Imprecise probabilistic subset simulation for rigorous bounding of the failure probability

V. M. Nikolov, P. O. Hristov
GATE Institute, Sofia University, 5 James Bourchier Blvd., 1164 Sofia, Bulgaria.
Email: {valentin.nikolov, petar.hristov}@gate-ai.eu

Abstract. One of the most challenging tasks in reliability engineering, namely the estimation of small failure probabilities can be efficiently performed through subset simulation (SuS). Given a performance function and a critical level, which defines failure, SuS uses Markov chain Monte Carlo sampling, to gradually explore the input domain of the performance function with a nested sequence of larger domains, until samples over the critical threshold and into the failure domain are generated.

P-SuS, proposed by (Hristov and DiazDelaO, 2023), generalises the original subset simulation algorithm by adapting it to work with probabilistic numerical models (PNM). P-SuS was shown to be able to discover and populate the failure domain, only based on partially converged simulations. However, the probabilistic description of its estimator for the failure probability is not calibrated, in that it does not cover the failure probability for the deterministic model, estimated by SuS or does so only at very small numerical uncertainties and with inappropriately large bounds.This problem is strongly influenced by the assumptions of independence of the intermediate failure events and the independence of the probabilities of the levels. The result of these assumptions has a knock-on simplification effect on how the final estimator is computed.

The work described in this paper relaxes these assumptions by reformulating parts of the P-SuS algorithm through the use of dependence bounds analysis to compute uncertain number convolutions without assuming a particular dependence. We show that the resulting probability-box estimator more appropriately accounts for the uncertainty present in the output of the PNM.

Keywords: Probabilistic numerical models; Reliability analysis; Subset simulation; Dependence bounds analysis; Imprecise probability

1. Introduction

Computational models or simulators have established themselves as an irreplaceable part of any engineering process, as they significantly reduce the cost of conducting the equivalent physical experiments when exploring the state space of a complex physical system. Computer modelling, however advanced, is only an idealised approximation of any process considered. Often, simplifications and assumptions about the form of the model are required to make it practical or even feasible to use. Moreover, due to their numerical nature, discretisation errors also occur and accumulate during calculations because of the limited precision of any computer. Nevertheless, the purpose of

simulators remains to be to inform engineering decisions, and as such their inherent uncertainties need to be rigorously quantified to make them reliable to use.

Probabilistic numerics (Hennig et al., 2015) have been developed to provide an explicit quantification of the numerical uncertainty in the output of the model. Numerical uncertainty typically includes uncertainties stemming from the use of finite discretisation schemes and number of iterations, the use of finite precision floating-point arithmetic and, for non-deterministic models, ensemble uncertainty (Oberkampf and Roy, 2025). By treating the numerical modelling problem as that of statistical inference, probabilistic numerical methods produce a distributional output that includes an estimate of the numerical error. A well-designed probabilistic numerical model (PNM) should provide a calibrated measure about the numerical uncertainty and should converge on the deterministic model result in the limit of increasing computational budget (such as using a finer mesh and allowing for more iterations).

Probability theory also plays a central role in reliability analysis (RA), which aims to evaluate whether a physical system or structure is safe to use under a set of different conditions. A performance function, usually in the form of a computational model, which maps the space of inputs to outputs, characterises the behaviour of the investigated system. When assessing the reliability of a physical system, the failure domain $\mathcal{F}$, that is the set of configurations of the input that result in outputs of the performance function that exceed a predefined, critical threshold, is of particular interest. Characterising $\mathcal{F}$ is challenging for most practical applications, because its volume is multiple orders of magnitude smaller than that of the input space. This, together with the potentially complex geometry of $\mathcal{F}$, makes the task of reliably estimating the probability of failure $p_F = \mathbb{P}(\mathcal{F})$ difficult.

Many methods have been developed to estimate p_F, including first and second order reliability methods (Zhao and Ono, 1999), sampling approaches based on Monte Carlo simulation (MCS) (Padmanabhan et al., 2006; Naess et al., 2009), as well as variance reduction methods improving on MC procedures by employing more efficient strategies for sampling, such as importance sampling (Ibrahim, 1991) and line sampling (Pradlwarter et al., 2007). Another method, which has stood the test of time is subset simulation (SuS) (Au and Beck, 2001), because of its computational efficiency in terms of model evaluations needed, compared to MC and its ability to efficiently handle high-dimensional domains. The core idea of SuS is to represent $\mathcal{F}$ as the intersection of a set of nested intermediate failure domains, thus transforming the problem into the simulation of conditional events with constant probabilities, much higher than that of $\mathcal{F}$. Markov chain Monte Carlo algorithms are then employed to sample from the intermediate failure domains, given that the input already lies in the previous, larger intermediate failure region, and compute an estimator for p_F, denoted as $\hat{p}_F^{SuS}$.

Many advancements of SuS have been developed since its inception to address some of its assumptions and deficiencies. For instance, Kinnear and DiazDelaO, 2025 proposed the use of an evolutionary optimisation concept called niching to deal with rapidly changing model outputs and multimodality of the performance function. Miao and Low, 2026 take intrachain, interchain, and interlevel correlations into account, to more accurately estimate the variance of the $\hat{p}_F^{SuS}$. Other modifications include the development of imprecise SuS, used to quantify the uncertainty in the estimate for the probability of failure arising from not having perfect knowledge of the probability distributions for the input parameters (Giovanis and Shields, 2021).

A modification of SuS, to enable its use with PNM was proposed in (Hristov and DiazDelaO, 2023). Even though the authors show that the method, called P-SuS, converges to $\hat{p}_F^{SuS}$ as the uncertainty in the output of the PNM decreases, they note that the calibration of the estimator uncertainty remains poor.

In this work, we conjecture that the lack of calibration in the uncertainty of $\hat{p}_F^{P-SuS}$ is at least in part due to the assumptions of independence used throughout P-SuS. We use probability bounds analysis to relax this assumption and show how this change is reflected on the final estimator of the algorithm that we term imprecise probabilistic subset simulation (IP-SuS).

The remainder of the paper is organised as follows: Section 2 gives an overview of SuS, P-SuS and Fréchet convolutions necessary to follow the proposed improvements in IP-SuS, presented in Section 3. In Section 4, the new algorithm is demonstrated and benchmarked on the the modified Branin function, also featuring in the original work, introducing P-SuS (Hristov and DiazDelaO, 2023). Finally, Section 5 draws some conclusions and outlines directions for future work.

2. Methodology overview

2.1. Subset simulation

Let $h : \mathcal{X} \subseteq \mathbb{R}^d \to \mathbb{R}$ be the performance function of the system under investigation, such that for some particular input combination of system properties, $\mathbf{x} \in \mathcal{X}$ and some predefined threshold t^*, $y = h(\mathbf{x}) > t^*$ signifies the failure of the system. The failure domain, corresponding to t^*, is the set of all input combinations leading to failure, $\mathcal{F} = \{\mathbf{x} \in \mathcal{X} : h(\mathbf{x}) > t^*\}$. In probabilistic reliability analysis, any uncertainty over the system properties domain, $\mathcal{X}$ is described as a random variable, X with some joint probability distribution, $G_X(\cdot)$. Projecting $G_X(\cdot)$ through $h(\cdot)$ results in an output random variable $Y \sim G_Y(\cdot)$, even for a deterministic $h(\cdot)$. Under this formulation, a failure event is defined as $F = \{Y > t^*\}$ and the associated probability of failure is thus

$$p_F \equiv \mathbb{P}(F) = \int_{\mathcal{F}} dG(\mathbf{x}) = \mathbb{E}[\mathbb{I}_F(\mathbf{x})] \tag{1}$$

where $\mathbb{I}_F$ is the indicator function: $\mathbb{I}_F(\mathbf{x}) = 1$ if $h(\mathbf{x}) > t^*$ and 0 otherwise.

If the volume of $\mathcal{F}$, is much smaller than that of $\mathcal{X}$, as is usually the case, direct Monte Carlo simulation approaches would require a large number of samples, $N_{MCS} = O(p_F^{-1})$, to even discover $\mathcal{F}$. If in addition $h(\cdot)$ takes even modest times to run, reliability analysis may become impractical.

To alleviate this problem, SuS represents the failure domain as $\mathcal{F} = \mathcal{F}_m \subset \mathcal{F}_{m-1} \subset \cdots \subset \mathcal{F}_1 \subset \mathcal{F}_0 = \mathcal{X}$, where each intermediate failure domain $\mathcal{F}_k$ is associated with a failure event, $F_k = \{Y > t_k\}$, defined by an intermediate critical threshold. Using the property of the nestedness of F_k, the probability of failure can be evaluated as

$$p_F = \mathbb{P}\left(\bigcap_{k=1}^{m} F_k\right) = \prod_{k=1}^{m} \mathbb{P}(F_k \mid F_{k-1}). \tag{2}$$

Thus, the task of sampling from $\mathcal{F}$ to estimate p_F is reduced to generating $N << N_{MCS}$ samples from each intermediate failure domain, according to that domain's own conditional distribution,

progressively steering the samples towards $\mathcal{F}$. In SuS, this is done through two main constructs. First, the intermediate thresholds t_k are chosen adaptively such that $t_k \leq t_{k+1}$ and $\mathbb{P}(F_{k+1}|F_k) = \mathbb{P}(F_{k+2}|F_{k+1}) = p_0$ for all $1 < k < m-2$. Second, $p_0 N$ samples are chosen from $\mathcal{F}_k$ as seeds for a component-wise modified Metropolis-Hastings algorithm, which uses them to start independent chains and generate the remaining $(1-p_0)N$ samples for $\mathcal{F}_{k+1}$. The procedure for generating nested intermediate failure events continues until at least $p_0 N$ samples are obtained from $\mathcal{F}$. Subsequently, p_F is estimated as:

$$\hat{p}_F^{SuS} = p_0^{m-1} \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(h(\mathbf{x}_i) > t^*) \tag{3}$$

2.2. Subset simulation for probabilistic numerical models

To extend SuS to PNM, P-SuS needs to account for the fact that each response from $h(\cdot)$ is an entire probability distribution. For the remainder of this section, only those details of how this accounting is done that are relevant to the imprecise extension of P-SuS will be presented.

Without loss of generality, it is assumed in P-SuS that the distributions of $y = h(\mathbf{x})$ has finite first and second moments. When each y_i for $i \in \{1, \ldots, N\}$ is a distribution the probability of the failure event $F_{ik} = \{y_i > t_k\}$ turns from a dichotomous variable in $\{0, 1\}$ into a random variable. The probability of each intermediate failure domain, p_{F_k} as well as the number of samples in that level N_k are also random variables. Similarly to SuS, p_0 is still a parameter chosen by the analyst, but it serves as a nominal probability of the intermediate failure domains. At each level, p_{F_k} can be estimated as

$$p_{F_k} \approx \frac{1}{N_k} \sum_{j=1}^{N_k} \mathbb{I}_{F_k}(\mathbf{x}_j) \tag{4}$$

where the indicator function is a random variable $\mathbb{I}_{F_k}(\mathbf{x}_j) \sim \text{Bernoulli}(\mathbb{P}(h(\mathbf{x}_j) \geq t_k))$. and $\mathbf{x}_j$ still come from the input distribution, conditional on the preceding failure event, $G_X(\cdot|F_{k-1})$. In P-SuS, the Bernoulli random variables describing the indicator functions are assumed to be independent. Under this assumption, p_{F_k} is modelled as:

$$p_{F_k} \sim \mathcal{N}\left(\frac{\mu_k}{N_k}, \frac{{\sigma_k}^2}{N_k^2}\right) \tag{5}$$

The failure domain is deemed to be populated well enough when $N_F \geq p_0 N$ samples have been generated in $\mathcal{F}$, according to a double acceptance procedure (Hristov and DiazDelaO, 2023). At this point, the expression in Eq. (3) is used to compute an estimate of p_F, denoted $\hat{p}_F^{P-SuS}$, as a product of the normal random variables for each p_{F_k}. However, even under an assumption of independence among p_{F_k}, only the mean and variance of $\hat{p}_F^{P-SuS}$ are available in closed form as:

$$\mathbb{E}[\hat{p}_F^{P-SuS}] = \mathbb{E}\left[\prod_{k=1}^{m} p_{F_k}\right] = \prod_{k=1}^{m} \mathbb{E}[p_{F_k}] \tag{6}$$

$$\mathbb{V}\left[\hat{p}_F^{P-SuS}\right] = \mathbb{V}\left[\prod_{k=1}^{m} p_{F_k}\right] = \\ \mathbb{V}\left[p_{F_m}\right]\mathbb{V}\left[\prod_{k=1}^{m-1} p_{F_k}\right] + \mathbb{V}[p_{F_m}]\,\mathbb{E}\left[\prod_{k=1}^{m-1} p_{F_k}\right]^2 + \mathbb{E}[p_{F_m}]^2\,\mathbb{V}\left[\prod_{k=1}^{m-1} p_{F_k}\right] \tag{7}$$

which also assumes independence between levels.

2.3. Convolutions under unknown dependence

To convolve two random variables, probability theory requires that the dependence between them is specified. This can be done by specifying the joint distribution of the variables, a copula, or a family of copulas, with all required parameters, in addition to the marginals, a functional dependence, or assuming a particular crisp dependence model. These models include independence, perfect and opposite dependence.

Adequately specifying the correct dependence for convolutions of uncertain numbers is a task that is commonly more challenging than comfortably proposing a model for the marginals. To compound this, in most practical applications, the effect of the dependence model on the result of convolutions is marked (Ferson et al., 2004).

When the dependence between the uncertain numbers is unknown, or imprecisely known, dependence bounds analysis (DBA) provides the mathematical machinery to enable convolutions (Williamson, 1989). Under DBA, the assumptions of perfect or opposite dependence can be relaxed to positive and negative dependence, respectively. Convolutions under no particular dependence can also be performed. The latter are referred to as Fréchet convolutions, as their bounds are based on the Fréchet-Hoeffding copula bounds (Nelsen, 2006). The upper and lower bounds on the result, Z, of a binary operation, $L(\cdot,\cdot)$, between two random variables, X and Y, with cumulative distribution functions (CDF), given by $G_X(\cdot)$ and $G_Y(\cdot)$, respectively are:

$$\begin{aligned} \tau_{W,L}(G_X, G_Y)(z) &= \sup_{z=L(x,y)} \max(G_X(x) + G_Y(y) - 1, 0) \\ \rho_{W,L}(G_X, G_Y)(z) &= \inf_{z=L(x,y)} \min(G_X(x) + G_Y(y), 1) \end{aligned} \tag{8}$$

In Eqs. 8, τ is the lower bound on the probability (resp. the right bound on the quantiles) and ρ is the upper (resp. left) bound. The results of Fréchet convolutions are sure to bound the result under the true, but unknown dependence, and they may or may not be as tight as possible under the provided information (Ferson et al., 2004). It must be noted that Fréchet convolutions are not limited to aleatory uncertain numbers only, but readily extend to mixed uncertain structures, such as probability boxes (p-boxes for short), too (Williamson, 1989). This extension is used as a principal construct of IP-SuS.

3. Imprecise probabilistic subset simulation

The assumptions of independence of the responses at each level and between the levels themselves are instrumental for the estimation of $\hat{p}_F^{P-SuS}$. This however, is an idealisation, as the indicator functions $\mathbb{I}_{F_k}(\mathbf{x})$ and their associated Bernoulli random variables arise from the same underlying

model within each chain of the MCMC algorithm, and as such have some complex form of correlation between them. Similarly, the levels are nested and each level is generated from the seeds from the previous level, making the failure probabilities at each level dependent. In this section we propose to replace the operations under independence with Fréchet convolutions.

The normal distribution in Eq. (5) is only suitable assuming that $\mathbb{I}_{F_k}(\mathbf{x}_j)$ in the sum of Eq. (4) are mutually independent, for all $1 < j < N_k$. However, when calculating that sum, no knowledge about the dependence structure of the indicator functions is available. The summation can be performed repeatedly using Fréchet convolutions, resulting in p_{F_k} taking the shape of a p-boxes of no particular distributional family, instead of crisp normal distributions. These p-boxes contain all the information about the possible cumulative distribution functions of p_{F_k}. The Fréchet convolution of two or more random variables, even with exact moments, results in a p-box, whose width represents the uncertainty arising from the lack of knowledge about the dependence.

Because Fréchet-Hoeffding copula bounds are valid only for binary operations, the sum in Eq. (4) needs to be done in a recursive manner. Repeatedly summing, random variables rigorously represented on a computer incurs additional uncertainty due to finite floating-point representations. The size of the uncertainty depends directly on the level of discretisation of the uncertain numbers and on the number of operations, that is on N_k. To control the accumulation of this kind of uncertainty, we set a parameter ϵ to impose a cut-off for the addends in Eq. (4) to those Bernoulli random variables, whose probability of success is greater than 1×10^{-4}. This choice is arbitrary and should be driven by the implementation of uncertainty arithmetic used.

Analogously, in Eqs. (6) and (7) the mean and the variance of the final estimator are also computed under the assumption of independence among the probabilities of the intermediate failure events, where the second equality in both equations holds only under this condition. The product can again be calculated under no particular dependence utilising Fréchet convolutions. As a result $\hat{p}_F^{IP-SuS}$ is no longer a distribution, for which only two moments are known, but a p-box completely defined by a pair of bounds.

Although the bounds are sufficient to fully define the final estimator of the probability of failure its mean and variance are calculated for the sake of comparison with P-SuS. The mean of the p-box is interval-valued and straightforward to calculate by taking the means of the lower and upper bounds. The variance is also interval-valued and calculated by combining its left and right step endpoints in all possible split configurations: the lower bound is either 0 (if a deterministic number can fit in the p-box) or the minimum variance of the endpoints, whereas the upper bound is the maximal variance over these combinations. The bounds for the moments can be further tightened using different methods, for example, by using information about the moments of the objects convolved.

4. Numerical experiments

To test the performance of IP-SuS, an experiment analogous to that in (Hristov and DiazDelaO, 2023) was conducted, in which a modification of the the well-known Branin function:

$$h(\mathbf{x}) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10(1-t)\cos(x_1) + 10 + 5x_1 \tag{9}$$

where $h : [-5, 10] \times [0, 15] \to \mathbb{R}$ was used as a basis for a probabilistic numerical model surrogate. The formulation of the PNM surrogate is as follows:

$$\begin{aligned} \mu_y(\mathbf{x}) &= h(\mathbf{x}) - c\sigma(\mathbf{x}) \\ \sigma_y^2(\mathbf{x}) &= e^{-\ell}|h(\mathbf{x})| \end{aligned} \tag{10}$$

where c is an appropriately chosen scaling constant or a function, which controls the calibration of the PNM surrogate uncertainty, and ℓ is a parameter, which controls the available computational budget by adjusting the scale of the uncertainty of the output. Through the formulation in Eqs. (10), the computational budget is reflected not only on the spread of the output distribution, but also on the distance between its mean and the response of the deterministic model. In the limit of infinite computational budget the uncertainty vanishes, that is $\lim_{\ell \to \infty} dG_y \to \delta(h(\mathbf{x}))$.

Hristov and DiazDelaO, 2023 observed that setting $t^* = 230$, gives a small failure domain $\mathcal{F}$, with two disjoint subdomains. The authors further noted that at low to moderate computational budgets, $\mathcal{F}$ cannot be reliably discovered by SuS using the means of the PNM predictions. In the original paper, it has already been shown that P-SuS can detect both failure sub-domains at any ℓ. Because IP-SuS uses the same MCMC algorithm for generating seeds and exploring the space, as P-SuS, the details about these properties of the algorithm will not be discussed further. However, the P-SuS estimator heavily underestimates p_F for outputs with large uncertainty. Moreover, the variance of the estimator is not well calibrated, that is, it does not contain the value of the original SuS estimator for all ℓ and it does not decrease in proportion to the reduction in the distance between the SuS and P-SuS estimators. This results in a final estimator of the probability of failure whose uncertainty is either too narrow or too wide.

The experiment was set up with $p_0 = 0.1$, $N = 1000$ and $\epsilon = 1 \times 10^{-4}$. Depending on ℓ, which was taken as $\ell = \{0.01, 0.05, 0.10, 0.15, 0.25, 0.50, 0.75, 1.0, 2.5, 5.0, 7.5, 10\}$, IP-SuS required anywhere between two and six conditional levels. We emphasise again that this part of the algorithm is entirely inherited from P-SuS. Unlike P-SuS, however, where the probability of each conditional level is described by a normal distribution, in IP-SuS the levels probabilities are p-boxes. Figure 1 shows the p-boxes of $\hat{p}_{F_k}$ (left and centre) and the p-box for $\hat{p}_{F_n}$ which is the estimation of p_F conditional on the last intermediate level, for $\ell = 5$. It can be seen that $\hat{p}_{F_1}$ and $\hat{p}_{F_2}$ contain the value of $p_0 = 0.1$, set at the beginning of the algorithm.

Once all p-boxes for $\hat{p}_{F_k}$ and $\hat{p}_{F_n}$ are multiplied under Fréchet rules the final estimator, $\hat{p}_F^{IP-SuS}$ is another p-box like those shown on Figure 2. There, $\hat{p}_F^{IP-SuS}$ for three different budget levels are depicted, starting from the lowest level of computational budget, $\ell = 0.01$, through a moderate budget, $\ell = 5$ to a high-fidelity response at $\ell = 10$. It can be seen that the probability of failure $\hat{p}_F^{SuS} \approx 0.0066$ estimated using the deterministic version of $h(\cdot)$, and shown as a red, dashed line, is contained within the p-boxes for all three values of ℓ, which is not the case for P-SuS at $\ell = 0.01$. In addition, it is evident that the bounds of the p-boxes become tighter as the uncertainty of the output decreases. This effect can further be observed in Figure 3, which shows the convergence of $\hat{p}_F^{IP-SuS}$ with increasing computational budget. The deterministic failure probability estimated using regular SuS and depicted as a red line is contained between the dashed lines representing the extent of the 95% central probability interval and the black solid lines depicting the imprecise median of the p-box of $\hat{p}_F^{IP-SuS}$ for most values of ℓ. For large uncertainties the bounds are wide,

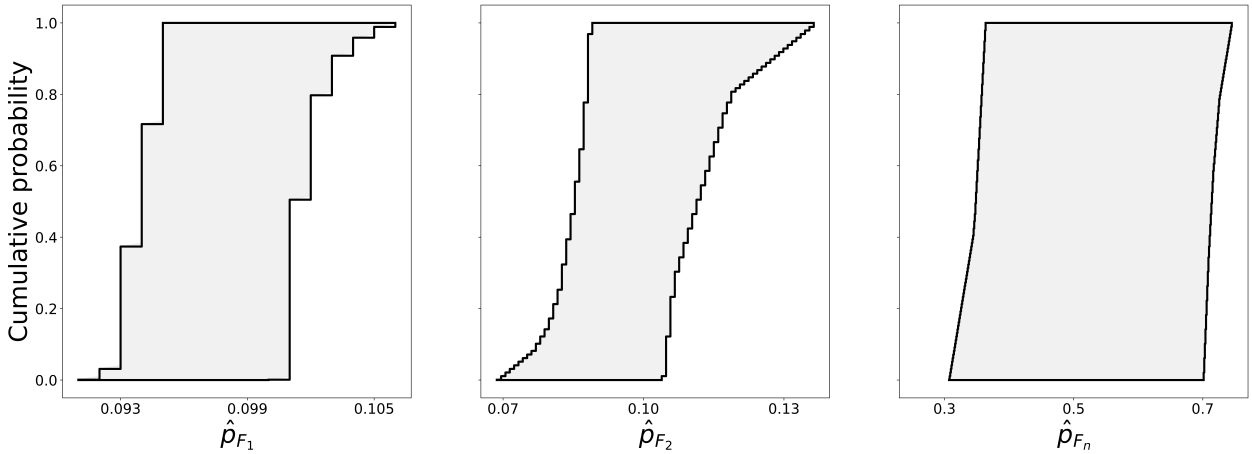
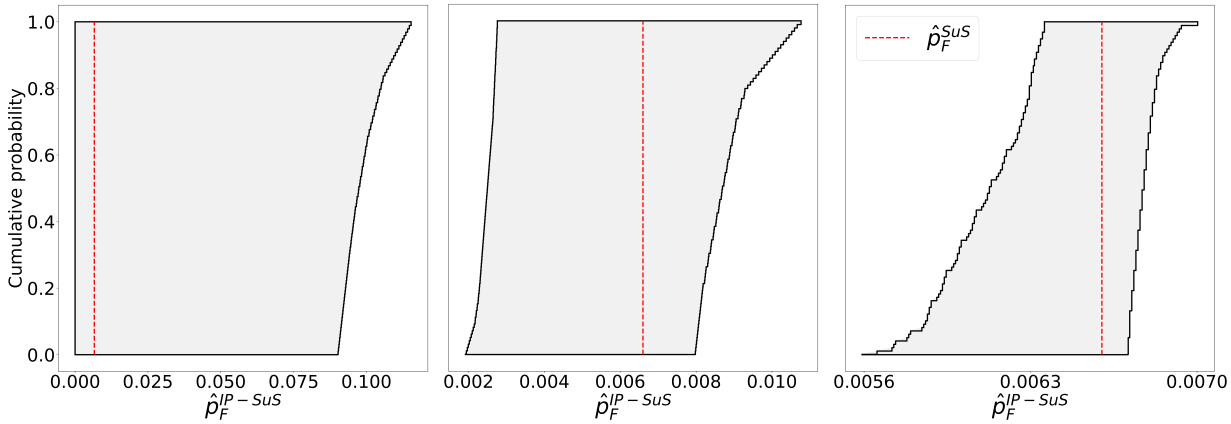


Figure 1. Probability boxes for each of the two conditional levels of IP-SuS and the p-box for $p_{F_n} = \mathbb{P}(F|F_2)$. Number of samples per level $N = 1000$; nominal level probability, $p_0 = 0.1$.



(a) (b) (c)

Figure 2. Probability boxes for $\hat{p}_F^{IP-SuS}$ at budget levels corresponding to $\ell = \{0.01, 5, 10\}$ in (Hristov and DiazDelaO, 2023). The probability of failure from subset simulation, $\hat{p}_F^{SuS}$ of the deterministic code, shown with a red, dashed line is contained in the probability boxes. Number of samples per level $N = 1000$; nominal level probability, $p_0 = 0.1$.

and as the uncertainty in the responses decreases, the bounds become tighter. It can be seen, however, that the bounds miss $\hat{p}_F^{SuS}$ for $\ell = 7.5$ but contain it again for $\ell = 10$.

To investigate the possible causes for this problem, we ran 100 repeated analyses for $\ell = 7.5$. Figure 4(a) presents the results, with error bars depicting 95% central probability intervals. It can immediately be observed that the random component of the algorithm, that is due to generating the samples across all intermediate levels up until $\mathcal{F}$, has a marked effect on the performance of $\hat{p}_F^{IP-SuS}$. Error bars in black contain the target $\hat{p}_F^{SuS}$, while those in red do not. The former constitute 51% of all repeats. This means that in 49% of the cases IP-SuS will not contain $\hat{p}_F^{SuS}$. To benchmark this, a similar analysis was carried out with P-SuS. However, because this algorithm only returns

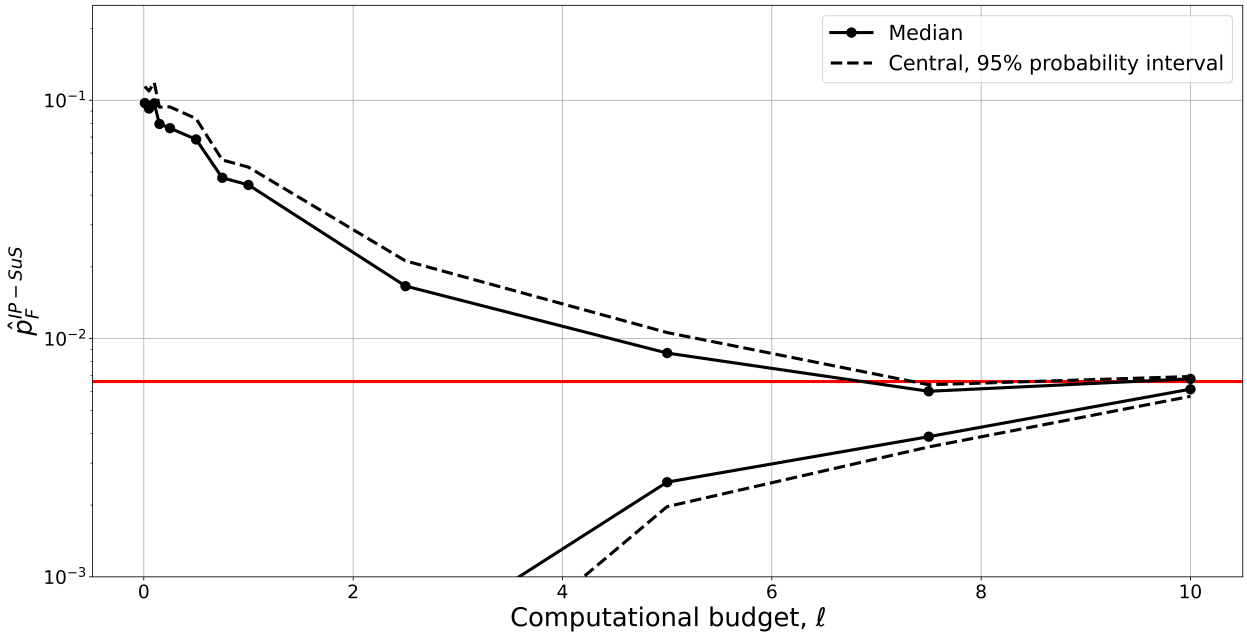
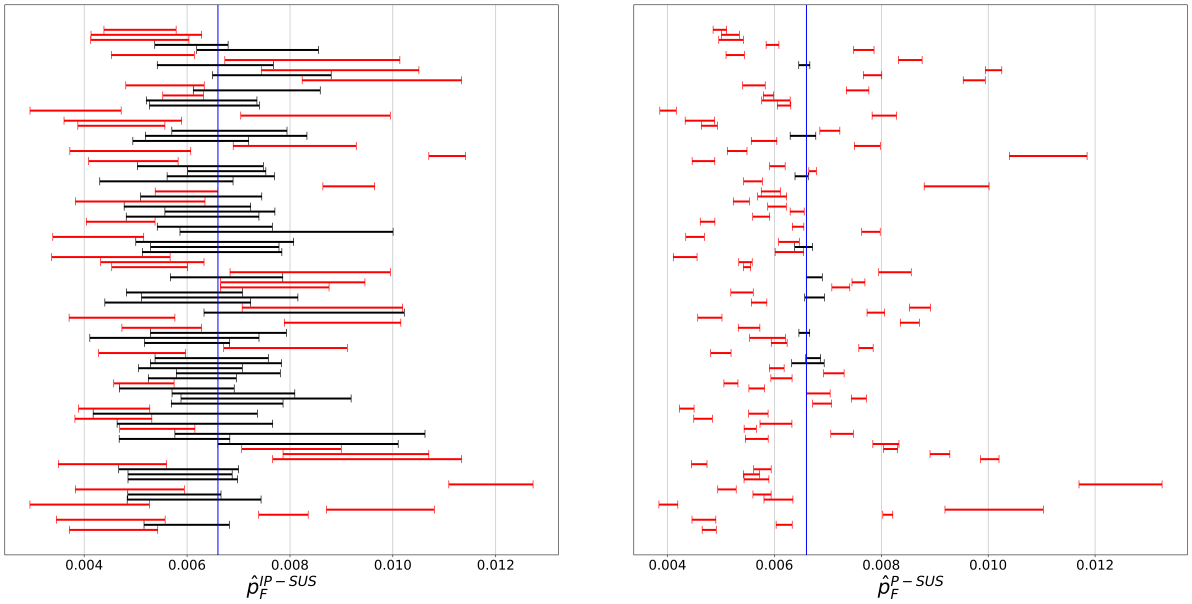


Figure 3. Convergence of $\hat{p}_F^{IP-SuS}$ with increasing computational budget. Black, solid lines show the imprecise median of the final estimator p-boxes; black, dashed lines depict the extent of the 95% central probability interval; red, solid line shows $\hat{p}_F^{SuS}$. Number of samples per level $N = 1000$; nominal level probability, $p_0 = 0.1$.

mean and variance about $\hat{p}_F^{P-SuS}$, we constructed bounds, using the celebrated Chebyshev-Cantelli one-sided inequality (Ghosh, 2002) and then computed the 95% central probability interval in a way analogous to that for $\hat{p}_F^{IP-SuS}$. The results are shown on Figure 4(b). In comparison, $\hat{p}_F^{P-SuS}$ contains $\hat{p}_F^{SuS}$ in less than 10% of the time. Regardless of this large discrepancy between the two methods, the coefficient of variation for the results from the two algorithms is very similar, coming at ≈23.7% for IP-SuS and ≈24.4% for P-SuS. This strongly suggests that the problem comes from P-SuS base, but this needs to be investigated further.

5. Conclusion

This paper presented an application of dependence bounds analysis to improve the uncertainty calibration of the failure probability estimator of P-SuS - an extension to subset simulation designed to work with probabilistic numerical models. The resulting method, termed imprecise P-SuS (IP-SuS) relaxes the assumptions of independence at two levels in the original P-SuS algorithm, among the seeds at each conditional level and among the levels themselves. The summation and product, respectively, between uncertain numbers are redefined to use Fréchet-Hoeffding bounds on convolutions, thus making no assumption about particular dependence among the operands. As a result the estimator produced by IP-SuS is a probability box, whose bounds tighten as the computational budget increases, accounting for the decrease in the uncertainty of the outputs produced by the



(a) (b)

Figure 4. Comparison between 95% probability intervals of (a) $\hat{p}_F^{IP-SuS}$ and (b) $\hat{p}_F^{P-SuS}$ from 100 repeated analyses for $\ell = 7.5$. Black error bars depict intervals which contain the value of the SuS estimator, $\hat{p}_F^{SuS}$, shown as a blue vertical line, while red ones depict intervals that do not. Number of samples per level $N = 1000$; nominal level probability, $p_0 = 0.1$.

PNM. For large uncertainties the width of the p-box of $\hat{p}_F^{IP-SuS}$ is large, but contains the credible region, which is a more desirable property than having tight bounds away from it. The effectiveness of the proposed modification was evaluated on the modified Branin function, which is often used as a benchmark for methods in uncertainty quantification.

The main direction for future work is aimed at properly calibrating the final estimator of the probability of failure, whose p-box should contain the true value for any computational budget. Moreover, the variability of the estimator should also be investigated and efforts are ongoing to make $\hat{p}_F^{IP-SuS}$ more stable.

Acknowledgements

This research was conducted with the support of the Swiss National Science Foundation and the Bulgarian Ministry of Education and Science under Grant Agreement No. IZPYZ0_228861 for the project “Automatic maximally rigorous uncertainty quantification to enable design for certification” (unCertify). The work also received funding from the GATE project funded by the Horizon 2020 WIDESPREAD-2018-2020 TEAMING Phase 2 programme under grant agreement no. 857155 and the programme ”Research, Innovation and Digitalization for Smart Transformation” 2021-2027 (PRIDST) under grant agreement no. BG16RFPR002-1.014-0010-C01.

References

Au, S.-K. and Beck, J. L. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277.

Ferson, S., Nelsen, R. B., Hajagos, J., Berleant, D. J., Zhang, J., Troy Tucker, W., Ginzburg, L. R., and Oberkampf, W. L. (2004). Dependence in probabilistic modeling, Dempster-Shafer theory and probability bounds analysis. Technical Report SAND2002-4015, Sandia National Laboratories (SNL), Albuquerque, (NM), United States.

Ghosh, B. K. (2002). Probability inequalities related to markov's theorem. *The American Statistician*, 56(3):186–190.

Giovanis, D. G. and Shields, M. D. (2021). Imprecise subset simulation for reliability analysis. In *Conference: 9th International Workshop on Reliable Engineering Computing (REC2021)*.

Hennig, P., Osborne, M. A., and Girolami, M. (2015). Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142.

Hristov, P. and DiazDelaO, F. (2023). Subset simulation for probabilistic computer models. *Applied Mathematical Modelling*, 120:769–785.

Ibrahim, Y. (1991). Observations on applications of importance sampling in structural reliability analysis. *Structural Safety*, 9(4):269–281.

Kinnear, H. J. and DiazDelaO, F. (2025). Niching subset simulation. *Probabilistic Engineering Mechanics*, 79:103729.

Miao, Q. and Low, Y. M. (2026). Accurate variance estimation for subset simulation incorporating intrachain, interchain, and interlevel correlations. *Structural Safety*, 120:102690.

Naess, A., Leira, B., and Batsevych, O. (2009). System reliability analysis by enhanced Monte Carlo simulation. *Structural Safety*, 31(5):349–355.

Nelsen, R. B. (2006). *Definitions and Basic Properties*, pages 7–49. Springer New York, New York, NY.

Oberkampf, W. L. and Roy, C. J. (2025). *Verification, Validation, and Uncertainty Quantification in Scientific Computing*. Cambridge University Press, 2 edition.

Padmanabhan, D., Agarwal, H., Renaud, J. E., and Batill, S. M. (2006). A study using Monte Carlo simulation for failure probability calculation in reliability-based optimization. *Optimization and Engineering*, 7(3):297–316.

Pradlwarter, H., Schuëller, G., Koutsourelakis, P., and Charmpis, D. (2007). Application of line sampling simulation method to reliability benchmark problems. *Structural Safety*, 29(3):208–221.

Williamson, R. (1989). *Probabilistic Arithmetic*. PhD thesis, University of Queensland.

Zhao, Y.-G. and Ono, T. (1999). A general procedure for first/second-order reliabilitymethod (form/sorm). *Structural Safety*, 21(2):95–112.