XLVI CILAMCE
VITÓRIA - ES - BRAZIL - 2025
XLVI IBERO-LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGENEERING

Structural Reliability Analysis of Reinforced Concrete: A Comparison of MCS, IS, FORM and Subset Simulation in Julia

Kristian Yuiti Matsushita[1], Herbert Martins Gomes[2], Felipe Schaedler de Almeida[1], Marius Bittner[3], Lukas Fritsch[3]

[1]*Department of Civil Engineering, Federal University of Rio Grande do Sul*
Av. Osvaldo Aranha, 99, Porto Alegre, RS 90035-190, Brazil
kristian.matsushita@ufrgs.br, felipe.almeida@ufrgs.br
[2]*Mechanical Engineering Department, Federal University of Rio Grande do Sul*
Av. Sarmento Leite, 425, Porto Alegre, RS 90150-170, Brazil
herbert@mecanica.ufrgs.br
[3]*Institute for Risk and Reliability, Leibniz University Hannover*
Callinstr. 34, Hannover, 30167, Lower Saxony, Germany
bittner@irz.uni-hannover.de, fritsch@irz.uni-hannover.de

Abstract. In civil engineering, estimating the probability of rare events is essential for ensuring the safety and reliability of structural systems. Scenarios such as failures in reinforced concrete elements, progressive collapse, and performance-based design require accurate evaluation of low-probability events. Traditional methods like Monte Carlo Simulation (MCS), Importance Sampling (IS) and the First-Order Reliability Method (FORM) are widely used due to their theoretical foundation and practical simplicity. However, MCS becomes computationally prohibitive for rare and high dimensional events, while FORM may lose accuracy in nonlinear or non-Gaussian problems, which are common in structural analysis. Subset Simulation (SuS) overcomes these limitations by decomposing the rare event into a sequence of intermediate events, sampled efficiently through Markov Chain Monte Carlo (MCMC) techniques. This work evaluates the performance of MCS, IS, FORM, and SuS within a package for uncertainty quantification in within the Julia programming language, which offers high-performance computing capabilities. The methods are compared based on accuracy, computational cost, and robustness across a range of examples, including benchmark problems and structural models of reinforced concrete. Results indicate that SuS delivers accurate estimates with significantly fewer simulation samples.

Keywords: Structural Reliability, Reinforced Concrete, Subset Simulation, Monte Carlo Simulation

1 Introduction

Uncertainties in material properties, geometric parameters and loading conditions play a critical role in determining the safety and serviceability of Reinforced Concrete (RC) structures. While deterministic safety factors have historically been employed to account for these variability, Ellingwood and Galambos [1] advocate for probability-based approaches as a more rigorous and rational alternative. These methods enable designers to quantify the reliability of structural systems by explicitly modeling the randomness inherent to structural behavior. Several studies have contributed to the advancement of reliability analysis techniques in reinforced concrete structures. These investigations encompass comparisons between design codes, evaluation of seismic performance, modeling of spatial variability, and system-level reliability under unforeseen loading conditions. Stucchi and Santos [2] presented probabilistic comparisons between design standards and demonstrated the feasibility of reliability-based assessments in conventional design procedures. Abdelouafi et al. [3] have explored the application of nonlinear finite element models and different reliability methods to evaluate structural performance under seismic actions. The spatial variability of concrete properties was accounted by Gomes and Awruch [4] using stochastic field. Junior et al. [5] discussed system-level evaluations of reinforced concrete frames subjected to differential settlements, where illustrate the importance of accounting for post-construction effects that may compromise structural reliability. This study contributes to the advancement of reliability analysis of reinforced concrete structures in two main aspects: (i) the efficiency of the Subset Simulation (SuS) method is compared to more traditional approaches, such as Monte Carlo (MC), First-Order Reliability Method (FORM), and Importance Sampling (IS), for estimating the

failure probability of rectangular RC sections under pure bending, for which an analytical limit state function is available; and (ii) shows the application of the `UncertaintyQuantification.jl` package, implemented in the Julia programming language, for the uncertainty quantification of RC elements.

2 Reliability Methods

In structural engineering, reliability analysis aims to quantify the probability that a system or component will fulfill its intended function without failure, taking into account the inherent uncertainties in loads, material properties, and geometry. Typically, these uncertainties are modeled as Random Variable (RV) X_i with a specific probability density function. Structural safety is assessed using a limit state function $g(\mathbf{X})$, where $\mathbf{X} = [X_1, X_2, \ldots, X_n]$ is a vector with n random variables respresenting the uncertain quantities. The system is considered safe if $g(\mathbf{X}) > 0$ and the probability of failure is given by $P_f = \mathbb{P}[g(\mathbf{X}) \leq 0]$.

Monte Carlo Simulation is one of the most straightforward and robust method for reliability analysis. The probability of failure is estimated by [6]

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[g(\mathbf{X}^{(i)}) \leq 0], \tag{1}$$

where $\mathbb{I}[\cdot]$ is the indicator function. Although conceptually simple and robust, MCS becomes computationally prohibitive for small P_f, since, $N > 10/P_f$ is required for convergence, resulting in simulations with millions of samples in typical structural failure problems, which motivates the development of more efficient methods.

Importance Sampling is a variance reduction technique that improves Monte Carlo estimates by drawing samples from a biased distribution concentrated near the failure domain [7]. Instead of sampling directly from the original distribution $f_{\mathbf{X}}(\mathbf{x})$, samples $\mathbf{V}^{(i)}$ are drawn from an alternative distribution $f_h(\mathbf{x})$, and the failure probability is estimated as

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[g(\mathbf{V}^{(i)}) \leq 0] \cdot \frac{f_{\mathbf{X}}(\mathbf{V}^{(i)})}{f_h(\mathbf{V}^{(i)})}. \tag{2}$$

FORM [8] provides a more efficient approximation based on transforming the random space $\mathbf{X}$ into a standard normal space $\mathbf{U}$. A search for the point on the failure surface closest to the origin (Most Probable Point) is performed to obtain the reliability index $\beta = \min\{\|\mathbf{u}\| \mid g(\mathbf{u}) = 0\}$. The failure probability is estimated by $P_f \approx \Phi(-\beta)$, where $g(\mathbf{U})$ is linearized at the Most Probable Point and $\Phi(\cdot)$ denotes a standard normal cumulative distribution function. Inaccurate results may be obtained in problems with highly nonlinear limit state functions.

Subset Simulation [9] is particularly effective for estimating small failure probabilities in high-dimensional spaces. Unlike MCS, which suffers from inefficiency for rare events, SuS decomposes the rare event $\{g(\mathbf{X}) \leq 0\}$ into a sequence of more probable intermediate events: $\mathcal{F}_m \subset \mathcal{F}_{m-1} \subset \cdots \subset \mathcal{F}_1 \subset \mathcal{F}_0 = \mathbb{R}^n$. The total failure probability is expressed as a product of conditional probabilities: $P_f = \mathbb{P}(\mathcal{F}_m) = \prod_{j=1}^{m} \mathbb{P}(\mathcal{F}_j \mid \mathcal{F}_{j-1})$. The procedure starts with a standard Monte Carlo Simulation (MCS) at level $j = 0$, generating N samples. The first threshold b_1 is defined so that a fixed fraction p_0 of samples satisfy $g(\mathbf{X}) \leq b_1$, forming the intermediate subset $\mathcal{F}_1$. In the following levels ($j \geq 2$), samples are generated conditionally using Markov Chain Monte Carlo (MCMC), starting from the $p_0 N$ samples that met the previous threshold. Each new level defines a threshold b_j such that approximately p_0 of the samples satisfy $g(\mathbf{X}) \leq b_j$, progressively approaching the rare failure domain. The final estimate of the failure probability is given as:

$$\hat{P}_f = \prod_{j=1}^{m} \frac{n f_j}{N}, \tag{3}$$

where m is the number of levels and $n f_j$ is the number of samples in the last level satisfying $g(\mathbf{X}) \leq 0$.

3 Numerical Examples

The application of structural reliability methods is illustrated through three numerical examples involving Reinforced Concrete (RC) rectangular cross-sections subjected to bending. The probability of failure is computed based on the limit state function

$$g(X) = \theta_R M_R(X) - \theta_S M_S(X), \tag{4}$$

where θ_R and θ_S are random variables representing model uncertainties associated with bending moment strength (M_R) and load effects (M_S), respectively. The applied moment $M_S = M_G + M_Q$ comprises dead (M_G) and live (M_Q) load components, which are defined based on a target design moment M_d. The characteristic values are given by $M_{G_k} = M_d/(\gamma_g + \gamma_q\chi/(1-\chi))$ and $M_{Q_k} = M_d/(\gamma_g(1-\chi)/\chi + \gamma_q)$ with $\gamma_g = \gamma_q = 1.4$, according to Brazilian standards, and a predefined load ratio $\chi = M_{Q_k}/(M_{G_k} + M_{Q_k})$.

RV	Unit	Example 1	Example 2	Example 3
b	cm	$\sim$ N(12, 0.6)	$\sim$ N(20, 1.20)	$\sim$ N(20, 1.20)
h	cm	–	$\sim$ N(50, 2.25)	$\sim$ N(50, 2.25)
d	cm	$\sim$ N(30, 1.5)	–	–
d_l	cm	–	$\sim$ LN(4.5, 1.1)	$\sim$ LN(4, 1.1)
d_u	cm	–	–	$\sim$ LN(3.315, 1.1)
k	–	$\sim$ N(0.5, 0.05)	–	–
f_c	MPa	$\sim$ N(40, 6)	$\sim$ N(1.17f_{ck}, 0.15f_{ck})	$\sim$ N(1.17f_{ck}, *)
f_y	MPa	$\sim$ N(360, 36)	$\sim$ N(1.08f_{yk}, 0.05f_{yk})	$\sim$ N(1.08f_{yk}, 0.05f_{yk})
A_S	cm^2	$\sim$ LN(2.26, 0.113)	–	–
M_S	kN·cm	$\sim$ EX-I(10, 3)	–	–
M_G	kN·m	–	$\sim$ N(G_k, 0.1G_k)	$\sim$ N(G_k, 0.1G_k)
M_Q	kN·m	–	$\sim$ EX-I(0.93Q_k, 0.186G_k)	$\sim$ EX-I(0.93Q_k, 0.186G_k)
θ_S	–	–	$\sim$ LN(1.00, 0.05)	$\sim$ LN(1.00, 0.05)
θ_R	–	–	$\sim$ LN(1.00, 0.05)	$\sim$ LN(1.00, 0.05)

*$\sigma_{f_c} = 0.15\exp[-0.036(f_{ck} - 20\text{MPa})]1.17f_{ck}$

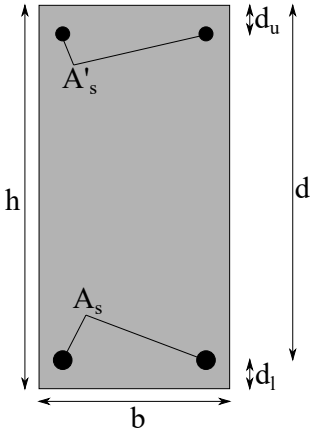


Figure 1. Random variables and cross section

Three numerical examples of increasing complexity and slightly increasing number of involved random variables are solved for the evaluation of the performances of the structural reliability methods considered in this work. The random variables and the geometric configuration of the cross-section are shown in Fig. 1, where N(a, b), LN(a, b), EX-I(a, b) denote Normal, Log-Normal and Exponential-I distributions, respectively, with mean a and standard deviation b.

Example 1 is adapted from Gong et al. [10], where no model uncertainties are considered ($\theta_R = \theta_S = 1$) and the applied moment is given by the random variable M_S (see Fig. 1). The RC section has only bottom reinforcement ($A'_S = 0$) and the bending moment strength is given by

$$M_R(\mathbf{X}) = A_s f_y \left(d - k\frac{A_s f_y}{b f_c}\right), \tag{5}$$

Example 2 was presented by Santos et al. [11] for the reliability assessment of the design code NBR 6118:2014. For the case with design parameters $\chi = 0.2$ and $M_d = 159.14\,\text{kN.m}$, the the cross-section is designed with a single bottom reinforcement layer with $A_s = 9.45\,\text{cm}^2$. The bending moment strength is given by

$$M_R(\mathbf{X}) = A_s f_y \left(h - d_l - 0.5\frac{A_s f_y}{0.85 b f_c}\right), \tag{6}$$

Example 3 reproduces the formulation by Leite and Gomes [12], which investigates an RC section reinforced on both the top ($A'_s = 0.63\,\text{cm}^2$) and bottom ($A_s = 6.30\,\text{cm}^2$). The bending strength is evaluated by verifying stress compatibility between tensile ($\sigma_{s,lower}$) and compressive ($\sigma'_{s,upper}$) reinforcement layers. The loading is composed of $M_S = M_G + M_Q$, with $\chi = 0.15$ and $M_d = 115.5\,\text{kN.m}$. The resistance is determined based on the conditions of axial and moment equilibrium. The axial equilibrium is given by $A_s\sigma_{s,\text{lower}} - A'_s\sigma'_{s,\text{upper}} = 0.85 f_{cd} \cdot b \cdot y$, while the moment equilibrium is expressed as $M_u = 0.85 f_{cd} \cdot b \cdot y(h - d_l - y/2) + A'_s\sigma'_{s,\text{upper}}(h - d_l - d_u)$. The unknowns in the formulation are the position of the neutral axis y, and the stresses in the reinforcement layers $\sigma_{s,\text{lower}}$ and $\sigma'_{s,\text{upper}}$. The RC domain limits are: $y_{23} = 0.207(h - d_l)$ and $y_{\text{lim}} = (0.8(h - d_l)\varepsilon_{cu})/(\varepsilon_{cu} + \varepsilon_{yd})$. The procedure to determine the resistance is outlined below:

1. Assuming $\sigma_{s,\text{lower}} = \sigma'_{s,\text{upper}} = f_{yd}$, and solving the axial equilibrium to obtain an initial estimate of y;
2. For $y \le y_{23}$, use $\varepsilon' = \frac{0.01(y - 0.8d_u)}{0.8(h - d_l) - y}$;

3. For $y_{23} < y \leq y_{\text{lim}}$, use $\varepsilon' = \frac{\varepsilon_{cu}(y-0.8d_u)}{y}$;
4. For $y > y_{\text{lim}}$, compute both $\varepsilon = \frac{\varepsilon_{cu}(0.8(h-d_l)-y)}{y}$ and $\varepsilon' = \frac{\varepsilon_{cu}(y-0.8d_u)}{y}$.

The corresponding stresses are then obtained as $\sigma'_{s,\text{upper}} = \min(E_s\varepsilon', f_{yd})$ and, when applicable, $\sigma_{s,\text{lower}} = \min(E_s\varepsilon, f_{yd})$. The axial equilibrium is recalculated to find y then the design moment resistance M_u is computed from the moment equilibrium equation.

4 Computational Framework: Julia and UncertaintyQuantification.jl

All simulations were performed using the Julia programming language (Bezansony et al. [13]), chosen for its balance between high performance and code simplicity. Julia offers execution speeds comparable to compiled languages while maintaining a syntax familiar to users of scientific computing. The reliability analyses were performed using the methods implemented in the `UncertaintyQuantification.jl` package, which is an open-source framework designed for uncertainty quantification (Behrensdorf et al. [14]).

Figure 2 presents the code used to estimate the probability of failure for Example 1. Lines 4 to 6 define the distribution parameters of the random variables, including both Lognormal and Gumbel distributions. These variables are instantiated using the `RandomVariable` type in lines 9 to 15 and grouped into a vector in line 18, which serves as input for the reliability analysis. The problem model is defined in lines 21 to 22, and the limit state function where failure is identified when $f(x) < 0$ is implemented in line 25.

```
1  using UncertaintyQuantification
2
3  # Define distribution parameters for lognormal and gumbel
4  p1, p2 = distribution_parameters(0.01, 0.003, Gumbel)
5  p3, p4 = distribution_parameters(226e-6, 11.3e-6, LogNormal)
6  p5, p6 = distribution_parameters(40, 6, LogNormal)
7
8  # Define probabilistic variables (units in comments)
9  Ms = RandomVariable.(Gumbel(p1, p2), :Ms)            # unit: MNm
10 d  = RandomVariable.(Normal(0.3, 0.015), :d)         # unit: m
11 fy = RandomVariable.(Normal(360, 36), :fy)           # unit: MPa
12 As = RandomVariable.(LogNormal(p3, p4), :As)         # unit: m2
13 k  = RandomVariable.(Normal(0.5, 0.05), :k)
14 b  = RandomVariable.(Normal(0.12, 0.006), :b)        # unit: m
15 fc = RandomVariable.(LogNormal(p5, p6), :fc)         # unit: MPa
16
17 # Define the Random Variable
18 x = [Ms, d, fy, As, k, b, fc]
19
20 # Define the performance function
21 f = Model(df -> df.d .* df.fy .* df.As
22         .- df.k .* df.fy.^2 .* df.As.^2 ./ (df.b .* df.fc) .- df.Ms, :f)
23
24 # Limit state function: failure occurs when f is negative
25 limitstate(df) = df.f
26
27 # Application for reliability methods:
28 method = FORM()
29 pf_FORM, beta, dp = probability_of_failure(f, limitstate, x, method)
30
31 method = MonteCarlo(N_samples)
32 pf_MCS, std_MCS, samples_MCS = probability_of_failure(f, limitstate, x, method)
33
34 method = ImportanceSampling(N_samples)
35 pf_IS, std_IS, samples_IS = probability_of_failure(f, limitstate, x, method)
36
37 method = SubSetSimulation(N_samples, 0.1, 10, Uniform(-1,1))
38 pf_SuS, std_SuS, samples_SuS = probability_of_failure(f, limitstate, x, method)
```

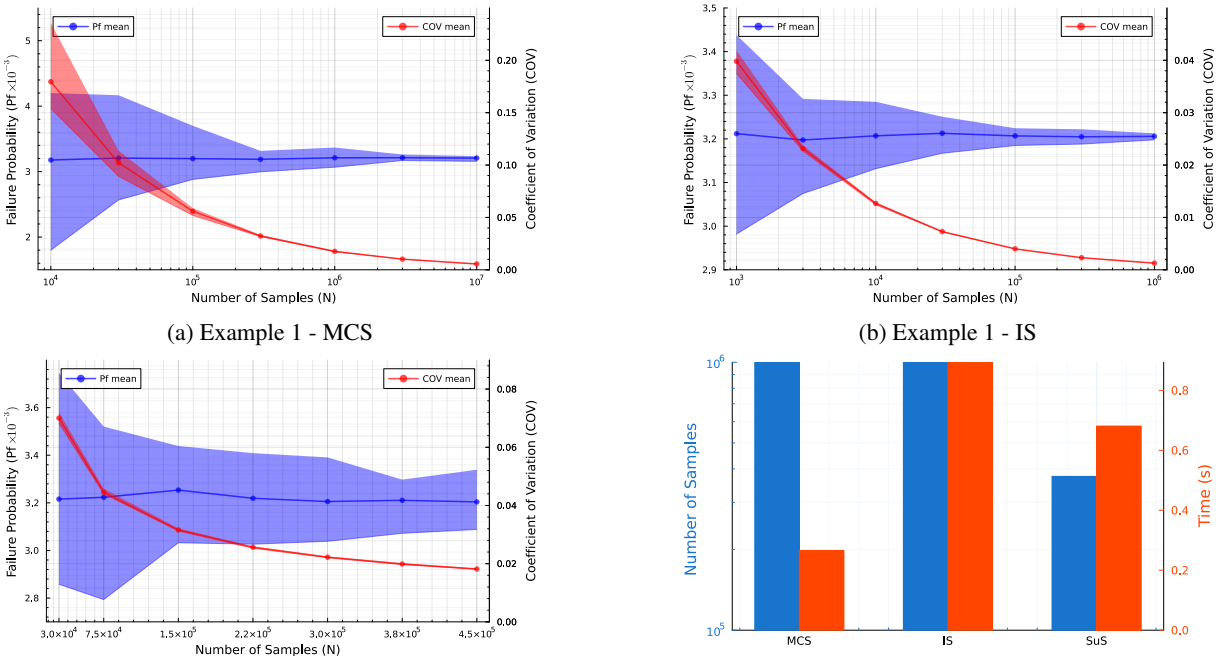
Figure 2. Example of the code to define the variables for Example 1 in Julia

Several reliability methods are then applied: FORM (lines 28–29), MCS (lines 31–32), IS (lines 34–35), and SuS (lines 37–38). In each case, the function `probability_of_failure` returns the estimated probability of failure (`pf`), for sampling methods, it also returns the standard deviation (`std`) and generated samples. From these outputs, the coefficient of variation is obtained as COV $=$ std/pf. In the case of FORM, as an approximation method, the MPP (`dp`) is also computed.

5 Results

A robust statistical comparisons of the reliabilty analysis methods considered in this work is performed based on 30 independent runs of each method for each example. The results are summarized in terms of the probability of failure (P_f), coefficient of variation (COV), and processing time (on an I7 13700HX laptop with 16 GB of RAM).The mean, minimum an maximum values of P_f and COV obtained with MCS, IS and SuS are presented in Figure 3, Figure 4 and Figure 5 for simulations with increasing number of samples.

In Example 1, all methods estimated $P_f \approx 3.21 \times 10^{-3}$, except for FORM with a slightly lower value of 3.01×10^{-3}, corresponding to $\beta = 2.75$. MCS (Figure 3a) served as a reference but required 10^7 samples and over 3 seconds of computation. IS (Figure 3b) achieved high precision ($COV = 0.0013$) with 10^6 samples and completed in less than one second. SuS (Figure 3c) further reduced the count of samples to 4.5×10^5, completing the analysis in 0.8 seconds with a modest increase in COV (0.018). In Figure 3d, SuS required the lowest number of samples to achieve $COV \leqslant 0.04$. Despite this, MCS reached the same level of precision in a slightly shorter computational time, indicating a more efficient processing per sample in this problem.



(a) Example 1 - MCS

(b) Example 1 - IS

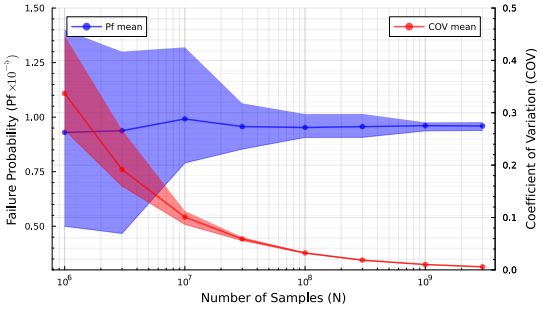
(c) Example 1 - SuS

(d) Example 1 - $COV \leqslant 0.02$

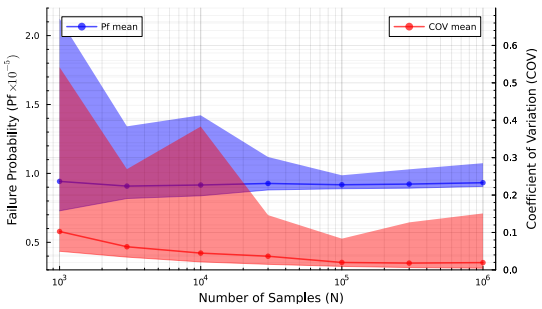
Figure 3. Results for Example 1

In Example 2, the failure probability is in the order of 10^{-6}. FORM estimated $P_f = 4.11 \times 10^{-6}$ ($\beta = 4.46$), which is slightly lower than the prediction of simulation methods: 9.59×10^{-6} for MCS, 9.33×10^{-6} for IS, and 9.76×10^{-6} for SuS. As shown in Figure 4a, MCS became computationally expensive, requiring 3×10^9 samples and over 10 minutes. IS (Figure 4b) delivered results in just over one second with a COV of 0.019. SuS (Figure 4c) proved more efficient, using only 5.6×10^4 samples and finishing in less than two seconds, with a COV of 0.034. The figure highlights how SuS progressively concentrates samples near the failure domain through intermediate thresholds, maintaining accuracy with minimal computational effort. In Figure 4d shows that IS has achieved the $COV \leqslant 0.04$ with significantly fewer samples and shorter computational time compared to both MCS and SuS. However, SuS showed a similar order of magnitude in terms of both samples generated and runtime, indicating comparable efficiency.

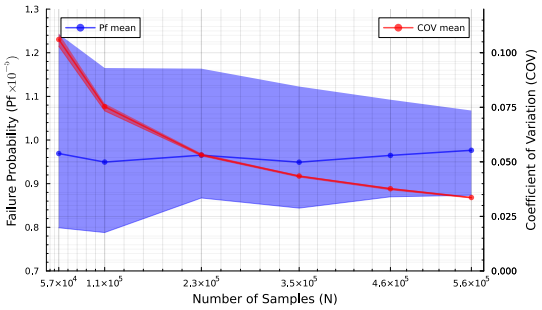
In Example 3, a very low failure probability ($\sim 10^{-8}$) was predicted by the simulation methods. Again, FORM estimation was below with $P_f = 1.07 \times 10^{-8}$ ($\beta = 5.60$). For certain number os samples, MCS (Figure 5a) returned $P_f = 0$, making COV is undefined. To obtain a meaningful estimate, it was required 10^{10} samples and over 40 minutes of computation, yielding a COV of 0.067 and confirming its impracticality in this context . IS (Figure 5b) delivered accurate results ($P_f = 2.23 \times 10^{-8}$, ($COV = 0.0069$) with 10^7 samples and a runtime of 35 seconds. SuS (Figure 5c) matched the same estimate with fewer samples (8×10^6), slightly higher COV (0.0135), and similar wall-clock time, effectively balancing accuracy and efficiency. The figures illustrate the method's ability to adaptively refine sampling near the failure region, even under highly nonlinear conditions. Figure 5d shows to achieve the precision of $COV \leqslant 0.015$, both IS and SuS performed remarkably well, showing strong



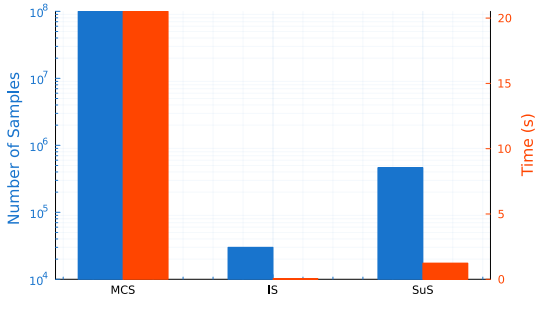
(a) Example 2 - MCS



(b) Example 2 - IS



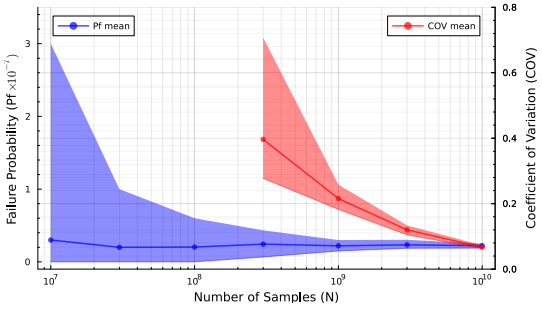
(c) Example 2 - SuS



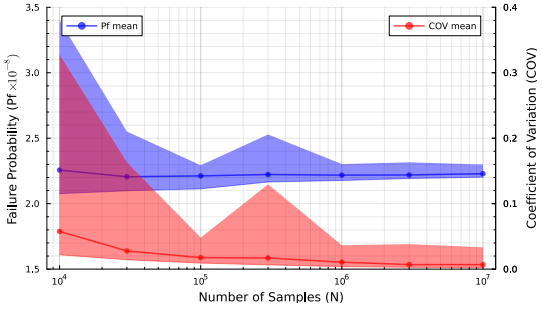
(d) Example 2 - $COV \leqslant 0.04$

Figure 4. Results for Example 2

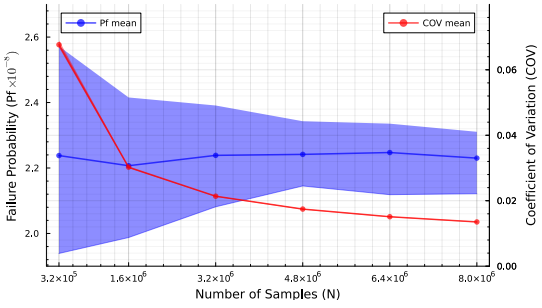
potential for application in complex structural reliability problems. In contrast, MCS required a large sample sizes, reinforcing its limitations in estimating very small probabilities of failure.



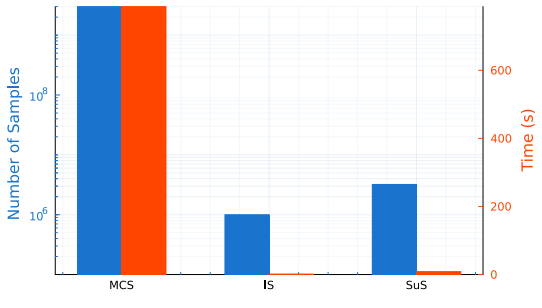
(a) Example 3 - MCS



(b) Example 3 - IS



(c) Example 3 - SuS



(d) Example 3 - $COV \leqslant 0.015$

Figure 5. Results for Example 3

Overall, the results confirm that while MCS is reliable from a statistical standpoint, its computational cost grows prohibitively in rare-event scenarios. IS is effective when the sampling distribution is well calibrated but may require tuning for complex problems. SuS consistently combines accuracy and efficiency, using fewer samples and shorter runtime across all examples. Its adaptive sampling strategy, supported by both visual and numerical evidence, reinforces its suitability for structural reliability analysis of reinforced concrete elements, particularly when implemented in high-performance computing environments such as Julia.

6 Conclusion

Four methods of reliability analysis were evaluated in terms of accuracy, computational cost, and robustness based on the solution of three problems of failure of reinforced concrete cross-sections subjected to bending moment. Although simple and statistically rigorous, MCS proved computationally expensive for rare-event scenarios. FORM was efficient in problems with nearly linear limit state functions but underestimated failure probabilities in more nonlinear cases. IS offered improved performance in moderately complex settings but relied heavily on the appropriate selection of the sampling distribution. SuS emerged as the most versatile and efficient method for estimating low failure probabilities, combining accuracy with a significant reduction in the number of simulations.

Julia high-performance capabilities, combined with the modular and intuitive structure of framework, enabled the development of reproducible and flexible simulations with minimal programming overhead. These tools proved particularly advantageous for structural reliability applications, allowing rapid prototyping, integration with optimization routines, and easy extension to different types of structural systems and probabilistic models.

Future work will explore hybrid approaches that integrate simulation-based methods with surrogate models or machine learning techniques to further reduce computational cost, especially in problems involving high-dimensional input spaces or real-time design requirements and computationaly expensive structural models.

Acknowledgements. The authors would like to express their gratitude to the National Council for Scientific and Technological Development (CNPq) and Coordination for the Improvement of Higher Education Personnel (CAPES) for their support in this research.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] B. Ellingwood and T. V. Galambos. Probability-based criteria for structural design. *Structural Safety*, vol. 1, n. 1, pp. 15–26, 1982.

[2] F. R. Stucchi and S. H. C. Santos. Reliability based comparison between aci 318-05 and nbr 6118. *Revista Brasileira de Engenharia Civil*, vol. 3, 2007.

[3] E. G. Abdelouafi, K. Benaissa, and K. Abdellatif. Reliability analysis of reinforced concrete buildings: Comparison between form and ism. *Procedia Engineering*, vol. 114, pp. 650–657, 2015.

[4] H. M. Gomes and A. M. Awruch. Reliability of reinforced concrete structures using stochastic finite elements. *Engineering Computations*, vol. 19, n. 7, pp. 764–786, 2002.

[5] W. M. P. Junior, M. H. M. Moraes, A. T. Beck, D. L. Araújo, A. P. Sarmento, and M. A. P. Sousa. Reliability analysis of reinforced concrete frames subjected to post-construction settlements. *Revista IBRACON de Estruturas e Materiais*, vol. 16, n. 5, pp. e16053, 2023.

[6] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Series in Statistics. Springer New York, 2004.

[7] R. E. Melchers. Importance sampling in structural systems. *Structural Safety*, vol. 6, n. 1, pp. 3–10, 1989.

[8] A. M. Hasofer and N. C. Lind. Exact and invariant second-moment code format. *Journal of the Engineering Mechanics Division*, vol. 100, n. 1, pp. 111–121, 1974.

[9] S.-K. Au and J. L. Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, vol. 16, n. 4, pp. 263–277, 2001.

[10] J.-X. Gong, P. Yi, and N. Zhao. Non-gradient–based algorithm for structural reliability analysis. *Journal of Engineering Mechanics*, vol. 140, n. 6, 2014.

[11] D. M. Santos, F. R. Stucchi, and A. T. Beck. Reliability of beams designed in accordance with brazilian codes. *Revista IBRACON de Estruturas e Materiais*, vol. 7, n. 5, pp. 723–746, 2014.

[12] D. D. L. Leite and H. M. Gomes. Reliability analysis of reinforced concrete sections for ultimate limit states. In *CILAMCE*, pp. 1–7, Rio de Janeiro, Brazil, 2021.

[13] J. Bezansony, A. Edelmanz, S. Karpinskix, and V. B. Shahy. Julia: A fresh approach to numerical computing. *IAM Rev.*, vol. 59, n. 1, pp. 65–98, 2017.

[14] J. Behrensdorf, A. Gray, A. Perin, J. Grashorn, M. Luttmann, M. Broggi, G. Agarwal, L. Fritsch, F. Mett, and L. Knipper. FriesischScott/UncertaintyQuantification.jl: v0.12.0 (v0.12.0). Zenodo, 2025.