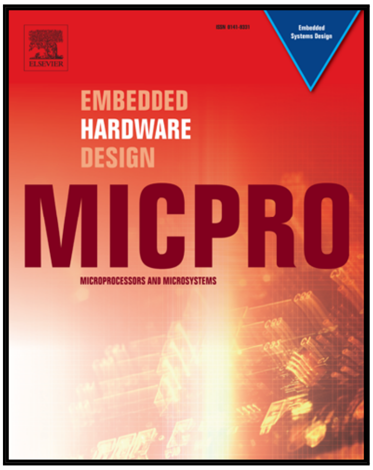
Journal Pre-proof



Probabilistic-WCET Reliability: Statistical Testing of EVT hypotheses

Federico Reghenzani, Giuseppe Massari, William Fornaciari

PII: S0141-9331(20)30302-1
DOI: https://doi.org/10.1016/j.micpro.2020.103135
Reference: MICPRO 103135

To appear in: *Microprocessors and Microsystems*

Received date: 10 August 2019
Revised date: 21 March 2020
Accepted date: 14 May 2020

Please cite this article as: Federico Reghenzani, Giuseppe Massari, William Fornaciari, Probabilistic-WCET Reliability: Statistical Testing of EVT hypotheses, *Microprocessors and Microsystems* (2020), doi: https://doi.org/10.1016/j.micpro.2020.103135

Probabilistic-WCET Reliability: Statistical Testing of EVT hypotheses

Federico Reghenzani[a], Giuseppe Massari[a], William Fornaciari[a]

[a]DEIB, Politecnico di Milano, via Ponzio 34/5, Milano, IT

Abstract

In recent years, the interest in probabilistic real-time has grown, as a response to the limitations of traditional static Worst-Case Execution Time (WCET) methods, in performing timing analysis of applications running on complex systems, like multi/many-cores and COTS platforms. The probabilistic theory can partially solve this problem, but it requires strong guarantees on the execution time traces, in order to provide safe probabilistic-WCET estimations. These requirements can be verified through suitable statistical tests, as described in this paper. In this work, we identify also challenges and problems of using statistical testing procedures in probabilistic real-time computing, proposing a unified test procedure based on a single index called Probabilistic Predictability Index (PPI). An experimental campaign has been carried out, considering both synthetic and realistic datasets, and the analysis of the impact of the Linux PREEMPT_RT patch on a modern complex platform as a use-case of the proposed index.

1. Introduction

The *Worst-Case Execution Time (WCET)* analysis is an essential part of hard real-time systems design, in order to properly validate tasks schedule, and thus guarantee the timing constraints are satisfied at run-time. Failing to meet these constraints leads the tasks to possibly misbehave, often with unacceptable consequences for hard real-time systems, especially in the case of safety-critical applications. Consequently, the timing analysis of critical tasks requires the WCET estimation to be safe, i.e, greater or equal to the actual WCET experienced at run-time. On the other hand, this estimation must be as tight as possible to the real WCET, to minimize over-provisioning in the resource assignment. Recently, getting a safe but tight WCET has become a challenging problem. The growing capabilities of embedded systems, in addition, but opposed to, the reaching of technology limits is increasing the hardware complexity of processors – such as the introduction of many-cores, multi-level caches, complex pipelines, etc. This hinders the use of traditional WCET estimation techniques [9] [28] [31], that either require an unfeasible amount of computational effort or produce an extremely pessimistic over-estimation. This over-estimation can even lead to the inability of computinga feasaible

Email addresses: `federico.reghenzani@polimi.it` (Federico Reghenzani), `giuseppe.massari@polimi.it` (Giuseppe Massari), `william.fornaciari@polimi.it` (William Fornaciari) tasks schedule. The problem is even magnified when dealing with Commercial-Off-The-Shelf (COTS) components and general-purpose operating systems [42], that have not been designed to be timing-predictable and they consequently lack a proper worst-case analysis.

1.1. Probabilistic Real-Time Computing

Given the aforementioned scenario, *probabilistic (hard) real-time* has been proposed as a possible solution to the WCET estimation problem. This approach is founded on the well-known *Extreme Value Theory (EVT)*, which is widely applied to the prediction of natural disasters. The theory is briefly introduced in Section 2. The use of EVT in real-time systems has been proposed since the beginning of the 2000s by Burns et al. [11] and Bernat et al. [7]. The first paper presented EVT and the possibility of using this theory for probabilistic real-time analysis. The latter instead, focused on the algebraic properties needed to combine several probabilistic-WCET estimations. Generally, probabilistic real-time based approaches can be divided into two main classes [1]: *Static Probabilistic Time Analyses (SPTA)* and *Measurement-Based Probabilistic Time Analyses (MBPTA)*. MBPTA, which we focus on, has been proposed to estimate the so-called probabilistic-WCET (pWCET) by Edgar and Burns [17] by directly sampling the execution times of the tasks. Opposite to the classical WCET estimations, the pWCET is not a single value, rather it is a statistical distribution, usually expressed by the complement of its *cumulative distribution function (cdf)*:

$$p = P(X > \overline{WCET}) \tag{1}$$

where X is the random variable representing the task execution time. By using this distribution, it is possible to compute the probability of violation (p) of a given $\overline{WCET}$ or, vice versa, the $\overline{WCET}$ given the probability of violation (p). The probability of violation represents how likely is the event of observing an execution time larger than $\overline{WCET}$. The selection of a value for the probability p is a trade-off between WCET reliability and tightness. The pWCET is considered *safe* if the estimated distribution "upper-bounds" the worst-case execution time with a probability value equal or higher than the real one[1].

1.2. Related works

The literature on probabilistic real-time computing was summarized in 2017 by Santinelli et al. [50] and more recently, in 2019, by two comprehensive surveys by Cazorla et al. [13] and by Davis and Cucu-Grosjean [16]. The research efforts focused on both the methodology of the application of EVT to probabilistic real-time, and on the design of computing architectures capable of generating execution time traces that fulfills the EVT requirements (see Section 2.1). Regarding the second point, several works have been published [14] [29]. Other works focused more on the theoretical aspects of MBPTA [2] [36], which still present several challenges to address [21] [44]. This work is neither based on a particular architecture nor wants to propose a new EVT-compliant one. Rather, we focus on the analysis of the tasks execution time traces, independently from the nature of the system generating them. The goal is to provide a test suite and a unified index, to verify the EVT hypotheses under the pWCET reliability requirement.

[1] Formal definitions for pWCET comparison are available in [50].

1.3. Paper contributions and structure

Most of the articles in the literature assume the EVT hypotheses verified or do not follow a systematic approach to assess them. Some works applied improper hypothesis tests, erroneously run multiple tests on the same data, or reached conclusions without a proper evaluation of the statistical effects. Strategies based on expert knowledge instead, like graphical plot analysis, do not offer a systematic approach and thus quantitative information on the pWCET reliability. For these reasons, in this work we aim at:

1. clearly stating the problems and the statistical aspects affecting the pWCET reliability;
2. analyzing and making a selection of the statistical tests fitting the probabilistic real-time computing case for the independent and identically distributed hypothesis;
3. proposing a single metric, called PPI, to use both as a decision rule and for the analysis of the execution time series, as a comparison metric, to check how different systems and task configurations may affect the validity of the probabilistic real-time results;
4. highlighting some common errors recurring in previous works when statistical tests are applied to MBPTA.

Finally, as use-case, we show how the theoretical results can be exploited to verify whether the real-time patch (PREEMPT_RT) of the Linux kernel is able to fulfill the EVT hypotheses or not.

To the best of our knowledge, this is the first attempt to systematically analyze these reliability theoretical problems on the application of EVT to probabilistic real-time computing. In fact, the conclusions drawn from the EVT results cannot exclude the context on which the statistical theory is used. Real-time computing is for sure an unique and particular scenario, compared to the usual applications of this theory. As subsequently explained, the reliability of the EVT results depends on three factors. The most interesting and studied from a real-time standpoint is the independent and identically distributed hypothesis: this work focuses, consequently, on this hypothesis, aiming at improving the reliability of EVT results, essential steps towards a possible future certification process of probabilistic real-time.

The article is structured as follows: in Section 2 the description of the EVT theory and its applicability to the probabilistic real-time problem is reviewed. Section 3 describes the statistical testing procedures, with a special focus on reliability aspects and common errors. The mathematical foundations of the proposed index PPI are described in Section 4. Finally, an experimental evaluation has been performed and presented in Section 5, followed by conclusions and future works considerations in Section 6.

2. Extreme Value Theory in Real-time Computing

The statistical theory of extremes has been developed to study the “tails” of a distribution, i.e. the events for which we have maximum (or minimum) probability values. In this regard, the aforementioned *Extreme Value Theory (EVT)* is in the opposite direction with respect to the well-known Central Limit Theorem (CLT), which is instead focused on the behavior of the distribution around the mean value.

Given a sequence of independent and identically distributed (i.i.d.) random variables $X_1, X_2, ..., X_n$, the EVT deals with the limit distribution at the extremes, i.e. the $\max(X_1, X_2, ..., X_n)$ or $\min(X_1, X_2, ..., X_n)$. In the real-time computing scenario, the sequence of random variables $X_1, X_2, ..., X_n$ is the execution times of a given task, that is the direct time measurements in case of MBPTA. In the WCET estimation case, we are interested in the maximum value, therefore it is possible to formalize the probability of not incurring in a execution time longer than a certain threshold x as follows:

$$\begin{aligned} &P(\max(X_1, X_2, ..., X_n) \leq x) = P(X_1 \leq x, X_2 \leq x, ..., X_n \leq x) \\ &\overset{\text{iid}}{=} P(X_1 \leq x)P(X_2 \leq x) \cdots P(X_n \leq x) = F^n(x) \end{aligned} \tag{2}$$

$F(x)$ is the cumulative distribution function (cdf) and its complement is the pWCET formula of Equation 1. Without entering in statistical details, it is possible to demonstrate that [12]:

$$\exists a_n, b_n \text{ s.t. } \lim_{n \to \infty} F^n(a_n x + b_n) = G(x) \tag{3}$$

where $G(x)$ is the cdf of the so-called *Extreme Value Distribution*, for some a_n and b_n. The form of this distribution can be generalized, as subsequently described, and its parameters can be estimated from data. The traditional methods to estimate the parameters of $G(x)$ are the *Block-Maxima (BM)* or the *Peak-over-Threshold (PoT)* approaches. In the first case, the time values are grouped inside blocks of constant size B, to then compute the maximum value for each block. Formally:

$$\begin{aligned} X^{BM} &= \{X_1^{BM}, X_2^{BM}, ..., X_{n/B}^{BM}\} \\ X_i^{BM} &= \max(X_{B\cdot(i-1)+1}, X_{B\cdot(i-1)+2}, ..., X_{B\cdot i}) \end{aligned} \tag{4}$$

for $i = 1, ..., \frac{n}{B}$. The PoT case, instead, discards values by removing any sample featuring a value lower than a predefined threshold P:

$$X^{PoT} = \{X_i \text{ s.t. } X_i > P\} \tag{5}$$

According to the Fisher-Tippett-Gnedenko theorem [19] [22], regardless of the original distribution, X^{BM} and X^{PoT} converge respectively to the *Generalized Extreme Value Distribution (GEVD)* and to the *Generalized Pareto Distribution (GPD).* These distributions can be then exploited to compute the pWCET, as shown in Figure 1 and described in [43]. The so obtained pWCET is representative of the real distribution of the extremes, and consequently safe for real-time computing, if and only if the following EVT hypotheses hold:

1. the execution time samples must be *identically and independently distributed (i.i.d.)*;
2. the original distribution must be in the *domain of attraction* of an extreme distribution;
3. the inputs provided to the task are representative of the real worst-case behaviour.

As explained in the next sections, all the hypotheses are necessary to obtain a reliable pWCET. The hypothesis on representativity (3) is related to the task itself and to the procedure that selects the input samples to generate the execution time trace. We do not explore this hypothesis in this work, since it does not depend on how the pWCET estimation process is performed.

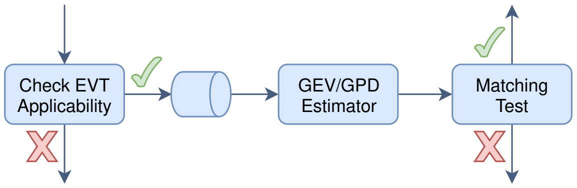


Figure 1: pWCET estimation flow based on the EVT.

2.1. The i.i.d. hypothesis

In common with many other statistical theories, the classical formulation of EVT requires the random samples to be identically and independently distributed (i.i.d.). In real-time computing, this hypothesis is mainly dependent on the processor and the system architecture. For example, a multi-core processor including a cache memory would not probably be able to fulfill the independence requirement, due to the time locality principle. Time samples from consecutive executions of the same task, in fact, are affected by the data locality given by the cache, making the execution times not independent. In practice, the i.i.d. requirement can be relaxed in favor of the stationary property and weaker independence properties [33] [51]. Such hypotheses must hold [50] and can be formalized as follows:

Stationarity. Given a random sequence $X_1, X_2, ..., X_n$ of size n, the process is said to be *strict stationary* iff for any choice of k, l, m with $0 < k + l + m < n$ the following condition is true: $F(X_k, X_{k+1}, ..., X_{k+l}) = F(X_{k+m}, X_{k+m+1}, ..., X_{k+m+l})$, where F is the cdf of the joint distribution. This condition implies identical distribution of the random variables. In real-time computing, the stationary hypothesis indicates a flat distribution of execution times, with constant variance. For instance, a task that drastically changes the job execution time after some runs violates this property.

Short-range independence. Given a sequence of random variables $X_1, X_2, ..., X_n$ of size n, the sequence is said to be short-range independent if for any $i_1 < i_2 < \cdots < i_p < j_1 < \cdots < j_p \leq n$ s.t. $j_1 - i_p \geq s > 1$, defining F_{IJ} the cdf of $X_{i_1,...,i_p,j_1,...,j_n}$, F_I the cdf of $X_{i_1,...,i_p}$, F_J the cdf of $X_{j_1,...,j_p}$ we have $|F_{IJ} - F_I F_J| \leq \alpha_{n,s}$ where $\alpha_{n,s}$ is a sequence with non-decreasing values with respect to s and $\alpha_{n,s} \to 0$ for $n \to \infty$. The intuition behind this property can be noticed looking at the content of the absolute value operator, that is zero if F_I and F_J are perfectly independent, otherwise the dependency has to be upper-bounded by (a function of) the distance among the random sequences. In real-time computing, an example of a cause of short-independence property violation is the presence of processor cache effects between two job instances.

Long-range independence. According to this property, the time series does not show a significant correlation across large time-spans. We define this property by defining its opposite. A long-range dependent sequence can be defined as: a random sequence $X_1, X_2, ..., X_n$ of size n is said to have long-range dependence if its auto-correlation function $\rho(\tau)$ decays exponentially: $\rho(\tau) \sim \frac{L(\tau)}{\tau^{1-2d}}$ with $0 < d < \frac{1}{2}$ where $L(\tau)$ verifies $\lim_{t\to\infty} \frac{L(at)}{L(t)} = 1$ for some $a > 0$.

It is worth noting that the short-range hypothesis is a sufficient but not necessary condition for EVT applicability [34]. If the dataset presents a short-range dependence, it can still be considered valid if other conditions hold. The statistical paper [34] proposes some diagnostic methods to check these properties, however, they require a non-trivial in-depth analysis of the dataset and the used estimation method. For this reason, we suggest using such techniques only when the short-range independence hypothesis is false and it is not possible to improve the system to be adherent to this hypothesis.

2.2. The domain-of-attraction hypothesis

The last property to be satisfied, introduced in [50], is called *matching* and it is related to the *domain of attraction* hypothesis. This hypothesis requires that the original distribution of extremes actually converges to one of the EVT distribution classes. When the timing samples are represented with random variables having continuous distribution functions, the domain of attraction hypothesis is true in the overwhelming majority of the times [49]. This is, instead, not necessarily true when discrete distributions are considered. The matching property is usually checked with *a posteriori* statistical tests, that verify whether the resulting distribution actually matches the input data. Typical tests are the Kolmogorov-Smirnov [27] and Anderson-Darling [55]. Moreover, checking the matching hypothesis with these tests has another advantage, i.e. verifying if the BM (or PoT) procedure and, in general, the whole EVT process correctly estimates the distribution.

It is hard to provide a generalization of the domain-of-attraction hypothesis, being a statistical detail that cannot be easily linked to specific hardware or software characteristics. A recent paper carried out a preliminary study on how to deal with this hypothesis in the context of probabilistic real-time [48]. In real-time computing, the execution time is usually expressed as clock cycles, which is a discrete measure. However, the cardinality of clock cycles is usually so large that we can approximately consider it in a continuous domain. An alternative is to measure the execution time in seconds, to avoid this issue and making this assumption almost certainly true.

3. Statistical Testing in MBPTA

All the hypotheses previously described can be verified through suitable statistical tests. The results are reject/not-reject responses that correspond to the adherence or not to the EVT hypothesis. This, in turn, can represent a true/false boolean response to the problem of verifying the pWCET reliability. Therefore, performing proper statistical tests in the correct way is fundamental, other than being a necessary step towards the certifiability of the probabilistic approaches.

3.1. Assessing the EVT hypotheses via hypothesis testing

A statistical test is typically described by its hypothesis scheme. Usually, the symbol H_0 represents the null hypothesis, while the symbol H_1 or H_a the alternative hypothesis. The result of a test can be “reject the null hypothesis” or “unable to reject the null hypothesis”. In the first case, the test detects strong evidence that the null hypothesis is probably false, while the alternative hypothesis is probably true. The outcome of a statistical test (reject/not-reject) comes from the evaluation of the *p-value* or the *critical value*. As the two approaches are exactly equivalent, we have decided to consider only the second one. The critical value is a constant value, i.e. not dependent on the input, but derived from the significance level α. It is compared against the *statistic* computed over the input data to take the reject/not-reject decision. How the critical value and the statistic are computed depends on the specific test.

3.2. Selection of the statistical tests

Stationarity. In literature, several studies on stationary processes are available, along with related statistical test procedures. In particular, there is a large availability of unit-root tests – a particular case of non-stationarity – but a lower number of general stationary tests. Given a time series $X = \{X_1, X_2, ..., X_n\}$, we are looking for a test with the following hypothesis scheme:

H_0 : the time series X is stationary
H_1 : the time series X is not stationary

In this regard, the most used one is the Kwiatkowsky, Phillips, Schmidt and Shin (KPSS) test [32]. A variant of KPSS considering the relaxed null hypothesis "the time series is stationary or trend stationary" exists. For the EVT hypothesis of stationarity, we are interested in the tightest one, thus we do not consider this variant. The formula for KPSS statistic is available in Appendix A.1. The critical values can be computed through the interpolation of the tabular data proposed in [32] or by using Monte Carlo approaches.

Short-Range dependence. To test the short-range dependence of data, we selected the Brock, Dechert, Scheinkman and LeBaron (BDS) test [10]. For probabilistic real-time, we decided to select this test because it is a *portmanteau test*, i.e. the null hypothesis is well specified, but the alternative hypothesis is not. Given a time series $\{X_i\} = X_1, X_2, ..., X_n$:

H_0 : the time series $\{X_i\}$ is independent
H_1 : the time series $\{X_i\}$ has some sort of dependency

Most of the other available tests detect specific sort of dependency (e.g. serial correlation or deterministic chaos). Therefore, we decided to choose the test with the most general detection capability. The formula for BDS statistic is available in Appendix A.2. The critical values can be computed via numerical methods.

Long-Range dependence. The *Hurst Exponent* (H) is the traditional index used to measure the long-term memory of a time series in financial applications [41]. H is a number in the range $[0; 1]$ indicating the degree of long-term dependency: $H = 0.5$ means a perfectly random and uncorrelated time series, while $H < 0.5$ or $H > 0.5$ indicates a negative or positive correlated time series, respectively. However, performing a statistical test on H is nontrivial [15] and, to the best of our knowledge, it does not exist a well-assessed test. The Hurst index is computed from the R/S statistic equation [25] instead, that can be directly used as a test:

H_0 : the time series has no long-range dependency
H_1 : the time series has long-range dependency

This test is sensitive to long-range dependency but also to short-range dependency. An alternative to this formulation is the Lo's modified version [38] that has been developed to limit the influence of short-range dependency in the R/S equation and it is commonly used. However, this version reduces the statistical power of the test [56], which is not desirable in this context (see Section 3.4). Therefore, by using the unmodified R/S statistic, this test may detect a short-range dependency partially overlapping the BDS test, thus providing pessimistic but safe results. The formula for R/S statistic is available in Appendix A.3. The critical values can be computed via numerical methods.

3.3. The significance level - False positive

Once the statistical test procedure is defined, the next critical step is to set the sample size and the significance level α. The sample size, i.e. the number of time measurements, composing a time trace used in the estimation process, is a parameter affecting both pWCET accuracy and safety, as discussed in the next Section 3.4. The significance level α is a parameter chosen by the designer of the statistical test procedure and it corresponds to the false-positive ratio of the test, i.e. the probability that a test rejects the null hypothesis even if it is actually true. This is often called *Type I error* and in pWCET terms it corresponds to discard the sample when it is instead compliant with EVT hypotheses. To check the sub-hypotheses presented in the previous section, the experimenter usually performs a sequence of three statistical tests. In general, executing multiple hypothesis tests on the same data increases the false-positive rate on the null hypothesis rejection of the overall test [5]:

$$\alpha_{\text{global}} = 1 - (1 - \alpha)^n \tag{6}$$

where n is the number of tests (in our case $n = 3$).

For common values $\alpha = 0.05$ and $\alpha = 0.01$, the resulting global significance levels are respectively $\alpha_{\text{global}} \approx 0.14$ and $\alpha_{\text{global}} \approx 0.03$. The real significance level is thus higher than the single test levels, entailing a higher false-positive rate in rejection. Rejecting a sample implies that the pWCET estimation process stops, because it detects that not all the hypotheses are satisfied, preventing the estimation of an unsafe pWCET. The false-positive rate makes difficult to characterize the capability of a hardware-software architecture to fulfill the EVT hypotheses: obtaining a rejection result, by running one single time a statistical test, does not necessarily mean that the architecture is non-compliant with EVT hypotheses. To perform a correct evaluation, the test has to be run multiple times and the final outcome has to be decided by looking at the overall reject/not-reject ratio: a rejection ratio close to α identifies a system that verifies the EVT hypotheses, while a higher ratio represents a violation of the EVT hypotheses. This problem is subsequently discussed in Section 3.5.

In statistical literature, several methods exist to reduce the α_{global} value when multiple tests are performed. The most famous one is the Bonferroni correction [8]. However, all of such approaches have the negative effect of reducing the statistical power [40] that, as explained in the subsequent paragraphs, may hinder the reliability of our results. Because the number of tests is fixed and low (3), it does not worth to trade a lower α_{global} value with lower statistical power.

3.4. The sample size - False negatives

In addition to the previously cited *Type I error*, the test can fail with the so-called *Type II error*, i.e. a false-negative result. In this case, the test retains the null hypothesis when it is actually false. Unlike the α, the *Type II error* can not be directly managed by the experimenter, rather it depends on the statistical power W of the test: $W = 1 - \beta$ where $\beta = P(\text{Accept } H_0 | H_0 \text{ is false})$. Unfortunately, the statistical power is neither simple to control nor to estimate. In fact, the statistical power W depends on several parameters, including the significance level, the input data distribution, the test statistic itself and the sample size. It can, however, easily be increased by enlarging the sample size.

In our scenario, the Type II error represents the inability to detect a violation of EVT hypotheses, which consequently generates an incorrect extreme value distribution, that may lead to unsafe pWCET computation. For hard real-time systems, a preliminary study on the statistical power is therefore necessary, to both select the proper sample size and to estimate the statistical power. A recent paper studied the statistical power in the context of pWCET [48], providing some insights on the minimum sample size value.

3.5. Fulfilling the hypotheses is a property of the system and not of the single time trace

In order to produce correct results, the EVT theory requires the original statistical process generating the data, in our scenario generating the execution times, to be compliant with the aforementioned hypotheses. In the computing scenario, this is a property of both the hardware and software. The compliance proof can be provided by construction, by describing the underlying statistical process that governs the execution times. This is the behind idea of randomized cache approaches [14]: the cache is an important source of violation of the EVT hypotheses, thus such approaches try to randomize the cache behavior to make this component compliant. However, to prove by construction that a whole system is compliant requires an in-detail hardware and software analysis that may vanish the advantage of measurement-based approaches. Consequently, the use of statistical tests has been proposed to assess the compliance *a posteriori* of the time measurements, without looking at the system description.

The correct assessment of such hypotheses with a statistical testing inference requires to acquire several time traces of execution time, run the statistical tests for each time trace, and, finally, look at the results. In particular, the ratio of rejection/non-rejection of the null hypothesis provides the deduction of the statistical property searched for. A single time traces meeting the EVT hypotheses can not provide any insights about the statistical process generating it. In fact, even if a single time trace may be able to pass the checks, this must not be interpreted as the system being compliant with the EVT hypotheses. In fact, the hypotheses can be considered fulfilled when the rejection/non-rejection ratio settles around the significance level for the test, or the α_{global} in case of multiple testing as described in Section 3.4.

3.6. Analysis of the previous literature

Most of the scientific literature in probabilistic real-time checks the EVT applicability by directly verifying the i.i.d. hypothesis. The execution times independence is usually checked by performing a Ljung-Box test [3] [4] [6] [18] [52]. This is problematic for two reasons: we already described that pure independence is a too strict requirement [51] and the Ljung-Box test checks for the presence of a particular form of independence, i.e. the serial correlation. Other approaches use the Wald-Wolfowitz test (also called *runs test*), e.g. [30] [52], that suffers from the same problems: on one hand it is used as a ”pure” independence test, on the other hand, its detection capabilities refer to a particular dependence around the median value. Finally, to test the identically distributed hypothesis, the Kolmogorov-Smirnov (KS) two-sample test has been extensively used[2] [4]

[2] Not to be confused with the one-sample KS test used as Goodness-of-Fit test for the maximum domain of attraction hypothesis.

[6] [18] [30] [52] [54] [57] [58]. This test consists of dividing the sample into two parts of equal size and then comparing each other with the KS test to check whether they have the same distribution. While this can be effective against some form of violation of the identically distributed hypothesis (e.g., the later presented dataset B1 in Figure 3 of the experimental evaluation), it is easy to build counter-examples that show this test is ineffective and improperly used in our scenario. For example, let consider $x_1, x_2, ..., x_{200}$ as our execution time trace, drawn from a sequence of random variable distributed as follow: $X_{20k+1}, ..., X_{20k+10}$, for $k = 0, ..., 9$ from a distribution D_1, while the values $X_{20k+11}, ..., X_{20k+20}$, for $k = 0, ..., 9$ from a distribution D_2. Applying the KS test using the first 100 elements as the first sample and the last 100 elements as the second sample, it would result in a false-negative, being unable to detect the non-identically distributed hypothesis. This because, while the two joint cumulative distribution functions of the samples of 100 elements each are similar, their inside random variables are not identically distributed.

In Section 3.4 we showed why the sample size plays a critical role in the amount of false-negative of the statistical tests. For example, for a sample size of 200, the KS test's statistical power can be lower than 50% [47]. Previous works in probabilistic real-time used different values for the sample size: in some cases very low, e.g. 50-100 range [3] [54] [58], making the test result unreliable. Other works used a higher number of samples, about ~ 500 [23] [39] [57]. Without a proper power analysis, it is difficult to estimate the confidence in the results of these papers. Conversely, papers like [6] [52] used 10^5 or 10^6 number of samples, getting rid of the statistical power problem. Our recommendation, as suggested by the preliminary results[3] of [47], is to use at least 1000 samples in academic research, while for industrial applications a preliminary statistical power analysis must be performed. Please note that even if 500 and 1000 are sizes of the same order of magnitude, the effect on statistical power differs for several orders of magnitude.

Finally, to the best of our knowledge, none of the previously cited papers considered the significance level problem described in Section 3.3 nor the systems have been tested with a proper sequence of analyses described in Section 3.5. The only exceptions are represented by the papers of Arcaro et al. [4] and Silva et al. [54], that analyzed the p-value distributions across different time traces.

4. Towards a unified index

The *reject/not-reject* result and the absolute values of the statistics of the three previously described tests do not provide a clear and straightforward information about the time predictability coming out of the traces or about the system in general.

In this regard, the goal of our work is to introduce a much more meaningful single unified index, through which to provide a quantitative value, to express the fulfillment of the statistical hypotheses, given time traces samples. We refer to this index as *Probabilistic Predictability Index* or *PPI*. The PPI has been designed by merging the three tests discussed while maintaining their statistical properties to be able to use PPI as a hypothesis test as well. The PPI is defined over the continuous range $(0; 1)$. For PPI values near 0

[3] The cited paper performed an analysis of Goodness-of-Fit tests for pWCET and not specifically of the tests exploited in PPI. However, it is reasonable, or at least conservative, to think that the statistical power of PPI tests is in the same order of magnitude.

the time samples present strong evidence that the time series is not analyzable, because it violates EVT hypotheses. Vice versa, for PPI values near 1 the time series presents good properties and adherence to EVT hypotheses. The time series should be rejected if the PPI is lower than the predefined critical value C_{PPI}, maintaining the original statistical tests' significance. In particular, if $PPI > C_{PPI}$ the hypotheses are true and the pWCET can be safely estimated, while if $PPI \leq C_{PPI}$ at least one hypothesis is violated and any pWCET estimation would lead to unreliable results. The PPI is obtained with a set of transformations that maintain the statistical foundations of the original hypothesis tests. The capability of rejecting or not rejecting the null hypothesis is unchanged, as well as the statistical power. This is extremely important in probabilistic real-time context, due to the critical aspect of the pWCET reliability.

4.1. Probabilistic Predictability Index (PPI) construction

The statistics of the previously described tests have the following ranges[4]:

$$S_{KPSS} \in (0; +\infty) \quad S_{BDS} \in (-\infty; +\infty) \quad S_{R/S} \in (0; +\infty)$$

In order to level off these statistics we need to define a common domain $D = (0; 1)$ that will be the PPI domain. By taking into account the described desired meaning for PPI and the statistics analytical formulation available in the Appendix, we have to find the following functions:

$$\begin{gathered} f_{KPSS} : (0; +\infty) \to D \\ f_{BDS} : (-\infty; +\infty) \to D \\ f_{R/S} : (0; +\infty) \to D \end{gathered}$$

under the following constraints:

$$\begin{aligned} \lim_{x \to +\infty} f_{KPSS} &= 0 \quad \lim_{x \to 0} f_{KPSS} = 1 \\ \lim_{x \to \pm\infty} f_{BDS} &= 0 \quad \lim_{x \to 0} f_{BDS} = 1 \\ \lim_{x \to +\infty} f_{R/S} &= 0 \quad \lim_{x \to 0} f_{R/S} = 1 \end{aligned}$$

Moreover, the rejection property of the test statistics against the critical value must be maintained. Let $C_{KPSS}, C_{BDS}, C_{R/S}$ be the critical values of the respective tests, each null hypothesis has to be rejected if

$$|S_i| > C_i \quad \forall i \in \{\mathrm{KPSS}, \mathrm{BDS}, \mathrm{R/S}\} \tag{7}$$

For this reason, the Equation 7 must hold whatever transformation we apply. In order to satisfy this requirement, the $f_{KPSS}, f_{BDS}, f_{R/S}$ transformations must be continuous, positive and monotonic functions. The following functions satisfy the aforementioned properties:

$$\begin{aligned} f_{KPSS}(x) &= e^{-K_{KPSS} \cdot x} \\ f_{BDS}(x) &= e^{-K_{BDS} \cdot |x|} \\ f_{R/S}(x) &= e^{-K_{R/S} \cdot x} \end{aligned} \tag{8}$$

[4] We omit the $(\{X_i\})$ parameter of statistics S_i for brevity.

Table 1: Example of PPI values for three cases of statistical tests. The critical value is $C_{PPI} = 0.89$.

$f_{KPSS}(S_{KPSS})$	$f_{BDS}(S_{BDS})$	$f_{R/S}(S_{R/S})$	PPI	
0.96	0.91	0.92	0.92	✓
0.50	0.91	0.92	0.50	✗
0.50	0.91	0.70	0.425	✗

Now we have to select $K_{KPSS}, K_{BDS}, K_{R/S}$ in order to be compliant with the previous constraints and to get the same critical value for each test. We assign an empirical value to $K_{KPSS} = \frac{1}{4}$, then the critical value for KPSS test can be computed $C^*_{KPSS} = e^{-\frac{1}{4}C_{KPSS}}$. Since we want the same critical value for the other two tests, their constants have to be computed as follow:

$$\begin{aligned} k_{BDS} &= -\frac{\log C^*_{KPSS}}{|C_{BDS}|} \\ k_{R/S} &= -\frac{\log C^*_{KPSS}}{C_{R/S}} \end{aligned} \tag{9}$$

eventually obtaining

$$C_{PPI} := C^*_i = f_i(C_i) \quad \forall i \in \{\text{KPSS}, \text{BDS}, \text{R/S}\}$$

Assigning different values to k_{KPSS} would produce different statistic PPI values. However, it does not change its statistical meaning, because the critical values would change in a consistent manner. Choosing a higher value of k_{KPSS} shifts the PPI to produce values towards 0, vice versa a lower value of k_{KPSS} shifts the PPI to produce values towards 1. We selected the value of $k_{KPSS} = \frac{1}{4}$ such that the obtained C_{PPI} is 0.89 for $\alpha = 0.05$, i.e. about of 10% of fraction of PPI values $(0.9 - 1.0)$ are dedicated to values representing valid hypothesis, while the remaining 90% fraction (values $0.0 - 0.9$) can represent the violation degree of the hypotheses. The experimenter can change k_{KPSS} at will, without losing statistical properties, but modifying the human perception of the index value. In Table 1, we can see an example of PPI value computation, for three cases of statistical tests statistics.

Having uniformed the three statistics in the $(0, 1)$ range with the same critical values, it is possible to merge them into a unique index, by applying the following conservative approach:

- if *all* test statistics are higher than the critical value, PPI must be higher than the critical value;
- if *any* of the three test statistics is lower than the critical value, PPI must be lower than the critical value;
- if *more than one* test statistics are lower than the critical value, PPI must be lower than the minimum statistic.

This approach ensures that the statistical test meaning is not changed: we can compare PPI with the critical value to assess all three hypotheses, assuming that if any hypothesis is violated the test will reject the null hypothesis.

Thus, we applied the following merging transformation:

$$PPI := \begin{cases} \min_{\forall i} f_i(S_i) \cdot \prod_{i \in v^*} [1 - (C_{PPI} - f_i(S_i))] & v \neq \varnothing \\ \frac{1}{3} \sum_{\forall i} f_i(S_i) & v = \varnothing \end{cases} \tag{10}$$

where v is the violation set, i.e. $v = \{i | f_i(S_i) < C_{PPI}\}$, and v^* is the violation set without the minimum, i.e. $v^* = \{v \setminus \arg\min_{\forall i} f_i(S_i)\}$. If no violation occurs in the three tests, the result is the arithmetic mean of the three values, that is greater than C_{PPI}. Otherwise, the PPI is equal to the minimum statistic potentially multiplied by other statistics that violate C_{PPI}. This leads to a PPI value lower than C_{PPI}, guaranteeing the statistical hypothesis testing property. The arithmetic mean computes PPI using the same *weight* for the three tests because each test verifies one different hypothesis.

To summarize, the PPI value can be computed by using Equation 10 and compared with the critical value C_{PPI}. In probabilistic real-time scenarios, if $PPI < C_{PPI}$ the following null hypothesis (H_0) has to be rejected in favor of the alternative one (H_1):

H_0 : the time trace verifies the EVT hypotheses
H_1 : at least one EVT hypothesis is violated

5. Experimental Evidences

In this section, we present the experimental evaluation of the chosen statistical tests and the proposed index PPI. The datasets have been analyzed thanks to the open-source software *chronovise* [43], where we implemented the algorithm to compute the PPI index. The algorithm is also available separately as MATLAB script [46]. The expectation is to get high rejection rates for time traces that do not satisfy the conditions described in Section 2.1. On the other hand, if the source of the samples is a distribution that verifies the EVT hypotheses, then the rejection rate should settle around the significance value α.

5.1. Time trace sources

For characterizing the properties of the proposed test, we used both synthetic time samples and real benchmark executions. The first class of time traces has been designed to stress the detection capability of each statistical test, by using synthetic distributions with well-known statistical properties. The real benchmarks are instead executed on different hardware platforms, with known real-time capabilities, to show an evaluation of the probabilistic predictability of the target system.

Without losing generality, we evaluated the tests with a level of significance $\alpha = 0.05$. This means that we expected for each test a type I error (i.e. false-positive rate) of 5%. In our scenario, this is a conservative error: each test excludes 5% of the times a dataset that is actually valid for EVT estimation. The overall type I error can be computed using Equation 6, obtaining 14% ($\alpha_{global} = 0.14$).

Synthetic sources. Let $X_{a:b}$ be an ordered subset of the full time trace $X_{1:n}$. For synthetic and controlled time traces we used both i.i.d. and non i.i.d. sources. For the former, we selected the following EVT-compliant distributions, that are expected to pass the statistical tests:

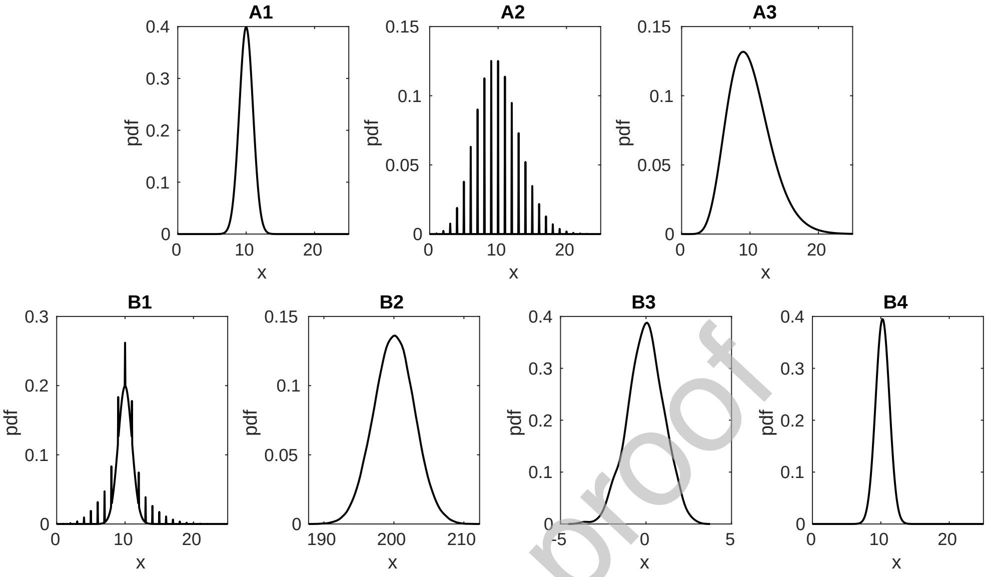


Figure 2: The Probability Distribution Functions of the synthetic benchmarks considered.

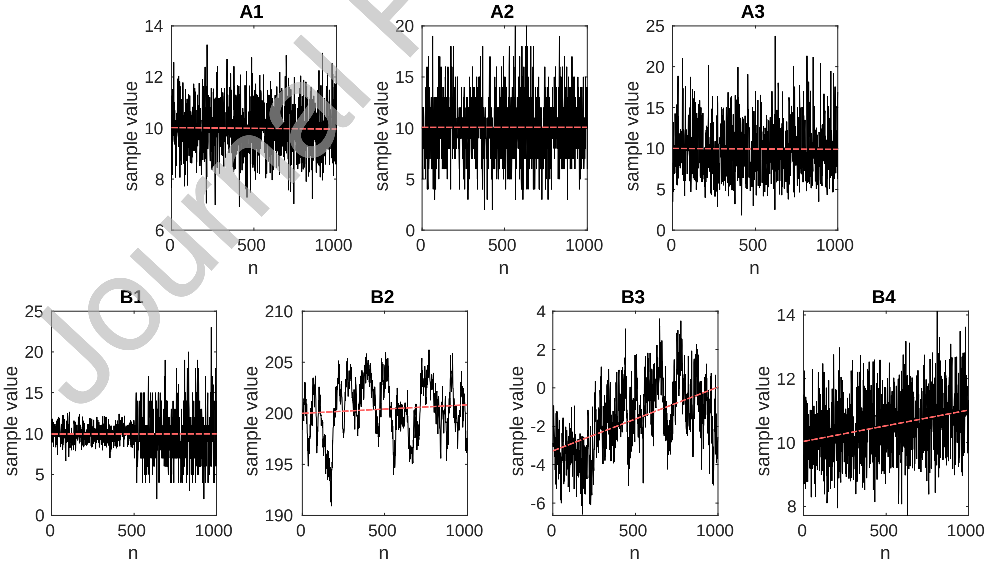


Figure 3: Plots of the realization of 1000 random variables with the distributions of the synthetic benchmarks considered. The red dashed line shows the long-term trend, computed as a linear interpolation of all the points.

A1 $X_{1:n} \sim \mathcal{N}(10, 1)$: Gaussian (normal)

A2 $X_{1:n} \sim \mathcal{P}(10)$: Poisson

A3 $X_{1:n} \sim \Gamma(10, 1)$: Gamma

Then we tested three non-compliant distributions, that are expected to violate at least one of the i.i.d. hypotheses:

B1 $X_{1:\frac{n}{2}} \sim \mathcal{N}(10, 1); X_{\frac{n}{2}+1:n} \sim \mathcal{P}(1)$: a normally distributed time trace for the first half part and then a Poisson distribution; it represents a sequence of independent but not identically distributed samples.

B2 $X_{1:n} \sim AR(2)$: an auto-regressive model of order 2, with constant 10 and auto-regressive coefficients $(0.7, 0.25)$. This class represents a short-range dependent time source.

B3 $X_{1:n} \sim ARFIMA(\frac{1}{2}, 0, 0, 0, \frac{1}{4})$: an auto-regressive fractionally integrated moving average model with AR, MA, and I coefficients zero, constant $\frac{1}{2}$ and $d = \frac{1}{4}$. This class represents a time source with a long memory.

B4 $X_{1:n} = \{\forall i \in [1; n] | X_i \sim N(10 + 0.001 \cdot i, 1)\}$: non identically distributed samples with long-range dependence, but short-range independent.

We have drawn a total of 1 000 000 samples for each distribution and then we split into groups of size 1 000 for a total of 1 000 evaluations. The pdfs of these distributions have been plotted in Figure 2, as well as an example of traces in Figure 3. It is possible to note that A1, A2, A3 appear as random, B1 is composed of two modes, B2 presents a clear short-range dependence, while B3 and B4 have long-term trends.

Real sources. Concerning the experimental evaluation on real platforms, we run four state-of-the-art benchmarks of the WCET Mälardalen suite [24]: `sqrt`, `minver`, `fdct`, `complex`. We implemented each benchmark onto five different platforms, whose well-known architecture characteristics introduce different degrees of unpredictability:

R1 `PIC`: a PIC18F45K50 microcontroller without operating system;

R2 `STM`: time-deterministic platform with a L1D and L1I cache: STM32F7 board programmed bare-metal without operating system;

R3 `MIO`: time-deterministic platform with a real-time operating system: the STM32F4 with Miosix operating system[5];

R4 `ODR`: embedded development-board unpredictable platform: multi-core Odroid XU-3 with a Linux OS (vanilla kernel);

R5 `INT`: a desktop system, completely unpredictable platform: multi-core Intel i7 with a Linux OS (vanilla kernel).

[5] `http://miosix.org/`

	E[PPI]	VAR[PPI]	Reject$_{PPI}$	Reject$_{KPSS}$	Reject$_{BDS}$	Reject$_{R/S}$
A1	0.9374	1e-3	13.9%	6.8%	5.5%	4.5%
A2	0.9388	1e-3	12.3%	5.3%	4.9%	4.3%
A3	0.9393	1e-3	11.4%	4.9%	5.3%	3.4%
B1	0.6302	2e-3	100%	4.7%	100%	6.9%
B2	0.0058	1e-5	100%	100%	100%	100%
B3	0.5228	3e-2	100%	83.1%	99.2%	98%
B4	0.1319	3e-3	100%	100%	5.5%	100%

Table 2: Tests rejection results of synthetic time traces analysis.

R1 is a simple processor, time-deterministic and constant instruction timing. R2 and R3 also are time-deterministic platforms, with no features that can affect the execution time predictability, with the exception of the L1 caches of R2, which introduce a timing dependence among the benchmark execution. R4 is, instead, an embedded development board with several advanced features, making the execution time unpredictable. R5 is even more unpredictable because it is a general-purpose machine and, consequently, contains several unpredictable hardware features, such as System Management Interrupts.

The benchmarks have been slightly modified to add: (1) a PRNG for input data generation (except for `complex` where the input is constant), (2) an external loop to run the benchmark multiple times, (3) a toggling mechanism for a GPIO to signal the start and stop of a benchmark execution. To maintain consistency among all platforms, the PRNG has been initialized with the same seed. This way each platform generates the same sequence of pseudo-random inputs to the benchmarks. The time measurements have been acquired by measuring the GPIO interval between the rising edge (start of the computation) and the falling edge (end of the computation), using a commercial logical analyzer with a 10ns resolution. Each benchmark then has been executed 100 000 times by using time series of size 1 000 for statistical testing, for a total number of 100 estimations for each benchmark.

5.2. Results

5.2.1. Synthetic samples

The results on time traces from synthetic sources are shown in Table 2. For i.i.d. datasets (A1-A3) it is possible to notice a rejection rate based on evaluations of single tests around 5%, that actually matches the chosen significance level α. The rejection rate of the composed index PPI is slightly below 14%, that is the significance level value computed by using Equation 6. This value represents the false-positive error rate, i.e. the percentage of time series that is discarded even if they are generated by compliant sources.

Regarding the results of time traces that do not satisfy at least one EVT condition (B1-B4), we can notice the PPI rejection rate is always 100%. We can observe that the power of BDS is high for B1-B3 but it is not for B4, where KPPS and R/S are able to reject the hypothesis. On the contrary, for B1 only BDS appears to be sufficiently

		Reject$_{PPI}$	Reject$_{KPSS}$	Reject$_{BDS}$	Reject$_{R/S}$
sqrt	R1	23%	7%	15%	6%
	R2	24%	2%	20%	3%
	R3	29%	6%	23%	7%
	R4	14%	1%	13%	1%
	R5	34%	20%	12%	21%
minver	R1	16%	6%	6%	6%
	R2	15%	9%	5%	6%
	R3	17%	10%	7%	8%
	R4	44%	1%	44%	1%
	R5	78%	61%	31%	66%
fdct	R1	8%	2%	5%	2%
	R2	20%	4%	15%	4%
	R3	100%	99%	100%	99%
	R4	62%	16%	48%	15%
	R5	81%	67%	45%	71%
complex	R1	79%	19%	72%	16%
	R2	56%	5%	52%	3%
	R3	100%	0%	100%	0%
	R4	92%	5%	92%	1%
	R5	100%	60%	95%	71%

Table 3: Tests rejections of R1-R5 real hardware time traces.

powerful. Moreover, it is worth highlighting that B1 is a non-identically distributed time series, but KPSS is not able to detect it, while BDS provides for it. This is due to the lack of statistical power of KPSS in case of weak stationary, but not strict stationary time series [37].

5.2.2. Real platforms samples

Table 3 and Figure 4 show the results when time traces are generated by executing benchmark applications on the aforementioned platforms. It is possible to observe the expected trend of generating *less-compliant* time traces, with the increasing of the hardware complexity. The traces generated by the `complex` benchmark are hardly analyzable for all platforms, due to the lack of variability. This is in contrast with the common logic behind the WCET analysis for which a more stable timing is preferable. The statistical tests described and the EVT in general instead, require a minimal degree of variability, as also shown by Lima et al. [35]. The benchmark `complex` lacks variability as it is the only benchmark one – out of the four benchmarks – that performs simple computation on the same input data for each iteration. For example, in the PIC microcontroller case (R1), the variability of time measurements of `complex` benchmark is due only to the measurement errors of the instrumentation: the input is constant for this benchmark and the PIC microcontroller has a constant instruction timing architecture, making the actual execution time constant. The same measurement errors affect the other experiments, but the variability from software and hardware causes dominates the measurement errors.

For all the other benchmarks, the simple PIC microcontroller generates deterministic time traces that lead to a low rejection rate, close to the significance level, i.e. the false

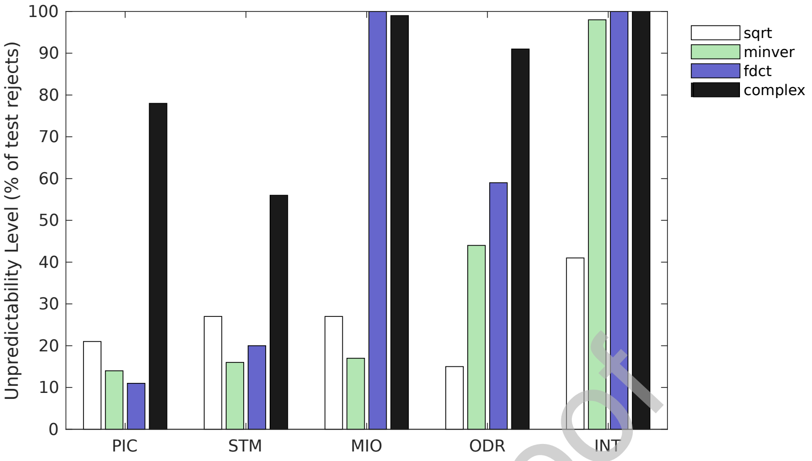


Figure 4: The reject rates of the four benchmarks executed onto the described platforms.

positive rate. The MIO and STM platforms present higher values of rejection, caused by the presence of the operating system and cache memories, respectively. As expected, and with the only exception of `sqrt` case, the probabilistic theory cannot be used for the Odroid, and least of all, the Intel CPU based machine. The only unexpected outlier is the `fdct` benchmark on the Miosix board. Here the rejection rate is 100% without a clear reason. By observing the time traces, we hypothesize that the instruction prefetcher, the board is equipped with, causes large recurrent variations compared to the intrinsic variability of the `fdct` benchmark, that triggers the detection of a short-range dependence. However, this conclusion requires a more in-depth analysis of the specific system, that falls outside the scope of this paper.

Generally, the results shown in Table 3 highlight an important fact: the single PPI hypothesis test result depends on both the system and the considered workload. Moreover, the result is a random variable, that consequently requires to be sampled several times to assess the capability of analyzing a system by using EVT. Again, the PPI hypothesis test result (or the result of any single test) is a property of the time traces provided, not of the system generating such traces.

5.3. Linux PREEMPT_RT analysis

The Linux kernel has been built as a general operating system and thus not appropriate for real-time computing, since its main performance goal can be considered maximizing the average throughput. For this reason, the PREEMPT_RT patch has been developed since the first decade of the 2000s, in order to add predictability and low-latency to the Linux kernel. A comprehensive survey of scientific works related to PREEMPT_RT is available [45].

In this experimental evaluation, we run the same benchmarks used for R1-R5 but on an Odroid H2, a quad-core x86-64 platform based on COTS components and consequently subject to unpredictable latencies. The goal is to verify if the introduction of

		ACET	WCOT	DI	Rej$_{PPI}$	Rej$_{KPSS}$	Rej$_{BDS}$	Rej$_{R/S}$
sqrt	P1	408 262 ns	938 217 ns	7 508.3	89%	39%	67%	48%
	P2	390 508 ns	426 728 ns	77.5	88%	0%	88%	0%
	P3	388 413 ns	413 123 ns	65.6	6%	1%	5%	1%
minver	P1	485 854 ns	1 114 823 ns	6 938.3	94%	48%	72%	47%
	P2	466 040 ns	1 914 380 ns	128.5	89%	0%	89%	0%
	P3	464 227 ns	542 293 ns	69.4	43%	0%	43%	0%
fdct	P1	470 208 ns	789 290 ns	6 482.7	92%	39%	67%	48%
	P2	450 845 ns	478 049 ns	46.8	100%	0%	100%	0%
	P3	450 561 ns	487 578 ns	35.5	100%	0%	100%	0%
complex	P1	429 737 ns	764 678 ns	7 206.3	87%	20%	71%	22%
	P2	410 894 ns	447 453 ns	63.9	97%	0%	97%	0%
	P3	410 585 ns	443 200 ns	46.7	100%	13%	100%	33%

Table 4: Linux PREEMPT_RT result for the four WCET benchmarks considered.

PREEMPT_RT improves the predictability of execution times and which effects PREEMPT_RT has on the applicability of the probabilistic theory. This is done exploiting the PPI index previously defined. The tests have been performed on three scenarios:

P1 On a plain vanilla Linux and the task having no special configuration (like the previous R5 case);

P2 On a plain vanilla Linux but the task configured with real-time priority and with core pinning;

P3 On a PREEMPT_RT kernel and the task configured with real-time priority and a core pinning;

The task under analysis runs together with contenders on other cores that cause interferences at both architecture and operating system levels. The contention has been generated thanks to the *stress-ng* tool[6]. Like the previous tests, each benchmark has been executed 100 000 times by using time series of size 1 000 for statistical testing, for a total number of 100 estimations for each benchmark and scenario.

The results are shown in Table 4. As common in PREEMPT_RT works, we also computed the Average-Case Execution Time (ACET), the Worst-Case Observed Time (WCOT), the Dispersion Index (DI), the PPI value and its components. DI is computed as follows: $\frac{\mu}{\sigma}$, where μ is the mean value of the time trace and σ the standard deviation. Looking at the traditional ACET, WCOT and DI values, it is possible to notice that applying the correct task real-time priority configuration and the $PREEMPT\_RT$ patch are essential to obtain low variability. In particular, the dispersion index of P2 is at least one order of magnitude lower than P1, and P3 is slightly better than P2. This means that the execution times vary in a smaller interval, making it more predictable. It is

[6] `http://kernel.ubuntu.com/~cking/stress-ng/`

possible to notice that the WCOT is much lower in P3 than the other cases. Setting real-time priority is not sufficient to reduce sporadic high-level latencies, as proved by the large WCOT of `minver` benchmark, even larger than the P1 scenario. To reduce these latencies is crucial the inclusion of the PREEMPT_RT patch in Linux kernel. Focusing on PPI index, there is no a direct link between DI, WCOT or ACET with respect to the satisfaction of EVT hypotheses. In fact, even if PREEMPT_RT seems to improve the satisfaction of hypothesis for `sqrt` and `minver`, this is not true for `fdct` and `complex`. Comparing the PPI index with its critical value, it is possible to conclude that only the scenario sqrt running on PREEMPT_RT satisfies the EVT hypothesis and makes possible the estimation of a correct distribution.

This example of PREEMPT_RT shows that it is not sufficient to improve the average-case, worst-case, nor the predictability of the platform to improve the satisfaction of the EVT hypotheses. A future possible work may be investigating why PREEMPT_RT is not able to improve the `fdct` and `complex` cases and which are the internal kernel mechanisms and/or architecture components that prevent this.

5.4. Summary

Considering the synthetic time traces, the PPI resulted to be very effective in the detection of the violation of i.i.d. property. Indeed, the non-compliant time traces have been rejected with a 100% rate, while the rejection rate of the compliant ones settled in the range 11.4% – 13.9%. This range represents the false-positive rate of the test, which is, however, lower than the expected theoretical value of 14%.

For the real-time traces from real benchmark applications, we can notice that the trend of the PPI rejection rate is coherent with the index meaning. In fact, for predictable platforms, we experienced low rejection rates, while for complex platforms this is very close to 100%, as expected.

Finally, we computed the PPI index for the same benchmarks running on a Linux embedded platform to observe if the PREEMPT_RT patch can make the system EVT-compliant, concluding that it is not possible to claim a priori satisfiability of the hypotheses, nor we can generally conclude that improving the usual average-case or worst-case metrics improves the satisfiability as well. This makes the PPI analysis essential.

6. Conclusion

Statistical hypothesis testing plays a key role in the reliability of probabilistic real-time estimation and the consequent safety of critical systems. However, some state-of-the-art works do not follow a systematic procedure in performing statistical tests. This paper aimed at highlighting the problems affecting part of the scientific literature in probabilistic real-time and discussed which factors affect the reliability of statistical test procedures. The PPI absolute values can be used to compare the predictability of time traces, from a probabilistic real-time standpoint. More in general, we can use PPI to compare different systems and workloads, as also shown by experimental evaluation. A use-case of this index has been proposed to check the real-time capability improvement when the PREEMPT_RT patch is applied to a Linux kernel.

Acknowledgement

This work has been partially funded by the RECIPE H2020 Project [20] (Grant no. 801137).

References

[1] J. Abella, D. Hardy, I. Puaut, E. Qui nones, and F. J. Cazorla. 2014. On the Comparison of Deterministic and Probabilistic WCET Estimation Techniques. In *2014 26th Euromicro Conference on Real-Time Systems*. IEEE, 266–275. `https://doi.org/10.1109/ECRTS.2014.16`

[2] Jaume Abella, Maria Padilla, Joan Del Castillo, and Francisco J. Cazorla. 2017. Measurement-Based Worst-Case Execution Time Estimation Using the Coefficient of Variation. *ACM Trans. Des. Autom. Electron. Syst.* 22, 4, Article 72 (June 2017), 29 pages. `https://doi.org/10.1145/3065924`

[3] Jaume Abella, Maria Padilla, Joan Del Castillo, and Francisco J. Cazorla. 2017. Measurement-Based Worst-Case Execution Time Estimation Using the Coefficient of Variation. *ACM Trans. Des. Autom. Electron. Syst.* 22, 4, Article Article 72 (June 2017), 29 pages. `https://doi.org/10.1145/3065924`

[4] Luís Fernando Arcaro, Karila Palma Silva, and Rômulo Silva De Oliveira. 2018. On the Reliability and Tightness of GP and Exponential Models for Probabilistic WCET Estimation. *ACM Trans. Des. Autom. Electron. Syst.* 23, 3, Article Article 39 (March 2018), 27 pages. `https://doi.org/10.1145/3185154`

[5] R. Bender and S. Lange. 2001. Adjusting for multiple testing - when and how? *Journal of Clinical Epidemiology* 54, 4 (April 2001), 343–349.

[6] Kostiantyn Berezovskyi, Luca Santinelli, Konstantinos Bletsas, and Eduardo Tovar. 2014. WCET Measurement-Based and Extreme Value Theory Characterisation of CUDA Kernels. In *Proceedings of the 22nd International Conference on Real-Time Networks and Systems (RTNS 14)*. Association for Computing Machinery, New York, NY, USA, 279288. `https://doi.org/10.1145/2659787.2659827`

[7] G. Bernat, A. Colin, and S. M. Petters. 2002. WCET analysis of probabilistic hard real-time systems. In *23rd IEEE Real-Time Systems Symposium, 2002. RTSS 2002.* IEEE, 279–288. `https://doi.org/10.1109/REAL.2002.1181582`

[8] C.E. Bonferroni. 1936. *Teoria statistica delle classi e calcolo delle probabilit.* Vol. 8. Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze. 3–62 pages.

[9] C. Brandolese, S. Corbetta, and W. Fornaciari. 2011. Software energy estimation based on statistical characterization of intermediate compilation code. In *IEEE/ACM International Symposium on Low Power Electronics and Design*. 333–338. `https://doi.org/10.1109/ISLPED.2011.5993659`

[10] W. A. Broock, J. A. Scheinkman, W. D. Dechert, and B. LeBaron. 1996. A test for independence based on the correlation dimension. *Econometric Reviews* 15, 3 (1996), 197–235. https://doi.org/10.1080/07474939608800353

[11] A. Burns and S. Edgar. 2000. Predicting computation time for advanced processor architectures. In *Proceedings 12th Euromicro Conference on Real-Time Systems. Euromicro RTS 2000*. 89–96. https://doi.org/10.1109/EMRTS.2000.853996

[12] Enrique Castillo, Ali S Hadi, Narayanaswamy Balakrishnan, and José-Mariá Sarabia. 2005. *Extreme value and related models with applications in engineering and science.* Wiley Hoboken, NJ.

[13] Francisco J. Cazorla, Leonidas Kosmidis, Enrico Mezzetti, Carles Hernandez, Jaume Abella, and Tullio Vardanega. 2019. Probabilistic Worst-Case Timing Analysis: Taxonomy and Comprehensive Survey. *ACM Comput. Surv.* 52, 1, Article 14 (Feb. 2019), 35 pages. https://doi.org/10.1145/3301283

[14] F. J. Cazorla, E. Quiñones, T. Vardanega, L. Cucu, B. Triquet, G. Bernat, E. Berger, J. Abella, F. Wartel, M. Houston, L. Santinelli, L. Kosmidis, C. Lo, and D. Maxim. 2013. PROARTIS: Probabilistically Analyzable Real-Time Systems. *ACM Trans. Embed. Comput. Syst.* 12, 2s, Article 94 (May 2013), 26 pages. https://doi.org/10.1145/2465787.2465796

[15] Michel Couillard and Matt Davison. 2005. A comment on measuring the Hurst exponent of financial time series. *Physica A: Statistical Mechanics and its Applications* 348 (2005), 404 – 418. https://doi.org/10.1016/j.physa.2004.09.035

[16] Robert Ian Davis and Liliana Cucu-Grosjean. 2019. A Survey of Probabilistic Schedulability Analysis Techniques for Real-Time Systems. *LITES: Leibniz Transactions on Embedded Systems* (May 2019), 1–53. http://eprints.whiterose.ac.uk/146181/ © Robert I. Davis and Liliana Cucu-Grosjean.

[17] S. Edgar and A. Burns. 2001. Statistical analysis of WCET for scheduling. In *Proceedings 22nd IEEE Real-Time Systems Symposium (RTSS 2001) (Cat. No.01PR1420)*. 215–224. https://doi.org/10.1109/REAL.2001.990614

[18] M. Fernandez, D. Morales, L. Kosmidis, A. Bardizbanyan, I. Broster, C. Hernandez, E. Quinones, J. Abella, F. Cazorla, P. Machado, and L. Fossati. 2017. Probabilistic timing analysis on time-randomized platforms for the space domain. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2017*. 738–739. https://doi.org/10.23919/DATE.2017.7927087

[19] R. A. Fisher and L. H. C. Tippett. 1928. Limiting forms of the frequency distribution of the largest or smallest member of a sample. *Mathematical Proceedings of the Cambridge Philosophical Society* 24, 2 (1928), 180–190.

[20] William Fornaciari, Giovanni Agosta, David Atienza, and Carlo et al. Brandolese. 2018. Reliable Power and Time-constraints-aware Predictive Management of Heterogeneous Exascale Systems. In *Proceedings of the 18th International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '18)*. ACM, New York, NY, USA, 187–194. https://doi.org/10.1145/3229631.3239368

[21] S. Jiménez Gil, I. Bate, G. Lima, L. Santinelli, A. Gogonel, and L. Cucu-Grosjean. 2017. Open Challenges for Probabilistic Measurement-Based Worst-Case Execution Time. *IEEE Embedded Systems Letters* 9, 3 (Sept 2017), 69–72.

[22] Boris Vladimirovich Gnedenko. 1948. On a local limit theorem of the theory of probability. *Uspekhi Matematicheskikh Nauk* 3, 3 (1948), 187–194.

[23] Fabrice Guet, Luca Santinelli, and Jérôme Morio. 2016. On the reliability of the probabilistic worst-case execution time estimates. In *8th European Congress on Embedded Real Time Software and Systems (ERTS 2016)*.

[24] Jan Gustafsson, Adam Betts, Andreas Ermedahl, and Björn Lisper. 2010. The Mälardalen WCET Benchmarks – Past, Present and Future. In *10th International Workshop on Worst-Case Execution Time Analysis, WCET 2010, July 6, 2010, Brussels, Belgium*, Björn Lisper (Ed.). OCG, Brussels, Belgium, 137–147.

[25] H. E. HURST. 1951. Long term storage capacity of reservoirs. *ASCE Transactions* 116, 776 (1951), 770–808. https://ci.nii.ac.jp/naid/10011004012/en/

[26] BelaireFranch Jorge and Contreras Dulce. 2002. How to compute the BDS test: a software comparison. *Journal of Applied Econometrics* 17, 6 (2002), 691–699. https://doi.org/10.1002/jae.679 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/jae.679

[27] Frank J. Massey Jr. 1951. The Kolmogorov-Smirnov Test for Goodness of Fit. *J. Amer. Statist. Assoc.* 46, 253 (1951), 68–78.

[28] R. Kirner and P. Puschner. 2008. Obstacles in Worst-Case Execution Time Analysis. In *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. 333–339.

[29] L. Kosmidis, C. Curtsinger, E. Quiones, J. Abella, E. Berger, and F. J. Cazorla. 2013. Probabilistic timing analysis on conventional cache designs. In *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*. 603–606. https://doi.org/10.7873/DATE.2013.132

[30] L. Kosmidis, E. Quiones, J. Abella, T. Vardanega, I. Broster, and F. J. Cazorla. 2014. Measurement-Based Probabilistic Timing Analysis and Its Impact on Processor Architecture. In *2014 17th Euromicro Conference on Digital System Design*. 401–410. https://doi.org/10.1109/DSD.2014.50

[31] O. Kotaba, J. Nowotsch, M. Paulitsch, S. M. Petters, and H. Theiling. 2013. Multicore in real-time systems–temporal isolation challenges due to shared resources. In *Workshop on Industry-Driven Approaches for Cost-effective Certification of Safety-Critical, Mixed-Criticality Systems*. Grenoble, France, 6.

[32] Denis Kwiatkowski, Peter C.B. Phillips, Peter Schmidt, and Yongcheol Shin. 1992. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics* 54, 1 (1992), 159 – 178. https://doi.org/10.1016/0304-4076(92)90104-Y

[33] M. R. Leadbetter and Holger Rootzen. 1988. Extremal Theory for Stochastic Processes. *Ann. Probab.* 16, 2 (04 1988), 431–478.

[34] Anthony W. Ledford and Jonathan A. Tawn. 2003. Diagnostics for Dependence within Time Series Extremes. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 65, 2 (2003), 521–543. `http://www.jstor.org/stable/3647519`

[35] G. Lima and I. Bate. 2017. Valid Application of EVT in Timing Analysis by Randomising Execution Time Measurements. In *2017 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*. 187–198. `https://doi.org/10.1109/RTAS.2017.17`

[36] G. Lima, D. Dias, and E. Barros. 2016. Extreme Value Theory for Estimating Task Execution Time Bounds: A Careful Look. In *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*. 200–211. `https://doi.org/10.1109/ECRTS.2016.20`

[37] L.R. Lima and B. Neri. 2013. A Test for Strict Stationarity. In *Uncertainty Analysis in Econometrics with Applications*, Van-Nam Huynh, Vladik Kreinovich, Songsak Sriboonchitta, and Komsan Suriya (Eds.). Springer Berlin Heidelberg, 17–30.

[38] Andrew W Lo. 1989. *Long-term memory in stock market prices.* Technical Report. National Bureau of Economic Research. `https://doi.org/10.3386/w2984`

[39] Y. Lu, T. Nolte, I. Bate, and L. Cucu-Grosjean. 2012. A Statistical Response-Time Analysis of Real-Time Embedded Systems. In *2012 IEEE 33rd Real-Time Systems Symposium*. 351–362. `https://doi.org/10.1109/RTSS.2012.85`

[40] Shinichi Nakagawa. 2004. A farewell to Bonferroni: the problems of low statistical power and publication bias. *Behavioral Ecology* 15, 6 (11 2004), 1044–1045. `https://doi.org/10.1093/beheco/arh107` arXiv:http://oup.prod.sis.lan/beheco/article-pdf/15/6/1044/17274115/arh107.pdf

[41] Bo Qian and Khaled Rasheed. 2004. Hurst exponent and financial market predictability. In *Proceedings of the Second IASTED International Conference on Financial Engineering and Applications.*

[42] Petar Radojković, Sylvain Girbal, Arnaud Grasset, Eduardo Quiñones, Sami Yehia, and Francisco J. Cazorla. 2012. On the Evaluation of the Impact of Shared Resources in Multithreaded COTS Processors in Time-critical Environments. *ACM Trans. Archit. Code Optim.* 8, 4, Article 34 (Jan. 2012), 25 pages. `https://doi.org/10.1145/2086696.2086713`

[43] F. Reghenzani, G. Massari, and W. Fornaciari. 2018. chronovise: Measurement-Based Probabilistic Timing Analysis framework. *Journal of Open Source Software* 3, 28 (2018), 711. `https://doi.org/10.21105/joss.00711`

[44] F. Reghenzani, G. Massari, and W. Fornaciari. 2018. The Misconception of Exponential Tail Upper-Bounding in Probabilistic Real-Time. *IEEE Embedded Systems Letters* (2018), 1–1. `https://doi.org/10.1109/LES.2018.2889114`

[45] Federico Reghenzani, Giuseppe Massari, and William Fornaciari. 2019. The Real-Time Linux Kernel: A Survey on PREEMPT_RT. *ACM Comput. Surv.* 52, 1, Article 18 (Feb. 2019), 36 pages. https://doi.org/10.1145/3297714

[46] Federico Reghenzani, Giuseppe Massari, and William Fornaciari. 2020. Probabilistic Predictability Index MATLAB Script. https://doi.org/10.5281/zenodo.3596957

[47] Federico Reghenzani, Giuseppe Massari, Luca Santinelli, and William Fornaciari. 2019. Statistical power estimation dataset for external validation GoF tests on EVT distribution. *Data in Brief* 25 (2019), 104071. https://doi.org/10.1016/j.dib.2019.104071

[48] Federico Reghenzani, Luca Santinelli, and William Fornaciari. 2019. Why Statistical Power Matters for Probabilistic Real-time: Work-in-progress. In *Proceedings of the International Conference on Embedded Software Companion (EMSOFT '19)*. ACM, New York, NY, USA, Article 3, 2 pages. https://doi.org/10.1145/3349568.3351555

[49] R.D. Reiss and M. Thomas. 2007. *Statistical Analysis of Extreme Values: with Applications to Insurance, Finance, Hydrology and Other Fields.* Birkhäuser Basel. https://books.google.it/books?id=I-g-I_I2OZIC

[50] L. Santinelli, F. Guet, and J. Morio. 2017. Revising Measurement-Based Probabilistic Timing Analysis. In *2017 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*. 199–208. https://doi.org/10.1109/RTAS.2017.16

[51] L. Santinelli, J. Morio, G. Dufour, and D. Jacquemart. 2014. On the Sustainability of the Extreme Value Theory for WCET Estimation. In *14th International Workshop on Worst-Case Execution Time Analysis (OpenAccess Series in Informatics (OASIcs))*, Vol. 39. 21–30. https://doi.org/10.4230/OASIcs.WCET.2014.21

[52] Luca Santinelli, Jérôme Morio, Guillaume Dufour, and Damien Jacquemart. 2014. On the Sustainability of the Extreme Value Theory for WCET Estimation. In *14th International Workshop on Worst-Case Execution Time Analysis (OpenAccess Series in Informatics (OASIcs))*, Heiko Falk (Ed.), Vol. 39. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 21–30. https://doi.org/10.4230/OASIcs.WCET.2014.21

[53] G. William Schwert. 1989. Tests for Unit Roots: A Monte Carlo Investigation. *Journal of Business & Economic Statistics* 7, 2 (1989). https://doi.org/10.1198/073500102753410354

[54] K. P. Silva, L. F. Arcaro, D. B. de Oliveira, and R. S. de Oliveira. 2018. An Empirical Study on the Adequacy of MBPTA for Tasks Executed on a Complex Computer Architecture with Linux. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, Vol. 1. 321–328. https://doi.org/10.1109/ETFA.2018.8502513

[55] M. A. Stephens. 1974. EDF Statistics for Goodness of Fit and Some Comparisons. *J. Amer. Statist. Assoc.* 69, 347 (1974), 730–737. https://doi.org/10.1080/01621459.1974.10480196

[56] Vadim Teverovsky, Murad S Taqqu, and Walter Willinger. 1999. A critical look at Lo's modified R/S statistic. *Journal of Statistical Planning and Inference* 80, 1 (1999), 211 – 227. `https://doi.org/10.1016/S0378-3758(98)00250-X`

[57] F. Wartel, L. Kosmidis, A. Gogonel, A. Baldovino, Z. Stephenson, B. Triquet, E. Quiones, C. Lo, E. Mezzetta, I. Broster, J. Abella, L. Cucu-Grosjean, T. Vardanega, and F. J. Cazorla. 2015. Timing analysis of an avionics case study on complex hardware/software platforms. In *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*. 397–402. `https://doi.org/10.7873/DATE.2015.0189`

[58] F. Wartel, L. Kosmidis, C. Lo, B. Triquet, E. Quiones, J. Abella, A. Gogonel, A. Baldovin, E. Mezzetti, L. Cucu, T. Vardanega, and F. J. Cazorla. 2013. Measurement-based probabilistic timing analysis: Lessons from an integrated-modular avionics case study. In *2013 8th IEEE International Symposium on Industrial Embedded Systems (SIES)*. 241–248. `https://doi.org/10.1109/SIES.2013.6601497`

Appendix A. Test statistics

Appendix A.1. KPSS

Omitting the mathematical proofs available in the original paper [32], the test statistic S_{KPSS} can be computed as:

$$\eta(X) = \left(\sum_{i=1}^{n}(X_i - \bar{X})\right)^2 - n^2$$

$$S_{\text{KPSS}}(X) = \frac{\eta(X)}{\sigma_{X,l}}$$

where σ_X is the consistent estimate of the error variance computed for lags $1, ..., l$. The value of l can be computed with the following well-known formula [53]:

$$l = 12\sqrt[4]{\frac{n}{100}}$$

Appendix A.2. BDS

The test statistic can be computed with the following formula:

$$S_{\text{BDS}} = \sqrt{n-m+1}\,\frac{c_{m,n} - c^{m}_{1,m-n+1}}{\sigma_{m,n}} \tag{A.1}$$

where n is the sample size, m is the *embedding dimension*, $\sigma_{m,n}$ the consistent variance estimator and $c_{a,b}$ is defined as follow [26]:

$$c_{a,b} = 2\frac{1}{(b-a+1)(b-a)}\sum_{s=a}^{b}\sum_{t=s+1}^{b}\prod_{j=0}^{a-1} I(X_{s-j}, X_{t-j})$$

where

$$I(X_{s-j}, X_{t-j}) = \begin{cases} 1 & \text{if } |X_{s-j} - X_{t-j}| < \varepsilon \\ 0 & \text{otherwise} \end{cases}$$

for some $\varepsilon > 0$. We do not provide here a detailed explanation of the above formulas, leaving the reader to examine them in detail in the cited statistical articles.

Under independence conditions, the S_{BDS} is normally distributed, therefore the critical region is obtained using the well-known t-student inverse-cdf.

Appendix A.3. R/S Statistic

Given the time series $X = \{X_1, X_2, ..., X_n\}$ and its mean value $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ the function *cumulative sum* is defined as:

$$Z_c(X) = \sum_{i=1}^{c}(X_i - \bar{X})$$

The test statistic is the defined as:

$$S_{R/S}(X) = \frac{1}{\sqrt{n}} \frac{\max_c(Z_c(X)) - \min_c(Z_c(X))}{\sigma_X}$$

where σ_X is the sample standard deviation. If the values are uncorrelated the statistic follows the distribution having the following cdf:

$$F(v) = 1 + 2\sum_{i=1}^{\infty}(1 - 4k^2v^2) \cdot e^{-2(kv)^2}$$

from which the critical values can be computed by numerical methods.

Declaration of Competing Interest

All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.

This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.

The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript

The following authors have affiliations with organizations with direct or indirect financial interest in the subject matter discussed in the manuscript:

Federico Reghenzani received his Master degree from Politecnico di Milano in September 2016. He is currently a PhD student working on HPC and embedded computing. His research interests are on resource management on heterogeneous platforms, embedded Linux systems and real-time analyses, in particular on probabilistic WCET estimation and mixed-criticality systems.