Adaptive Simulation-based Framework for Error Characterization of Inexact Circuits

Justine Bonnot, Vincent Camus, Karol Desnos, Daniel Menard

Adaptive Simulation-based Framework for Error Characterization of Inexact Circuits

Justine Bonnot[†], Vincent Camus[‡], Karol Desnos[†], Daniel Menard[†]
[†]Univ. Rennes, INSA Rennes, CNRS, IETR - UMR 6164, F-35000 Rennes, France
[‡]ICLAB, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

***Abstract*—To design faster and more energy-efficient systems, numerous inexact arithmetic operators have been proposed, generally obtained by modifying the logic structure of conventional circuits. However, as the quality of service of an application has to be ensured, these operators need to be precisely characterized to be usable in commercial or real-life applications. The characterization of the error induced by inexact operators is commonly achieved with exhaustive or stochastic bit-accurate gate-level simulations. However, for high bit-widths, the time and memory required for such simulations become prohibitive. To overcome these limitations, a new characterization framework for inexact operators is proposed. The proposed framework characterizes the error induced by inexact operators in terms of mean error distance, error rate and maximum error distance, allowing to completely define the error probability mass function. By exploiting statistical properties of the approximation error, the number of simulations needed for precise characterization is minimized. From user-defined confidence requirements, the proposed method computes the minimal number of simulations to obtain the desired accuracy on the characterization for the error rate and mean error distance. The maximum error distance value is then extracted from the simulated samples using the extreme value theory. For 32-bit adders, the proposed method reduces the number of simulations needed up to a few tens of thousands points.**

***Index Terms*—Approximate computing, error modelization, statistics, extreme values, inexact circuits, quality of service.**

I. Introduction

Real-time and energy constraints for the current design of embedded systems increase the need for developing new techniques to save resources during the implementation phase. Approximate computing is one of the main approaches for post-Moore's Law computing. It exploits the error resilience of numerous applications in order to save energy or accelerate processing. The numerical accuracy of an application is now taken as a new tunable parameter to design more efficient systems.

Approximations have been introduced at different levels. At data level, approximation can be introduced by reducing the volume of stored or processed data [1] or by optimizing the data representation. For example, one may carefully choose the type of arithmetic used, which can be floating-point, fixed-point, logarithmic etc. or carefully tune the precision in optimizing the bit-widths of the data [2], [3]. At algorithmic level, the processing complexity can be reduced, for instance, by skipping or approximating a fraction of the computations [4]–[6]. Finally, at hardware level, approximations have been exploited in different manners such as overclocking [7] or by modifying the logic structure of original exact operators into an inexact version [8]–[12] with a lower logic complexity or shorter critical paths. Inexact operators generate errors with varied amplitude and error rate. The error amplitude depends on the location of the erroneous bits of the operator output.

Before analyzing the effects of the errors induced by the chosen approximations on the application quality metric, the errors induced by the inexact operator have themselves to be modeled, to avoid exhaustive simulation. A thorough characterization of the approximation error allows to choose the most suitable operator with respect to the implementation constraints and to quantify the impact of the approximation on the application quality metric.

The error induced by inexact operators can be evaluated with two types of approaches: 1) Analytical methods [13]–[16] mathematically express error statistics as the mean error distance or the error rate, but are dedicated to specific logic structures and can become really complex to implement in terms of computation time and memory for high bit-widths operators. 2) Functional simulation techniques [17]–[19] simulate the inexact operator on a representative set of data and computes statistics on the approximation error. To mimic the inexact operator behavior, bit-accurate simulations at the logic-level (BALL simulations) are required to catch the internal structure modifications of the operator. Nevertheless, BALL simulations are two or three orders of magnitude more complex than classical simulations with native data types. Thus, exhaustively testing the operator for all the input value combinations is not feasible for high bit-widths because of the required simulation time.

Commonly, the error statistics are computed by simulating a given number of random inputs [17]–[19]. The quality of the statistical characterization obtained from a random sampling is highly dependent on the number of samples taken and on the chosen input distribution. Besides, classical simulation-based analysis do not provide any confidence information on the obtained statistical estimation. Using a great number of samples can be ineffective in terms of simulation time. Furthermore, to the best of our knowledge, no generic method has been proposed to evaluate the upper bound of the error distance induced by inexact operators, which is a critical characteristic to know when implementing inexact circuits.

In this paper, we propose a characterization method for inexact operators according to three different metrics: the mean error distance, the error rate and the upper bound of

the error distance, called maximum error distance in the rest of the paper. This framework extends the preliminary work proposed in [20]. To estimate the mean error distance and the error rate, the proposed method derives the minimal number of samples to simulate, to get an accuracy on the estimation according to a given user-defined confidence interval. Our approach drastically reduces the number of simulations needed and thus the characterization time, by exploiting the statistical properties of the approximation error. The simulated samples are then analyzed with the extreme value theory to derive the maximum error distance according to user-defined confidence information. Reducing the characterization time allows to characterize high bit-widths operators. The efficiency of our method is evaluated on several inexact adders of different bit-widths, from 8 to 32 bits.

The remainder of this paper is organized as follows: Section II reviews the existing analytical and simulation-based techniques to characterize the approximation error induced by inexact operators. Section III details the metrics used for the characterization of inexact operators and the methods to estimate them according to user-defined confidence parameters. Section IV presents the proposed framework combining the estimation of the mean error distance, error rate and maximum error distance. Section V presents the experimental setup and the obtained results in terms of number of simulated samples and quality of the obtained estimation.

II. Context and Related Works

Inexact arithmetic operators generate varied error profiles. When implementing inexact operators in an application, the objective is to derive the impact of the induced approximations on the application quality metric.

The evaluation of the impact of the inexact operator on the application quality metric is done in two steps as presented in Figure 1. The errors induced by the inexact operator have first to be modeled (Block 1) for different error metrics. Then, the error metrics derived are used to evaluate the output application Quality of Service (QoS) (Block 2) despite the induced approximations.

A. Error Metrics

Performance metrics have been proposed to evaluate the savings offered by the use of this functional approximation, as the energy/area reduction. Nevertheless, inexact arithmetic operators have also to be characterized in terms of approximation error.

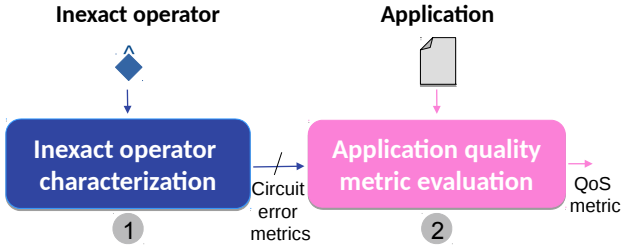


Fig. 1: Proposed framework for evaluating the impact of inexact operators on an application.

1) Metrics for Quality of Service: The characterization of the error induced by the approximations allows to know the impact of the approximation on the application QoS. This step corresponds to Block 2 in Figure 1. The application quality metric, whose measurement depends on the application, quantifies the output quality of the application. For instance, for a signal processing application, the application quality metric can be the Signal-to-Noise Ratio (SNR), whereas for an image processing application, the application quality metric can be the Structural Similarity Index Measure (SSIM). Nevertheless, to derive the QoS at the output of an application, the errors induced by inexact operators have first to be modeled.

2) Circuit Error Metrics: Numerous error metrics for inexact arithmetic operators have been proposed (Block 1 in Figure 1). Inexact arithmetic circuits are traditionally characterized based on the absolute Error Distance (e) of the calculation output, expressed as:

$$e = |\widehat{z} - z| \tag{1}$$

where $\widehat{z}$ and z are the erroneous and exact outputs of the computation, respectively. Then, statistical error characteristics, the mean Error Distance (mean ED) μ_{e}, the Standard Deviation (e_{rms}) and the Error Rate (f), are derived from e, defined as:

$$\mu_e = \frac{1}{N} \sum_{i \in \mathcal{I}} e_i \tag{2}$$

$$f = \tfrac{1}{N} \sum_{i \in \mathcal{I}} f_{e_i}, \text{with } f_{e_i} = \begin{cases} 1 \text{ if } e_i = 0 \\ 0 \text{ else} \end{cases} \tag{3}$$

$$e_{\text{rms}} = \sqrt{\frac{1}{N} \sum_{i \in \mathcal{I}} {e_i}^2} \tag{4}$$

where e_i is the Error Distance of the i^{th} stimuli on a sample set $\mathcal{I}$ of size N.

The error can also be characterized in terms of maximum Error Distance (maximum ED) M_{e} defined as:

$$M_{\text{e}} = \max_{i \in \mathcal{I}} e_i \tag{5}$$

Finally, the Probability Mass Function (PMF) of the error induced by the inexact operator can also be used as a metric. The PMF of the approximation error is the function indicating the probability that the error distance is exactly equal to a particular value.

B. Application Quality Metric Determination

Currently, two types of state-of-the-art approaches can be used to evaluate the quality metric of an application implementing inexact operators: analytical and simulation-based approaches.

1) Analytical Techniques: Analytical methods mathematically express statistics on the error at the output of the application. Based on the error propagation method proposed with Interval Arithmetic (IA) or Affine Arithmetic (AA), in [21] the authors proposed an adaptation of these methods to inexact circuits. IA derives guaranteed error bounds at the output of an application by propagating intervals representing the different variables. For instance, IA models a variable x by $[\underline{x}; \overline{x}]$ where $\underline{x}$ represent the minimum value of the variable x and $\overline{x}$ its maximum value. Then, rules are proposed in [22] to propagate the intervals through simple and non recursive systems. IA gives guaranteed but pessimistic error bounds.

AA has then been proposed to take into account first order correlations between variables contrary to IA. AA models the variables by affine forms. For instance, a variable x is modeled by $x_0 + x_1 \cdot \epsilon_1 + ... + x_n \cdot \epsilon_n$ where x_0 is the mean value of the distribution of x, x_i are the coefficients of the affine form and ϵ_i are independant and identically distributed (i.i.d.) variables in $[-1; 1]$. The asset of AA compared to IA is that the relationship between variables is kept when computing the error bounds leading to less conservative error bounds.

When applying IA or AA to inexact arithmetic circuits, the asymmetric nature of the error profile induces poor results. Thus, in [21], Modified Interval Arithmetic (MIA) and Modified Affine Arithmetic (MAA) are proposed as variants of IA and AA. In this case, intervals/affine forms are used to represent the bars of the PMF of the approximation error. Finally, rules are used to propagate the PMF through simple blocks (addition, substraction, multiplication and division) so as to get the output error profile. When implementing MIA or MAA in Block 2, the error metric required from Block 1 is the PMF of the inexact operator implemented. Nevertheless, as presented in [21], these techniques are suffering from range explosion and pessimistic bounds in the case of MIA, and storage explosion when it comes to MAA. Besides, the PMF of the inexact operator has first to be characterized by simulations, which becomes prohibitive in case of large bit-widths inexact operators. The application of such techniques in complex applications is then questionned.

In [13], an analytical framework is proposed to compute the Peak Signal-to-Noise Ratio (PSNR) metric for image processing applications implementing specific types of inexact arithmetic adders. When implementing this analytical technique, the error metric required from Block 1 is the mean ED. Nevertheless this analytical framework is specific to particular inexact adder types.

Sengupta et al. [23] proposed an analytical framework to characterize the variance of the error at the output of a Directed Acyclic Graph (DAG) composed of inexact adders. In the proposed method, the approximation is supposed to be applied only on the Least Significant Bits (LSBs). For each node of the DAG potentially composed of inexact adders, the variance of the error is computed depending on the number of LSBs approximated and on the error distribution. Nevertheless, to determine the relationship between the variance and the number of LSBs approximated, for each considered inexact adder, exhaustive simulations are required.

Finally, in [24], a more generic analytical approach based on the Fourier and Mellin transforms has been proposed to evaluate the PMF of the error at the output of a circuit implementing inexact operators. The asset of the proposed method is to be applicable to varied inexact operators while giving results very close to the results obtained with Monte-Carlo simulations, but the gain in terms of time are not enough to answer to the growing space to explore when implementing inexact operators in a complex application, considering its exponential theoretical complexity.

2) Simulation-based Techniques: When implementing an approximation in an application, functional simulation techniques are mainly used to link the errors induced by the approximation to the application QoS.

Functional simulation techniques run the approximate application on data and checks the obtained QoS. Finally, statistics on the impact of the approximation on the application QoS can be computed. Functional simulation is used in [17]–[19] with inexact arithmetic operators. Nevertheless, the simulation of inexact arithmetic operators is complex. To mimic the behavior of inexact arithmetic operators, Bit-Accurate Logic-Level (BALL) simulations are required to catch the internal structure modifications of the operator at the logic-level. The BALL simulation time of a 16-bit inexact adder is around 300 times longer than the one of a native accurate processor instruction, and even 4000 times longer in the case of an inexact multiplier, which makes exhaustive simulation impossible.

To reduce the simulation time of applications implementing inexact operators, a technique to accelerate the simulation of inexact arithmetic operators, "Fast and Fuzzy", is proposed in [25]. The "Fast and Fuzzy" simulator simplifies the approximation error model to fasten the simulation. The BALL simulation of an inexact operator is replaced by the exact operation to which is added a pseudo-random variable modeling the approximation error. The "Fast and Fuzzy" simulator is designed to quickly evaluate the impact of different approximations at the hardware level on the QoS of an application. When implementing the "Fast and Fuzzy" simulator in Block 2, the error metrics required from Block 1 are the mean ED, the Error Rate and the maximum ED of the inexact operator implemented.

C. Inexact Operator Characterization

1) Analytical Techniques: Analytical techniques have been proposed to evaluate error metrics of inexact operators. In [13], estimated values for the error rate and the mean error distance of several block-based inexact adders are analytically derived. The derivation of error metrics for the different adders is handled separately. For instance, to derive the error metrics of the Almost Correct Adder (ACA), the authors form the universal error set composed by all the possible error patterns in the inexact operator. With a n-bit ACA, it is possible to derive n disjoint subsets whose union form the universal error set. The total mean ED of the operator is then defined by the sum of the mean ED in each subset, and the mean ED in

each subset is approximately equal to $2^i \cdot q_i$ where q_i is the probability to be in the considered subset. The Error Rate can be derived as $\sum_i q_i$. Through the probabilistic analysis of the inexact operator, the values of q_i are analytically derived.

When it comes to an n-bit Equally-Segmented Adder (ESA) divided into $r = \lceil \frac{n}{k} \rceil - 1$ sub-adders, since all the sub-adders have an equal size except the first sub-adder which is exact, the Error Rate is equal to $1-(\frac{1}{2})^r$. An approximation is then used to compute the mean ED. A similar method is applied for the Error-Tolerant Adder type II (ETAII) also giving approximate values of the Error Rate and mean ED.

As an improvement of the method proposed in [13], [14] derived a method to compute the exact error profile of block-based inexact adders. Another improvement brought by [14] is to provide a generic method to compute the error statistics of block-based adders. Making the assumption that the inputs are uniformely distributed, the authors compute the probabilites of the signals propagating, generating and killing the carry.

Given these probabilites, the computation of the Error Rate is possible. To derive the error distribution, the binary representation of the Error Distance, named the "error pattern" is analyzed. All the possible error patterns are enumerated and their probability of occurrence is computed.

In [15], a method to compute the exact error distribution of inexact arithmetic adders is also proposed. The method trades off complexity for genericity, not only targetting block-based adders. Nevertheless, this method is particularly long to analyze large bit-width adders. Again, the conditions on the inputs that led to an error are identified and treated as independent events using probabilities.

Roy and Dhar [26] complemented the method proposed in [15], deriving the accurate value of the mean ED of inexact Lower Significant Bit (LSB) adders. This method is based on the structure of these adders decomposed into several approximate sub-adders and an accurate one on the Most Significant Bit (MSB). Matrices storing the different error amplitudes for each sub-adder are built to finally compute the mean ED. Nevertheless, the proposed method targets only the estimation of the mean ED which is not enough to characterize the error generated by an inexact operator, and is also particular to a class of operator.

Finally, the methods in [13]–[15], [26] are all dedicated to specific structures of inexact operators. If the application designer is willing to test inexact operators belonging to different types, the analytical method to compute the error statistics requires a new mathematical derivation. Besides, to compute the desired metrics, the number of computations to do becomes really high with the bit-width and an important memory storage is required. To end with, no estimation has been proposed on the maximum ED, which is a critical parameter when implementing an approximation in an application.

2) Simulation-based Techniques: To characterize the error induced by inexact operators, simulation-based techniques are massively used.

For instance, before using MIA or MAA to propagate the errors through an application, a characterization phase based on simulations is required. This characterization phase is required to derive the PMF of the error-free input data, the PMF of the error generated by the inexact operator and if the input is noisy, the PMF of the error on the input data. Once the different PMF have been derived, they are stored in Look-Up Tables (LUTs).

Functional simulation can also be used to compute the statistics of the error induced by the approximations. The inexact operator can be simulated exhaustively, i.e. for all possible inputs. For instance, if the considered inexact operator has two unsigned inputs x and y coded on N_x-bit and N_y-bit respectively, the exhaustive input set $\mathcal{I} = \mathcal{I}_x \times \mathcal{I}_y$ is composed of 2^{Nx+Ny} values. Consequently, for high bit-widths in the case of inexact operators, and more generally if the input design space is large, exhaustive simulations are not feasible because of the required simulation time.

Commonly, functional simulation is applied on a given number of random inputs. Inexact operators are generally simulated with 5 million random inputs as proposed in [19], which is the typical inexact circuit characterization method. Nevertheless, the quality of the statistical characterization obtained from a random sampling is highly dependent on the number of samples taken and on the chosen input distribution. Besides, the quality of the estimation of the statistics is not evaluated, and the random sampling based on a fixed number of samples can be ineffective in terms of simulation time. To be used in a real application, a method to characterize the error induced by inexact arithmetic operators with a user-defined confidence interval is proposed in this paper.

III. Estimation of the error of approximation

The proposed method is intended to be used as Block 1 from Figure 1 to estimate the following metrics for inexact operators: the mean ED μ_{e}, the Error Rate f and the maximum ED M_{e}. The proposed method demonstrates that the statistical study of the approximation error can lead to a significant reduction in the size of the sample set to simulate in order to characterize an inexact operator given user-confidence information. The proposed framework is not specific to a class of inexact operators and can be applied to adders, multipliers or even more sophisticated operators, which has not been proposed yet. First, statistical parameters as the mean ED and f are estimated with inferential statistics. Then, the extremum bounds on the approximation error are derived with the extreme value theory.

A. Statistical Estimation of the mean ED and the Error Rate

Inferential statistics, presented in [27], aim at reproducing the behavior of a large population using a subset of this population. This statistical analysis is particularly interesting in the case of high bit-width inexact arithmetic operators, where the exhaustive characterization is not feasible. Using inferential statistics, the input operands set is sampled to give an estimation with an accuracy h and a probability p that the estimation is contained within the estimated confidence interval, instead of simulating exhaustively all the possible

input operands combinations in $\mathcal{I}$. This method is used to compute confidence intervals on the mean ED μ_{e} and the Error Rate f. Since the probabilistic laws used to estimate those parameters are centered, the obtained confidence intervals also are. In this case, the accuracy h on the estimation of the confidence interval $I = [a, b]$ is expressed as $h = \frac{b-a}{2}$. The objectives of the proposed method are: 1) to estimate the error characteristics more efficiently, using a reduced but sufficient number of samples, 2) to provide the estimated error characteristics with a given confidence information. The proposed method computes the minimal number of samples to simulate, to estimate the error characteristics μ_{e} and f according to (h, p). N_{μ_e} and N_f represent the minimal number of samples to estimate μ_{e} and f, respectively.

1) Computation of the minimal number of samples N_{μ_e} to estimate μ_e: The empirical mean $\overline{\mu_e}$, a punctual estimator of μ_{e}, is used to estimate the real value of the mean error distance, μ_e. That is to say, $\overline{\mu_e}$ is an estimation of μ_e computed over a given number of samples. $\overline{\mu_e}$ is used to compute the theoretical number of samples N_{μ_e} to simulate to get an estimation according to the confidence parameters (h, p). To estimate N_{μ_e}, the standard deviation of the simulated samples is needed. The empirical mean $\overline{\mu_e}$ and the empirical standard deviation $\tilde{S}^2$, a biased estimator of the standard deviation σ_e, are computed over T samples as:

$$\overline{\mu_e} = \frac{1}{T}\sum_{i=1}^{T} e_i \tag{6}$$

$$\tilde{S}^2 = \frac{1}{T}\sum_{i=1}^{T}(e_i - \overline{\mu_e})^2 \tag{7}$$

The estimators $\overline{\mu_e}$ and $\tilde{S}^2$ are associated to confidence intervals IC_{μ_e} and IC_{σ_e} respectively, defined such that they include μ_e and σ_e with a probability p. Then, according to the Central Limit Theorem, since (e_1, e_2, ..., e_T) are belonging to the same probability set, are independent and identically distributed, the property in Equation 8 is verified if the number of samples N_{μ_e} is higher than 30 [27]. Consequently, no assumption has to be made on the distribution of the population. In Equation 8, $\mathcal{N}(0, \sigma)$ represents a gaussian distribution whose mean is 0 and standard deviation is σ.

$$\sqrt{N_{\mu_e}}(\overline{\mu_e} - \mu_e) \xrightarrow{\text{law}} \mathcal{N}(0, \sigma) \tag{8}$$

The confidence interval $\text{IC}^p_{\mu_e}$ is developed in Equation 9 and contains μ_e with a probability p. The term $a^\alpha_{\mu_e}$ embodies the accuracy on the estimation and is computed as in Equation 10.

$$\text{IC}^p_{\mu_e} = [\overline{\mu_e} - a^\alpha_{\mu_e}; \overline{\mu_e} + a^\alpha_{\mu_e}] \tag{9}$$

In Equation 10, z_α is given by the table of the standard normal distribution given p [27]. N_{μ_e} is the minimal number of samples to simulate to get an estimation respecting the user-defined parameters (h, p).

$$a^\alpha_{\mu_e} = z_\alpha \cdot \frac{\tilde{S}}{\sqrt{N_{\mu_e} - 1}} \tag{10}$$

The desired accuracy h on the estimation of the mean ED impacts the number of samples to simulate as expressed in Equation 11. To get a desired accuracy of h, $a^\alpha_{\mu_e}$ must be lower or equal to h.

$$N_{\mu_e} > \frac{z^2_\alpha \cdot \tilde{S}^2}{h^2} \tag{11}$$

According to Equation 11, if the standard deviation of the error generated by the inexact adder is very large, N_{μ_e} can be very high. Inexact operators with a large standard deviation renders circuits with poor interest. In the proposed method, a maximal number of simulated points $N_{\max}$ has been set. If the required number of points is higher than $N_{\max}$, the estimated mean ED and Error Rate f are given according to p but with a precision h depending on $N_{\max}$.

2) Computation of the minimal number of samples N_f to estimate f: The proportion of input operands in $\mathcal{I}$ that generate an error is embodied by the Error Rate f. f follows a hypergeometric law [27]. The estimator used for the error rate is f_e, the proportion of samples generating an error in the random sampling. The estimator is computed as in Equation 3, applied on the sampled set. Such an estimator can also be associated to a confidence interval IC^p_f that is defined such that the real error rate f of the population $\mathcal{E}$ is contained in this confidence interval with a probability p. The confidence interval IC^p_f is defined in Equation 12.

$$\text{IC}^k_f = [f_e - a^\alpha_f; f_e + a^\alpha_f] \tag{12}$$

In Equation 12, a^α_f represents the accuracy on the estimation of f, z_α is given by the table of the standard normal distribution [27] and N_f represents the minimal number of samples to simulate, to get an estimation with the user-defined parameters (h, p).

$$a^\alpha_f = z_\alpha \cdot \sqrt{\frac{f_e(1 - f_e)}{N_f}} \tag{13}$$

To get a desired accuracy of h, a^α_f must be lower or equal to h, which impacts N_f as in Equation 14.

$$N_f > \frac{z^2_\alpha \cdot f_e(1 - f_e)}{h^2} \tag{14}$$

B. Estimation of the maximum ED with Extreme Value Theory

The proposed method aims at estimating the maximum ED according to an in-range probability p. An interesting approximate computing technique rarely generates the maximum ED which can consequently be considered as a rare event. Currently, simulation-based techniques are used to estimate the maximum ED but no guarantee is obtained that the real maximum value is not higher than the observed maximum value.

The user-defined confidence parameter p allows to be more or less conservative on the estimation depending on the critical nature of the application. Our approach exploits the statistical properties of the maximum approximation error, using the Extreme Values Theory (EVT). The probability p corresponds to the probability that the real value of the maximum ED M_{e}

is lower or equal to the estimated value of the maximum ED $\tilde{M}$. The higher p, the more conservative the estimation. The studied population for estimating the maximum ED is the set (e_1, e_2, ..., e_T) of error distance values, that are independent and identically distributed events.

EVT [29], [30] aims at describing the stochastic behavior of minima or maxima, and is particularly useful in domains such as finance or insurance. EVT aims at predicting the occurrence or amplitude of rare events even though no observation is available.

The Cumulative Distribution Function (CDF) of the set of error distance values is called G and its associated survivor function is $\overline{G} = 1 - G$. The ordered statistics on a sample of size T can be defined as $e_{1,T} \leq e_{2,T} \leq ... \leq e_{T,T} = M_T$. The proposed method aims at estimating the value $\tilde{M}$ such that:

$$\tilde{M} = \overline{G}^{-1}(\alpha_T)$$

where $\alpha_T = 1 - p < \frac{1}{T}$ when $\lim_{T \to \infty} \alpha_T = 0$, i.e. estimating the extreme quantile value for α_T. Nevertheless, the CDF and its survivor function are unknown. To estimate $\tilde{M}$, the following property from [29] and [30] is used:

Property III-B.1: The distributions of extremum values converge towards an extreme value distribution.

Three types of extreme value distributions exist, the Gumbel, Weibull and Fréchet distributions. For estimating the upper bound on the error induced by inexact operators, the followed distribution is the Gumbel distribution [31]. Contrary to the estimation of mean ED or f, no confidence interval can be computed on the estimation of M_e. The value $\tilde{M}$ estimated corresponds to the value that encompasses M_e with a given user-defined probability p. The proposed method is inspired of the dynamic range determination processed in fixed-point theory as presented in [31].

To estimate the real value of the maximum ED, the value $\tilde{M} = \overline{G}^{-1}(\alpha_T)$ is computed. So as to compute $\tilde{M}$ for a given probability p, the distribution of the maximum error values has to be studied and identified to a Gumbel distribution.

To derive the maximum error values distribution, T samples are simulated k times. The maximum error value over each sample of size T is extracted, and the obtained list of maximum error values models the experimental maximum error values distribution. It can then be identified to a Gumbel distribution defined hereafter by its density function g in Equation 15 and its CDF G in Equation 16:

$$g(x) = \frac{1}{\sigma}\text{exp}(-\frac{(x-\mu_G)}{\sigma_G})\text{exp}(-\text{exp}(-\frac{-(x-\mu_G)}{\sigma_G})) \quad (15)$$

$$G(x) = \text{exp}(-\text{exp}(-\frac{(x-\mu_G)}{\sigma_G})) \quad (16)$$

The parameters (σ_G, μ_G) are used to fit the Gumbel distribution to the experimental distribution of maximum error values. The term σ_G is called the scale parameter and is used to stretch or shrink the distribution. The term μ_G is called the location parameter and is used to shift the distribution on the horizontal axis. The computation of (σ_G, μ_G) is detailed in Equations 17, 18:

$$\sigma_G = \frac{1}{\pi} \cdot \sqrt{6}\tilde{S}_G \quad (17)$$

$$\mu_G = \overline{\mu}_G - \sigma_G \cdot \lambda \quad (18)$$

where $\tilde{S}_G$ is the empirical standard deviation of the experimental maximum error values, $\overline{\mu}_G$ is the empirical mean of the experimental maximum error values, and λ is the Euler constant. The parameters (σ_G, μ_G) completely define the Gumbel distribution fitting the maximum error values distribution.

Once the distribution of maximum error values has been completely defined, the goal of our proposed method is to compute the value $\tilde{M}$ such that $G(\tilde{M}) = P(X \leq \tilde{M}) = p$ where p is the in-range probability. Equations 19, 20 can then be derived.

$$p = P(X \leq \tilde{M}) = \text{exp}(-\text{exp} - \frac{(\tilde{M}-\mu_G)}{\sigma_G}) \quad (19)$$

$$\tilde{M} = \mu_G - \sigma_G \cdot \ln(\ln(\frac{1}{p})) \quad (20)$$

IV. Proposed Algorithm

Algorithm 1 presents the estimation of the mean ED and the Error Rate f with a fair number of samples. From the simulated samples, the maximum ED is also estimated. The population on which inferential statistics are applied is the set $\mathcal{E} = \{e_i / i \in \mathcal{I}\}$. The statistical variables mean ED μ_e, the Error Rate f and the Standard Deviation (STD) σ_e are describing the population $\mathcal{E}$ and are consequently characterized by probability laws. To sample the population $\mathcal{E}$, a random sampling method without replacement is used. So that the exhaustive sampling behaves like a non exhaustive sampling, T, the initial number of simulated samples, is taken higher or equal to 30.

To characterize an inexact arithmetic operator, the user provides the following information: the desired accuracy on the estimation h, the probability p that the estimated interval contains the real value for μ_e and f, and that the real maximum is lower than the estimated maximum, and the refreshment period T. T is used to refine the number of samples required. A first sampling extracts T samples from the population $\mathcal{E}$, on which are computed the empirical mean $\overline{\mu_e}$, standard deviation $\tilde{S}^2$ and empirical error rate f_e. From these estimations, the theoretical minimal numbers of samples to compute to estimate μ_e and f according to the user's precision constraints is obtained. The maximum error value of the simulated samples is also extracted, and appended to the set of maximum error values J.

Then, to estimate μ_e and f, the empirical standard deviation $\tilde{S}$, empirical mean $\overline{\mu_e}$ and error rate f_e of the samples are used. Those three estimators are computed to derive the theoretical numbers of samples to simulate to estimate μ_e and f, N_{μ_e} and N_f respectively. The maximum of these two values, N, is taken as the reference number of samples to simulate. The same process is refined every T samples to converge

towards a minimized value of N, and every T samples, the maximum error value is extracted and appended to the set J. Consequently, the higher T, the more the computations of N_{μ_e} and N_f are accurate. If N is higher than $N_{\max}$, $N_{\max}$ points are simulated but the estimated results are not fulfilling the accuracy requirement, embodied by h. In this case, the obtained accuracy h can be computed depending on $N_{\max}$ as:

$$h = \frac{z_\alpha \cdot \tilde{S}}{\sqrt{N_{\max} - 1}} \tag{21}$$

Algorithm 1 Characterization of μ_e, f and M_e of population $\mathcal{E}$

procedure CHARACTERIZE$\mu_e, f, M_E(\mathcal{E}, h, p, T, N_{\max})$
  $\alpha = 1 - p$
  $J = \varnothing$
  $E = (e_1, .., e_T) = \text{sampling}(\mathcal{E}, T)$
  $M = \max(E)$
  $J = J \cup M$
  $\overline{\mu_e} = \text{computeMean}(E, T)$ ▷ Equation 6
  $\tilde{S}^2 = \text{computeSD}(E, T, \overline{\mu_e})$ ▷ Equation 7
  $f_e = \text{computeFreq}(E, T)$ ▷ Equation 3
  $N_{\mu_e} = \text{computeNMean}(\tilde{S}^2, h)$ ▷ Equation 11
  $N_f = \text{computeNFreq}(f_e, h)$ ▷ Equation 14
  $N = \max(N_{\mu_e}, N_f)$
  $n = T$
  $\mathcal{E} = \mathcal{E} \backslash E$
  while $n < N$ **do**
    $E' = (e_n, .., e_{n+T}) = \text{sampling}(\mathcal{E}, T)$
    $M = \max(E)$
    $J = J \cup M$
    $E = E \cup E'$
    $\overline{\mu_e} = \text{computeMean}(E, n + T)$
    $\tilde{S}^2 = \text{computeSD}(E, n + T, \overline{\mu_e})$
    $f_e = \text{computeFreq}(E, n + T)$
    $n+ = T$
    $N_{\mu_e} = \text{computeN}(\tilde{S}^2, h)$
    $N_f = \text{computeNFreq}(f_e, h)$
    $N = \max(N_{\mu_e}, N_f)$
    $\mathcal{E} = \mathcal{E} \backslash E$
    if $N \geq N_{\max}$ **then**
      $N = N_{\max}$
    end if
  end while
  $\sigma_G = \text{computeScale}(J, n)$ ▷ Equation 17
  $\mu_G = \text{computeLocation}(J, n, \sigma_G)$ ▷ Equation 18
  $\hat{M} = \text{computeMax}(\mu_G, \sigma_G, p)$ ▷ Equation 20
end procedure

Once the N points have been simulated, the set of maximum error values J is used to identify the obtained distribution of maximum error values to a Gumbel distribution. The parameters σ_G and μ_G are computed and used to compute the estimation of the maximum ED according to the in-range probability p.

V. Experimental Study

A. Inexact Adders under Consideration

For this experimental study, inexact adders have been selected among three major kinds of topology explored in the literature: timing-starved adders [9], speculative adders [10], [11] and carry cut-back adders [12].

The ACA [9] is the most known timing-starved adder. It is composed of an array of overlapping and translated sub-adders, so that each sum bit is constructed using exactly the same amount of preceding carry stages, except the first ones. The critical-path delay is thus limited, but the circuit cost is fairly high. The ACA is an interesting case study due to its very low Error Rate. Errors occur when carry chains are longer than the ACA sub-adder size, which is the main ACA design parameter. Thus, ACA designs have a very low frequency of errors, but of high arithmetic distance.

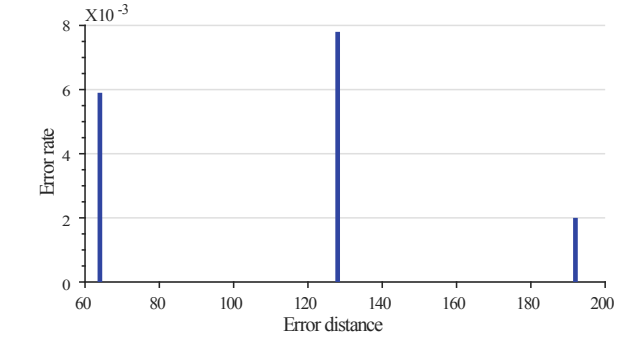
The Inexact Speculative Adder (ISA) [10] is the leading architecture of speculative adders. As an evolution of the ETAII [11], it also segments the addition into several sub-adders with carry speculated from preceding sub-blocks. The ISA features a shorter speculative overhead that improves speed and energy efficiency, and introduces a dual-direction error correction-reduction scheme that lowers mean and worst-case errors. ISA designs typically display higher Error Rates than ACA but with lower error values, depending of the number of sub-blocks and error compensation level, which are the main ISA design parameters.

The Carry Cut-Back Adder (CCBA) [12] exploits a novel idea of artificially-built *false paths* (i.e. paths that cannot be logically activated) [19], co-optimizing arithmetic precision together with physical netlist delay. To guarantee floating-point-like precision, high-significance carry stages are monitored to cut the carry chain at lower-significance positions. These cuts prevent the critical-path activation, thus relaxing timing constraints and enabling energy efficiency levels out of reach from conventionally designed circuits. The Error Rate ranges similarly as for the ISA, but the error values are lower than those generated by the ACA and the ISA, depending of the number of cuts and cutting distance, which are the main CCBA design parameters.

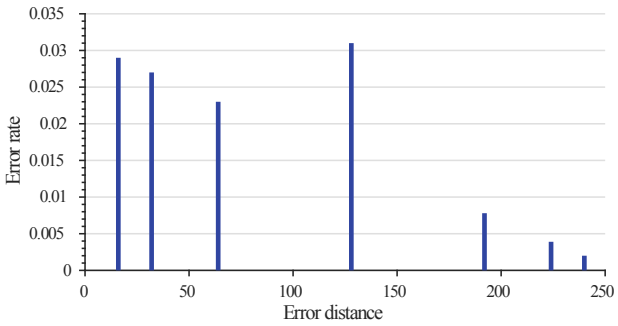
The important error characteristics when implementing inexact operators are the mean ED, the Error Rate and the maximum ED. The PMF of the ACA with two different carry-chain lengths are presented in Figure 2.

B. Estimation of the mean ED and the Error Rate

The proposed experimental study aims at showing that 1) the proposed method correctly estimates the error characteristics mean ED μ_e and Error Rate f of circuits for various bit-widths, 2) this estimation remains consistent for higher bit-widths where exhaustive simulation is not possible, and 3) for the majority of inexact adders, the proposed method outperforms naive stochastic simulation with a fixed-number of samples



(a) 8-bit ACA_6.



(b) 8-bit ACA_4.

Fig. 2: Probability mass functions of 8-bit inexact adders.

(FNS simulation). Indeed, two cases are shown: the proposed method requires less samples and thus converges faster towards an accurate error estimation, or the proposed method requires more samples than the traditional FNS simulation which is, in this case, not accurate enough.

Each above-mentioned adder architecture have been synthesized, with different bit-widths, from 8 to 32 bits, and varying main design parameters, in order to cover a large spectrum of error behaviors. The proposed characterizations have been completed with $h = 5\,\%$ and $p = 95\,\%$ on an Intel Core i7-6700 processor. The consistency of the error characterization remains the same even with varied h and p. The higher p and the lower h, the larger the sample set to simulate.

1) Quality of the estimation for small bit-widths: To first check the quality of the proposed method, small bit-width inexact adders have been characterized with our method, as well as with an exhaustive characterization using BALL simulations to obtain their real error characteristics. That is to say that the Error Distance values generated by the inexact operators under consideration have been computed for all their possible input values. For instance, for 16-bit inexact operators, the exhaustive characterization requires the simulation of 2^{32} operations.

Table I reports the confidence intervals on μ_e and f obtained by the proposed method, compared to their real values, and the numbers of samples N used for the proposed characterization. The ratio N_ratio between the number of simulations N done using the proposed characterization method, and the number of simulations done when using an exhaustive characterization is also indicated. For 8-bit operators, the number of simulations when using an exhaustive characterization is 2^{16}, while for 16-bit operators, the number of simulations when using an exhaustive characterization is 2^{32}. For both 8-bit and 16-bit adders, the obtained confidence intervals almost always contain the real values, demonstrating that our method is accurate. The 16-bit ACA_8 is the only design for which the obtained confidence intervals do not contain the real values (c.f. bold numbers), but the relative error between confidence interval bound and real value is extremely small.

For most operators, only a few tens of thousands of simulated samples were required to get precise error characteristics. For both 16-bit ACA_{12} and ACA_8, the number of simulated samples has been saturated with $N_\text{max} = 25$ millions (c.f. bold sample number). This is due to the fact that ACA adders have a large standard deviation in error values. Though, the proposed method outputs very accurate estimated values of f and mean ED. The largest relative error on the estimated values compared to the exhaustive characterization is on the estimation of f of the operator ACA_8, and is equal to 1.27%.

2) Consistency of the estimation for 32-bit operators: To check the consistency of the proposed method for this larger bit-width, the proposed characterization has been compared to random FNS simulation with 5 million samples from [19], which is the typical inexact circuit characterization method as exhaustive simulation is not feasible. The chosen CCBA and ACA adders are Pareto-optimal designs shown in the comparative study of [19]. Those adders are realistic designs to be implemented, and thus represent ideal subjects for the proposed characterization.

Table II reports the results for 32-bit inexact adder characterization. In the case of 32-bit operators, it is to be noted that both characterizations (the proposed characterization and the one obtained with FNS simulation) are statistical estimates. In case the two methods do not converge towards the same estimation, bold numbers represent values obtained with higher amount of samples, assumed more accurate. The ratio N_ratio between the number of simulations N done using the proposed characterization method, and the total number of simulations (2^{64}) is also indicated.

For 2 out of 8 designs ($\text{CCBA}_{1,5}$ and $\text{ISA}_{2,8}$), the obtained confidence intervals obtained with less simulation samples than the FNS simulation do not contain the error values from this latter. Nevertheless, the obtained estimated values of f and mean ED are very close from the random characterization. Inversely, for 3 of them ($\text{CCBA}_{1,6}$, $\text{ISA}_{2,2}$ and ACA_{17}), the proposed method has converged into different confidence intervals than the BALL simulation, as it has determined that more samples were required for safe estimation. This is coherent, as by user decision, the confidence interval has only 95 % chance to contain the real value. The most critical case concerns ACA_{17}. For this characterization, naive BALL simulation has dangerously underestimated mean ED compared to the proposed method. This is due to the very low error rate of the 32-bit ACA, for which 5 million samples is insufficient to make good statistics on errors.

TABLE I: Estimation results and comparison with exhaustive characterization for operators of small word-lengths (bold numbers if confidence intervals do not contain the real values).

N_{bits}	Op. type	Name	IC_{μ_e}		μ_e	IC_f		f	N	N_{ratio}
8	ISA	$ISA_{2,2}$	$8.63{\cdot}10^{-1}$	$9.55{\cdot}10^{-1}$	$8.75\cdot10^{-1}$	$1.08{\cdot}10^{-1}$	$1.19{\cdot}10^{-1}$	$1.09\cdot10^{-1}$	11,765	0.180
		$ISA_{2,4}$	$4.16\cdot10^{-2}$	$1.38\cdot10^{-1}$	$9.38{\cdot}10^{-2}$	$1.04{\cdot}10^{-2}$	$3.46{\cdot}10^{-2}$	$2.34{\cdot}10^{-2}$	578	0.009
	ACA	ACA_6	1.67	1.99	1.75	$1.51{\cdot}10^{-2}$	$1.77{\cdot}10^{-2}$	$1.56{\cdot}10^{-2}$	35,873	0.547
16	CCBA	$CCBA_{1,6}$	$7.30{\cdot}10^{-1}$	$8.18{\cdot}10^{-1}$	$7.50\cdot10^{-1}$	$1.83{\cdot}10^{-1}$	$2.04\cdot10^{-1}$	$1.88{\cdot}10^{-1}$	5041	$1.175{\cdot}10^{-6}$
	ISA	$ISA_{2,4}$	1.95	2.06	1.97	$3.05{\cdot}10^{-2}$	$3.21{\cdot}10^{-2}$	$3.08{\cdot}10^{-2}$	178,930	$4.166{\cdot}10^{-5}$
		$ISA_{2,6}$	$1.73{\cdot}10^{-1}$	$2.69{\cdot}10^{-1}$	$2.42{\cdot}10^{-1}$	$5.40{\cdot}10^{-3}$	$8.40{\cdot}10^{-3}$	$7.60{\cdot}10^{-3}$	11,602	$2.701{\cdot}10^{-6}$
	ACA	ACA_{12}	9.50	9.94	9.69	$4.86{\cdot}10^{-4}$	$4.91{\cdot}10^{-4}$	$4.88{\cdot}10^{-4}$	**25M**	0.006
		ACA_8	$1.71{\cdot}10^{2}$	$1.72{\cdot}10^{2}$	$\mathbf{1.70\cdot10^{2}}$	$1.57{\cdot}10^{-2}$	$1.58{\cdot}10^{-2}$	$\mathbf{1.56\cdot10^{-2}}$	**25M**	0.006

TABLE II: Estimation results and comparison with 5-million BALL simulations for 32-bit operators (bold numbers if confidence intervals do not contain the FNS values).

Op. type	Name	IC_{μ_e}		$\overline{\mu_e}$ 5M	IC_f		$\overline{f}$ 5M	N	N_{ratio}
CCBA	$CCBA_{1,5}$	$1.564{\cdot}10^{1}$	$1.574{\cdot}10^{1}$	$\mathbf{1.576\cdot10^{1}}$	$1.222{\cdot}10^{-1}$	$1.230{\cdot}10^{-1}$	$\mathbf{1.231\cdot10^{-1}}$	2,792,512	10^{-13}
	$CCBA_{1,6}$	$1.877{\cdot}10^{1}$	$\mathbf{1.889\cdot10^{1}}$	$1.897{\cdot}10^{1}$	$\mathbf{2.867\cdot10^{-2}}$	$2.880\cdot10^{-2}$	$2.860{\cdot}10^{-2}$	17,008,400	10^{-12}
	$CCBA_{1,7}$	$2.132{\cdot}10^{-1}$	$2.613{\cdot}10^{-1}$	$2.420{\cdot}10^{-1}$	$6.700{\cdot}10^{-3}$	$8.200{\cdot}10^{-3}$	$7.600{\cdot}10^{-3}$	50,176	10^{-15}
	$CCBA_{1,9}$	$4.421{\cdot}10^{-1}$	$5.482{\cdot}10^{-1}$	$5.017{\cdot}10^{-1}$	$1.700{\cdot}10^{-3}$	$2.100{\cdot}10^{-3}$	$2.000{\cdot}10^{-3}$	172,676	10^{-14}
ISA	$ISA_{2,2}$	$8.166{\cdot}10^{3}$	$\mathbf{8.183\cdot10^{3}}$	$8.189{\cdot}10^{3}$	$1.246{\cdot}10^{-1}$	$\mathbf{1.249\cdot10^{-1}}$	$1.250{\cdot}10^{-1}$	**25M**	10^{-12}
	$ISA_{2,8}$	3.826	3.933	**3.763**	$7.505{\cdot}10^{-3}$	$7.698{\cdot}10^{-3}$	$7.600{\cdot}10^{-3}$	3,130,201	10^{-13}
	$ISA_{2,10}$	$9.125{\cdot}10^{-1}$	1.012	1.003	$4.566{\cdot}10^{-4}$	$5.104{\cdot}10^{-4}$	$4.954{\cdot}10^{-4}$	3,084,740	10^{-13}
ACA	ACA_{17}	$\mathbf{1.433\cdot10^{4}}$	$1.812{\cdot}10^{4}$	$1.391{\cdot}10^{4}$	$4.999{\cdot}10^{-5}$	$5.004{\cdot}10^{-5}$	$5.002{\cdot}10^{-5}$	**25M**	10^{-12}

3) Number of simulations required for accurate estimation: Algorithm 1 refines the estimation of mean ED and f given a refreshment period T. Figures 3a and 3b illustrates the convergence of the estimation on f and mean ED respectively. The different curves, corresponding to the different operators, have different starting points depending on the chosen refreshment period T. The relative error of estimation of mean ED and f depending on the simulation length are represented. To compute the relative error of estimation ϵ of the confidence interval on mean ED, $IC_{\mu_e} = [a; b]$, the computation of the center of the estimated interval μ_e is required and is computed as:

$$\mu_e = a + \frac{b-a}{2} \tag{22}$$

Finally, the center of the estimated interval μ_e is compared to the FNS value obtained with 5-million BALL simulation $\overline{\mu_{e,5M}}$ as:

$$\epsilon = \frac{|\mu_e - \overline{\mu_{e,5M}}|}{\overline{\mu_{e,5M}}} \tag{23}$$

The same process is applied to compute the relative error of estimation of f.

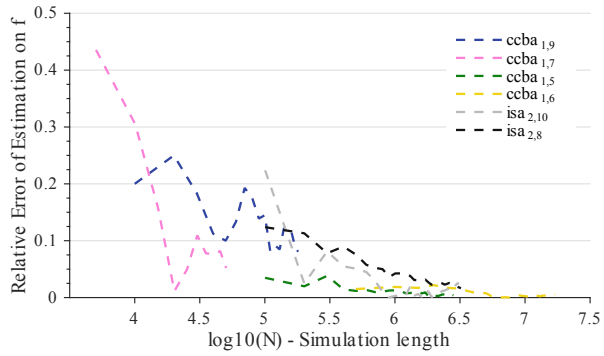
The final estimated values are all very accurate since the relative error of estimation is always lower than 0.1 %. Small bumps can be noted in the convergence of the estimated values due to the random sampling processed in each iteration of the algorithm. Besides, the speed of convergence strongly varies depending of the chosen operator. This is why the proposed method, which is an adaptive sample-size method, better fits any operator rather than naive FNS simulations.

C. Estimation of the maximum ED

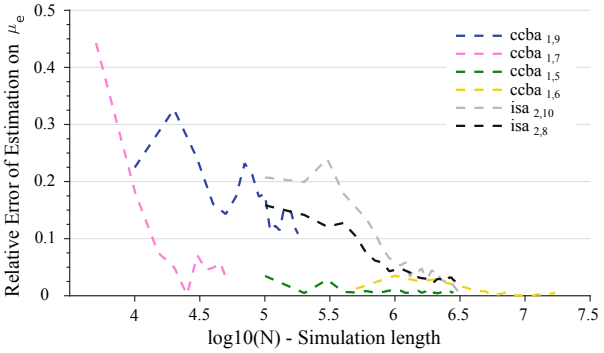
The proposed experimental study aims at showing that the proposed method correctly estimates the maximum ED of circuits for various bit-widths, and that the quality of the estimated maximum error value is configurable depending on the in-range probability p, the size of the sample sets T and the number of times the T samples are simulated, k. The total number of simulated points is then $k \times T$. Two cases are shown: the dependency of the quality of the estimation on the total number of simulated samples $k \times T$, and on the in-range probability p. maximum ED estimations have been completed with varying p, T and k.

1) Quality of the estimation for small bit-widths: To first check the quality of the proposed estimation method, the maximum ED of small bit-width inexact adders has been compared to an exhaustive characterization using BALL simulations which shows the real maximum error distance characteristics. Table III reports the estimated values $\tilde{M}$ of the maximum ED obtained by the proposed method, compared to their real values M_e, depending on the parameters (k, T, p).

For most 8-bit adders, only $k \times T = 1000$ simulations are required to correctly estimate the maximum ED encompassing the real value M_e. The in-range probability p can be used to be more or less conservative on the estimation. For the example of the ACA_6, the in-range probability can also be used to adjust the accuracy of the estimation. If p is lower than 95%, the obtained estimation $\tilde{M}$ does not always encompass the real maximum M_e. For an in-range probability $p = 95\%$, the estimated maximum value always encompasses the real maximum M_e, demonstrating that the proposed estimation is



(a) Estimation of f.



(b) Estimation of mean ED.

Fig. 3: Convergence of the estimation of mean ED and f depending on the number of simulated samples N, with $p = 95\%$ and $h = 0.5\%$ for different 32-bit adders.

conservative.

For most 16-bit adders, the estimation of the maximum ED is accurate with only $k \times T = 10^4$ simulations. The ACA_8 still requires an in-range probability of 95% to encompass the real maximum value M_e. Nevertheless, the ACA_{12} is the only design for which the estimation is accurate only for $p > 95\%$. This operator has very scattered error values and the chance to catch the real maximum during the determination of the error distance values distribution is lower than for the other inexact operators. However, this renders a poor quality inexact operator.

2) Consistency of the estimation for 32-bit operators: Table IV reports the results for 32-bit inexact adder maximum ED estimation. To check the consistency of the proposed estimation method for this larger bit-width, the obtained estimations have been compared to random BALL simulation with 5 million samples from [19].

In the case of 32-bit operators, it is to be noted that both obtained values $\tilde{M}$ and M_e (5M) are estimates. For most 32-bit adders excepted the $\text{CCBA}_{1,6}$ and $\text{ISA}_{2,10}$, the proposed method gives conservative estimates even with an in-range probability of 90%. For the operator $\text{ISA}_{2,10}$, the in-range probability has to be greater or equal to 95% to obtain a correct estimation. However, the $\text{CCBA}_{1,6}$ requires to set the in-range probability up to 99.9% to encompass the maximum error distance estimated with 5 million samples.

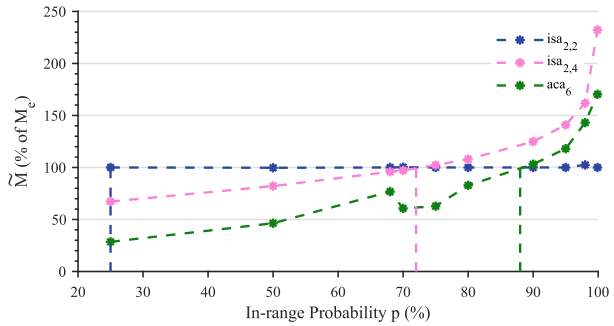
3) Accuracy of the estimation depending on the in-range probability: The in-range probability allows to be more or

TABLE III: Estimation results of maximum ED and comparison with exhaustive characterization for operators of small word-lengths (bold numbers if $\tilde{M} < M_e$).

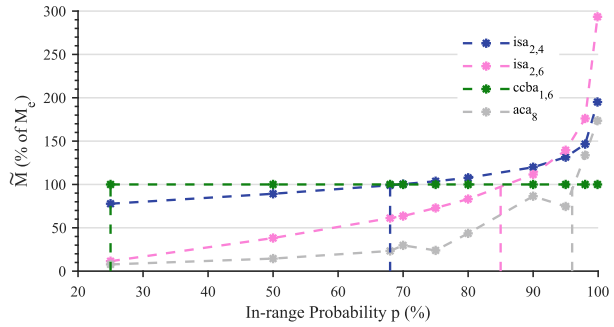
N_{bits}	Op. type	Name	p	k	T	$\tilde{M}$	M_e
8	ISA	$\text{ISA}_{2,2}$	90	10	100	8	8
			95	10	100	8	8
			98	10	100	8	8
		$\text{ISA}_{2,4}$	90	10	100	4	4
			95	10	100	6	4
			98	10	100	7	4
	ACA	ACA_6	90	10	100	**151**	192
			90	10	100	199	192
			95	10	100	210	192
			98	10	100	249	192
16	CCBA	$\text{CCBA}_{1,6}$	90	10	1000	4	4
			95	10	1000	4	4
			98	10	1000	4	4
	ISA	$\text{ISA}_{2,4}$	90	10	1000	64	64
			95	10	1000	64	64
			98	10	1000	64	64
		$\text{ISA}_{2,6}$	90	10	100	32	32
			95	10	1000	32	32
			98	10	1000	32	32
	ACA	ACA_{12}	90	10	1000	**28467**	61440
			95	10	1000	**41646**	61440
			98	10	1000	66068	61440
		ACA_8	90	10	1000	**63305**	65280
			95	10	1000	67008	65280
			98	10	1000	84188	65280

TABLE IV: Estimation results of maximum ED and comparison with Monte-Carlo characterization (5M) for 32-bit operators (bold numbers if $\tilde{M} < M_e$).

N_{bits}	Op. type	Name	p	k	T	$\tilde{M}$	M_e (5M)
32	CCBA	$\text{CCBA}_{1,5}$	90	500	1000	128	128
			95	500	1000	128	128
			98	500	1000	128	128
		$\text{CCBA}_{1,6}$	90	500	1000	**1349**	1641.6
			95	500	1000	**1409**	1641.6
			98	500	1000	**1491**	1641.6
			99.9	500	1000	1743	1642
		$\text{CCBA}_{1,7}$	90	500	1000	32	32
			95	500	1000	32	32
			98	500	1000	32	32
		$\text{CCBA}_{1,9}$	90	500	1000	325	256
			95	500	1000	373	256
			98	500	1000	439	256
	ISA	$\text{ISA}_{2,2}$	90	500	1000	65536	65536
			95	500	1000	65536	65536
			98	500	1000	65536	65536
		$\text{ISA}_{2,8}$	90	500	1000	16384	16384
			95	500	1000	16384	16384
			98	500	1000	16384	16384
		$\text{ISA}_{2,10}$	90	500	1000	**1945**	2048
			95	500	1000	2614	2048
			98	500	1000	3289	2048
	ACA	ACA_{17}	90	500	1000	$4{\cdot}10^9$	$2{\cdot}10^9$
			95	500	1000	$4{\cdot}10^9$	$2{\cdot}10^9$
			98	500	1000	$4{\cdot}10^9$	$2{\cdot}10^9$

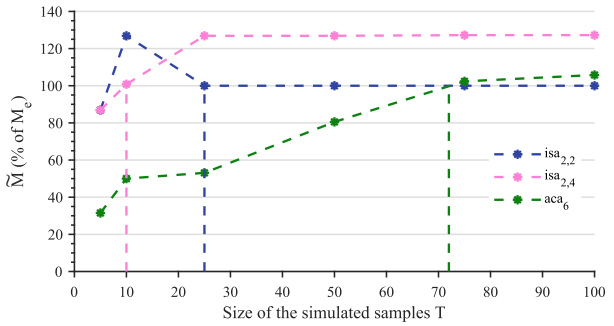


(a) 8-bit adders.

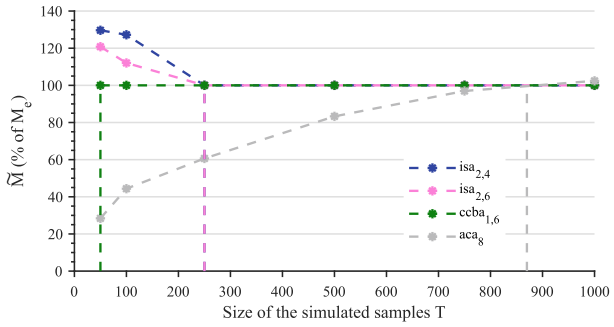


(b) 16-bit adders.

Fig. 4: Estimation of maximum ED as a percentage of M_e depending on the in-range probability p for a fixed number of simulated samples $k \times T$, $k = 10, T = 100$. Vertical lines indicate $\tilde{M} = M_e$.



(a) 8-bit adders.



(b) 16-bit adders.

Fig. 5: Estimation of maximum ED as a percentage of M_e depending on the T with $k = 10$, $p = 90\%$. Vertical lines indicate $\tilde{M} = M_e$.

less conservative on the estimate of M_e. Figures 4a and 4b depict the link between the parameter p and the accuracy of estimation. Indeed, when implementing an approximate computing technique, the maximum ED must not be underestimated. However, if the obtained value overshoots the real maximum ED, the application designer may wrongly discard a technique, hence the importance of adjusting the parameter p.

The estimated maximum error distance values $\tilde{M}$ are represented as a percentage of the accurate maximum ED values M_e for each inexact adder in Figures 4a and 4b. Vertical lines indicate for each inexact adder, when p is high enough to accurately estimate $\tilde{M}$ ($\tilde{M} = M_e$). The proposed method correctly estimates the maximum ED for both inexact adders 8-bit ISA$_{2,2}$ and 16-bit CCBA$_{1,6}$ and for p varying from 25% to 100% since these adders frequently generate the maximum error distance.

For the 8-bit ISA$_{2,4}$, $\tilde{M}$ encompasses M_e when $p \geq 72\%$ and for the 16-bit ISA$_{2,6}$, when $p \geq 68$. Nevertheless, for the ISA$_{2,6}$, ACA$_6$ and ACA$_8$, the in-range probability p has to be very high to encompass the real maximum error distance (higher than 85%, 88% and 97% respetively). Indeed, the generated errors are scattered and local maxima may be found in the different samples, leading to a lower value of $\tilde{M}$. When p increases, the estimated maximum ED becomes very conservative. Small bumps can be observed for the ACA$_6$ and ACA$_8$, also caused by the large standard deviation generated by this type of inexact adder. It is still to be noted that for 8-bit and 16-bit estimations, the number of simulated samples is small, since equal to 1000 samples which represents 1.5% of the whole input space for 8-bit operators, and only $2.3 \cdot 10^{-5}\%$ of the whole input space for 16-bit operators.

4) Accuracy of the estimation depending on the number of simulated samples: The accuracy of the estimation can also be controlled with the total number of simulated points $k \times T$ taken to derive the distribution of the maximum error distance values. Figures 5a and 5b represent the evolution of the accuracy of estimation depending on the size of the simulated samples T, with k set to 10, for different 8-bit and 16-bit adders respectively.

Figures 5a and 5b represent the estimated value $\tilde{M}$ as a percentage of M_e depending on T. In this case, T samples are simulated and their maximum is extracted. This operation is done $k = 10$ times. For 8-bit adders, the estimates converge towards a value for 8-bit ISA$_{2,2}$ and ISA$_{2,4}$ as soon as $T \geq 25$. For 16-bit adders, the estimates converge towards a value for ISA$_{2,4}$, ISA$_{2,6}$ as soon as $T \geq 250$ and for CCBA$_{1,6}$ as soon as $T \geq 50$. As soon as the size of the samples exceeds 25 for 8-bit adders, and 250 for 16-bit adders, simulating additional samples does not impact the estimated maximum value $\tilde{M}$. The adders ISA$_{2,2}$, 16-bit ISA$_{2,4}$, ISA$_{2,6}$ and CCBA$_{1,6}$ are converging towards the real value M_e when $T \geq 25$ for the

8-bit adder and $T \geq 250$ for the 16-bit adders. For the 8-bit adder $ISA_{2,4}$, the estimation is conservative since $M_e = 4$ and the estimate converges towards $\tilde{M} = 5$. This case is not problematic since the relative error of estimation is equal to 25%.

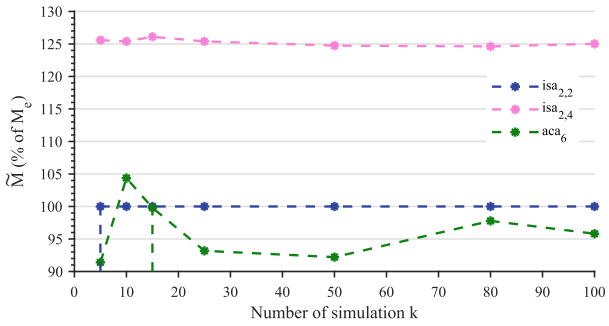
Nevertheless, to estimate correctly $\tilde{M}$ for the ACA_6 and ACA_8, the size of the simulated samples has to be really high compared to the other considered inexact operators, 72 and 870 respectively. Indeed, as shown in Figure 4a and 4b, for a fixed sample size $T = 100$, the estimation of $\tilde{M}$ for both inexact operators ACA_6 and ACA_8 needs a high in-range probability to reach the accurate value M_e.

Figures 6a and 6b represent the estimated value $\tilde{M}$ as a percentage of M_e depending on k. In this case, T samples ($T = 100$ for 8-bit adders, $T = 250$ for 16-bit adders) and their maximum is extracted. This operation is done a varying number of times k. For 8-bit adders, the $ISA_{2,2}$ and $ISA_{2,4}$ converge towards a value as soon as $k = 5$. As shown in Figures 5a, the ACA_6 would require more simulations to converge. Nevertheless, contrary to the impact of T on the quality of the estimation, in this case, a single adder ($ISA_{2,2}$) has converged towards the exact value M_e. This is due to the frequent generation of the maximum error value with this inexact adder. For the ACA_6, the estimated maximum $\tilde{M}$ is underestimated. Indeed, if the maximum extracted in the samples of size T is a local maximum, which induces parasite results when computing the Gumbel distribution. For the $ISA_{2,4}$, the estimated maximum $\tilde{M}$ is overestimated, with a relative error of estimation of 25%. For 16-bit adders, the estimates converge towards a value as soon as $k = 25$. The curve representing the $ISA_{2,4}$ is overlapping the curve representing the $CCBA_{1,6}$. In this case, for both adders $ISA_{2,4}$ and $CCBA_{1,6}$, only $k = 5$ simulations are required to correctly estimate the value $\tilde{M}$. For the $ISA_{2,6}$, the value $\tilde{M}$ is slightly overestimated. Finally, for the same reasons as for the 8-bit ACA_6, the ACA_8 underestimate the value $\tilde{M}$. However, as stated for the estimation of the $f$and mean ED, inexact operators with a large standard deviation renders circuits with poor interest.

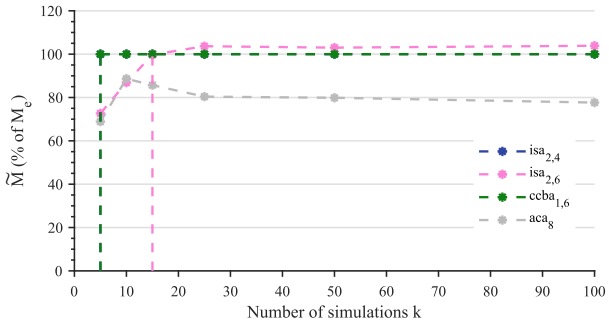
To draw a conclusion, to correctly estimate the maximum error distance for an inexact adder, the user will mainly modify the in-range probability p, allowing to be more or less conservative on the estimation without simulating additional samples, or the size of the samples T to ensure to converge towards global and not local maximum values to derive the Gumbel distribution.

VI. Conclusion

In this paper, we propose a characterization method of the approximation error induced by inexact arithmetic circuits, that exploits the statistical properties of the error. The benefits of the proposed method are demonstrated on different inexact arithmetic adders (ACA, ISA and CCBA) and the mean error distance, error rate and maximum error distance are estimated. From user-defined confidence requirements, the proposed method automatically adjusts the number of simulations



(a) 8-bit adders, $T = 100$.



(b) 16-bit adders, $T = 250$.

Fig. 6: Estimation of maximum ED as a percentage of M_e depending on k, $p = 90\%$. Vertical lines indicate $\tilde{M} = M_e$.

required by using statistical properties of the approximation error. Validated by its accurate estimation of error characteristics on 8 to 16-bit circuits, the proposed method has been proven coherence and consistency on larger bit-widths, with 32-bit circuits, where exhaustive simulation is not feasible. This experimental study has demonstrated that the proposed method outperforms naive stochastic BALL simulations with a fixed number of samples, either by converging towards a more accurate characterization, or by drastically reducing the amount of samples required for an accurate estimation, saving time and resources.

VII. Acknowledgments

This project has received funding from the French Agence Nationale de la Recherche under grant ANR-15-CE25-0015 (ARTEFaCT project).

References

[1] R. Airoldi, F. Campi, and J. Nurmi, "Approximate computing for complexity reduction in timing synchronization," *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, p. 155, 2014.

[2] J. Park, J. H. Choi, and K. Roy, "Dynamic bit-width adaptation in dct: an approach to trade off image quality and computation energy," *IEEE transactions on very large scale integration (VLSI) systems*, vol. 18, no. 5, 2010.

[3] H.-N. Nguyen, D. Menard, and O. Sentieys, "Dynamic precision scaling for low power wcdma receiver," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*. IEEE, 2009, pp. 205–208.

[4] A. Mercat, J. Bonnot, M. Pelcat, K. Desnos, W. Hamidouche, and D. Menard, "Smart search space reduction for approximate computing: A low energy hevc encoder case study," *JSA*, vol. 80, pp. 56–67, 2017.

[5] J. Bonnot, E. Nogues, and D. Menard, "New non-uniform segmentation technique for software function evaluation," in *Application-specific Systems, Architectures and Processors (ASAP), 2016 IEEE 27th International Conference on*. IEEE, 2016, pp. 131–138.

[6] S. Misailovic, S. Sidiroglou, H. Hoffmann, and M. Rinard, "Quality of service profiling," in *Software Engineering, 2010 ACM/IEEE 32nd International Conference on*, vol. 1. IEEE, 2010, pp. 25–34.

[7] K. Shi, D. Boland, and G. A. Constantinides, "Accuracy-performance tradeoffs on an FPGA through overclocking," in *IEEE FCCM, 21st Annual International Symposium on*, 2013, pp. 29–36.

[8] T. Liu and S.-L. Lu, "Performance improvement with circuit-level speculation," in *IEEE/ACM MICRO 2000*, 2000, pp. 348–355.

[9] A. K. Verma, P. Brisk, and P. Ienne, "Variable latency speculative addition: A new paradigm for arithmetic circuit design," in *Design, Automation and Test in Europe (DATE)*. IEEE, 2008, pp. 1250–1255.

[10] V. Camus, J. Schlachter, and C. Enz, "Energy-efficient inexact speculative adder with high performance and accuracy control," in *Circuits and Systems (ISCAS), IEEE International Symposium*, 2015.

[11] N. Zhu, W.-L. Goh, and K.-S. Yeo, "An enhanced low-power high-speed adder for error-tolerant application," in *Integrated Circuits (ISIC), 12th IEEE International Symposium on*, Dec. 2009, pp. 69–72.

[12] V. Camus, J. Schlachter, and C. Enz, "A low-power carry cut-back approximate adder with fixed-point implementation and floating-point precision," in *Design Automation Conference (DAC)*, 2016.

[13] C. Liu, J. Han, and F. Lombardi, "An analytical framework for evaluating the error characteristics of approximate adders," *IEEE Transactions on Computers (TC)*, vol. 64, no. 5, May 2015.

[14] Y. Wu, Y. Li, X. Ge, and W. Qian, "An accurate and efficient method to calculate the error statistics of block-based approximate adders," *arXiv preprint arXiv:1703.03522*, 2017.

[15] S. Mazahir, O. Hasan, R. Hafiz, M. Shafique, and J. Henkel, "Probabilistic error modeling for approximate adders," *IEEE Transactions on Computers (TC)*, vol. 66, no. 3, pp. 515–530, 2017.

[16] C. Yu and M. Ciesielski, "Analyzing imprecise adders using bdds–a case study," in *VLSI (ISVLSI), 2016 IEEE Computer Society Annual Symposium on*. IEEE, 2016, pp. 152–157.

[17] K. Du, P. n, and K. Mohanram, "High performance reliable variable latency carry select addition," in *IEEE DATE*, 2012, pp. 1257–1262.

[18] H. Jiang, C. Liu, L. Liu, F. Lombardi, and J. Han, "A review, classification and comparative evaluation of approximate arithmetic circuits," in *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 2017.

[19] V. Camus, M. Cacciotti, J. Schlachter, and C. Enz, "Design of approximate circuits by fabrication of false timing paths: The carry cut-back adder," in *IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS)*, 2018.

[20] J. Bonnot, V. Camus, K. Desnos, and D. Menard, "Cassis: Characterization with adaptive sample-size inferential statistics applied to inexact circuits," in *Signal Processing Conference (EUSIPCO), 2018 26th European*, 2018.

[21] J. Huang, J. Lach, and G. Robins, "A methodology for energy-quality tradeoff using imprecise hardware," in *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012, pp. 504–509.

[22] R. E. Moore, "Interval arithmetic and automatic error analysis in digital computing," Stanford Univ Calif Applied Mathematics And Statistics Labs, Tech. Rep., 1962.

[23] D. Sengupta, F. S. Snigdha, J. Hu, and S. S. Sapatnekar, "Saber: Selection of approximate bits for the design of error tolerant circuits," in *Proceedings of the 54th Annual Design Automation Conference 2017*. ACM, 2017, p. 72.

[24] ——, "An analytical approach for error pmf characterization in approximate circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2018.

[25] J. Bonnot, K. Desnos, M. Pelcat, and D. Menard, "A fast and fuzzy functional simulator of inexact arithmetic operators for approximate computing systems," in *Proceedings of the 2018 on Great Lakes Symposium on VLSI*. ACM, 2018, pp. 195–200.

[26] A. S. Roy and A. S. Dhar, "A novel approach for fast and accurate mean error distance computation in approximate adders," in *Circuits and Systems (ISCAS), 2018 IEEE International Symposium on*. IEEE, 2018, pp. 1–5.

[27] R. Lowry, "Concepts and applications of inferential statistics," 2014.

[28] G. Saporta, *Probabilités, analyse des données et statistique*. Editions Technip, 2006.

[29] R. R. Kinnison, *Applied extreme value statistics*. Battelle, 1985.

[30] R.-D. Reiss, M. Thomas, and R. Reiss, *Statistical analysis of extreme values*. Springer, 2007, vol. 2.

[31] E. Özer, A. P. Nisbet, and D. Gregg, "A stochastic bitwidth estimation technique for compact and low-power custom processors," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 7, no. 3, p. 34, 2008.

[32] A. Chapoutot, L.-S. Didier, and F. Villers, "Range estimation of floating-point variables in simulink models," in *Design and Architectures for Signal and Image Processing (DASIP), 2012 Conference on*. IEEE, 2012, pp. 1–8.