A Reliable Statistical Analysis of the Best-Fit Distribution for High Execution Times

Xavier Civit[†,‡], Joan del Castillo[‡], Jaume Abella[†]
[†]Barcelona Supercomputing Center (BSC)
[‡]Universitat Autonoma de Barcelona (UAB)

***Abstract*—Extreme Value Theory has been used to model the WCET probabilistically, relying on the assumption that probabilistic WCET (pWCET) estimates can be upper-bounded with exponential distributions, but this is only assessed on execution time samples with pass/fail hypothesis tests. However, the degree of fulfilment of this hypothesis for the execution time sample has a direct impact on the tightness of the pWCET estimate.**

This paper tackles this limitation of pass/fail tests by applying 3 alternative methods to model the distribution of high execution times through the analysis of the number of finite moments of execution time samples. These methods provide information on the degree of fulfilment of the exponentiality hypothesis, rather than a simple pass/fail response. Hence, whenever the number of finite moments is shown to be low, despite pass/fail tests are passed, these methods indicate that pWCET estimates may be untight. We show that those methods complement each other and the information obtained – number of finite moments proven to exist – can be used to increase the execution time sample size opportunistically to obtain tighter pWCET estimates.

***Keywords*-WCET, MBPTA, Execution time distribution, Statistical analysis**

I. Introduction

The estimation of the Worst-Case Execution Time (WCET) for mixed-criticality tasks in critical real-time embedded systems (CRTES) is an increasingly complex problem in multiple domains, spanning from automotive, to avionics, railway or space among others. Such complexity emanates from the automation of complex functions such as, for instance, autonomous driving in automotive and unmanned vehicles in space/avionics. The level of performance needed to execute these mixed-critical functions timely (100x increase to enable autonomous driving [4]) can only be delivered by higher performance hardware than that used in current systems. However, complex software running on complex hardware challenges WCET estimation [2].

Measurement-Based Probabilistic Timing Analysis (MBPTA) [11], [17], [7], [3] responds to this challenge by enabling measurement-based timing analysis, widely adopted by industry [24], and enabling probabilistic and statistical means to characterize (high) execution time distributions. In particular, a branch of research builds upon already-commercial time-randomized processor designs [30], [18] that relieve the user from having to control many low-level details of the hardware/software behavior, thus facilitating timing analysis regardless of the criticality level of the task under analysis.

Appropriate measurement collection protocols on time-randomized processors deliver independent and identically distributed (i.i.d.) execution times, thus enabling a body of statistical techniques to be used atop. Recently, MBPTA-CV [3], a variant of MBPTA, has been proposed to deliver probabilistic WCET (pWCET) estimates building on known properties of the execution time distributions modelled. In particular, real-time programs are characterized by having an – often unknown – maximum execution time. Hence, their high execution times can be reliably upper-bounded with exponential tail distributions, which are the limit tail distributions for finite distributions. However, execution time samples may exhibit statistical characteristics different to those of the actual execution time distribution from where they are sampled. So far, MBPTA methods (and MBPTA-CV in particular) only test whether those samples pass or fail exponentiality tests, without further analyzing to what degree the sample is compatible with the exponential assumption. As we illustrate later in this paper, in some cases where exponentiality tests are passed, pWCET estimates obtained with exponential tails may be unnecessarily pessimistic. However, in those cases, our analysis reveals that execution time samples have particular characteristics: they have a relatively low number of finite moments.

In this paper we introduce the use of the *number of finite moments* (*nfm*) of statistical samples to characterize high execution times of a distribution through the analysis of a sample of it. We show how *nfm* can be estimated and how they can be used to determine whether pWCET estimates may be overly pessimistic, thus indicating that a larger sample will reduce the pessimism. In particular, we build on the fact that, the higher the *nfm*, the higher the degree of exponentiality of the sample tail. Hence, we use three different models to estimate the *nfm*: (1) the coefficient of variation (CV) estimator [10], (2) the *Group Estimator* [8] and the *Ratio Max Sum* [25]. We show that no method delivers the highest *nfm* in all cases. We note that the *nfm* estimates provided by those methods do not imply that a larger *nfm* does not exist, but only that the particular methods are unable to prove the existence of further finite moments. Hence, we propose applying the three of them and using the largest *nfm* found to assess whether the pWCET estimates can be regarded as sufficiently tight or else, a increasing the sample size is convenient. In detail, our contributions are as follows:

- We introduce the use of three methods to estimate the *nfm* of a statistical sample of execution times in the context of real-time programs, thus with finite execution times.
- We compare them quantitatively on several benchmark suites and a railway case study on top of a time-randomized platform, showing that no method is superior to the others in all cases and hence, the best one for each case must be used.
- Whenever those methods can only prove the existence of a relatively low *nfm*, and given that the studied distribution has $nfm = \infty$ by construction, we apply data transformations to prove the existence of further *nfm*, and show that increasing the sample size can lead to tighter pWCET estimates if the *nfm* is still low.

Our results show that building on the *nfm* we can identify effectively those cases where pWCET estimates are untight. Thus, by increasing the number of measurements opportunistically in those cases, tighter pWCET estimates are obtained.

Although we apply our approach on time-randomized platforms, the method is compatible with any platform and measurement collection protocol as long as the distribution modelled is i.i.d. and has compact support (i.e. has a maximum value), leaving on the hands of the end user building an argument on the representativeness of the measurements collected w.r.t. the behavior of the system during operation.

II. Background

MBPTA. MBPTA is intended to deliver a distribution upper-bounding high execution times rather than a single WCET. MBPTA builds on the concept that the likelihood of the WCET can be ridiculously low (or even zero) and hence, it may be simply dismissed. Thus, MBPTA aims at delivering WCET estimates whose residual risk of exceedance can be set to arbitrarily low levels. For instance, we could set it to be up to 10^{-8} deadline misses per hour. In practice, this is an upper-bound probability and, in the context of MBPTA, this threshold indicates that the true deadline miss rate is below 10^{-8} per hour, being potentially zero, but no evidence is had below such threshold.

MBPTA builds upon statistical methods to deliver a pWCET distribution such that each execution time value has an associated exceedance probability. Therefore, we can use as WCET estimate the execution time value whose exceedance probability is below the acceptable failure rate according to the criticality of the task under analysis and the integrity requirements imposed by the corresponding safety standard.

MBPTA requires the platform used to have specific properties that relate to delivering i.i.d. execution times for programs. This has been proven doable in real multicores and more complex manycores by controlling conveniently the sources of jitter [30], [18]. Evaluation on avionics [31] and space [12] case studies, among others, have proven the effectiveness and reliability of this approach.

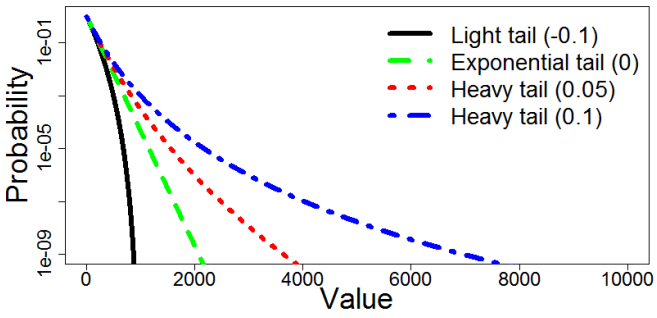

Fig. 1. Complementary cumulative distribution function for light, exponential and two heavy tail GPD distributions with $\xi = -0.1$, $\xi = 0$, $\xi = 0.05$ and $\xi = 0.1$ respectively, and ($\mu = 0$, $\sigma = 100$) for all of them.

Most MBPTA methods devised so far build upon Extreme Value Theory (EVT) [6] to model the pWCET distribution. While the Central Limit Theory is intended to model the central behavior of a distribution, EVT is intended to model its tails. In the case of WCET estimation, EVT is used to model the right tail of the execution time distribution (so high execution times). Based on an execution time sample with R measurements, one could predict, without using EVT, the execution times occurring with probabilities down to $1/R$ with decreasing confidence as the probability decreases due to the use of fewer observations to raise such a prediction. By using EVT, the tail of the distribution is modelled with a suitable distribution that allows predicting (high) execution times for arbitrarily low exceedance probabilities.

EVT. There are two main families of EVT distributions: Generalized Extreme Value (GEV) distributions and Generalized Pareto Distributions (GPD). They differ on how they operate on the input sample to deliver a tail distribution, but their main aspects relevant to our work are mostly common. Therefore, for the sake of economy we only focus on one of them, GPD. Both GEV and GPD are characterized by three parameters: shape (ξ), scale (σ) and location (μ). The shape describes the type of tail: light tail, exponential tail or heavy tail for GPD, and Reverse Weibull, Gumbel or Fréchet for GEV respectively. σ relates to the slope of the tail whereas μ indicates the value at which the exceedance probability starts dropping. Since GPD and GEV use different methods to select maxima, both σ and μ are similar but not identical across GPD and GEV. Instead, ξ is identical across both distribution families.

Example GPD distributions are depicted in Figure 1 for a light tail ($\xi = -0.1$), an exponential tail ($\xi = 0$) and two heavy tails ($\xi = 0.05$ and $\xi = 0.1$ respectively). As shown, the shape parameter is critically important for the rate at which tails fall. ① Light tails fall sharply approaching a maximum value asymptotically. They are regarded as appropriate when a maximum value exists and, due to fitting constraints, it is known to be close to those in the input sample, which cannot be proven true in general. Hence, fitting light tails without knowing whether values close to the

absolute maximum belong to the sample is intrinsically risky. ② Exponential tails ($\xi = 0$) fall at an exponential rate, thus lacking compact support, but their exceedance probability quickly decreases for increasingly higher values. They are regarded as convenient when a maximum value exists but it is unknown. In other words, exponential tails are the limit distribution for light tails removing constraints related to having values close to the absolute maximum. ③ Finally, heavy tails fall at a polynomial rate and are convenient for unbounded distributions that are clearly non-normal.

The parameters of a random variable (execution times in our case) are typically referred to as *moments*, being the first moment the mean, the second central moment the variance, etc. Exponential and light tail distributions have infinite moments since light tails have compact support, and so infinite moments [29], and exponential tails are the limit distribution for light tails. In general, $\xi = 1/nfm$ if $nfm \neq \infty$, and $\xi \leq 0$ otherwise. Hence, *we are interested in proving that the* nfm *of a given execution sample is as high as possible*. If *nfm* is low (so $\xi \gg 0$) then some tail values are far away from the other values retained for tail modelling, which makes that the best fitting exponential distribution has a overly high σ value (so the slope is too gentle) to minimize the distance w.r.t. all (distant) values in the fitting process. In statistics it is well-known that using larger samples may include further information, thus enabling better predictions. If $\xi \gg 0$, then we know that a larger sample is very likely to deliver tighter estimates. This occurs because more values close to those far away will be collected and used for a tighter estimation. Hence, fitting will eventually occur only with the group of highest values, so σ will be lower and the slope of the pWCET distribution steeper. If distant values did not exist, increasing the sample size will not alter σ meaningfully. We refer the interested reader to [3] for details on the causes of having few distant values with a limited sample on time-randomized architectures.

In the remaining of this work we bring methods used for *nfm* estimation in other domains to the case of pWCET estimation. We show that their appropriate use can allow proving that high *nfm* values exist, thus providing evidence on the tightness of pWCET estimates. Whenever the *nfm* proven is rather low, this indicates that the exponential tail fit may be unnecessarily pessimistic.

III. *NFM* Estimation for pWCET Modelling

This section presents three methods to estimate the *nfm* of a distribution through the analysis of a sample of it: the Group Estimator (GE), the ratio max sum (RMS) and the coefficient of variation (CV). Next we describe the methods and how they have been applied for the particular problem at hand.

A. Group Estimator (GE)

The GE was introduced by Davydov et al. [8]. GE builds on the assumption that the distribution analyzed is heavy-tailed. Hence, if it is exponential or light-tailed, GE will return a very high *nfm*, but never ∞. To introduce GE, let us consider a sample $X^n = \{X_1, X_2, \ldots, X_n\}$ of size n taken from a heavy-tailed distribution $F(x)$. We assume that $X_1, X_2, \ldots X_n$ correspond to i.i.d. random variables. As shown later in the evaluation section, this is achievable for some appropriate platforms [18].

In order to compute the estimator, the sample is divided into l groups $V_1, V_2, \ldots, V_l$, each group containing m random variables, that is, $n = l \cdot m$. The practical approach when n cannot be divided by l is simply discarding the remaining values of the sample. Let $M_{li}^{(1)} = \max\{X_j | X_j \in V_i\}$ (the largest element in the group) and let $M_{li}^{(2)}$ denote the second largest element in the same group V_i. Let us denote

$$k_{li} = \frac{M_{li}^{(2)}}{M_{li}^{(1)}} \qquad \text{and} \qquad z_l = \frac{1}{l}\sum_{i=1}^{l} k_{li}$$

so that k_{li} corresponds to the ratio between the second largest and the largest element in group V_i out of l groups, and z_l is the average across all k_{li} values.

Based on some assumptions related to the characteristics of the distribution being modelled and assuming $l = m = [\sqrt{n}]$, it has been proven that

$$z_l \xrightarrow{a.s.} \frac{\alpha}{\alpha + 1}$$

where $nfm = \alpha$. Note that $\xrightarrow{a.s.}$ stands for *almost surely*[1]. However, those assumptions, which relate to α being neither too close to 0 (so $z_l \approx 0$) nor to ∞ (so $z_l \approx 1$), do not necessarily hold for execution time distributions, which can be arbitrary. We refer the interested reader to [8] for further details on those assumptions. Hence, an alternative method is needed to determine the most convenient value of m (and so l) to estimate z_l, and ultimately *nfm* (α).

1) A Bootstrap Method to Select m: When choosing m and l there is a trade-off between bias and variance. If m is too small, so data is split into many small groups, the asymptotic basis of the model may be violated leading to potentially significant bias. Conversely, if m is too high, then very few groups are obtained, thus leading to high variance. In order to avoid these problems, we build upon the bootstrap method proposed by Markovich [22], whose goal is minimizing both, variance and bias.

The bootstrap estimate is obtained by drawing B samples with replacement from the original data set X^n. Hence, some observations from X^n may appear more than once whereas others may not appear. One can use smaller resamples $\{X_1^*, X_2^*, \ldots, X_{n_1}^*\}$ of size $n_1 < n$ from X^n to avoid the situation where the bootstrap estimate of the bias is equal to 0 regardless of the true non-zero bias of the estimator [16]. The values n_1 and n may be related by:

$$n_1 = n^d, \qquad 0 < d < 1 \tag{1}$$

[1]Almost surely, in maths terminology, means that its probability is 1, despite there could be some exceptions whose accumulated probability is zero.

The re-sample is divided into l_1 subgroups so that $l_1 = [n_1/m_1]$ holds. The size of subgroups m_1 and m are related by:

$$m = m_1(n/n_1)^c, \qquad 0 < c < 1 \tag{2}$$

Since the distribution function $F(x)$ is unknown, one can find m_1 by minimizing the empirical bootstrap estimate of the min square error (MSE), exploring all possible integer values of m_1 in the range $[2, n_1]$, as shown below:

$$MSE^*(l_1, m_1) = \left(\hat{b}^*(l_1, m_1)\right)^2 + \widehat{Var}^*(l_1, m_1)$$

with

$$\hat{b}^*(l_1, m_1) = \frac{1}{B}\sum_{b=1}^{B} z_{l_1}^b - z_l$$

and

$$\widehat{Var}^*(l_1, m_1) = \frac{1}{B-1}\sum_{b=1}^{B}\left(z_{l_1}^b - \frac{1}{B}\sum_{b=1}^{B} z_{l_1}^b\right)^2$$

that are the empirical bootstrap estimates of the bias and the variance, respectively. Then, one can determine m based on the value of m_1 obtained minimizing the MSE as described in Equation 2. In Equations 1 and 2, c and d must be chosen appropriately. Based on asymptotic theory, Hall [16] concludes that $d = 1/2$ and $c = 2/3$ lead to the most accurate results for this bootstrap method.

B. Ratio Max Sum (RMS)

The ratio between the maximum and the sum (RMS) of a sample has a number of properties that allow concluding whether a sample has a finite expected value. The RMS belongs to classic Probabilistics theory and has been considered for heavy tail analysis by Novak [25]. In particular, we are interested in studying the RMS for positive random variables and increasing values of *nfm* (t in the canonical description of RMS below). The RMS is formulated as follows:

$$R_n(t) = \frac{\left(M_n^{(1)}\right)^t}{S_n^{(t)}} \tag{3}$$

where $S_n^{(t)}$ denotes $\sum_{i=1}^{n} |X_i|^t$, and $M_n^{(1)}$ stands for the largest value in the sample. Note that for each $t \in \mathbb{N}$, the statistic $R_n(t)$ provides information of the moment t of the modelled random variable.

The properties of interest of $R_n(t)$ that allow determining whether the particular moment t exists, are as follows: (1) It is well known that the way maxima grows in increasingly larger samples is tightly related to the *nfm*. Hence, (2) $M_n^{(1)}/n^{1/t}$ tends to 0 if and only if the t^{th} moment is finite, so if and only if $\mathbb{E}|X|^t < \infty$. Hence, $R_n(t) = \left(M_n^{(1)}\right)^t / S_n^{(t)}$ is asymptotically small if $\mathbb{E}|X|^t < \infty$. Conversely, (3) in the case of heavy tails $\left(M_n^{(1)}\right)^t$ is comparable to $S_n^{(t)}$, and hence $R_n(t)$ tends to 1 for increasing values of t. This is formally described with the following theorem:

Theorem 1. *As* $n \to \infty$

- $R_n(t) \xrightarrow{a.s.} 0 \Leftrightarrow \mathbb{E}|X|^t < \infty$
- $R_n(t) \xrightarrow{p} 0 \Leftrightarrow \mathbb{E}|X|^t \not\Vdash (|X| \leq x)$ *is slowly varying*
- $R_n(t) \xrightarrow{p} 1 \Leftrightarrow \mathbb{P}(|X| > x)$ *is slowly varying*

Given that $R_n(t)$ is asymptotically small if $\mathbb{E}(X) < \infty$, we can fix a confidence threshold c (i.e. $c = 0.05$) and estimate the $nfm = t$ where $t \in \mathbb{N}$ and $R_n(t) < c$. Note that $R_n(t)$ builds upon exponentiation (see Equation 3), which may bring numerical problems when the exponent (t) is high. Therefore, RMS is used to obtain *nfm* up to a given maximum threshold. In this paper we limit the estimation of *nfm* to 50 for RMS. In any case, whenever *nfm* are above 50, differences in terms of ξ become statistically irrelevant in practice, so obtaining higher *nfm* values does not provide any benefit in practice.

C. Coefficient of Variation (CV)

The coefficient of variation (CV) has been used to study the tail of distribution functions [9]. The CV is obtained as the ratio between the standard deviation and the mean of a distribution. Therefore, the CV is independent of the location and scale of the distribution. Formally stated, the CV of a random variable X is obtained as follows:

$$CV(X) = \frac{\sqrt{V(X)}}{M(X)} \tag{4}$$

Using the CV, we can obtain ξ, and hence *nfm* based on the following equation:

$$CV(X) = \frac{1}{\sqrt{1 - 2 \cdot \xi}} \tag{5}$$

Note that this equation holds as long as $\xi < 0.5$, thus meaning that the random variable has at least 2 moments (the mean and the variance).

In practice, however, the mean and the variance of a distribution are unknown and we can only use the mean ($\overline{x}$) and variance (sd^2) of a particular sample, where sd stands for the standard deviation. Therefore, the estimator of the CV, cv, is obtained as follows:

$$cv(x) = \frac{sd(x)}{\overline{x}} \tag{6}$$

and so we can obtain ξ, and hence the *nfm* building on Equation 6, which is the empirical version of Equation 4. In particular, we obtain cv with Equation 6, replace CV by cv in Equation 5, and then we obtain ξ. Finally, as explained before, $nfm = 1/\xi$ if $\xi > 0$, and $nfm = \infty$ otherwise.

D. Joint Considerations

The three methods to estimate the *nfm* build upon the assumption that input samples correspond to i.i.d. random variables. Therefore, this hypothesis needs hold before the application of those methods. In this work we build upon a hardware platform that provides those properties by construction. In particular we use a hardware platform whose execution time variations for a given program are produced

by time-randomized hardware components such as random placement and replacement cache memories, as well as randomly arbitrated shared resources [18]. The i.i.d. hypothesis is empirically corroborated for all experiments as explained in the evaluation section.

While the three methods, namely GE, RMS and CV, allow estimating the *nfm*, GE and RMS can only deliver *nfm* positive values arbitrarily high, but not infinite[2]. Thus, GE and RMS are appropriate for heavy tails only. In the case of exponential or light tails, both GE and RMS are expected to deliver high (but not infinite) *nfm* values. Conversely, as explained before, CV estimates ξ directly, thus being appropriate for any type of distribution.

Finally, those methods indicate the existence of a given *nfm*. However, higher moments may also exist. Hence, if results across methods differ, the maximum *nfm* across the different methods may provide the best approach.

IV. Evaluation

In this section we evaluate the three estimators for the *nfm*. First, we describe the evaluation framework. Second, we compare the three estimators in terms of *nfm*. Finally, we quantify the benefits that their joint utilization can provide to identify cases where the *nfm* is low and hence, there is margin for obtaining tighter pWCET estimates.

A. Experimental Framework

We collect execution time measurements in a cycle-accurate simulator based on SoCLib [28] modelling the probabilistically-analyzable LEON3 multicore in [30], which implements random placement and replacement cache policies, random arbitration in shared resources and time upper-bounding for variable-latency arithmetic operations [18].

We evaluate several sets of relevant software. First, we conduct our study on two sets of benchmarks: (1) the EEMBC AutoBench benchmark suite [26], which is representative of a number of critical real-time functions used in automotive embedded systems. (2) A subset of the Mälardarlen benchmark suite [15], which includes benchmarks intended for the evaluation of WCET analysis tools and methods. Then, we assess our methodology on a railway case study implementing a safety-related real-time function from the European Train Control System (ETCS) reference architecture. In particular, this application is in charge of the safety functions related to travelling distance and speed control, and is graded as Safety Integrity Level (SIL) 4 according to appropriate safety standards [1]. We evaluate 10 different input vectors leading to 10 different execution paths, as provided by the end user (`TEST0` to `TEST9`). Each benchmark and case study test case is run till completion 1,000 times, thus collecting 1,000 execution time measurements per benchmark/test case.

[2]Infinite values could only occur numerically, in the case of GE, for degenerate distributions, which are generally disregarded. Thus, neither GE nor RMS can provide an infinite *nfm*.

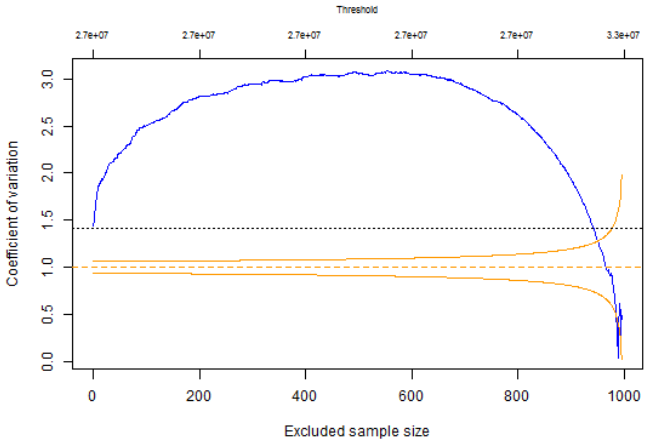

Fig. 2. CV-plot for `matrix`.

I.i.d. properties hold by construction due to the way measurements are collected in our platform, since each measurement for any given program is obtained identically on the same processor with the same inputs and initial state, changing only the (random) seeds that determine placement and replacement decisions in the caches. However, the use of EVT imposes the assessment of statistical independence for the data sample. Hence, we use the powerful Ljung-Box independence test [5] to test autocorrelation for 20 different lags simultaneously. All samples have passed the test for a $\alpha = 0.05$ significance level. Note that, statistically, we could expect some (few) test fails (5% with $\alpha = 0.05$). Since we used 34 different programs and benchmarks, the probability of all of them passing the test is $\approx 17.5\%$. If the test was failed for any of the samples, the default solution would have been increasing the sample size until the test was passed, which would eventually occur since the random variables sampled are independent by construction.

B. Results for nfm

Next, we evaluate the *nfm* on the different samples for the different methods. Table I shows the *nfm* for the three methods as well as the maximum *nfm* value obtained across them, which is the best value that can be obtained using the three methods simultaneously. In the case of RMS, as explained before, we obtain up to 50 *nfm* due to numerical limitations of the exponentiation operation in Equation 3. Hence, whenever $nfm = 50$ for RMS, it means that at least 50 moments exist. $nfm > 50$ provides limited benefits w.r.t. $nfm = 50$ since $\xi = 0$ ($nfm = \infty$) and $\xi = 0.02$ ($nfm = 50$) are too close to tell apart distributions whose samples provide so similar shape values. Hence, obtaining up to $nfm = 50$ for RMS is regarded as an acceptable constraint.

EEMBC. In the case of CV, the method either does not converge or obtains very few *nfm* for 4 of the 16 cases (shaded gray cells in the table). In particular, the method fails to converge for `aifirf`, `canrdr`, `matrix` and `pntrch`. For those, we have obtained the CV-plot [10], which draws

TABLE I
nfm FOR THE EEMBC AND MÄLARDARLEN BENCHMARKS WHEN USING GEB, RMS AND CV, MAXIMUM *nfm* ACROSS THE THREE METHODS, AND PWCET ESTIMATE.

	nfm				**pWCET**
EEMBC bench	GEB	RMS	CV	**Max.**	10^{-9}
a2time	47	33	∞	∞	2.11
aifftr	28	16	21	28	1.66
aifirf	16	12	0	16	1.29
aiifft	28	21	44	44	1.70
basefp	227	50	∞	∞	1.45
bitmnp	54	50	∞	∞	1.69
cacheb	5	5	∞	∞	1.29
canrdr	54	28	∞	∞	1.45
idctrn	114	50	∞	∞	1.28
iirflt	116	50	∞	∞	1.59
matrix	20	21	∞	∞	1.10
pntrch	41	23	∞	∞	1.40
puwmod	163	50	∞	∞	1.49
rspeed	46	40	19	46	1.76
tblock	57	50	∞	∞	2.06
ttsprk	53	42	26	53	1.79
Mälardarlen bench	GEB	RMS	CV	**Max.**	10^{-9}
adpcm	239	50	∞	∞	1.27
bs	89	50	0	89	1.01
cnt	181	50	8	181	1.17
crc	284	50	31	284	1.15
fir	123	50	8	123	1.14
lcdnum	51	31	0	51	1.04
ns	353	50	156	353	1.05
prime	223	50	∞	∞	1.05

the residual coefficient of variation (rcv) as observations are excluded from the sample, where $rcv = 1$ for exponential tails, $rcv < 1$ for light tails and $rcv > 1$ for heavy tails. Such estimator is regarded as inaccurate whenever $rcv > 1.41$. For instance, Figure 2 shows the CV-plot for `matrix`. As shown, the rcv (blue line in the plot) is sustainedly above 1.41, the value above which the CV estimator is regarded as unreliable.

In those cases, we have directly tested whether the exponentiality assumption (and so $nfm = \infty$) is acceptable with alternatives methods. In order to obtain the *nfm* for all those cases where the CV method does not converge, data should be transformed as suggested in [9][3]. Such data transformation makes heavy tails be non-heavy for their analysis. Then, the sign of the ξ' value obtained with transformed data needs to be changed to obtain a reliable estimate of ξ for the original data so that $\xi = -\xi'$. Based on the ξ obtained following this approach, we obtain *nfm* for those 4 benchmarks. Those are the values reported in Table I for these benchmarks. As shown, the data transformation allows to prove the existence of infinite *nfm* in three cases. In the remaining one, `aifirf`, even the data transformation fails to converge, so CV fails to support the existence of *nfm*.

As shown, the best method changes across applications. CV shows to be the best choice for 12 out of 16 benchmarks since it is the only one able to prove that *nfm* is infinite (11 out of those 12 benchmarks). However, for 4 benchmarks it delivers lower *nfm* values (or no *nfm* at all as for `aifirf`) than, at least, one of the other methods. In those cases, GEB delivers the highest *nfm* results. RMS, instead, is not the best method for any individual application, but for some applications it delivers better results than one of the other methods. Hence, RMS may potentially outperform both, GEB and CV, for other applications.

[3]The new variable, Y is obtained from the original X, as follows: $Y = -1/(X+c)+1/c$ where $c = \psi/\xi$. Note that ψ and ξ are the two parameters describing a GPD when the two-parameter formulation is used [6].

Mälardarlen. As for EEMBC, CV does not manage to prove the existence of any *nfm* in some cases (`bs` and `lcdnum`), and neither does the aforementioned data transformation. When considering the 3 methods together, we realize that, differently to the case of EEMBC, GEB provides the best results for Mälardarlen since it deliver the highest *nfm* in 6 out of 8 cases. CV is the best technique in the remaining 2 cases. As before, RMS does not provide the highest *nfm* count in any of the benchmarks considered, but this may be related to the numerical limitations that prevent from obtaining $nfm > 50$ for RMS.

C. pWCET Estimation

In Table I (rightmost column) we show the results in the form of ratio of the pWCET value obtained for the exponential fit with an exceedance probability of 10^{-9} per run w.r.t. the maximum observed execution time (MOET) in the sample.

EEMBC. As shown, pWCET estimates are, on average, 1.59x higher than their MOET. In some cases, the different methods for estimating *nfm* have difficulties to prove the existence of many *nfm*, as it is the case for `aifirf` and `aifftr`, for which we can only prove the existence of 16 and 28 *nfm* respectively. In these cases, we expect that the pWCET estimates obtained may be potentially untight. Based on the reasoning provided before, by increasing the size of the sample we should obtain tighter pWCET estimates.

We have collected 10,000 execution time measurements for these two benchmarks, `aifirf` and `aifftr`, and, as expected, pWCET estimates decreased noticeably. In particular, the pWCET for `aifirf` decreased from 1.29x to 1.13x (w.r.t. the MOET in 1,000 runs), thus being much tighter. Similarly, the pWCET for `aifftr` decreased from 1.66x to 1.51x, showing also a noticeable improvement.

When analyzing the *nfm*, we realize that in the case of `aifftr` the value increased from 28 to 40. In this case, the CV method is the one obtaining the 40 *nfm*. Note that this value is already high since $\xi = 1/nfm = 0.025$. In the case of `aifirf`, despite the improved tightness of the pWCET estimate, the *nfm* only grew from 16 to 17, being still GEB the method providing the highest *nfm*. This relates to the fact that the number of high measurements increased, thus leading to a lower σ and hence, a steeper pWCET distribution. However, some of the highest values still stay far away from the other measurements, as illustrated in Figure 3 with the tail distribution (aka empirical complementary cumulative distribution function or simply ECCDF) of the 10,000 measurements sample, which does not allow to prove the existence of a higher *nfm*. As explained before,

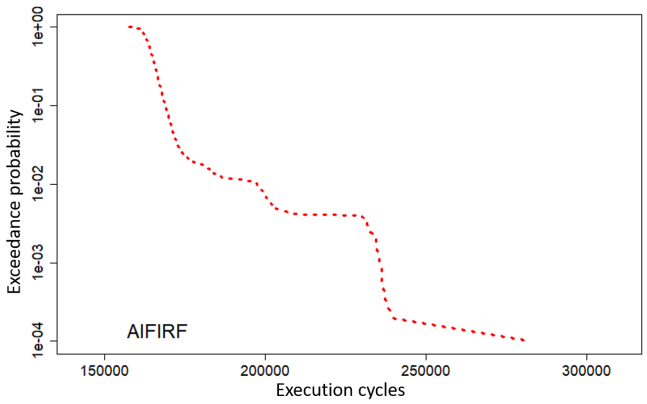

Fig. 3. ECCDF for `aifirf` with 10,000 runs.

the different methods provide evidence of the existence of a *nfm* value, but do not tell anything about whether more *nfm* exist. In the case of `aifirf`, despite the *nfm* is still relatively low, the fact that the pWCET estimate is already close to the MOET limits the gains that could be obtained by further increasing the sample size.

Mälardarlen. pWCET estimates for Mälardarlen are much lower than for EEMBC and, on average, they are only 1.11x higher than their respective MOET values, thus showing that they are pretty tight. This is completely consistent with the *nfm* obtained, which is at least 51 for all benchmarks. Therefore, none of these benchmarks requires increasing the sample size, as indicated by the estimation of their *nfm*.

D. Railway Case Study

TABLE II
nfm FOR THE RAILWAY CASE STUDY WHEN USING GEB, RMS AND CV, MAXIMUM *nfm* ACROSS THE THREE METHODS, AND PWCET ESTIMATE.

	nfm				**pWCET**
Test case	GEB	RMS	CV	**Max.**	10^{-9}
TEST0	67	40	∞	∞	1.64
TEST1	107	50	∞	∞	1.68
TEST2	40	44	∞	∞	1.60
TEST3	67	40	∞	∞	1.66
TEST4	73	41	∞	∞	1.64
TEST5	121	44	∞	∞	1.68
TEST6	75	43	∞	∞	1.65
TEST7	198	50	∞	∞	1.66
TEST8	271	43	∞	∞	1.73
TEST9	42	30	∞	∞	1.72

We have conducted the analysis in terms of *nfm* for the 10 test cases of the railway case study. Results are shown in Table II, where we can see that all methods are able to prove the existence of a high number of *nfm*. In particular, CV proves the existence of an infinite number of *nfm* for all test cases. Due to the characteristics of this case study, we know that execution time variability can be pretty small due to the fact that most data fits in first level caches and hence, little variability can occur due to cache conflicts in both, relative and absolute terms. This leads to compact execution time distributions in all samples, where highest execution times observed are very close to the bulk of measurements and none of them departs away from the median. Hence, it is easy to prove the existence of large *nfm* for all methods, as shown in the table.

pWCET estimates are consistently in the range 1.6x to 1.75x w.r.t. the MOET in all test cases, but due to the existence of an infinite number of *nfm* we do not expect further reductions by increasing the sample size. In order to assess this point, we have repeated the analysis on enlarged samples with 5,000 measurements instead of 1,000 and results showed pWCET variations within 5% in all cases, thus corroborating our expectations.

We have also analyzed the source for the relatively high pWCET estimates w.r.t. the MOET. We realized that, the execution time variability of these test cases is low and, in fact, actual measurements are very close to the absolute WCET by construction. This makes that a tight pWCET fit could be obtained with a light tail (see black straight line in Figure 1) instead of with an exponential tail (see green dashed line in Figure 1). However, as discussed in Section II, light tails (GPD) and Reverse Weibull distributions (GEV) are intrinsically risky if no guarantees are had on whether the absolute maximum value is close enough to the data in the sample, and such guarantees cannot be had in the general case. Thus, we resorted to exponential tails, which provide relatively high pWCET estimates in this case, but that cannot be made tighter by increasing the sample size.

V. Related work

The use of statistical methods to model WCET estimates has received increasing attention during last years. As discussed in Section II, some efforts have been made to model pWCET distributions with either exponential tails only [11], [17], [7], [3] or with more general EVT distributions [20], [19].

The application of EVT requires input data to be i.i.d. [13], [17], [14]. However, the use of some platforms or some data collection methods, as the ones in [27], may break the independence requirement. Melani et al. [23] investigate the factors that may cause dependencies, including scheduling policies and processor state, and conclude that they can be conveniently leveraged in pWCET estimates. In that line, Santinelli et al. [27] investigate stationary processes and also analyze how to account for dependencies in pWCET estimation. Yue et al. [21] propose alternative methods to collect measurements, based on retaining only maxima, to remove dependencies. Also, Lima and Bate [19] propose methods to deal with discrete data that, to some extent, may mitigate dependencies.

The platform to use has also been subject of discussion. Some authors build upon Commercial Off-The-Shelf (COTS) platforms [21], [27] whose timing behavior may fail to provide i.i.d. properties and where representativeness between analysis conditions and operation conditions is hard to sustain. Instead, other platforms, such as those building

upon time randomization and time upper-bounding [18], have been shown to address these concerns by construction. Thus, these are the platforms considered in this work which, by construction, avoid dependencies across measurements.

Different methods exist that use EVT to model the pWCET of programs. Some of them target programs whose execution time may not be upper-bounded [20], [19]. Therefore, any tail model is possible, including very heavy tails. In this paper we target real-time programs, which have a finite execution time by construction. Therefore, their execution time distributions can be reliably modelled with exponential tails. This is the approach followed already in some works [11], [17], [7], [3].

VI. Conclusions

The use of EVT for WCET estimation has become popular. A family of MBPTA techniques building upon EVT has proven that exponential tails are appropriate to model high execution times, and platforms providing the required properties have reached commercial products and industrial case studies. However, while the required properties hold for the modelled distributions, the actual properties of the execution time samples used to model high execution times have only been assessed against pass/fail tests, which provide limited information.

In this paper we analyze and apply different methods to quantify the number of finite moments to statistical samples. This allows obtaining evidence on whether samples are sufficiently close to the exponential distribution to regard pWCET estimates as tight enough. We have applied the different methods, with appropriate data transformations whenever needed, which allowed to prove the existence of an infinite number of finite moments for almost all benchmarks and test cases for a railway case study, thus supporting the tightness of their pWCET estimates. In few cases our these methods indicate that pWCET estimates can be untight. We, therefore, increased the sample size in those cases and, as expected, pWCET estimates became tighter.

Overall, the approach presented in this paper allows assessing the tightness of pWCET estimates with multiple methods, which allows devoting effort to collect larger samples only in those cases where gains are foreseen based on the number of finite moments.

Acknowledgements

This work was partially funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 772773), the Spanish Ministry of Economy and Competitiveness (MINECO) under grant TIN2015-65316-P and the HiPEAC Network of Excellence. Jaume Abella has been partially funded by MINECO under Ramon y Cajal postdoctoral fellowship number RYC-2013-14717.

References

[1] IEC 61508: Functional safety of electrical /electronic /programmable electronic safety-related systems, 2010.

[2] J. Abella et al. WCET analysis methods: Pitfalls and challenges on their trustworthiness. In *SIES*, 2015.

[3] J. Abella et al. Measurement-based worst-case execution time estimation using the coefficient of variation. *ACM Trans. Des. Autom. Electron. Syst.*, 22(4), June 2017.

[4] ARM. ARM Expects Vehicle Compute Performance to Increase 100x in Next Decade. Technical report, https://www.arm.com/about/newsroom/arm-expects-vehicle-compute-performance-to-increase-100x-in-next-decade.php, 2015.

[5] G.E.P. Box and D.A. Pierce. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American Statistical Association*, 1970.

[6] S. Coles. *An Introduction to Statistical Modeling of Extreme Values*. Springer, 2001.

[7] L. Cucu-Grosjean et al. Measurement-based probabilistic timing analysis for multi-path programs. In *ECRTS*, 2012.

[8] Y. Davydov et al. More on p-stable convex sets in banach spaces. *Journal of Theoretical Probability*, 13(1):39–64, 2000.

[9] J. del Castillo and M. Padilla. Modelling extreme values by the residual coefficient of variation. *Statistics and Operations Research Transactions*, 40(2):303–320, 2016.

[10] J. del Castillo et al. Methods to distinguish between polynomial and exponential tails. *IEEE Scandinavian Journal of Statistics*, 41(2):382–393, 2014.

[11] S. Edgar and A. Burns. Statistical analysis of WCET for scheduling. In *RTSS*, 2001.

[12] M. Fernandez et al. Probabilistic timing analysis on time-randomized platforms for the space domain. In *DATE*, 2017.

[13] R.A. Fisher and L.H.C. Tippett. 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.

[14] D. Griffin and A. Burns. Realism in Statistical Analysis of Worst Case Execution Times. In *WCET Workshop*, 2010.

[15] Jan Gustafsson, Adam Betts, Andreas Ermedahl, and Björn Lisper. The Mälardalen WCET benchmarks – past, present and future. In Björn Lisper, editor, *the International Workshop on Worst-case Execution-time Analysis*, pages 137–147, Brussels, Belgium, July 2010. OCG.

[16] P. Hall. Using the bootstrap to estimate mean squared error and select smoothing parameter in nonparametric problems. *Journal of Multivariatte Analysis*, (32):177–203, 1990.

[17] J.P. Hansen et al. Statistical-based WCET estimation and validation. In *WCET Workshop*, 2009.

[18] L. Kosmidis et al. Fitting processor architectures for measurement-based probabilistic timing analysis. *Microprocessors and Microsystems*, 47:287 – 302, 2016.

[19] G. Lima and I. Bate. Valid application of EVT in timing analysis by randomising execution time measurements. In *RTAS*, 2017.

[20] G. Lima et al. Extreme value theory for estimating task execution time bounds: A careful look. In *ECRTS*, 2016.

[21] Y. Lu et al. A new way about using statistical analysis of worst-case execution times. *SIGBED Review*, 8(3), 2011.

[22] N. Markovich. *Nonparametric Analysis of Univariate Heavy-Tailed Data*. 1, 2007.

[23] A. Melani et al. Learning from probabilities: Dependences within real-time systems. In *ETFA*, 2013.

[24] E. Mezzetti and T. Vardanega. On the industrial fitness of WCET analysis. *WCET Workshop*, 2011.

[25] S. Novak. *Extreme Value Methods with Applications to Finance*. Chapman and Hall, 2012.

[26] J. Poovey. *Characterization of the EEMBC Benchmark Suite*. North Carolina State University, 2007.

[27] L. Santinelli et al. On the Sustainability of the Extreme Value Theory for WCET Estimation. In *WCET Workshop*, 2014.

[28] SoCLib. -, 2003-2012. http://www.soclib.fr/trac/dev.

[29] L. Trapani. Testing for (in)finite moments. *Journal of Econometrics*, 191(1):57 – 68, 2016.

[30] http://www.gaisler.com/index.php/products/processors/leon3. *Leon3 Processor*. Cobham Gaisler.

[31] F. Wartel et al. Timing analysis of an avionics case study on complex hardware/software platforms. In *DATE*, 2015.