} x_h^2 + 2\sum_{h=1}^{i-1}\sum_{q=h+1}^{i-1} x_h x_q$, where $x_h = (A_h+1)\cdot\widehat{s}_h$. Step (vii) rewrites $(a+b)^2 = a^2+2ab+b^2$ for $a = \widehat{s}_i$ and $b = \sum_{h=1}^{i-1}(A_h+1)\cdot s_h$. Finally, Step (viii) replaces A_h with its definition and Step (ix) is Eq. (3). □

Next we explain how to obtain Analysis Inputs 1–4.

VII. Statistical Inference

CAA depends on the ability to infer the per-task *statistical parameters* $\widehat{e}_k$, $\widehat{s}_k$, $\widehat{\upsilon}_{k,k}$, and $\widehat{\upsilon}_{k,q}$, in a robust and economical manner. Specifically, we require a *distribution-agnostic* statistical inference procedure, as execution-time distributions are expected to vary widely in practice, and incorrectly assuming the wrong underlying distribution can lead to unsound results [*e.g.*, 29, Ch. 1]. Additionally, for practical purposes, the procedure should have modest sampling requirements, meaning it must provide high-confidence estimates from relatively small sample sizes. A well-known method that satisfies both requirements is *non-parametric bootstrapping* [55]. We briefly review how it generally works and then apply it to our case.

A. Review: Bootstrap Confidence Intervals

Let X be a random variable with an unknown distribution. We are interested in estimating a *ground-truth parameter* θ of X, such as $\mathbb{E}[X]$, $\mathbb{V}[X]$, or a similar quantity. Nonparametric bootstrapping [55] is a classic method for inferring an *estimate* $\widetilde{\theta}$ of θ from a finite sample of X collected during a *random experiment*. Crucially, bootstrapping is more robust than simply computing $\widetilde{\theta}$ on the given sample, as it approximates the *sampling distribution* of $\widetilde{\theta}$. That is, it assesses how the sample estimate $\widetilde{\theta}$ of the true parameter θ varies due to sampling noise, and accounts for this uncertainty when determining the final estimate. To this end, bootstrapping with a *sample size* G and *bootstrap count* B works as follows:

- B1 Draw an *initial sample* $x_1, x_2, \ldots, x_G$ of G independent observations of the random variable X. No assumption is made on the ground-truth distribution of X besides that it does not change between observations. Thus, the samples are *independent and identically distributed* (i.i.d.).
- B2 Generate a *bootstrap sample* $x_1^*, x_2^*, \ldots, x_G^*$ by randomly resampling with replacement from the initial sample.
- B3 Compute the *bootstrap statistic* $\widetilde{\theta}^* = f(x_1^*, x_2^*, \ldots, x_G^*)$ on the bootstrap sample (*e.g.*, the mean to estimate $\mathbb{E}[X]$).
- B4 Repeat steps B2 and B3 to obtain B *bootstrap samples* $\beta_1, \ldots, \beta_B$ with corresponding estimates $\widetilde{\theta}_1^*, \ldots, \widetilde{\theta}_B^*$.
- B5 The *bootstrap distribution* $\widetilde{\theta}_1^*, \widetilde{\theta}_2^*, \ldots, \widetilde{\theta}_B^*$ approximates the sampling distribution of the estimated parameter $\widetilde{\theta}$.

Intuitively, the variance of the bootstrap distribution (for sufficiently large B) is a measure of how "noisy" the inference process is. More formally, from $\widetilde{\theta}_1^*, \widetilde{\theta}_2^*, \ldots, \widetilde{\theta}_B^*$ we can infer a *confidence interval* (CI) [*e.g.*, 17, 55] that is likely to contain the true population parameter θ with arbitrary given confidence. To compute a γ-CI for θ, where γ is a given *level of confidence* (*e.g.*, $\gamma = 0.95$ for a 95% CI), we proceed as follows:

P1 *Sort* the bootstrap estimates $\widetilde{\theta}_1^*, \ldots, \widetilde{\theta}_B^*$ in increasing order.
P2 Let $l = (1-\gamma)/2$ and $u = 1 - l$. Find the l-th and u-th *percentiles*, denoted Q_l and Q_u.
P3 The γ-CI for θ is $[Q_l, Q_u]$.

For sufficiently large G and $\gamma < 1$, the γ-CI has the property that, if the entire bootstrap procedure is repeated multiple times, then the fraction of inferred γ-CIs that contain the true value of θ is in expectation at least γ [*e.g.*, 53]. For example, if we bootstrap a 99% CI ($\gamma = 0.99$) with the above procedure (*i.e.*, steps B1–B5 and P1–P3) 100 times to obtain 100 (different) CIs, then we expect the true θ of the ground-truth distribution to lie outside of the computed γ-CI only once.

In the context of this paper, the goal is to derive an upper bound on θ. As the interval $[Q_l, Q_u]$ is likely to contain the parameter θ (with the desired confidence γ), we use the interval's upper bound as the final estimate: $\widehat{\theta} \triangleq Q_u$.

B. CAA Input Parameter Inference

A random experiment on a real system is simply a repeated trial of the system's execution behavior under predetermined conditions, producing a sample $x_1, x_2, \ldots, x_G$ consisting of G execution traces. However, there are some subtle nuances involved in applying bootstrapping to periodic tasks. First, we must set a fixed *horizon* (or *trace length*) H that determines the length of each trial (*i.e.*, how many jobs are measured). For example, H could be some number of hyperperiods (*e.g.*, in automotive systems with typically short hyperperiods [32]).

Second, recall from Sec. V that the execution cost of each job is a *separate* random variable. Each time we record a trace of execution times (for each job of each task released up to time H), we collect *one* observation for each job. We are thus effectively running many instances of the bootstrapping procedure simultaneously. The sample size G is the total number of traces that must be collected.

Third, and crucially, the measurement setup must be a *valid random experiment*, ensuring that the i.i.d. assumption underlying the bootstrapping method is met and that the sampled distributions are indicative of the real system's behavior. For example, to ensure independence, the system and all relevant parts of its environment can be rebooted (otherwise reset to an initial state) before a trial is recorded, and the tracing setup must ensure the absence of any harmful measurement bias in the observed execution-time distributions.

Finally, job abortion poses a problem as we cannot observe the full execution requirement of a job when it is aborted. Thus, if a job $J_{k,j}$ aborts or does not complete by time H in a given trace, we conservatively assume that its execution requirement was $D_{k,j} + 1$, which safely over-approximates its real demand.

Under these assumptions, let $J_{k,j}(x^*)$ denote the j-th job of τ_k in trace x^*, $\mathbb{J}_k$ all jobs of τ_k in a trace of length H, and $\mathcal{C}_{k,j}(x^*)$ the execution-time request of $J_{k,j}(x^*)$.

Bootstrapping the sample mean and standard deviation for each job following Sec. VII-A results in the upper bounds $\widehat{\mathbb{E}}[\mathcal{C}_{k,j}]$ and $\widehat{\sigma}[\mathcal{C}_{k,j}]$ for each $J_{k,j} \in \mathbb{J}_k$. Analysis Inputs 1 and 2 are directly computed from these.

$$\widehat{e}_k \triangleq \max_{J_{k,j} \in \mathbb{J}_k} \{\widehat{\mathbb{E}}[\mathcal{C}_{k,j}]\} \tag{6}$$

$$\widehat{s}_k \triangleq \max_{J_{k,j} \in \mathbb{J}_k} \{\widehat{\sigma}[\mathcal{C}_{k,j}]\} \tag{7}$$

Computing Analysis Inputs 3 and 4 is somewhat more complex, as it involves all pairs of jobs. Using the same sample $x_1, x_2, \ldots, x_G$ underlying Eqs. (6) and (7), we bootstrap a bound on the sample covariance for each pair of sampled jobs, denoted $\widehat{\mathrm{Cov}}[\mathcal{C}_{k,j}, \mathcal{C}_{q,l}]$ for $J_{k,j} \in \mathbb{J}_k$, $J_{q,l} \in \mathbb{J}_q$, and $\tau_k, \tau_q \in \tau$ (see Casella and Berger [9] for an extensive introduction to sample statistics). Finally, Analysis Inputs 3 and 4 are computed from the pairwise bounds as:

$$\widehat{v}_{k,k} \triangleq \max_{J_{k,j}, J_{k,l} \in (\mathbb{J}_k \times \mathbb{J}_k),\ j \neq l} \widehat{\mathrm{Cov}}[\mathcal{C}_{k,j}, \mathcal{C}_{k,l}], \tag{8}$$

$$\widehat{v}_{k,q} \triangleq \max_{J_{k,j}, J_{q,l} \in (\mathbb{J}_k \times \mathbb{J}_q)} \widehat{\mathrm{Cov}}[\mathcal{C}_{k,j}, \mathcal{C}_{q,l}]. \tag{9}$$

The inference procedure is agnostic to practical factors such as cache behavior, the number of cores, any effects of cross-core interference, *etc.*, and therefore applies equally on uniprocessors and within multicore processors under partitioned scheduling [8]. However, the conducted random experiment must ensure that the combined effects of all such factors are adequately reflected in the collected sample, which may require exploring different *scenarios of operation*. In this case, if a system's stochastic execution behavior is explored through multiple experiments to test various predetermined scenarios of operation (*e.g.*, different co-runners or memory stressors, cache pressure, environmental inputs, *etc.*), Eqs. (6)–(9) must reflect the maximum across all experiments.

Even then, no matter how many scenarios are sampled and how many observations are made, Eqs. (6)–(9) represent a crucial generalization step: the bootstrapped bounds, rooted in the observation of *finite* traces, are *lifted* to task-level bounds applicable to *all* jobs. This step relies on the *representative sampling* assumption at the heart of any statistical approach—the initial sample must accurately reflect the characteristics of the larger population under analysis—a concept long studied in both fundamental statistics [33] and measurement-based probabilistic analysis of real-time systems [15]. Ensuring representativeness is a complex challenge, but not specific to non-parametric bootstrapping and beyond the scope of this paper.

Ultimately, it is important to recognize that *no* statistical inference method can provide *absolute certainty*: there is always a minuscule, but non-zero chance that a ground-truth parameter lies outside the statistically estimated range, irrespective of the statistical method used. For applications that cannot tolerate such residual risk under any circumstances, classical WCET analysis remains the only viable option (where available).

For the wider class of modern systems with more flexible requirements, however, bootstrapped CIs provide an excellent means of estimating the ground truth that is statistically rigorous, distribution-agnostic, sample-efficient, and mathematically well understood. In such systems, it is crucial to understand the methods and parameters employed to infer the inputs underlying analysis results. DFP bounds should thus always be reported and understood in the context of the inference settings used. We explore the impact of the parameters G, B, and γ on the DFP bounds through a case study in Sec. IX.

VIII. Evaluation

We evaluated the accuracy of the proposed CAA and the baseline CTA [46] methods using synthetically generated workloads. Given the absence of real-world benchmarks or standard procedures for generating stochastic task parameters in the real-time systems community, we explored a wide range of potential parameters, following a setup similar to [46]. Specifically, we examined various relationships among the inputs of CTA and CAA to assess their accuracy in producing DFP bounds across different workload profiles. Starting from a base setup, we designed four distinct experiments, each altering a key parameter used to generate synthetic workloads. For each generated task set, we obtained DFP bounds for the lowest-priority task τ_ℓ using both CTA and CAA.

Experimental setup. For each combination of the workload-generation parameters n, U^{mean}, $r^{\max}$, and $\text{c}^{\max}$ defined below, we randomly generated 5000 periodic task sets with the following procedure. For a given task-set size n, we randomly selected n periods from the set $1, 2, 5, 10, 20, 50, 100, 200, 500, 1000$ms, which are typical in automotive systems [32], and assigned rate-monotonic priorities and implicit deadlines. Next, for a target utilization U^{mean}, we used the *Dirichlet-Rescale* algorithm [24] to generate n random mean utilization values $u_1, u_2, \ldots, u_n$ that sum to U^{mean}. From each mean utilization u_i, we inferred the corresponding mean execution-time bound $\widehat{e}_i = T_i \cdot u_i$.

In line with the CTA evaluation [46], the upper bound $\widehat{s}_i$ on the standard deviation was selected uniformly at random from the interval $[0.01 \cdot \widehat{e}_i, r^{\max} \cdot \widehat{e}_i]$, where $r^{\max}$ is the configurable maximum ratio of the standard deviation and the mean.

For CAA, we similarly varied the maximum covariance $\widehat{v}_{i,k}$ between each pair of tasks. This value was selected uniformly at random from the interval $[0, \text{c}^{\max} \cdot \sqrt{\widehat{s}_i^2 \cdot \widehat{s}_k^2}]$, where $\text{c}^{\max}$ is the configurable covariance coefficient. This range spans from independent execution (if $\widehat{v}_{i,k} = 0$) up to covariance equal to the theoretical upper bound used by CTA (if $\widehat{v}_{i,k} = \sqrt{\widehat{s}_i^2 \cdot \widehat{s}_k^2}$).

The configuration serving as a reference point for all subsequent experiments consisted of a task-set size $n = 25$ with $U^{\text{mean}} = 0.35$ (approximately half of Liu & Layland's utilization bound [38] for $n = 25$). In addition, we set $r^{\max} = 0.2$ (which is in the range used in [46]) and $\text{c}^{\max} = 0.2$ (*i.e.*, up to a moderate amount of positive correlation).

Results. Fig. 2 consists of 4×2 plots arranged in four rows **1**–**4** and two columns **A**–**B**. Each row reports the results for one of the four varying parameters: n, U^{mean}, $r^{\max}$, and $\text{c}^{\max}$. Plots in column **A** report the average DFP values estimated by both CTA and CAA as a function of the varying parameter, while column **B** shows the same underlying data sets as scatter plots. In particular, column **B** reports the CAA (x-axis) and CTA (y-axis) results as one point per task set, directly showing the magnitudes of the computed DFP values. A color scale in the far-right legend shows the magnitude of the varied parameter.

Influence of the task-set size n. In the first experiment, we varied n from 5 to 50 in increments of 5. Fig. $2_{\textbf{1,A}}$ shows CAA to realize an improvement of roughly one order of magnitude over the CTA baseline. The gap becomes slightly more pronounced as n increases. This trend is explained by Bienaymé's identity (Fact 2): as CTA is more pessimistic ($\widehat{s}_i$) than CAA ($\widehat{v}_{i,k}$) on a per-task basis, CTA accumulates pessimism with each additional higher-priority job. Fig. $2_{\textbf{1,B}}$ confirms the structural advantage of CAA, as it dominates CTA not just on average, but for every tested workload.

Influence of the total mean utilization. In the second experiment, we varied the mean utilization U^{mean} from 0.1 to 1.0 in steps of 0.1, as shown in Fig. $2_{\textbf{2,A}}$. Again, CAA yields a clear improvement over CTA, with a decreasing improvement at higher utilizations. This is particularly apparent in Fig. $2_{\textbf{2,B}}$, which reveals a fairly consistent offset for lower-utilization task sets (which naturally also have lower DFPs overall). At higher utilizations, as DFP approaches the trivial bound of 1, the difference diminishes. However, workloads with such extremely high DFPs are likely of little interest in practice.

Influence of standard deviation. Next, we varied the maximum ratio between standard deviation and mean, $r^{\max}$, from 0.01 to 0.2 in steps of 0.01. Fig. $2_{\textbf{3,A}}$ and Fig. $2_{\textbf{3,B}}$ show CAA to consistently offer a roughly one-magnitude improvement over CTA throughout the evaluated parameter range. At the same time, DFPs noticeably increase as the standard deviation rises, which is expected since workloads with more volatile execution-time distributions are less predictable than those with lower dispersion in execution times.

Influence of covariance. Finally, we varied $\text{c}^{\max}$ from 0.1 (at most weak correlation) to 1 (extreme correlation as upper-bounded by CTA is possible), thus affecting the range from which the randomly generated correlation $\widehat{v}_{i,k}$ parameters are drawn. As anticipated, the plot in Fig. $2_{\textbf{4,A}}$ illustrates that CAA achieves the greatest improvements if actual correlation is much lower than CTA's pessimistic upper bound. At the same time, Fig. $2_{\textbf{4,A}}$ clearly demonstrates a gradually diminishing advantage as covariance levels approach very high levels. The shading in Fig. $2_{\textbf{4,B}}$ confirms this observation, again showing that CAA is never worse than CTA for any tested workload.

In summary, across all experiments, CAA consistently provides significantly more accurate DFP estimates than CTA, often achieving an order-of-magnitude improvement. Next, we report on a case study in which we explored whether this advantage persists when task-model parameters are inferred statistically, rather than derived from idealized (ground-truth) parameters, which are obviously unknown in practice.

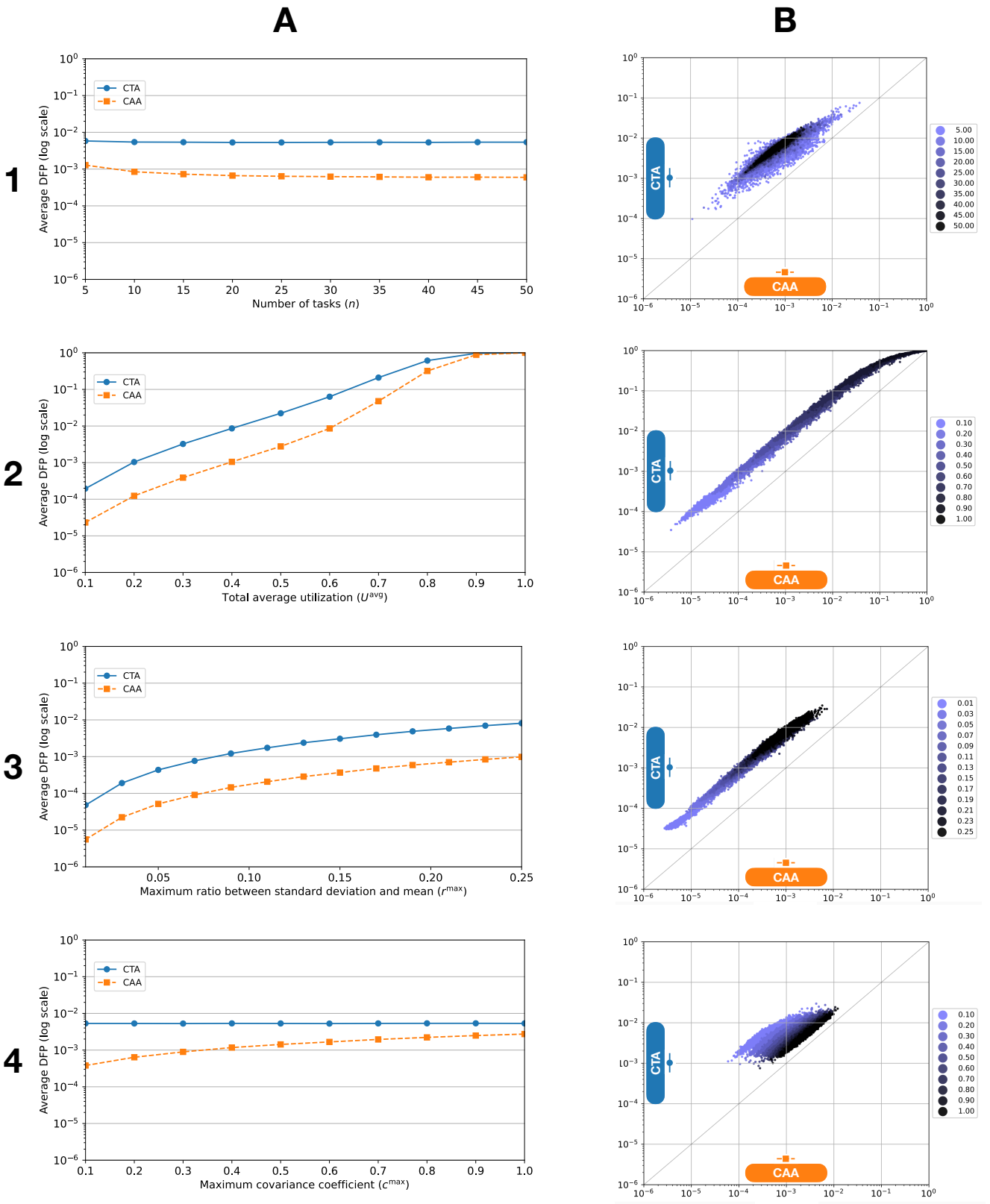

Fig. 2: Column **A** presents average DFP estimates by CTA and CAA. Column **B** shows the same data as scatter plots with a color gradient indicating the varying parameter. The following parameters were varied: n (row **1**), U^{mean}(row **2**), r^{max}(row **3**), and c^{max} (row **4**). The proposed CAA consistently outperforms the baseline CTA across all tested parameter ranges.

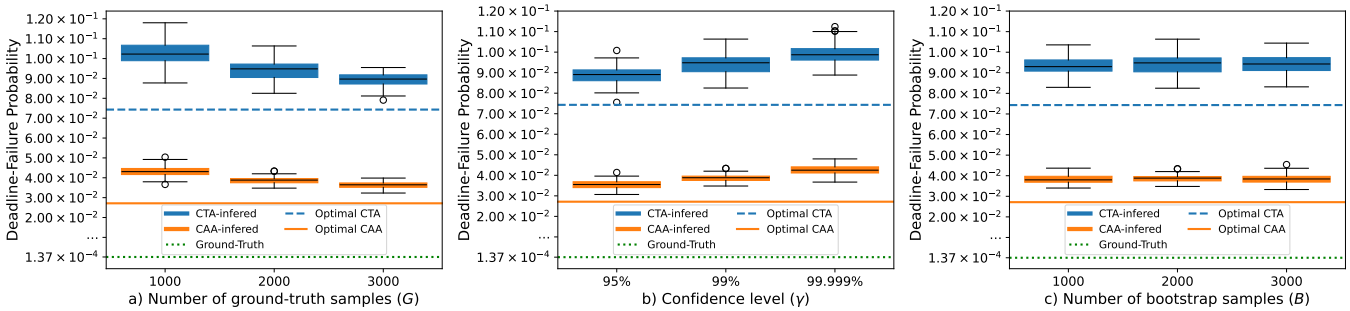

Fig. 3: Box plots of DFP estimates for τ_5 in Table II for varying statistical interference parameters: **(a)** varying the number of ground-truth samples G; **(b)** varying the confidence level γ; and **(c)** varying the number of bootstrap samples B.

IX. Case Study

We conducted a case study applying the distribution-agnostic statistical inference method proposed in Sec. VII to a workload extracted from the WATERS'17 industrial challenge [26]. The objectives were to **(i)** serve as a proof of concept and **(ii)** explore the impact of statistical inference on DFP estimation accuracy.

The WATERS'17 industrial challenge [26] features an automotive workload deployed on a partitioned multicore processor under partitioned fixed-priority scheduling. We focused on the five highest-priority periodic tasks on the challenge system's core 2, as summarized in Table II. The challenge description reports for each task both a maximum and a mean execution time. Notably, the workload is *classically infeasible*, as the reported WCET parameters result in a total utilization greater than one. However, it is *stochastically feasible* since the total mean utilization does not overload the system.

Task	T_i	D_i	WCET	$\mathbb{E}[\mathcal{C}_i]$	$\sigma[\mathcal{C}_i]$	$\mathrm{Cov}[\mathcal{C}_{i,j}, \mathcal{C}_{i,j'}]$
τ_1	2000 μs	2000 μs	404 μs	294 μs	25	639.16
τ_2	5000 μs	5000 μs	931 μs	635 μs	68	4623.84
τ_3	20000 μs	20000 μs	10468 μs	6686 μs	868	753175.39
τ_4	50000 μs	50000 μs	3084 μs	2019 μs	244	59796.99
τ_5	100000 μs	100000 μs	9418 μs	6465 μs	678	459129.35

TABLE II: Task set extracted from the WATERS'17 challenge, with the assumed ground-truth values (last two columns).

Ground-truth DFP. As the challenge description omits detailed statistical properties of the workload, we made simplifying assumptions to allow the ground-truth DFP of τ_5 (DFP_5) to be computed exactly. Specifically, we assumed that each task executes for its reported WCET $c_{i,\mathrm{wc}}$ with probability 0.05, and for its minimum execution time $c_{i,\mathrm{min}}$ with probability 0.95. We set $c_{i,\mathrm{min}} \triangleq \frac{\mathbb{E}[C_i] - 0.05 \cdot c_{i,\mathrm{wc}}}{0.95}$, thus ensuring a mean execution time consistent with the reported values [26]. To keep the computation of DFP_5 tractable, we assumed that all tasks are synchronously released and that each task experiences maximum intra-task correlation, but *no* inter-task correlation (*i.e.*, tasks are mutually independent, with inter-task covariances equal to 0). Under these assumptions, $DFP_5 = 1.36875 \times 10^{-4}$.

Inference. As explained in Sec. VII, the proposed inference method uses three parameters: the number of ground-truth samples G, the confidence level γ, and the number of bootstrap samples B. We report the DFP estimates computed by CAA and CTA for τ_5 as $\mathrm{DFP}^{\mathrm{caa}}_{G,B,\gamma}$ and $\mathrm{DFP}^{\mathrm{cta}}_{G,B,\gamma}$, respectively, indicating the specific parameters used.

We varied G and B across $\{1000, 2000, 3000\}$ and γ across $\{95, 99, 99.999\}$, using $G = B = 2000$ and $\gamma = 99$ as the base configuration. For each parameter combination, we ran the statistical inference procedure 100 times and observed the resulting distributions of $\mathrm{DFP}^{\mathrm{caa}}_{G,B,\gamma}$ and $\mathrm{DFP}^{\mathrm{cta}}_{G,B,\gamma}$ estimates, which are visualized in Fig. 3. For comparison, we also report the best attainable DFP for each method, computed with ground-truth task parameters rather than inferred ones.

Influence of the sample size. Fig. 3(a) shows that both methods tend towards their respective optimal DFP estimates as G increases. Importantly, in all trials, the DFP estimates did not underestimate their respective optima nor the ground-truth DFP. In fact, using fewer samples renders the predictions *more* pessimistic, rather than unsound, which is reassuring. Additionally, the symmetric nature of the $\mathrm{DFP}^{\mathrm{caa}}_{G,B,\gamma}$ and $\mathrm{DFP}^{\mathrm{cta}}_{G,B,\gamma}$ box plots suggests that using the median DFP estimate from repeated inference attempts is a robust choice.

Influence of the percentile. As expected, increasing γ leads to more conservative results, as evident in Fig. 3(b) for both CTA and CAA. This effect is more pronounced in CTA, where a larger γ amplifies the pessimism in the analysis.

Influence of the bootstrap count. Finally, Fig. 3(c) shows that the B setting has no significant impact on analysis accuracy (for the range of values considered here), which is expected, as discussed by Tibshirani and Efron [55, p. 52-53].

In summary, all estimated DFP values slightly exceeded their respective optima and, most importantly, the ground-truth DFP. In other words, not a *single* attempt at parameter inference resulted in an unsound DFP prediction. While any measurement-based approach inherently cannot achieve absolute certainty, this proof of concept demonstrates the promising potential of the proposed distribution-agnostic bootstrapping method. These encouraging initial results emphasize the need for further studies on real systems with more complex workloads.

X. Related Work

In 2019, Davis and Cucu-Grosjean provided comprehensive surveys of probabilistic schedulability [16] and timing [15] analysis techniques. As discussed in Sec. I, this paper is a direct response to some of the key challenges and open problems

highlighted by Davis and Cucu-Grosjean. Our proposal, CAA together with distribution-agnostic nonparametric bootstrapping, is the first solution to explicitly account for largely *unrestricted* inter- and intra-task covariance. Nonetheless, prior work has explored various approaches to addressing more restricted forms of execution-time dependence, which we briefly review below.

Hidden Markov models (HMMs) offer a natural approach to handling execution-time dependencies arising from a task's internal state. In particular, Frías et al. [20] and Abeni et al. [1] use HMMs to address execution-time dependence in periodic tasks provisioned in constant-bandwidth servers. The application of HMMs with continuous Gaussian emission distributions has been proposed by Friebe et al. [21–23]. Their approach bounds the deadline-miss probability for a reservation-based system with private reservations for each task. While the accuracy of HMMs heavily depends on the quality of the underlying data, these distributions are likely varying over time, as pointed out by Friebe et al. [22, 23]. Furthermore, HMMs can capture a limited degree of intra-task dependence, although still with independent emission distributions, but cannot account for inter-task dependence. We note that this line of work studies the deadline-miss probability in a long-run interpretation [16], which is a different metric than the DFP considered here.

The problem of unknown dependencies among jobs was examined by Ivers and Ernst [28], who provided a solution for fixed-priority preemptive scheduling, under the assumption that the entire probability distribution for each task is fully known and that each job follows the same distribution. Their method leverages copulas (as first used by Bernat et al. [3] in timing analysis) and Fréchet bounds to derive probabilistic response-time bounds. In contrast, CAA accounts for covariance regardless of whether all jobs follow the same distribution, which is not always the case, and uses only bounds on simple summary statistics that can be inferred via bootstrapping.

Extreme value theory (EVT) has been applied in statistical analyses of both measured execution times [14, 36, 37] and observed response times [41–43] in contexts with dependencies. Notably, EVT imposes nontrivial restrictions when applied to a series of dependent observations [13, Ch. 5]. Furthermore, the distributions must be extremal independent [51] or stationary [34] for dependent tasks. In contrast, our proposal is mathematically simpler and distribution-agnostic.

Other approaches considering restricted notions of dependence include von der Brüggen et al.'s analysis of DFP under EDF scheduling [57], which allows for dependencies among a small number of subsequent jobs. Liu et al. [39] proposed *independence thresholds*, where a per-task threshold splits a job's execution cost into a dependent and an independent part, and provided a related stochastic response-time analysis. Mills and Anderson [49] designed a multiprocessor scheduling policy ensuring bounded average-case tardiness despite arbitrary degrees of dependence, provided that any dependence is limited to time intervals of bounded length. CAA shares none of these restrictions: it imposes no limits on the dependent fraction of a job's total execution time nor on the maximum separation in time or job count over which dependencies manifest.

While considering various restricted forms of dependence, the approaches discussed above (with the exception of [57]) do not consider the DFP metric. The DFP metric is primarily analyzed by applying *independence-assuming methods* to properly padded pWCET distributions [5] under preemptive static-priority scheduling. Notably, in 2022, Chen et al. [12] rectified a mistaken critical-instant assumption common in earlier DFP analyses. The *convolution-based method* by Maxim and Cucu-Grosjean [48] bounds DFP by convolving discrete pWECT distributions of individual jobs. This method was improved by Marković et al. [44] with optimal resampling and more efficient circular convolution. Task-level convolution was proposed by von der Brüggen et al. [56] as an alternative. Furthermore, well-known *analytical bounds* have been used to bound DFP directly: Chen et al. [10, 11] adapt the Chernoff bound while von der Brüggen et al. [56] consider Hoeffding and Bernstein inequalities. The Berry-Esseen theorem was applied by Marković et al. [45] to estimate response-time distributions.

In a different direction also rooted in independence assumptions, Bozhko et al. [4] recently proposed to apply Monte-Carlo sampling to estimate response-time distributions. Many more non-analytical methods exist for the periodic task model [*e.g.*, 7, 18, 19, 40] and interested readers are referred to Davis and Cucu-Grosjean's surveys [15, 16] for further details.

XI. Conclusion

The field of probabilistic schedulability analysis promises to provide a sound and accurate characterization of the temporal behavior of modern real-time embedded systems deployed on complex hardware platforms with uncertain execution behavior. Despite significant progress in recent years, important open problems remain. In particular, Davis and Cucu-Grosjean in their survey of real-time probabilistic analysis [16] highlight the need for *"analyses* [...] *that can address dependencies"* as a key unsolved challenge in need of attention.

In response, we have introduced CAA, the first probabilistic schedulability analysis to explicitly account for task dependencies using bounds on maximum inter- and intra-task covariance. Using this information, CAA safely estimates a task's response-time distribution to upper-bound its DFP, *i.e.*, the risk of missing a deadline. By design, CAA dominates the earlier CTA [46], which, while sound in the presence of task dependencies, does not explicitly analyze them. As demonstrated also empirically, CTA therefore produces significantly more pessimistic bounds than CAA. Furthermore, we have addressed the challenge of obtaining the necessary model parameters through statistical inference and proposed a nonparametric bootstrapping approach, which was demonstrated in a proof-of-concept case study.

Finally, as Davis and Cucu-Grosjean [16] also identified computational efficiency as an unresolved challenge, it is worth noting that CAA offers a closed-form solution that is fast to compute and uses negligible memory.

CAA presents numerous opportunities for future work. In particular, it calls for experimental validation in practical settings with real workloads on complex hardware platforms, as well as extensions to a broader class of system models and schedulers.

ACKNOWLEDGEMENTS

We thank the reviewers for their insightful and constructive feedback, which has helped to significantly improve the paper.

This result is part of a project (PropRT) that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 865170).

Additionally, this project has received funding from the ERC under the European Union's Horizon 2020 research and innovation program (grant agreement No 803111).

REFERENCES

[1] L. Abeni, D. Fontanelli, L. Palopoli, and B. V. Frías, "A Markovian model for the computation time of real-time applications," in *IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2017, pp. 1–6.

[2] G. Bernat, A. Colin, and S. Petters, "pWCET: A tool for probabilistic worst-case execution time analysis of real-time systems," University of York, Department of Computer Science, Technical Report YCS-2003-123, 2003.

[3] G. Bernat, A. Burns, and M. Newby, "Probabilistic timing analysis: An approach using copulas," *Journal of Embedded Computing*, vol. 1, no. 2, pp. 179–194, 2005.

[4] S. Bozhko, G. von der Brüggen, and B. B. Brandenburg, "Monte Carlo response-time analysis," in *42nd IEEE Real-Time Systems Symposium (RTSS)*, 2021, pp. 342–355.

[5] S. Bozhko, F. Marković, G. von der Brüggen, and B. B. Brandenburg, "What Really is pWCET? A Rigorous Axiomatic Proposal," in *44th IEEE Real-Time Systems Symposium (RTSS)*, 2023, pp. 317–330.

[6] F. P. Cantelli, "Sui confini della probabilitá," in *Atti del Congresso Internazionale dei Matematici*, 1928, pp. 47–59.

[7] L. Carnevali, A. Melani, L. Santinelli, and G. Lipari, "Probabilistic deadline miss analysis of real-time systems using regenerative transient analysis," in *22nd International Conference on Real-Time Networks and Systems (RTNS)*, 2014, pp. 299–308.

[8] J. Carpenter, S. Funk, P. Holman, A. Srinivasan, J. Anderson, and S. Baruah, "A categorization of real-time multiprocessor scheduling problems and algorithms," in *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, J. Y. Leung, Ed. Chapman and Hall/CRC, 2004, pp. 30–1–30–19.

[9] G. Casella and R. Berger, *Statistical Inference*. CRC Press, 2024.

[10] K. Chen and J. Chen, "Probabilistic schedulability tests for uniprocessor fixed-priority scheduling under soft errors," in *12th IEEE International Symposium on Industrial Embedded Systems, (SIES)*, 2017, pp. 1–8.

[11] K.-H. Chen, N. Ueter, G. von der Brüggen, and J.-J. Chen, "Efficient computation of deadline-miss probability and potential pitfalls," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2019, pp. 896–901.

[12] K.-H. Chen, M. Günzel, G. von der Brüggen, and J.-J. Chen, "Critical instant for probabilistic timing guarantees: Refuted and revisited," in *43rd IEEE Real-Time Systems Symposium (RTSS)*, 2022, pp. 145–157.

[13] S. Coles, *An Introduction to Statistical Modeling of Extreme Values*. London, UK: Springer-Verlag, 2001, vol. 208.

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

[15] R. I. Davis and L. Cucu-Grosjean, "A survey of probabilistic timing analysis techniques for real-time systems," *Leibniz Transactions on Embedded Systems*, vol. 6, no. 1, pp. 03–1–03:60, 2019.

[16] ——, "A survey of probabilistic schedulability analysis techniques for real-time systems," *Leibniz Transactions on Embedded Systems*, vol. 6, no. 1, pp. 04:1–04:53, 2019.

[17] A. Davison, *Bootstrap Methods and Their Application*. Cambridge University Press, 1997.

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

[19] J. L. Díaz, J. M. López, M. García, A. M. Campos, K. Kim, and L. L. Bello, "Pessimism in the stochastic analysis of real-time systems: Concept and applications," in *25th IEEE Real-Time Systems Symposium (RTSS)*, 2004, pp. 197–207.

[20] B. V. Frías, L. Palopoli, L. Abeni, and D. Fontanelli, "Probabilistic real-time guarantees: There is life beyond the i.i.d. assumption," in *23rd IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2017, pp. 175–186.

[21] A. Friebe, A. V. Papadopoulos, and T. Nolte, "Identification and validation of Markov models with continuous emission distributions for execution times," in *26th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, 2020, pp. 1–10.

[22] A. Friebe, F. Marković, A. V. Papadopoulos, and T. Nolte, "Adaptive runtime estimate of task execution times using Bayesian modeling," in *27th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, (RTCSA)*, 2021, pp. 1–10.

[23] ——, "Continuous-emission Markov models for real-time applications: Bounding deadline miss probabilities," in *29th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2023, pp. 14–26.

[24] D. Griffin, I. Bate, and R. I. Davis, "Generating utilization vectors for the systematic evaluation of schedulability tests," in *41st IEEE Real-Time Systems Symposium (RTSS)*, 2020, pp. 76–88.

[25] G. Grimmett and D. Welsh, *Probability: An Introduction*. Oxford University Press, 2014.

[26] A. Hamann, D. Dasar, S. Kramer, M. Pressler, F. Wurst, and D. Ziegenbein, "WATERS industrial challenge 2017," in *8th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*, 2017.

[27] O. Hössjer and A. Sjölander, "Sharp lower and upper bounds for the covariance of bounded random variables," *Statistics & Probability Letters*, vol. 182, p. 109323, 2022.

[28] M. Ivers and R. Ernst, "Probabilistic network loads with dependencies and the effect on queue sojourn times," in *6th International ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine)*, 2009, pp. 280–296.

[29] N. Jayakrishnan, A. Wierman, and B. Zwart, *The Fundamentals of Heavy Tails: Properties, Emergence, and Estimation*. Cambridge University Press, 2022, vol. 53.

[30] Y. Jiang and Y. Liu, *Stochastic Network Calculus*. Springer, 2008, vol. 1.

[31] A. Klenke, *Wahrscheinlichkeitstheorie*. Springer, 2006, vol. 1.

[32] S. Kramer, D. Ziegenbein, and A. Hamann, "Real world automotive benchmarks for free," in *6th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*, vol. 130, 2015.

[33] W. Kruskal and F. Mosteller, "Representative sampling, iv: The history of the concept in statistics, 1895-1939," *International Statistical Review*, pp. 169–195, 1980.

[34] M. R. Leadbetter, G. Lindgren, and H. Rootzén, "Conditions for the convergence in distribution of maxima of stationary normal processes," *Stochastic Processes and their Applications*, vol. 8,

no. 2, pp. 131–139, 1978.

[35] E. A. Lee, "The past, present and future of cyber-physical systems: A focus on models," *Sensors*, vol. 15, no. 3, pp. 4837–4869, 2015.

[36] G. Lima and I. Bate, "Valid application of EVT in timing analysis by randomising execution time measurements," in *23rd IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2017, pp. 187–198.

[37] G. Lima, D. Dias, and E. Barros, "Extreme value theory for estimating task execution time bounds: A careful look," in *28th Euromicro Conference on Real-Time Systems (ECRTS)*, 2016, pp. 200–211.

[38] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *Journal of the ACM*, vol. 20, no. 1, pp. 46–61, 1973.

[39] R. Liu, A. F. Mills, and J. H. Anderson, "Independence thresholds: Balancing tractability and practicality in soft real-time stochastic analysis," in *35th IEEE Real-Time Systems Symposium (RTSS)*, 2014, pp. 314–323.

[40] J. M. López, J. L. Díaz, J. Entrialgo, and D. García, "Stochastic analysis of real-time systems under preemptive priority-driven scheduling," *Real-Time Systems*, vol. 40, no. 2, pp. 180–207, 2008.

[41] Y. Lu, T. Nolte, J. Kraft, and C. Norström, "Statistical-based response-time analysis of systems with execution dependencies between tasks," in *15th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS)*, 2010, pp. 169–179.

[42] ——, "A statistical approach to response-time analysis of complex embedded real-time systems," in *16th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, 2010, pp. 153–160.

[43] Y. Lu, T. Nolte, I. Bate, and L. Cucu-Grosjean, "A statistical response-time analysis of real-time embedded systems," in *33rd IEEE Real-Time Systems Symposium, (RTSS)*, 2012, pp. 351–362.

[44] F. Marković, A. V. Papadopoulos, and T. Nolte, "On the convolution efficiency for probabilistic analysis of real-time systems," in *33rd Euromicro Conference on Real-Time Systems (ECRTS)*, 2021, pp. 16:1–16:22.

[45] F. Marković, T. Nolte, and A. V. Papadopoulos, "Analytical approximations in probabilistic analysis of real-time systems," in *43rd IEEE Real-Time Systems Symposium (RTSS)*, 2022, pp. 158–171.

[46] F. Marković, P. Roux, S. Bozhko, A. V. Papadopoulos, and B. B. Brandenburg, "CTA: A correlation-tolerant analysis of the deadline-failure probability of dependent tasks," in *44th IEEE Real-Time Systems Symposium (RTSS)*, 2023, pp. 317–330.

[47] ——, "CTA: A correlation-tolerant analysis of the deadline-failure probability of dependent tasks (Python Jupyter notebook)," 2023. [Online]. Available: https://doi.org/10.5281/zenodo.8416355

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

[49] A. F. Mills and J. H. Anderson, "A multiprocessor server-based scheduler for soft real-time tasks with stochastic execution demand," in *17th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, 2011, pp. 207–217.

[50] M. Pagani, E. Rossi, A. Biondi, M. Marinoni, G. Lipari, and G. Buttazzo, "A bandwidth reservation mechanism for AXI-based hardware accelerators on FPGAs," in *31st Euromicro Conference on Real-Time Systems (ECRTS)*, 2019, pp. 24:1–24:24.

[51] L. Santinelli, J. Morio, G. Dufour, and D. Jacquemart, "On the sustainability of the extreme value theory for WCET estimation," in *14th International Workshop on Worst-Case Execution Time Analysis (WCET)*, 2014, pp. 21–30.

[52] T. Stark, "Real-time execution management in the ROS 2 framework," Ph.D. dissertation, Universität des Saarlandes, Fakultät für Mathematik und Informatik, Saarbrücken, 2022.

[53] S. H. Tan and S. B. Tan, "The correct interpretation of confidence intervals," *Proceedings of Singapore Healthcare*, vol. 19, no. 3, pp. 276–278, 2010.

[54] T.-S. Tia, Z. Deng, M. Shankar, M. Storch, J. Sun, L.-C. Wu, and J.-S. Liu, "Probabilistic performance guarantee for real-time tasks with varying computation times," in *16th Real-Time Technology and Applications Symposium (RTAS)*, 1995, pp. 164–173.

[55] R. J. Tibshirani and B. Efron, "An introduction to the bootstrap," *Monographs on Statistics and Applied Probability*, vol. 57, no. 1, pp. 1–436, 1993.

[56] G. von der Brüggen, N. Piatkowski, K.-H. Chen, J.-J. Chen, and K. Morik, "Efficiently approximating the probability of deadline misses in real-time systems," in *30th Euromicro Conference on Real-Time Systems (ECRTS)*, 2018, pp. 6:1–6:22.

[57] G. von der Brüggen, N. Piatkowski, K.-H. Chen, J.-J. Chen, K. Morik, and B. B. Brandenburg, "Efficiently approximating the worst-case deadline failure probability under EDF," in *42nd IEEE Real-Time Systems Symposium (RTSS)*, 2021, pp. 214–226.