

MathematicS
MathS in A.
In Action

XAVIER JONSSON & JÉRÔME LELONG

Rare event simulation for electronic circuit design

Volume 11 (2022), p. 91-108.

<https://doi.org/10.5802/msia.19>

© Les auteurs, 2022.



Cet article est mis à disposition selon les termes de la licence CREATIVE COMMONS ATTRIBUTION 4.0.

<http://creativecommons.org/licenses/by/4.0/>



MathematicS In Action est membre du
Centre Mersenne pour l'édition scientifique ouverte

<http://www.centre-mersenne.org/>

e-ISSN : 2102-5754

Rare event simulation for electronic circuit design

XAVIER JONSSON *
JÉRÔME LELONG **

* Siemens Electronic Design Automation, 110 Rue Blaise Pascal, 38330 Montbonnot France

E-mail address: xavier_jonsson@siemens.com

** Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

E-mail address: jerome.lelong@univ-grenoble-alpes.fr.

Abstract

In this work, we propose an algorithm to simulate rare events for electronic circuit design. Our approach heavily relies on a smart use of importance sampling, which enables us to tackle probabilities of the magnitude 10^{-10} . Not only can we compute very rare default probability, but we can also compute the quantile associated to a given default probability and its expected shortfall. We show the impressive efficiency of the method on real circuits.

1. Introduction

We consider the problem of computing an expectation in the context of rare event simulation using a Monte Carlo approach. It is well known that, in such a framework, the accuracy of the Monte Carlo estimator is quite bad because too few samples lead to a non zero value of the involved characteristics. Assume we want to compute $\mathbb{E}[\varphi(X)]$, where $\varphi : \mathbb{R}^d \mapsto \mathbb{R}$ and X is a random vector with values in \mathbb{R}^d . There are basically three well-known tools to improve the accuracy of the crude Monte Carlo estimator: control variates, importance sampling and stratified sampling. As we will see in Section 2, importance sampling can be implemented in a quite fully automated way while stratification requires an a priori good knowledge of the realization space.

As an application problem, we will study the efficient and accurate determination of the yield of an integrated circuit, through electrical circuit level simulation, under stochastic constraints due to the manufacturing process. Several orders of magnitude of variance reduction can be achieved, for example millions or billions of MC runs necessary for very small probability might be reduced to a few hundreds or thousands runs. Section 5 presents quantitative results and speedup factors obtained with our importance sampling algorithm. The performance will be demonstrated on several integrated circuits from SRAM bitcells, to digital standard cells, analog IP blocks as well as on some small toy circuits.

We first give an overview of the semiconductor industry, and of the integrated circuits in particular. The notion of yield is explained and how this translates to very low probability of failure estimation for some kind of circuits. We will also introduce shortly the concept of analog circuit simulation used in our context.

1.1. EDA market and its segmentation

The past decades saw a considerable evolution of the manufacturing processes of integrated circuits (IC). The beginning of the twenty-first century experiences the same trend but with the emergence of connected objects, applications linked to mobility, onboard systems (especially in the automotive domain), the processing of mass data or medical devices.

For circuit designers, it is necessary to reduce the time of putting new products on the market, while satisfying both the demands for increased performance (consumption, reduced weights,

sizes and costs). To these constraints related to the end user, there are the problems of safety and security (aviation, motor vehicles), connectivity and adaptivity to more and more standards: WLAN (Wireless Local Area Network), Bluetooth, Wi-Fi (Wireless Fidelity). The result is an exponential growth in terms of complexity of operation and size (in number of transistors) for ICs. The time to market of a product is a crucial factor in its success. These systems of great complexity have to be designed and validated efficiently. The manufacturing costs being high, multiple attempts cannot be accepted.

The initial design of complex circuits is obtained in particular by the use of standard cell libraries and blocks of intellectual property (IP) developed by specialized teams or companies. Simulation constitutes the initial step after the specification. From our perspective, we will consider only a limited number of segments of this market: memory circuit simulation and standard cells. The simulation of complete digital blocks (digital SoCs) or large AMS/RF circuits cannot be addressed with our techniques due to the prohibitive CPU time required for each individual simulation. For our experiments, electrical simulations will be achieved with the commercial SPICE simulator Eldo[®]. SPICE-type simulators (Simulation Program with Integrated Circuit Emphasis) are the industry-standard way to verify circuit operation at the transistor level before manufacturing an integrated circuit.

1.2. Principles of analog circuit simulation

The automatic analysis routines of a SPICE simulation program determine the numerical solution of a mathematical representation of the circuit under validation.

After each circuit element has been modeled, the system of equations describing the complete circuit is determined and topological constraints are determined by the interconnection of the elements. The equations of the circuit are generally a nonlinear differential algebraic system of the form

$$\begin{aligned}\Phi(u(t), \dot{u}(t), t) &= 0 \\ u(0) &= u_0\end{aligned}\tag{1.1}$$

where Φ is a nonlinear vector function, $u(t)$ is the vector of circuit variables (e.g. node voltages and charges or branch currents), $\dot{u}(t)$ is the derivative of this vector with respect to time t . The circuit analysis consists in determining the solution of Eq. (1.1) for the following different simulation domains: nonlinear quiescent point calculation (DC), time-domain large-signal solution of nonlinear differential algebraic equations (TRAN), linear small-signal frequency domain (AC), noise analysis (NOISE). SPICE circuit simulation programs are presented in [7], [17] and [13].

The Monte Carlo simulation in SPICE programs is performed on multiple model evaluations with randomly selected model process variables. These variables are noted X in this paper. It is worth mentioning that the input parameters in models are in general the various design parameters (geometry of transistors), the process parameters (random fluctuations due to manufacturing process), and the environmental conditions (voltages conditions and temperatures). The electrical models of all major semiconductor manufacturers contain statistical models of process parameters. The reader may consult [6] for a detailed introduction to the origin of the random variability at the transistor level. The distribution associated with each component of the random variables X belongs to normal or uniform families (more complex distributions such as the Pearson's family may be encountered if higher moments such as skewness or kurtosis are given.) The process parameters defined in the statistical models are in general independent random variables. It means that we can restrict our work to simulation models with independent Gaussian random variables X . In the context of circuit simulation, the problems can live in very large dimension. The number of parameters are in general proportional to the number of transistors. In Section 5.2 some circuit examples will be presented where the dimension of X ranges from a few tens to several thousands components.

A circuit performance may be any electrical characteristic that is extracted from the simulation. It will be noted $H = h(X)$. In our context, the simulator acts as a black-box. We do not know anything a priori about the intrinsic nature of the relation between the measured output and the statistical parameters. Typical measures of performance may be the time delay for signal propagation across a circuit, the gain of an amplifier or the oscillation frequency of an oscillator.

1.3. Yield losses and probability of failure

The yield of an Integrated Circuit (IC) plays a major role in the manufacturing cost of an Application-Specific IC (ASIC). This concept typically enters in a manufacturing flow called “Yield Learning Flow”. In such a flow, the yield is defined to be the proportion of the manufactured circuits that satisfy both the design engineers and the process engineers specifications. The yield of a manufactured product should increase from the initial design to the large-volume production. Yield is therefore related to the cost of production. In nanoscale technologies, yield losses can be entirely caused by the uncontrolled and unavoidable factors like random dopant fluctuation and line edge roughness (see [1]). And each new technology node tend to make the problem more difficult, since the size of the transistors within ICs has shrunk exponentially, following Moore’s Law. Over the past decades, the typical MOS transistors channel lengths were once several micrometers (Intel 8086 CPU launched in 1978 was using a 3.2 μm process), but today ICs are incorporating MOS with channel lengths of tens of nanometers. This empirical law predicted that the number of transistors would double every 2 years due to shrinking transistor dimensions and additional technical improvements. As a consequence, some authors predicted that power consumption per unit area would remain constant, and that computer chip performance would roughly double every 18 months. The leading manufacturers continue to satisfy this empirical trend. In 2020, TSMC have already started mass producing 5 nanometer (nm) chips for companies including Apple, Huawei and Qualcomm. TSMC and Intel both have planned 2 nm products on their road-maps, with TSMC scheduling earliest production for 2023 and Intel for 2029. In May 2021, IBM announced the development of the first chip with 2nm technology.

Yield is interpreted as the probability of failure, rather than the probability of success. In some applications, it is much simpler to determine the yield of a single type of circuit which is repeated a very large number of times, say several millions. A typical and important example is Static Random-Access Memory (SRAM) manufacturing yield. The simple block in SRAM circuits is the bitcell. Its role is to store one elementary bit of data using latching circuitry (flip-flop). The probability of failure is defined here to be the probability of a wrong operation (read, write, or retention) occurring in an SRAM circuit. For such problems, the yield may be quantitatively appreciated by considering an explicit value for the acceptable quality limit (AQL). For example, an AQL for 1-Megabit SRAM bit-cell circuits could be 100 ppm. If production of one million such circuits is targeted, the manufacturing process cannot allow more than 100 defective bit-cells globally. This means that the probability of failure must be at most:

$$\frac{100\text{-defective bit-cells}}{10^6\text{-circuits} \times 10^6\text{-bits}} = 10^{-10}.$$

This relations shows that yield drops exponentially with the number of elementary blocks, and imposes extremely small values for where a large number of repetitions is used.

The practical problem of estimating failure probabilities for a large class of circuits is then translated into a general mathematical framework. The goal is to compute the expectation $\mathbb{E}[\varphi(X)]$ where the function φ writes as an indicator function $\varphi(x) = \mathbf{1}_{\{h(x) \geq \gamma\}}$ with $x \in \mathbb{R}^d$, where h is a circuit performance of interest (evaluated through SPICE simulations) and γ is an upper bound specified by the circuit designer. Note also the practical importance of the inverse problem, e.g. the estimation of the quantile $\gamma = \Gamma(p)$ given the probability of failure $0 < p < 1$

such that the quantile function Γ is defined as follows:

$$\Gamma(p) = \inf\{y \in \mathbb{R} : p \leq F_H(y)\} \quad (1.2)$$

in terms of the output cumulative distribution function F_H . The quantile problem is fundamental in verification flows where the circuit designers have to validate the design of complete library of standard cells. The circuits must satisfy prescribed specifications, for example the time delay must be less than some upper bound $\gamma \leq \gamma_u$. We will show, with the examples, how this quantile estimator can be obtained in a fully automatic way with our algorithm.

The rest of the paper unfolds as follows. In Section 2, we present importance sampling in a Gaussian framework and further enhance it for very rare events. Then, in Section 3, we apply the methodology developed before to the computation of a default probability and its conditional loss. In Section 4, we explain how importance sampling and stratified sampling can be coupled to improve the performance of our algorithm on some particularly demanding cases. Finally, Section 5 presents intensive numerical experiments on real electronic circuits.

2. Importance sampling

2.1. General framework

Assume the random vector X has a density function $h : \mathbb{R}^d \mapsto \mathbb{R}_+$. Consider a family of density functions $(f(\cdot; \theta))_{\theta \in \mathbb{R}^p}$ parametrized by a finite dimensional parameter $\theta \in \mathbb{R}^p$ satisfying $\text{supp}(h) \subset \text{supp}(f(\cdot; \theta))$ for all $\theta \in \mathbb{R}^p$. Then, we can write

$$\mathbb{E}[\varphi(X)] = \int_{\text{supp}(h)} \varphi(x)h(x)dx = \int_{\text{supp}(f(\cdot; \theta))} \varphi(x) \frac{h(x)}{f(x; \theta)} f(x; \theta)dx = \mathbb{E} \left[\varphi(Y) \frac{h(Y)}{f(Y; \theta)} \right] \quad (2.1)$$

where Y is a random vector with density function $f(\cdot; \theta)$.

From a practical point of view, one has to find the best density function h in order to maximize the accuracy of the Monte Carlo estimator. First, we need to make precise in which sense “best” has to be understood.

As the convergence of the Monte Carlo estimator is governed by the central limit theorem, it is quite natural to try to find the density function $f(\cdot, \theta)$ which minimizes the variance of the estimator

$$\text{Var} \left(\varphi(Y) \frac{h(Y)}{f(Y; \theta)} \right) = \mathbb{E} \left[\varphi(Y)^2 \left(\frac{h(Y)}{f(Y; \theta)} \right)^2 \right] - (\mathbb{E}[\varphi(X)])^2.$$

Only the second moment depends on the density function h and moreover using Eq. (2.1) again, we obtain

$$\mathbb{E} \left[\varphi(Y)^2 \left(\frac{h(Y)}{f(Y; \theta)} \right)^2 \right] = \mathbb{E} \left[\varphi(X)^2 \frac{h(X)}{f(X; \theta)} \right].$$

If the optimization problem $\min_{\theta} \mathbb{E} \left[\varphi(X)^2 \frac{h(X)}{f(X; \theta)} \right]$ is well-posed (strongly convex for instance), we can solve this problem directly.

2.2. The Gaussian case

Assume the density function h stands for the standard Gaussian density function in \mathbb{R}^d . Consider a family of Gaussian distributions with values in \mathbb{R}^d parametrized by their expectations

$$f(x; \theta) = \frac{1}{(2\pi)^{d/2}} e^{-|x-\theta|^2/2} \quad \text{for } \theta \in \mathbb{R}^d.$$

Then, the importance sampling formula Eq. (2.1) writes as

$$\mathbb{E}[\varphi(X)] = \mathbb{E} \left[\varphi(Y) \frac{h(Y)}{f(Y; \theta)} \right] = \mathbb{E} \left[\varphi(X + \theta) e^{-X \cdot \theta - |\theta|^2/2} \right]. \quad (2.2)$$

The second moment can be also written in a simplified way

$$v(\theta) = \mathbb{E} \left[\left(\varphi(Y) \frac{h(Y)}{f(Y; \theta)} \right)^2 \right] = \mathbb{E} \left[\varphi(X)^2 e^{-X \cdot \theta + |\theta|^2 / 2} \right] \quad (2.3)$$

where the last equality comes from the application of Eq. (2.2).

Proposition 2.1. *Assume that for all $\theta \in \mathbb{R}^d$, $\mathbb{E}[\varphi(X)^2 e^{-\theta \cdot X}] < +\infty$ and that $\mathbb{P}(\varphi(X) \neq 0) > 0$. Then, the function v is infinitely differentiable and strongly convex. Moreover, we can exchange derivation and expectation.*

From this proposition, we can write

$$\nabla v(\theta) = \mathbb{E} \left[(\theta - X) \varphi(X)^2 e^{-X \cdot \theta + |\theta|^2 / 2} \right] \quad (2.4)$$

$$\nabla^2 v(\theta) = \mathbb{E} \left[(I_d + (\theta - X)'(\theta - X)) \varphi(X)^2 e^{-X \cdot \theta + |\theta|^2 / 2} \right]. \quad (2.5)$$

Hence, the value θ^* minimizing the variance is uniquely defined by $\nabla v(\theta^*) = 0$. As the Hessian matrix is tractable at roughly the same cost as the gradient itself, one can implement a Newton algorithm with optimal descent direction to minimize v .

2.3. Existing approaches in EDA

Importance Sampling is used with great success for many decades in multiple scientific fields, including structural safety engineering, aeronautics, communication networks, finance, nuclear safety. The world of EDA is actually rather behind, comparatively. IS with Gaussian mixtures has been used noticeably in the context of SRAM simulation (see [10]).

The mathematics of the basic idea are rather straightforward, although some users have difficulties with this notion of using a modified distribution. The performance of any IS strategy heavily depends on the chosen instrumental distribution, and implementation must be very thorough. However, incorrect choice of the instrumental distribution (as demonstrated in [8]) or bad implementation can lead to poor performance, and even to an increase of the variance.

However, at least compared to the field of mechanical engineering, the relation between the process parameters and the output is more often mildly non-linear than not. Given the difficulty, many approaches that yield only an approximation of the probability or quantile have been tried. Machine Learning techniques can help as well. None of these approaches can avoid the curse of dimensionality (see for example [14], [16]), and the validity of the approximated confidence intervals, when they can be estimated, is questionable.

2.4. Algorithm for the mean shift approach in the Gaussian case

We consider the framework detailed in Section 2.2 relying on the variance criterion to determine the optimal parameter θ^* uniquely defined by $\nabla v(\theta^*) = 0$ thanks to Proposition 2.1. The idea of the algorithm is to replace all the expectations by empirical means and then deterministic optimization techniques.

Algorithm 1 (Adaptive Sample Average Monte Carlo).

- (1) Draw X_1, \dots, X_n n i.i.d. samples following the normal distribution
- (2) Compute the minimizer θ_n of

$$v_n(\theta) = \frac{1}{n} \sum_{i=1}^n \varphi^2(X_i) e^{-\theta \cdot X_i + \frac{|\theta|^2}{2}}$$

by solving $\nabla v_n(\theta) = 0$ using Newton's method.

- (3) Draw $\bar{X}_1, \dots, \bar{X}_m$ m i.i.d. samples following the normal distribution and independent of (X_1, \dots, X_m) .
- (4) Compute the expectation $\mathbb{E}(\varphi(X))$ by Monte Carlo using this new sample

$$M_m(\theta_n) = \frac{1}{m} \sum_{i=1}^m \varphi(\bar{X}_i + \theta_n) e^{-\theta_n \cdot \bar{X}_i - \frac{|\theta_n|^2}{2}}.$$

The number of samples n used for solving the optimization problem may differ from the number of samples m used in the Monte Carlo procedure. The way to balance the number of samples between these two steps has to be further studied. For n large enough, v_n inherits its strong convexity and differentiability from the ones of v .

The convergence of the Sample Average approximation θ_n is known to converge to θ^* and to satisfy a central limit theorem under integrability conditions. We introduce the following condition

$$\exists \delta > 0, \mathbb{E}[\varphi^{4+\delta}(X)] < +\infty. \quad (2.6)$$

Proposition 2.2. *Assume that for all $\theta \in \mathbb{R}^d$, $\mathbb{E}[\varphi(X)^2 e^{-\theta \cdot X}] < +\infty$ and that $\mathbb{P}(\varphi(X) \neq 0) > 0$. Then, θ_n and $v_n(\theta_n)$ converge a.s. to θ^* and $v(\theta^*)$ as $n \rightarrow \infty$. If, moreover, (2.6) holds, then $\sqrt{n}(\theta_n - \theta^*) \xrightarrow{\mathcal{L}} \mathcal{N}_d(0, \Gamma)$ where*

$$\Gamma = [\nabla^2 v(\theta^*)]^{-1} \text{Cov} \left((\theta^* - X) \varphi^2(X) e^{-\theta^* \cdot X + \frac{|\theta^*|^2}{2}} \right) [\nabla^2 v(\theta^*)]^{-1}.$$

We refer to [9] and [15] for a proof of this result.

The convergence of the sequence $M_m(\theta_n)_{m,n}$ is governed by the following theorem. We can state the following result.

Proposition 2.3. *Let $n(m)$ be an increasing function of m tending to infinity with m . We assume the Assumptions of Proposition 2.2. The following results hold.*

- (i) *The estimator $M_m(\theta_{n(m)})$ is an unbiased and $M_m(\theta_{n(m)}) \rightarrow \mathbb{E}[\varphi(X)]$ a.s.;*
- (ii) *If the function φ is almost everywhere continuous and for some $\eta > 0$ and some compact neighborhood \mathcal{V} of θ^* , $\sup_{\theta \in \mathcal{V}} \mathbb{E}[|\varphi(X + \theta)|^{2+\eta}] < +\infty$, then $\sqrt{m}(M_m(\theta_{n(m)}) - \mathbb{E}[\varphi(X)]) \xrightarrow{\mathcal{L}} \mathcal{N}_d(0, (\sigma^*)^2)$ where*

$$(\sigma^*)^2 = \text{Var} \left(\varphi(X + \theta^*) e^{-\theta^* \cdot X - \frac{|\theta^*|^2}{2}} \right) = v(\theta^*) - \mathbb{E}[\varphi(X)]^2.$$

Proof. The proof of (i) can be found in [2].

We only give the guidelines to prove (ii).

$$\sqrt{m}(M_m(\theta_{n(m)}) - \mathbb{E}[\varphi(X)]) = \sqrt{m}(M_m(\theta_*) - \mathbb{E}[\varphi(X)]) + \sqrt{m}(M_m(\theta_{n(m)}) - M_m(\theta_*))$$

From the standard central limit theorem, $\sqrt{m}(M_m(\theta_*) - \mathbb{E}[\varphi(X)]) \xrightarrow[n \rightarrow +\infty]{\mathcal{L}} \mathcal{N}(0, v(\theta_*) - \mathbb{E}[\varphi(X)]^2)$. Therefore, it is sufficient to prove that $\sqrt{m}(M_m(\theta_{n(m)}) - M_m(\theta_*)) \xrightarrow[n \rightarrow +\infty]{\mathbb{P}} 0$. Let $\varepsilon > 0$.

$$\begin{aligned} & \mathbb{P} \left(\sqrt{n} |M_n(\theta_{n(m)}) - M_n(\theta_*)| > \varepsilon \right) \\ & \leq \mathbb{P}(n(m)^{1/4} |(\theta_{n(m)}) - (\theta_*)| > 1) + \frac{n}{\varepsilon^2} \mathbb{E} \left[|M_n(\theta_{n(m)}) - M_n(\theta_*)|^2 \mathbf{1}_{\{|\theta_{n(m)} - \theta_*| \leq n(m)^{-1/4}\}} \right]. \end{aligned} \quad (2.7)$$

We deduce from Proposition 2.2, that $\mathbb{P}(n(m)^{1/4} |\theta_{n(m)} - \theta_*| > 1) \rightarrow 0$. We introduce

$$Q(\theta) = e^{-\theta \cdot \bar{X}^1 - \frac{|\theta|^2}{2}}.$$

Conditionally on $\theta_{n(m)}$, $M_n(\theta_{n(m)}) - M_n(\theta_*)$ is a sum of i.i.d. centered random variables. Then, it is sufficient to monitor differences as

$$|\varphi(\bar{X}^1 + \theta_*)Q(\theta_*) - \varphi(\bar{X}^1 + \theta_{n(m)})Q(\theta_{n(m)})|^2$$

on the set $\{|\theta_{n(m)} - \theta_*| \leq n(m)^{-1/4}\}$, which is a subset of \mathcal{V} for large enough n . As φ is almost everywhere continuous, we deduce that the above difference a.s. goes to 0 when m tends to infinity. Let $\delta < \eta$, the conditional independence combined with Hölder's inequality yield that for large enough m

$$\begin{aligned} \mathbb{E} \left[|\varphi(\bar{X}^1 + \theta_{n(m)})Q(\theta_{n(m)})|^{2+\delta} \mathbf{1}_{\{\theta_{n(m)} \in \mathcal{V}\}} \right] &\leq \sup_{\theta \in \mathcal{V}} \mathbb{E} \left[|\varphi(\bar{X}^1 + \theta)Q(\theta)|^{2+\delta} \right] \\ &\leq \sup_{\theta \in \mathcal{V}} \mathbb{E} \left[|\varphi(\bar{X}^1 + \theta)|^{2+\eta} \right]^{\frac{2+\delta}{2+\eta}} \mathbb{E} \left[Q(\theta)^{\frac{(2+\delta)(2+\eta)}{\eta-\delta}} \right]^{\frac{\eta-\delta}{2+\eta}}. \end{aligned}$$

From the local boundedness of $\theta \mapsto \mathbb{E}[|\varphi(X + \theta)|^{2+\eta}]$, we deduce that

$$\sup_n \mathbb{E} \left[|\varphi(\bar{X}^1 + \theta_{n(m)})Q(\theta_{n(m)})|^{2(1+\delta)} \right] < \infty,$$

which proves the uniform integrability of the family $(|\varphi(\bar{X}^1 + \theta_{n(m)})Q(\theta_{n(m)})|^2)_n$. Then, we deduce that the second term in (2.7) tends to zero. \square

An immediate consequence of this result is that the accuracy of a Monte Carlo algorithm based on $M_n(\theta_{n(m)})$ can be monitored using the standard theory of confidence intervals.

A key issue from a practical and computational point of view is to fix $n(m)$ for a given value of m . As $n(m)$ has not impact on the convergence rate but only on how the empirical variance is close to the optimal variance. Hence, we are actually more interested in the convergence of $v_n(\theta_n)$ to $v(\theta^*)$ than in the one of θ_n itself. Monitoring the convergence of $v_n(\theta_n)$ would give us a hint on how to choose $n(m)$.

Proposition 2.4. *Under the assumptions of Proposition 2.2, $\sqrt{n}(v_n(\theta_n) - v(\theta^*)) \xrightarrow{\mathcal{L}} \mathcal{N}(0, (\gamma^*)^2)$ where*

$$(\gamma^*)^2 = \text{Var} \left(\varphi^2(X) e^{-\theta^* \cdot X + \frac{|\theta^*|^2}{2}} \right).$$

Proof.

$$\sqrt{n}(v_n(\theta_n) - v(\theta^*)) = \sqrt{n}(v_n(\theta_n) - v_n(\theta^*)) + \sqrt{n}(v_n(\theta^*) - v(\theta^*))$$

By the standard central limit theorem, $\sqrt{n}(v_n(\theta^*) - v(\theta^*))$ converges in distribution to a normal random variable with mean 0 and variance $(\gamma^*)^2$.

To handle the term $\sqrt{n}(v_n(\theta_n) - v_n(\theta^*))$, we use the following Taylor expansion

$$\sqrt{n}(v_n(\theta_n) - v_n(\theta^*)) = \sqrt{n} \nabla v_n(\tilde{\theta}_n)(\theta^* - \theta_n)$$

where $\tilde{\theta}_n$ is a point on the segment joining θ^* and θ_n . Since ∇v_n converges locally uniformly to the continuous function ∇v (see [9]), $\nabla v_n(\tilde{\theta}_n)$ converges a.s. to $\nabla v(\theta^*) = 0$. Then, we deduce from the convergence in distribution of $\sqrt{n}(\theta_n - \theta^*)$ (see Proposition 2.2) that $\sqrt{n}(v_n(\theta_n) - v_n(\theta^*))$ goes to zero in probability. \square

As usual, if we consider a convergent estimator γ_n^2 of $(\gamma^*)^2$, we obtain $\frac{\sqrt{n}}{\gamma_n}(v_n(\theta_n) - v(\theta^*)) \xrightarrow{\mathcal{L}} \mathcal{N}(0, 1)$, which can be used in practice to determine the magnitude of n . A natural estimator of γ_n can be computed online within the minimization algorithm

$$\gamma_n^2 = \frac{1}{n} \sum_{i=1}^n \varphi^4(X_i) e^{-2\theta_n \cdot X_i + |\theta_n|^2} - v_n(\theta_n)^2.$$

The convergence of γ_n^2 to $(\gamma^*)^2$ ensues from the locally uniform strong law of large numbers for the sequence of random functions $\theta \mapsto \varphi^4(X_i) e^{-2\theta \cdot X_i + |\theta|^2}$, see [15, Lemma A1] or [12, Corollary 7.10, p. 189].

2.5. Using importance sampling for *very* rare event simulation

Assume the function φ writes as an indicator function $\varphi(x) = \mathbf{1}_{\{h(x) \geq \gamma\}}$ with $x \in \mathbb{R}^d$ where γ makes the event almost never occur in practice, $\mathbb{P}(h(X) \geq \gamma)$ is much smaller than one over the number of samples. Then, the empirical updating formulae are not well defined and we will never manage to compute the mixture parameters straightaway. In such a situation, [11] suggest to use a ladder with finite number of levels to reach level γ . Consider some levels $\gamma_1, \dots, \gamma_K = \gamma$ where K is a priori unknown but computed online. In this scheme, the samples used at step k are drawn according to the importance sampling distribution obtained from step $k - 1$, which ensures that the level γ_k is never too rare. This is only meaningful if the value returned by the simulator, represented by the function h , takes a continuous range of values or at least is piecewise constant with sufficiently many different values, but it cannot be a binary response.

In this section, we only focus the mean shift approach. The multilevel approach is based on a step by step update of the importance density thus ensuring the targeted event is never too rare. At step k , the samples are generated according to the importance sampling density computed at step $k - 1$. Before stating the algorithm, we need to extend Eq. (2.2) to a wider setting.

Consider two \mathbb{R}^d -valued normal random vectors Y and Z with densities $f(\cdot; \theta_Y)$ and $f(\cdot; \theta_Z)$ respectively for some $\theta_Y, \theta_Z \in \mathbb{R}^d$. Then, from Eq. (2.1), we get

$$\mathbb{E}[\varphi(Y)] = \mathbb{E} \left[\varphi(Z) \frac{f(Z; \theta_Y)}{f(Z; \theta_Z)} \right] \quad (2.8)$$

$$\mathbb{E} \left[\left(\varphi(Z) \frac{f(Z; \theta_Y)}{f(Z; \theta_Z)} \right)^2 \right] = \mathbb{E} \left[\varphi(Y)^2 \frac{f(Y; \theta_Y)}{f(Y; \theta_Z)} \right]. \quad (2.9)$$

These two formulae can be made more explicit by expanding the density functions

$$\begin{aligned} \mathbb{E}[\varphi(Y)] &= \mathbb{E} \left[\varphi(Z) e^{-(\theta_Z - \theta_Y) \cdot Z + \frac{1}{2}(|\theta_Z|^2 - |\theta_Y|^2)} \right] \\ \mathbb{E} \left[\left(\varphi(Z) \frac{f(Z; \theta_Y)}{f(Z; \theta_Z)} \right)^2 \right] &= \mathbb{E} \left[\varphi(Y)^2 e^{-(\theta_Z - \theta_Y) \cdot Y + \frac{1}{2}(|\theta_Z|^2 - |\theta_Y|^2)} \right]. \end{aligned}$$

Let \bar{K} be the maximum number of levels allowed in the algorithm. For $k = 0, \dots, \bar{K}$, let n_k be the number of samples used at step k of the algorithm. The Algorithm can be written as

Algorithm 2 (Multilevel importance sampling).

Set $\theta_n^{(0)} = 0$ in \mathbb{R}^d and $k = 0$.

- (1) Draw some iid samples $X_1^{(k+1)}, \dots, X_{n_{k+1}}^{(k+1)}$ according to the distribution of $X^{(k)} \sim f(\cdot; \theta_{n_k}^{(k)})$. Conditionally on $\theta_{n_k}^{(k)}, X_1^{(k+1)}, \dots, X_{n_{k+1}}^{(k+1)}$ and $X_1^{(k)}, \dots, X_{n_k}^{(k)}$ are independent.
- (2) Using the previous samples, compute the quantile of order ρ of $h(X^{(k)})$ and set it to γ_{k+1} provided it is less than γ ; otherwise set $\gamma_{k+1} = \gamma$.
- (3) Compute the variance criterion

$$v_{n_{k+1}}^{(k+1)}(\theta) = \frac{1}{n} \sum_{j=1}^{n_{k+1}} \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} e^{-(\theta - \theta_{n_k}^{(k)}) \cdot X_j^{(k+1)} + (|\theta|^2 - |\theta_{n_k}^{(k)}|^2)/2}.$$

Then, $\theta_{n_{k+1}}^{(k+1)} = \arg \min_{\theta} v_{n_{k+1}}^{(k+1)}(\theta)$.

(4) If $\gamma_k < \gamma$ and $k < \bar{K}$, return to step 1 and proceed with $k \leftarrow k + 1$.

(5) Draw some new samples $\bar{X}_1, \dots, \bar{X}_m$ following the standard normal distribution and independent of the past. Approximate $\mathbb{E}[\varphi(X)]$ by

$$S_{m,n} = \frac{1}{m} \sum_{j=1}^m \mathbf{1}_{\{h(\bar{X}_j + \theta^{(k+1)}) \geq \gamma\}} e^{-\frac{1}{2} |\theta_n^{(k+1)}|^2 - \theta_n^{(k+1)} \cdot X_j}.$$

Remark 2.5. In practice, we use $\rho = 0.8$. The quantile of order of ρ of $h(X^{(k)})$ is obtained by sorting the samples $h(X_1^{(k+1)}), \dots, h(X_{n_{k+1}}^{(k+1)})$ from lowest to highest and taking in this ordered set the $\lceil \rho * n_{k+1} \rceil$ -th value.

When the function h is symmetric, $h(x) = h(-x)$ for $x \in \mathbb{R}^d$, we can show that the optimal value of θ is zero. Therefore, the sequence $\theta_n^{(k)}$ produced by the algorithm is constant and so is the sequence γ_k . The maximum number of levels \bar{K} is used to ensure that the algorithm will eventually stop and will not loop infinitely. The algorithm may also fail to produce an increasing sequence $(\gamma_k)_k$ when the function h has several modes with equivalent heights. In all such hedge cases, the upper bound \bar{K} will ensure the algorithm returns even though with no improvement.

Using intermediate levels has no theoretical impact on the convergence of the algorithm, so it is enough to study the theoretical properties of the last level: $\gamma_{k+1} = \gamma$. Though, it is a practically essential trick to circumvent the difficulty of sampling a rare event. If the levels $(\gamma_k)_k$ are chosen so that the events $\{h(X_j^{(k+1)}) \geq \gamma_k\}$ are never too rare, then a fairly small number of samples n can be used. We know from Proposition 2.3 that the convergence rate of $S_{m,n}$ is monitored by \sqrt{m} so the computational effort should be put on m .

Computation of $\theta_{n_{k+1}}^{(k+1)}$. The vector $\theta_{n_{k+1}}^{(k+1)}$ is defined as the solution of a minimization problem, which is known to be strongly convex. Hence, the solution can be computed using the Newton algorithm with optimal descent direction given the inverse of the Hessian matrix.

$$\begin{aligned} \nabla v_n^{(k+1)}(\theta) &= \frac{1}{n} \sum_{i=1}^n \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} (\theta - X_j^{(k+1)}) e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X_j^{(k+1)} + (|\theta|^2 - |\theta_{n_{k+1}}^{(k)}|^2)/2} \\ \nabla^2 v_n^{(k+1)}(\theta) &= \frac{1}{n} \sum_{i=1}^n \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} \left(I + (\theta - X_j^{(k+1)})(\theta - X_j^{(k+1)})' \right) \\ &\quad \times e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X_j^{(k+1)} + (|\theta|^2 - |\theta_{n_{k+1}}^{(k)}|^2)/2}. \end{aligned}$$

When n goes to infinity, the Hessian matrix $\nabla^2 v_n^{(k+1)}(\theta)$ converges a.s. to

$$\mathbb{E} \left[\mathbf{1}_{\{h(X^{(k+1)}) \geq \gamma_{k+1}\}} \left(I + (\theta - X^{(k+1)})(\theta - X^{(k+1)})' \right) e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X^{(k+1)} + (|\theta|^2 - |\theta_{n_{k+1}}^{(k)}|^2)/2} \left| \theta_{n_{k+1}}^{(k)} \right. \right]$$

whose smallest eigenvalue is larger than

$$\begin{aligned} &\mathbb{E} \left[\mathbf{1}_{\{h(X^{(k+1)}) \geq \gamma_{k+1}\}} e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X^{(k+1)} + (|\theta|^2 - |\theta_{n_{k+1}}^{(k)}|^2)/2} \left| \theta_{n_{k+1}}^{(k)} \right. \right] \\ &= \mathbb{E} \left[\mathbf{1}_{\{h(X^{(k+1)}) \geq \gamma_{k+1}\}} e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X^{(k+1)}} \left| \theta_{n_{k+1}}^{(k)} \right. \right] \mathbb{E} \left[e^{(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X^{(k+1)}} \left| \theta_{n_{k+1}}^{(k)} \right. \right] \\ &\geq \mathbb{E} \left[\mathbf{1}_{\{h(X^{(k+1)}) \geq \gamma_{k+1}\}} \right] \end{aligned}$$

since

$$\mathbb{E} \left[e^{(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X^{(k+1)}} \left| \theta_{n_{k+1}}^{(k)} \right. \right] = e^{\frac{1}{2} |\theta - \theta_{n_{k+1}}^{(k)}|^2 + (\theta - \theta_{n_{k+1}}^{(k)}) \cdot \theta_{n_{k+1}}^{(k)}} = e^{(|\theta|^2 - |\theta_{n_{k+1}}^{(k)}|^2)/2}.$$

Depending on the choice of the levels $(\gamma_k)_k$, the lower bound $\mathbb{E} \left[\mathbf{1}_{\{h(X^{(k+1)}) \geq \gamma_{k+1}\}} \right]$ can become quite small, which may lead to a badly performing Newton algorithm as the smallest eigenvalue of the Hessian matrix measures the convexity of the function and the less convex the function is, the slower the Newton algorithm converges. To circumvent this problem, we suggest to consider an alternative characterization of $\theta_n^{(k+1)}$. The equation $\nabla v_n^{(k+1)}(\theta_n^{(k+1)}) = 0$ can be rewritten

$$\theta_n^{(k+1)} - \frac{\frac{1}{n} \sum_{j=1}^n \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} X_j^{(k+1)} e^{-(\theta_n^{(k+1)} - \theta_{n_{k+1}}^{(k)}) \cdot X_j^{(k+1)}}}{\frac{1}{n} \sum_{j=1}^n \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} e^{-(\theta_n^{(k+1)} - \theta_{n_{k+1}}^{(k)}) \cdot X_j^{(k+1)}}} = 0.$$

If we introduce the function $u_n^{(k+1)}$ defined by

$$u_n^{(k+1)}(\theta) = \frac{1}{2} |\theta|^2 + \log \left(\frac{1}{n} \sum_{j=1}^n \mathbf{1}_{\{h(X_j^{(k+1)}) \geq \gamma_{k+1}\}} e^{-(\theta - \theta_{n_{k+1}}^{(k)}) \cdot X_j^{(k+1)}} \right)$$

then $\theta_n^{(k+1)}$ can be seen as the root of $\nabla u_n^{(k+1)}(\theta)$ and we can prove that the Hessian matrix $\nabla^2 u_n^{(k+1)}(\theta)$ is uniformly bounded from below by I_d . We refer the reader to [9] for more details on this subtle transformation of the original problem to ensure a minimum convexity.

Note that in practice, one needs to solve the linear systems $\nabla^2 u_n^{(k+1)}(\theta) \delta = -\nabla u_n^{(k+1)}(\theta)$ at each iteration of the Newton algorithm. The Hessian matrix being a dense matrix, it is more interesting to use an iterative algorithm such as the conjugate gradient algorithm.

The number d of input variables may also be high. This dimension is directly linked to number of electronic devices on a given circuit and specifically to the number of transistors. However this dimension does not impact the theoretical convergence properties of the approach. As this dimension grows, it becomes difficult to make sufficient progress during the multilevel algorithm. When the simulation budget is limited, we experienced non convergence (and therefore failure) of the Importance Sampling algorithm if the number of variables d is closed to 10^5 variables. A simple idea is to look for the importance sampling parameter in a subspace $\{A\vartheta : \vartheta \in \mathbb{R}^{d'}\}$ where $A \in \mathbb{R}^{d \times d'}$ is matrix with rank $d' \leq d$. In our situation, the matrix A needs to be identified automatically, and is a submatrix of the identity matrix after removing its $d - d'$ columns corresponding to the unimportant variables. Our approach falls into the category of sequential deterministic methods. It produces the same subset on a given problem every time, by starting with a single solution (a variable subset) and iteratively adding or removing dimensions until some termination criterion is met. This approach is clearly suboptimal, there is no guarantee for finding the optimal subset solution (because we do not examine all possible subsets). We made some tests to validate the approach with large scale problems see Section 5.1.

3. Computing a default probability and its expected shortfall

In this section, we assume that the function φ is given by $\varphi(x) = \mathbf{1}_{\{h(x) \geq \gamma\}}$ where $h : \mathbb{R}^d \rightarrow \mathbb{R}$ and $a \in \mathbb{R}$. We consider the problem of both estimating $\mathbb{P}(h(X) \geq \gamma)$ and $\mathbb{E}[h(X) | h(X) \geq \gamma]$, where X is a standard normal random vector in \mathbb{R}^d . The quantity $\mathbb{E}[h(X) | h(X) \geq \gamma]$ is the conditional loss beyond the level γ , we denote it by $\text{CL}(h(X))$. When γ is the value at risk at level α for some $\alpha \in (0, 1)$, which means that γ is the $(1 - \alpha)$ quantile of $h(X)$, the quantity $\text{CL}(h(X))$ is the conditional value at risk or expected shortfall at level α . The conditional loss can be seen as a mesure of the dispersion of the distribution tail. In particular, a CL close to γ means that the fail circuits are very concentrated and justifies the practitioners' habit to pick a single fail circuit at random when trying to analyse the reasons of a failure. We will give an practical example in Section 5.2.2.

The default probability is computed by applying the importance sampling approach developed in Section 2 to its standard Monte Carlo estimator.

Note that $\text{CL}(h(X))$ can be written

$$\text{CL}(h(X)) = \frac{\mathbb{E}[h(X)\mathbf{1}_{\{h(X) > \gamma\}}]}{\mathbb{P}(h(X) > \gamma)}.$$

Based on Section 2.2, it is natural to consider the importance sampling version of the CL for $\theta \in \mathbb{R}^d$

$$\text{CL}(h(X)) = \frac{\mathbb{E}\left[h(X + \theta)\mathbf{1}_{\{h(X+\theta) > \gamma\}} e^{-\theta \cdot X - |\theta|^2/2}\right]}{\mathbb{E}\left[\mathbf{1}_{\{h(X+\theta) > \gamma\}} e^{-\theta \cdot X - |\theta|^2/2}\right]}.$$
 (3.1)

Note that we could afford different importance sampling parameters for the numerator and the denominator, each being the solution of an optimization problem. Since we are computing both $\text{CL}(h(X))$ and $\mathbb{P}(h(X) \geq \gamma)$, we will anyway have to compute the optimal θ for the denominator. Considering the computational cost of solving such an optimization problem, we believe using two different importance sampling parameters in (3.1) is not worth it.

Then, a standard estimator writes as

$$\text{CL}_n(h(X)) = \frac{\sum_{i=1}^n h(X_i + \theta)\mathbf{1}_{\{h(X_i+\theta) > \gamma\}} e^{-\theta \cdot X_i - |\theta|^2/2}}{\sum_{i=1}^n \mathbf{1}_{\{h(X_i+\theta) > \gamma\}} e^{-\theta \cdot X_i - |\theta|^2/2}}.$$
 (3.2)

We use the convention $\frac{0}{0} = 0$. Although the estimations of $\mathbb{P}(h(X) \geq \gamma)$ and $\text{CL}(h(X))$ are presented one after the other, in practice they are computed simultaneously using the same runs of the simulator. Thus, the two estimators are actually obtained for the cost of one.

It is clear from the strong law of large numbers that $\text{CL}_n(h(X)) \rightarrow \text{CL}(h(X))$ a.s. when $n \rightarrow \infty$. Due the random nature of an estimator, it has to come with a confidence interval, which relies on a central limit theorem.

Proposition 3.1. *The following convergence holds*

$$\sqrt{n}(\text{CL}_n(h(X)) - \text{CL}(h(X))) \xrightarrow[n \rightarrow \infty]{\mathcal{L}} \mathcal{N}(0, \sigma^2)$$
 (3.3)

with

$$\sigma^2 = \frac{\mathbb{E}\left[\left(h(X) - \mathbb{E}[h(X)|h(X) > \gamma]\right)^2 \mathbf{1}_{\{h(X) > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2}\right]}{\mathbb{P}(h(X) > \gamma)^2}.$$
 (3.4)

Proof. We introduce the notation $Y_i^\theta = h(X_i + \theta)$, $\mathcal{E}_i^\theta = e^{-\theta \cdot X_i - |\theta|^2/2}$. For $\theta = 0$, we simply write $Y = h(X)$.

$$\begin{aligned} \sqrt{n}(\text{CL}_n(Y) - \text{CL}(Y)) &= \sqrt{n} \left(\frac{\sum_{i=1}^n Y_i^\theta \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta}{\sum_{i=1}^n \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta} - \frac{\mathbb{E}[Y^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta]}{\mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta]} \right) \\ &= \frac{\frac{1}{\sqrt{n}} \sum_{i=1}^n \left(Y_i^\theta \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta \mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] - \mathbb{E}[Y^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta \right)}{\frac{1}{n} \sum_{i=1}^n \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta \mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta]} \end{aligned}$$

From the strong law of large numbers, the denominator converges a.s. to $\mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta]^2 = \mathbb{P}(Y > \gamma)^2$. The random variables $\left(Y_i^\theta \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta \mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] - \mathbb{E}[Y^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta \right)_i$ are iid and centered. Hence, we deduce from the central limit theorem that the numerator

converges in distribution to a normal distribution with mean 0 and variance

$$\begin{aligned} & \mathbb{E} \left[\left(Y^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta \mathbb{E}[\mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] - \mathbb{E}[Y^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta] \mathbf{1}_{\{Y^\theta > \gamma\}} \mathcal{E}^\theta \right)^2 \right] \\ &= \mathbb{E} \left[\left(Y \mathbb{P}(Y > \gamma) - \mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}}] \right)^2 \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2} \right] \end{aligned}$$

where the exponential weight comes from (2.3). Then, by Slutsky's lemma, $\sqrt{n}(\text{CL}_n(Y) - \text{CL}(Y))$ converges in distribution to random normal variable with mean 0 and variance

$$\begin{aligned} & \frac{\mathbb{E} \left[\left(Y \mathbb{P}(Y > \gamma) - \mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}}] \right)^2 \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2} \right]}{\mathbb{P}(Y > \gamma)^4} \\ &= \frac{\mathbb{E} \left[(Y - \mathbb{E}[Y|Y > \gamma])^2 \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2} \right]}{\mathbb{P}(Y > \gamma)^2}. \quad \square \end{aligned}$$

Some comments on Proposition 3.1. It is interesting to analyse the convergence result stated in Proposition 3.1 in light of the behaviour of an other estimator of $\text{CL}(X)$. Let us introduce

$$\overline{\text{CL}}_n(h(X)) = \frac{\frac{1}{n} \sum_{i=1}^n Y_i^\theta \mathcal{E}_i^\theta \mathbf{1}_{\{Y_i^\theta > \gamma\}}}{\mathbb{P}(X > \gamma)}.$$

Clearly, $\mathbb{E}[\overline{\text{CL}}_n(X)] = \mathbb{E}[Y|Y > \gamma]$, which assesses that $\overline{\text{CL}}_n(h(X))$ is unbiased. From, the standard central limit theorem, we deduce that

$$\sqrt{n}(\text{CL}_n(h(X)) - \text{CL}(h(X))) \xrightarrow[n \rightarrow \infty]{\mathcal{L}} \mathcal{N}(0, \bar{\sigma}^2) \quad \text{with } \bar{\sigma}^2 = \frac{\text{Var}(Y^\theta \mathcal{E}^\theta \mathbf{1}_{\{Y^\theta > \gamma\}})}{\mathbb{P}(X > \gamma)^2}.$$

Note that $\bar{\sigma}^2$ can also be written

$$\begin{aligned} \bar{\sigma}^2 &= \frac{\mathbb{E} \left[\left(Y^\theta \mathcal{E}^\theta \mathbf{1}_{\{Y^\theta > \gamma\}} - \mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}}] \right)^2 \right]}{\mathbb{P}(X > \gamma)^2} \\ &= \frac{\mathbb{E} \left[Y^2 \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2} \right] - \mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}}]^2}{\mathbb{P}(X > \gamma)^2}. \end{aligned}$$

Easy calculations yield

$$\begin{aligned} \bar{\sigma}^2 - \sigma^2 &= \frac{-\mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}}]^2 - \mathbb{E}[\mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2}] \mathbb{E}[Y|Y > \gamma]^2}{\mathbb{P}(X > \gamma)^2} \\ &\quad + \frac{2\mathbb{E}[Y|Y > \gamma] \mathbb{E}[Y \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2}]}{\mathbb{P}(X > \gamma)^2} \\ &= \frac{\mathbb{E}[Y|Y > \gamma] \mathbb{E} \left[(2Y - \mathbb{E}[Y|Y > \gamma]) \mathbf{1}_{\{Y > \gamma\}} e^{-\theta \cdot X + |\theta|^2/2} \right]}{\mathbb{P}(Y > \gamma)^2} - \mathbb{E}[Y|Y > \gamma]^2. \end{aligned}$$

For $\theta = 0$, the difference simplifies into

$$\bar{\sigma}^2 - \sigma^2 = \mathbb{E}[Y|Y > \gamma]^2 \left(\frac{1}{\mathbb{P}(Y > \gamma)} - 1 \right) > 0.$$

The limiting variance $\bar{\sigma}^2$ is larger than the one appearing in (3.3) (at least for $\theta = 0$) but unlike $\overline{\text{CL}}_n(h(X))$, the estimator $\text{CL}_n(h(X))$ is biased, which is precisely the effect induced by replacing $\mathbb{P}(Y > \gamma)$ by $\frac{1}{n} \sum_{i=1}^n \mathbf{1}_{\{Y_i^\theta > \gamma\}} \mathcal{E}_i^\theta$ in $\overline{\text{CL}}_n(h(X))$ to obtain $\text{CL}_n(h(X))$.

The gap between the two limiting variances may become impressively large. Consider the following toy example with $h(x) = x$, X following a real valued standard normal distribution and $\gamma = 1.5$. Take $\theta = 0$. In this case, $\mathbb{P}(X > \gamma) \approx 0.067$ and we find $\sigma^2 \approx 2$ whereas $\bar{\sigma}^2 \approx 54$.

Proposition 3.2. *When $\theta = 0$, the estimator $\text{CL}_n(h(X))$ has bias*

$$-\mathbb{E}[h(X)|h(X) > \gamma]\mathbb{P}(h(X) \leq \gamma)^n.$$

Proof. We use the notation $Y = h(X)$. We define $T_n = \sum_{i=1}^n \mathbf{1}_{\{Y_i > \gamma\}}$. On $\{T_n \geq 1\}$, the tuple (Z_1, \dots, Z_{T_n}) holds the values of (Y_1, \dots, Y_n) for which $Y_i > \gamma$ for $i \in \{1, \dots, n\}$. Using the convention $\frac{0}{0} = 0$, we have

$$\text{CL}_n(Y) = \frac{1}{T_n} \sum_{i=1}^{T_n} Z_i. \tag{3.5}$$

$$\mathbb{E} \left[\frac{1}{T_n} \sum_{i=1}^{T_n} Z_i \right] = \sum_{t=1}^n \mathbb{E} \left[\frac{1}{t} \sum_{i=1}^t Z_i \mathbf{1}_{\{T_n=t\}} \right] = \sum_{t=1}^n \mathbb{E} [Z_1 \mathbf{1}_{\{T_n=t\}}].$$

Observe that Z_1 and T_n are independent. Hence,

$$\begin{aligned} \sum_{t=1}^n \mathbb{E} [Z_1 \mathbf{1}_{\{T_n=t\}}] &= \sum_{t=1}^n \mathbb{E}[Z_1] \mathbb{P}(T_n = t) \\ &= \mathbb{E}[Y|Y > \gamma] \sum_{t=1}^n \binom{n}{t} \mathbb{P}(Y > \gamma)^t (1 - \mathbb{P}(Y > \gamma))^{n-t} \\ &= \mathbb{E}[Y|Y > \gamma] (1 - (1 - \mathbb{P}(Y > \gamma))^n). \end{aligned}$$

Hence,

$$\mathbb{E}[\text{CL}_n(Y)] = \mathbb{E}[Y|Y > \gamma] - \mathbb{E}[Y|Y > \gamma] \mathbb{P}(Y \leq \gamma)^n. \quad \square$$

4. Stratified sampling

An other well-known tool for rare event simulation is stratified sampling. This technique can lead to very impressive results provided one has a good a priori knowledge of the strata and of their respective weights in the sampling.

4.1. Introduction to stratification

Consider a partition D_1, \dots, D_I of \mathbb{R}^d and assume we know how to sample according to $\mathcal{L}(X|X \in D_i)$ for $i = 1, \dots, I$. Let $p_i = \mathbb{P}(X \in D_i)$. We assume that the strata are chosen such that $p_i > 0$ for all i . For a fixed $1 \leq i \leq I$, let $(X_i^j, 1 \leq j \leq N_i)$ be i.i.d. samples according to $\mathcal{L}(X|X \in D_i)$ where N_i is the number of samples in stratum i . We assume that the samples in two different strata are independent, ie for different values of the index i .

We know that

$$\mathbb{E}[\varphi(X)] = \sum_{i=1}^I \mathbb{E}[\varphi(X)|X \in D_i] p_i.$$

Hence, we can use the following Monte Carlo estimator to approximation $\mathbb{E}[\varphi(X)]$

$$\sum_{i=1}^I p_i \frac{1}{N_i} \sum_{j=1}^{N_i} \varphi(X_i^j).$$

The variance of this estimator is given by

$$\sum_{i=1}^I \frac{p_i^2}{N_i} v_i^2$$

where $v_i^2 = \text{Var}(\varphi(X_i^1))$. Assume the total number of samples is N , then this variance is minimum for $N_i = q_i/N$ with

$$q_i = \frac{p_i v_i}{\sum_{\ell=1}^I p_\ell v_\ell}. \quad (4.1)$$

Using this optimal allocation yields to an estimator with minimum variance given by $\frac{v_*^2}{N}$ with

$$v_*^2 = \left(\sum_{i=1}^I p_i v_i \right)^2.$$

However, this allocation formula can be hardly used in practice as the v_i 's are usually unknown. To overcome this difficulty, [4] proposed an adaptive scheme to learn the $(v_i)_{i=1, \dots, I}$ and then the $(q_i)_{i=1, \dots, I}$.

4.2. Importance sampling as a stratifying direction

The key difficulty in applying stratified sampling techniques is to know which strata to be used. [5] suggested to use the optimal importance sampling vector as a direction along which to stratify. Assume some optimal shift μ^* has already been computed. The vector $u = \frac{\mu^*}{|\mu^*|}$ defines an hyperplane on which the strata are based. Consider an increasing sequence of levels $(a_i)_{0 \leq i \leq I}$ with $a_0 = -\infty$ and $a_I = +\infty$, we consider the partition $\cup_{i=1}^I D_i$ with $D_i = \{x \in \mathbb{R}^d : a_{i-1} \leq u \cdot x < a_i\}$. Since $u \cdot X$ follows the standard normal distribution with values in \mathbb{R} , the levels a_i 's are usually chosen as quantiles of the standard normal distribution. Then, simulating the normal random vectors conditionally to being in a given stratum can be easily implemented by using the following result.

Lemma 4.1. *Let $a < b$ be real numbers and $X \sim \mathcal{N}(0, I_d)$. Let U be a random variable with uniform distribution on $[0, 1]$ and $Y \sim \mathcal{N}(0, I_d)$ both independent of X . Set $Z = \Phi^{-1}(\Phi(a) + U(\Phi(b) - \Phi(a)))$, where Φ is the cumulative distribution function of the standard normal distribution. Then,*

$$uZ + Y - u(u \cdot Y) \sim \mathcal{L}(X | a \leq u \cdot X \leq b).$$

A natural approach is to proceed in two steps

- (1) *Importance Sampling step.* Apply Algorithm 2 to compute an almost optimal change of measure with parameters (μ^*, I_d) .
- (2) *Stratified sampling.* Set $u = \frac{\mu^*}{|\mu^*|}$. Fix some levels (a_0, \dots, a_I) based on the quantiles of the standard normal distribution and use the strata defined by $D_i = \{x \in \mathbb{R}^d : a_{i-1} \leq u \cdot x < a_i\}$ for $1 \leq i \leq I$ in the algorithm proposed by [4].

5. Rare events simulation for electronic circuits

5.1. Large scale analytic problems

We validate in this section the Importance Sampling multilevel approach with large scale problems. The test-cases were generated automatically for the purpose of this test, and not related to circuit simulation.

The response function is defined as $h(x) = a \cdot x_A + b \cdot x_B$, and $x = (x_A, x_B)$ is a vector of normal variables $\mathcal{N}(0, 1)$. The coefficients a and b were defined such as $\min_{i \in A} |a_i| \gg \max_{i \in B} |b_i|$. This makes the subset A the set of important variables. The remaining B -components may be considered as noisy variables.

For our comparisons, we fixed the cardinality $|A| = 10$ and the vector a , and we gradually increased the dimension of the noisy components. The size of the batches is fixed at 1000 runs.

$ B $	Prob.	CI@95%	Nb. Runs
1000	2.8039×10^{-5}	8.60%	5000
2000	2.9891×10^{-5}	8.91%	5000
10000	2.8239×10^{-5}	9.65%	10000
50000	2.9815×10^{-5}	9.83%	20200

The previous table indicates that the approach is rather not dependent on the input dimension of the problem. The differences, as the dimension grows, are due mainly to the difficulty of finding the *optimal* subset of important variables during the search of the final mean-shift. Increasing the batch size may help as well, because larger batches will improve the accuracy of the reduction dimension algorithms.

5.2. Circuit examples

We use systematically the same settings for the Importance sampling algorithm. The relative precision on the probability estimator is fixed at 10%. And the size of each batch is 1000 runs, the total number of runs is therefore a multiple of this size. We report the “Speedup” in our tests as the fraction of the theoretical number runs of the plain MC over the effective number of runs to obtain the same relative accuracy (based on the Central Limit theorem). For our experiments, electrical simulations will be achieved with the commercial SPICE simulators Eldo[®] and EldoRF from Siemens EDA.

5.2.1. VCO LC Tank

This is an example of an analog IP block showing the IS algorithm for low probability estimation and quantile estimation. The circuit is a simple LC-tank VCO oscillating at 1.8GHz (inspired from [3])

The circuit contains 28 Gaussian random variables. The oscillation frequency (FOSC) is obtained using the Steady-State Analysis of Eldo RF. The phase noise (PNOISE) at 1 MHz offset from the carrier is also extracted using the Steady-State noise analysis of Eldo RF. Both the oscillation frequency and the phase noise are strongly non-Gaussian. The distributions are heavily skewed (left and right respectively). This means that all kinds of Gaussian extrapolations using the estimated standard deviation with a few 100’s runs are completely wrong. The results for the analysis of ISMC are mostly read in the following table.

Measure	Tail	Prob	CI@95%	Nb. Runs	Speedup
FOSC	Left	7.9849×10^{-5}	8.89%	5000	1.2×10^3
PNOISE	Right	3.5240×10^{-6}	9.05%	9000	1.5×10^4

The Prob column indicates the tail probability and CI@95% the relative Confidence Interval at 95% level. The Nb. Runs column gives the total number of runs for each estimator. The Speedup column finally provides the speedup factor over a plain Monte Carlo simulation (for the same accuracy).

Similarly, one gives the results for the quantile estimation where we fixed the tail probability at level 1.0×10^{-4} .

Measure	Tail	Quantile	CI@95%	Nb. Runs	Speedup
FOSC	Left	1.75195×10^9	0.06%	4000	3.1×10^2
PNOISE	Right	-1.1239×10^2	0.16%	4000	2.2×10^2

5.2.2. 6T SRAM Bitcell

The circuit is a classical six transistors SRAM bit cell designed in CMOS 45nm. A subset of the device model parameters has variability information with Gaussian random variables. In this

example, the total number of statistical parameters is 36 (6 per device). The cell is configured for measuring the write delay. The purpose of the Monte Carlo simulation is to estimate the probability for the write delay to be larger than a certain threshold. This threshold might be used to define a “fail” behavior for yield estimation. Formally, the problem is to estimate the probability of failure and the associated confidence interval.

This kind of low probability estimation is difficult (not to say impossible) with plain Monte Carlo, because the number of runs required to obtain a decent accuracy (in terms of confidence interval on the probability) is huge. In this particular example, the IS flow estimates a probability of 9.1893×10^{-6} to within 9.99% (with 95% confidence level) in 8000 runs, whereas it would require approximately 4×10^7 runs with plain Monte Carlo.

Measure	Tail	Prob	CI@95%	Nb. Runs	Speedup
WDELAY	Right	9.1893×10^{-6}	9.99%	8000	5.2×10^3

We give the detail of the multilevel Importance Sampling phase with this circuit. We recall that we need to use intermediate levels to reduce the difficulty of sampling a rare event. If the levels $(\gamma_k)_k$ are chosen so that the events $\{h(X_j^{(k+1)}) \geq \gamma_k\}$ are never too rare, then a fairly small number of samples n can be used. The following table provides the first iterations of this exploration phase.

Iteration (k)	Nb. Runs	γ_k	$S_{m,n}(\gamma_k)$	CI@95%
0	1000	3.99328×10^1	1.51331×10^{-3}	21.9%
1	1000	4.69753×10^1	1.38538×10^{-4}	27.0%
2	1000	5.31605×10^1	8.51455×10^{-6}	34.7%

We now provide some results on CVaR computation for this circuit. As noted above, comparing the threshold γ and CVaR may give a hint about the dispersion of *Failed* population of runs.

Measure	γ	CVaR	CI@95%	Nb. Runs	Speedup
WDELAY	5.3120×10^1	5.3388×10^1	10.1%	9000	5.1×10^3

It is important to note here that the CVaR is a by-product of the IS flow which is mainly driven by the computation of the failure probability. The computation of the CVaR is based on the same samples. Here, practitioners may conclude that picking one point of the failed runs provides an accurate example for debugging or understanding the reasons that lead to this fail point. The analysis of the largest components of the mean-shift may help further to identify the variables and therefore the electronic devices at the origin of the typical failure point.

5.2.3. StdCell LH

This circuit is flip-flop or latch circuit extracted from a library of standard cells. It is a circuit that can be used to store state information, and a fundamental storage element in sequential logic of digital electronics systems used in computers, and many types of systems. The number of random variables is fixed at 66. The “LAB3” output represents the measure of a delay (in seconds).

Measure	Tail	Prob	CI@95%	Nb. Runs	Speedup
LAB3	Right	1.2493×10^{-9}	8.05%	7000	6.8×10^7

The speedup here is quite impressive, but note that in general the testbenches used for such standard cells contain multiple output measures. Because the samples cannot be mutualized across the different estimators, the total number of runs will be a multiple of the computed quantiles and/or probabilities.

5.2.4. StdCell MUX4

This circuit a multiplexer (or mux) is another fundamental block extracted from a library of standard cells. The number of random variables is fixed at 116. The output response chosen for this experiment is a measure of delay (in seconds) on a critical path. It is therefore a time-domain analysis.

We are interested here in the computation of quantiles at some predefined failure probabilities.

Measure	Failure Prob.	Quantile	CI@95%	Nb. Runs	Speedup
LAB179	1.34990×10^{-3}	2.55684×10^{-10}	0.72%	4000	3.2
-	9.86588×10^{-10}	2.78912×10^{-10}	0.22%	6000	3.9×10^6

As we noted in the previous example, if multiple circuit output need to be characterized then the speedup 3.2 obtained for the first failure probability (approx. 1.3×10^{-3}) may not compensate the total number of runs required for estimating each quantile individually. A plain Monte Carlo simulation may do the job because the quantiles can be computed on the same samples. This option may not be possible when very low failure probabilities are searched. In such cases, the IS algorithm may prevail over the plain MC.

5.2.5. Memory Cell

This circuit represents a block of memory simulated with its peripheral circuitry. The number of random variables is fixed at 2096. The simulation output is here the read delay measured in seconds.

Measure	Tail	Prob	CI@95%	Nb. Runs	Speedup
READ10	Right	3.4506×10^{-8}	8.96%	9000	1.5×10^6

This last example is provided to demonstrate the effectiveness of IS algorithm on a quite large size problem. This is a time-domain analysis which requires several minutes for a nominal simulation on a modern architecture. Our IS algorithm can run in full parallel mode, taking advantage of compute farms.

References

- [1] Kanak Agarwal and Sani Nassif. The Impact of Random Device Variation on SRAM Cell Stability in Sub-90-nm CMOS Technologies. *IEEE Trans. VLSI Syst.*, 16(1):86–97, 2008.
- [2] Laetitia Badouraly Kassim, Jérôme Lelong, and Imane Loumrhari. Importance sampling for jump processes and applications to finance. *J. Comput. Finance*, 2(19), 2015.
- [3] J. Craninckx and M. S. J. Steyaert. A 1.8-GHz low-phase-noise CMOS VCO using optimized hollow spiral inductors. *IEEE J. Solid-State Circuits*, 32(5):736–744, 1997.
- [4] Pierre Etoré and Benjamin Jourdain. Adaptive Optimal Allocation in Stratified Sampling Methods. *Methodol. Comput. Appl. Probab.*, 12(3):335–360, 2010.
- [5] Paul Glasserman, Philip Heidelberger, and Perwez Shahabuddin. Asymptotically optimal importance sampling and stratification for pricing path-dependent options. *Math. Finance*, 9(2):117–152, 1999.
- [6] T. Hiramoto. Variability in Scaled MOSFETs. In *Nanoscale Silicon Devices*. CRC Press, 2015.
- [7] Chung-Wen Ho, A. Ruehli, and P. Brennan. The modified nodal approach to network analysis. *IEEE Trans. Circuits Syst.*, 22(6):504–509, 1975.
- [8] D. E. Hocevar, M. R. Lightner, and T. N. Trick. A Study of Variance Reduction Techniques for Estimating Circuit Yields. *IEEE Trans. CAD Integr. Circuits Syst.*, 2(3):180–192, 1983.

- [9] Benjamin Jourdain and Jérôme Lelong. Robust Adaptive Importance Sampling for Normal Random Vectors. *Ann. Appl. Probab.*, 19(5):1687–1718, 2009.
- [10] R. Kanj, R. Joshi, and S. Nassif. Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events. In *2006 43rd ACM/IEEE Design Automation Conference*, pages 69–72, 2006.
- [11] Dirk P. Kroese, Sergey Porotsky, and Reuven Y. Rubinstein. The cross-entropy method for continuous multi-extremal optimization. *Methodol. Comput. Appl. Probab.*, 8(3):383–407, 2006.
- [12] Michel Ledoux and Michel Talagrand. *Probability in Banach spaces*, volume 23 of *Ergebnisse der Mathematik und ihrer Grenzgebiete. 3. Folge*. Springer, 1991.
- [13] William J. McCalla. *Fundamentals of Computer-Aided Circuit Simulation*. Kluwer Academic Publishers, 1987.
- [14] Trent McConaghy, Kristopher Breen, Jeffrey Dyck, and Amit Gupta. *Variation-Aware Design of Custom Integrated Circuits: A Hands-on Field Guide*. Springer, 2013.
- [15] Reuven Y. Rubinstein and Alexander Shapiro. *Discrete event systems*. Wiley Series in Probability and Mathematical Statistics: Probability and mathematical statistics. John Wiley & Sons, 1993.
- [16] Amith Singhee and Rob A. Rutenbar. Statistical Blockade: Very Fast Statistical Simulation and Modeling of Rare Circuit Events and Its Application to Memory Design. *IEEE Trans. CAD Integr. Circuits Syst.*, 28(8):1176–1189, 2009.
- [17] Jierâi Vlach and Kishore Singhal. *Computer Methods for Circuit Analysis and Design*. John Wiley & Sons, 2nd edition, 1993.