Stochastic gradient descent for risk optimization

André Gustavo Carlon¹, André Jacomel Torii², Rafael Holdorf Lopez³, and
José Eduardo Souza de Cursi⁴

¹ King Abdullah University of Science & Technology (KAUST), Computer, Electrical
and Mathematical Sciences & Engineering Division (CEMSE), Thuwal 23955-6900,
Saudi Arabia.

agcarlon@gmail.com

² Latin American Institute for Technology, Infrastructure and Territory (ILATIT),
Federal University for Latin American Integration (UNILA), Av. Tancredo Neves
6731, Foz do Iguaçu, Brazil

³ Center for Optimization and Reliability in Engineering (CORE), Department of
Civil Engineering, Federal University of Santa Catarina, UFSC, Rua João Pio
Duarte, s/n, Florianópolis, Brazil

⁴ Institut National des Sciences Appliquées (INSA) de Rouen, 76801 Saint Etienne
du Rouvray CEDEX, France

Abstract. This paper presents an approach for the use of stochastic gradient descent methods for the solution of risk optimization problems. The first challenge is to avoid the high-cost evaluation of the failure probability and its gradient at each iteration of the optimization process. We propose here that it is accomplished by employing a stochastic gradient descent algorithm for the minimization of the Chernoff bound of the limit state function associated with the probabilistic constraint. The employed stochastic gradient descent algorithm, the Adam algorithm, is a robust method used in machine learning training. A numerical example is presented to illustrate the advantages and potential drawbacks of the proposed approach.

Keywords: risk optimization, Stochastic gradient descent, Chernoff bound, stochastic optimization

1 Introduction

It is often the case in engineering that uncertainties need to be considered when designing a project. It becomes even more critical when optimization is applied for project design, leading to optimization under uncertainty problems. In the optimization under uncertainty literature, two design philosophies are of particular interest for the civil/mechanical engineering community: Reliability Based Design Optimization (RBDO) and Risk Optimization (RO) [1, 2]. In RBDO, reliability constraints are imposed to the design. On the other hand, in RO, expected costs of failure are explicitly incorporated in the objective function, and consequently, the reliability level is implicitly defined during the optimization procedure.

In this paper, we focus on the solution of RO problems, whose most critical step is the evaluation of the probabilities of failure (or reliability analysis) of the engineering systems under analysis. Usually, the methods used in reliability analysis rely on approximations of the limit-state equations or in Monte Carlo (MC) sampling. It well known that the former introduces bias in the optimization problem and the later leads high computational cost. Since a large number of reliability analyses are required for design optimization, if MC sampling is used, the optimization procedure can become inviable.

Sampling based schemes have been historically employed for the solution of RO problems [3–5], which normally require a huge computational cost. Recently, approximate models were developed for this purpose. For example, Torii, Lopez, and Miguel [6] successfully proposed a gradient-based polynomial chaos expansion as a surrogate model for the state-limit functions and their gradients in the solution of RO problems. However, these approaches may become inviable for large scale problems, due to, for instance, the curse of dimensionality inherent of surrogate models. On the other hand, considering approximate models for RO, another approach by Torii et al. [7], consists on casting the RO problem as a sequence of RBDO problems and employing a performance measure approach to solve them. Although computationally efficient, this approach is based on First Order Reliability Methods, thus, introducing bias in the solution if the limit state function is not linear or the random variables are not independent and Gaussian.

For the general case of RO, where linearity of the state-limit functions can not be assumed and the number of random variables is large, there are not many alternatives to using MC sampling for the computation of failure probabilities. In this context, we propose in this paper an initial study for the use of stochastic gradient descent methods [8] for the solution of RO problems. Stochastic gradient methods are iterative methods used to optimize the expected value of functions by using as little as one singleton gradient sample per iteration. In engineering problems, stochastic gradient methods were recently applied for the solution of optimal design of experimental problems [9] and the robust design of passive control devices [10], reaching promising results. In this paper, we propose the construction of the RO problem in a manner that it avoids the high-cost evaluation of the failure probability and its gradient at each iteration of the optimization process. Indeed, here we approximate the probability of failure by its Chernoff bound, thus, being able to use stochastic gradient methods. For the solution of the resulting problem, we employ the Adam algorithm [11], which is a robust stochastic descent algorithm used in machine learning training. A numerical example is then presented to illustrate the potential advantages and drawbacks of the use of the Chernoff bound and stochastic gradient for the solution of RO problems.

2 Risk optimization (RO)

RO can be presented in the general form as the search of the design parameters $\mathbf{d}$ in the search space $\mathcal{D}$ that minimize the objective function f , i.e.,

$$\text{Find } \mathbf{d}^* = \arg \min_{\mathbf{d} \in \mathcal{D}} f(\mathbf{d}), \quad (1)$$

where

$$f(\mathbf{d}) \stackrel{\text{def}}{=} C_0(\mathbf{d}) + \sum_{i=1}^m C_i(\mathbf{d}) P_{f,i}(\mathbf{d}), \quad (2)$$

for constants $\{C_i\}_{i=1}^m$ and probabilities of failure $\{P_{f,i}\}_{i=1}^m$. Each constant C_i and probability of failure $P_{f,i}$ are related to a mode of failure given by a state limit function g_i , hence, minimizing f implies in minimizing the weighted sum of the probabilities of failure added by C_0 . To account for uncertainties, we define a random vector $\mathbf{X} \in \mathcal{X}$ such that the probability of failure is

$$P_{f,i}(\mathbf{d}) \stackrel{\text{def}}{=} P(g_i(\mathbf{d}, \mathbf{X}) > 0) \quad (3)$$

$$= \int_{g_i(\mathbf{d}, \mathbf{X}) \geq 0} \mu_{\mathbf{X}}(\mathbf{d}, \mathbf{X}) d\mathbf{X}, \quad (4)$$

where $\mu_{\mathbf{X}}$ is the probability density of $\mathbf{X}$. For a limit state function $g_i \stackrel{\text{def}}{=} S_i - R_i$, as used in [12],

$$P_{f,i}(\mathbf{d}) = P(S_i(\mathbf{d}, \mathbf{X}) > R_i(\mathbf{d}, \mathbf{X})) \quad (5)$$

$$= P\left(\frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})} > 1\right). \quad (6)$$

Evaluating the probability of failure can be expensive, possibly requiring expensive methods like MC simulation, or approximations of the limit state functions.

3 Chernoff bound

The Chernoff bound is a direct application of the Markov inequality over the exponential of a random variable [13]. Assuming Y is a random variable, Chernoff's bound can be written as

$$P(Y \geq a) \leq \frac{\mathbb{E}(e^{tY})}{e^{ta}}, \quad t > 0. \quad (7)$$

We can use (7) to bound the probability of failure. Thus, from (6), we can write

$$P_{f,i}(\mathbf{d}) \leq \hat{P}_{f,i}(\mathbf{d}) \stackrel{\text{def}}{=} \min_{t_i > 0} \frac{\mathbb{E}\left[e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}}\right]}{e^{t_i}}, \quad t_i > 0. \quad (8)$$

In order to avoid direct evaluation of the probability of failure, we employ the bound $\hat{P}_{f,i}$ during the optimization procedure. Then, instead of directly minimizing f as presented in (2), we minimize $\hat{f}$ given by

$$\hat{f}(\mathbf{d}) \stackrel{\text{def}}{=} C_0(\mathbf{d}) + \sum_{i=1}^m C_i(\mathbf{d}) \hat{P}_{f,i}(\mathbf{d}). \quad (9)$$

The main advantage of minimizing (9) instead of (2) is that the first does not require the evaluation of the probability of failure and its sensitivities, which are known to be cumbersome to evaluate from the computational point of view.

4 Stochastic gradient descent (SGD)

The SGD is a method for minimizing the expected value of functions while avoiding MC sampling. SGD is a direct application of Robbins and Monro [8] stochastic approximation in convex stochastic optimization problems.

To use SGD, one necessary condition is that an unbiased estimator of the gradient of the function to be minimized is available. To minimize $\hat{f}$ in (9), we need to define a function F such that

$$\mathbb{E}[F(\mathbf{d}, \mathbf{X})] = \hat{f}(\mathbf{d}). \quad (10)$$

Thus, we define F as

$$F(\mathbf{d}, \mathbf{X}) \stackrel{\text{def}}{=} C_0(\mathbf{d}) + \sum_{i=1}^m C_i(\mathbf{d}) \frac{e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}}}{e^{t_i}}. \quad (11)$$

Note that, from (10),

$$\nabla_{\mathbf{d}} \hat{f}(\mathbf{d}) = \nabla_{\mathbf{d}} \mathbb{E}[F(\mathbf{d}, \mathbf{X})] \quad (12)$$

$$= \mathbb{E}[\nabla_{\mathbf{d}} F(\mathbf{d}, \mathbf{X}) + F(\mathbf{d}, \mathbf{X}) \nabla_{\mathbf{d}} \log(\mu_{\mathbf{X}}(\mathbf{d}, \mathbf{X}))] \quad (13)$$

(cf. Appendix B in [10]), hence, by defining the unbiased gradient estimator

$$\nabla_{\mathbf{d}} \mathcal{F}(\mathbf{d}, \mathbf{X}) \stackrel{\text{def}}{=} \nabla_{\mathbf{d}} F(\mathbf{d}, \mathbf{X}) + F(\mathbf{d}, \mathbf{X}) \nabla_{\mathbf{d}} \log(\mu_{\mathbf{X}}(\mathbf{d}, \mathbf{X})), \quad (14)$$

it can be seen that $\nabla_{\mathbf{d}} \mathcal{F}$ is an unbiased gradient estimator of $\hat{f}$. In what follows, we use $\nabla_{\mathbf{d}} \mathcal{F}$ as a search direction for SGD. For the minimization of the expected value of a function $F : \mathcal{D} \times \mathcal{X} \mapsto \mathbb{R}$ such that $\mathbb{E}[F(\mathbf{d}, \mathbf{X})] = \hat{f}(\mathbf{d})$, the SGD update is

$$\text{Sample } \mathbf{X} \sim \mu_{\mathbf{X}} \quad (15)$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - \alpha_k \nabla_{\mathbf{d}} \mathcal{F}(\mathbf{d}, \mathbf{X}), \quad (16)$$

where α_k is a sequence of decreasing step-sizes [10].

Developing the gradient of F in (11),

$$\nabla_{\mathbf{d}} F(\mathbf{d}, \mathbf{X}) = \nabla_{\mathbf{d}} C_0(\mathbf{d}) + \nabla_{\mathbf{d}} \sum_{i=1}^m C_i(\mathbf{d}) \frac{e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}}}{e^{t_i}} \quad (17)$$

$$= \nabla_{\mathbf{d}} C_0(\mathbf{d}) + \sum_{i=1}^m \left(\nabla_{\mathbf{d}} C_i(\mathbf{d}) \frac{e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}}}{e^{t_i}} + C_i(\mathbf{d}) \frac{\nabla_{\mathbf{d}} e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}}}{e^{t_i}} \right). \quad (18)$$

Derivatives of the constants C_i depend on the problem formulation and should not be expensive to calculate. The gradient inside the summation on the right hand side of (18) is obtained from S_i , R_i , and their gradients as

$$\nabla_{\mathbf{d}} e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}} = t_i e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}} \frac{\nabla_{\mathbf{d}} S_i(\mathbf{d}, \mathbf{X}) R_i(\mathbf{d}, \mathbf{X}) - S_i(\mathbf{d}, \mathbf{X}) \nabla_{\mathbf{d}} R_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})^2}. \quad (19)$$

Hence, to use SGD to minimize the Chernoff bound one uses only S_i , R_i , and its derivatives, without requiring any MC sampling. This results in an optimization procedure with a small cost per iteration when compared to a classical approach, e.g., using any classical optimization method with MC with sample average approximation (SAA) to evaluate the probability of failure.

4.1 Adam algorithm

In the numerical section, we use an algorithm of the family of SGD named Adam [11]. Adam uses the following update rule

$$\text{Sample } \mathbf{X} \sim \mu_{\mathbf{X}} \quad (20)$$

$$\mathbf{m}^{(k+1)} = \beta_1 \mathbf{m}^{(k)} + (1 - \beta_1) \nabla_{\mathbf{d}} \mathcal{F}(\mathbf{d}^{(k)}, \mathbf{X}) \quad (21)$$

$$\mathbf{v}^{(k+1)} = \beta_2 \mathbf{v}^{(k)} + (1 - \beta_2) (\nabla_{\mathbf{d}} \mathcal{F}(\mathbf{d}^{(k)}, \mathbf{X}))^2 \quad (22)$$

$$\hat{\mathbf{m}} = \frac{\mathbf{m}^{(k+1)}}{1 - \beta_1^{k+1}} \quad (23)$$

$$\hat{\mathbf{v}} = \frac{\mathbf{v}^{(k+1)}}{1 - \beta_2^{k+1}} \quad (24)$$

$$\mathbf{d}^{(k+1)} = \mathbf{d}^{(k)} - \alpha_k \frac{\hat{\mathbf{m}}}{\sqrt{\hat{\mathbf{v}}} + \epsilon}, \quad (25)$$

where $0 < \beta_1 < \beta_2 < 1$ and ϵ are constants to be defined. Adam is more robust than SGD, being efficient in dealing with noisy gradients, the reason why we chose to use it in the numerical section.

4.2 Setting the parameter t_i of Chernoff's bound

Note that tightness of Chernoff's bound depends on the parameter $t > 0$. In order to get an optimal Chernoff's bound, the parameter that minimizes the bound

should be employed. Thus, for each limit state, the parameter t_i at iteration (k) is obtained from the minimization problem

$$\text{Find } t_i^{(k)} = \arg \min_{t_i > 0} \frac{\mathbb{E} \left[e^{t_i \frac{S_i(\mathbf{d}, \mathbf{X})}{R_i(\mathbf{d}, \mathbf{X})}} \right]}{e^{t_i}}. \quad (26)$$

To solve the optimization sub-problem from (26), we independently sample t uniformly from a fixed interval between positive values many times and verify the value that minimizes the bound. Tuning t requires sampling $\mathbf{X}$ and evaluating S and R , what can be expensive. Thus, we update t if the distance of $\mathbf{d}^{(k)}$ to $\mathbf{d}^{(\bar{k})}$, being $\bar{k}$ the iteration when t was last tuned, exceeds a decaying value,

$$\text{Find } t^{(k)} \text{ if } \left\| \mathbf{d}^{(k)} - \mathbf{d}^{(\bar{k})} \right\| \geq \frac{\epsilon_t}{\sqrt{k}}, \quad (27)$$

with ϵ_t being a parameter to be set. Note that the decay in the right hand side of (27) is the same used for the step-size in stochastic optimization.

The parameter $t_i^{(k)}$ as defined above is expected to oscillate due to sampling errors. To mitigate the noise in t , we use a running average $\bar{t}$ instead of t , defined as

$$\bar{t}_i^{(k)} \stackrel{\text{def}}{=} \frac{2}{k} \sum_{j=\frac{k}{2}}^k t_i^{(j)}. \quad (28)$$

5 Numerical example

5.1 Cantilever beam

To test the proposed methodology, we solve one numerical example, a cantilever beam subject to a point load P at its end. This benchmark problem has been presented by Torii et al. [7]. We consider a rectangular cross-section whose dimensions, width b and height h , are the design parameters, i.e., $\mathbf{d} = (b, h)$. The failure modes considered are the elastic torsional buckling and the plastic yielding of the material with, respectively, state limit functions

$$g_1 = \frac{S_1}{R_1} = \frac{M_p}{PL} \quad (29)$$

$$g_2 = \frac{S_2}{R_2} = \frac{P_{cr}}{P}. \quad (30)$$

According to [14], the plastic moment M_p is given by

$$M_p = Z f_y, \quad (31)$$

where

$$Z = \frac{bh^2}{4} \quad (32)$$

is the plastic modulus and f_y is the yielding strength of the material. The critical torsional buckling load P_{cr} is given by [15] as

$$P_{\text{cr}} = 4.013 \frac{\sqrt{EI_y C}}{L^2} \quad (33)$$

for beam length L , Young modulus E , cross-sectional inertia over the vertical axis

$$I_y = \frac{b^3 h}{12}, \quad (34)$$

torsional constant of a thin cross section

$$C = \frac{1}{3} b^2 h G, \quad (35)$$

and shear modulus G .

The parameters f_y , G , L , and P are modeled as truncated Gaussian-distributed random variables with means and standard deviations as presented in Table 1.

Table 1: Mean and standard deviation for the random parameters of the cantilever beam example.

Parameter	mean	std
f_y (MPa)	35.0	3.5
G (MPa)	7700.0	385.0
L (mm)	200.0	10.0
P (kN)	8.0	1.6

Box constraints are imposed on b and h ,

$$0.1 \leq b, h \leq 30.0. \quad (36)$$

A Poisson constant of 0.364 is used to calculate E from G . The objective function to be minimized in this example is

$$f(\mathbf{d}) = bh \left(1 + 2 \sum_{i=1}^2 P_{f,i} \right) + 200 \sum_{i=1}^2 P_{f,i}. \quad (37)$$

5.2 Setup

We approximate (37) as

$$\hat{f}(\mathbf{d}) = bh \left(1 + 2 \sum_{i=1}^2 \hat{P}_{f,i} \right) + 200 \sum_{i=1}^2 \hat{P}_{f,i}. \quad (38)$$

From S_i and R_i , as defined in (29) and (30), and their gradients, we evaluate, at each iteration, the search direction $\nabla_{\mathbf{d}}\mathcal{F}$ using (14), (18), and (19).

We run the optimization using Adam until 10^7 model gradients are evaluated. The initial step-size used in this example is $\alpha_0 = 0.2$ and the tolerance for the distance between tunings is set as $\epsilon_t = 2.0$. For the tuning of t , we use a Monte Carlo sample of size 10^3 . We set Adam’s parameters as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-12}$. We start optimization from the same location as [7], at $\mathbf{d}_0 = (1.6608, 10.4370)$.

5.3 Results

Figure 1 presents the contours of f and $\hat{f}$ for the cantilever example, with the minimum value marked with an “ $\times$ ”. It can be seen that the objective function using the Chernoff bound in Figure 1b resembles the original objective function, presented in Figure 1a. Moreover, the optimum of $\hat{f}$ is close to the one of f , with a small difference due to the Chernoff bound gap.

In Figure 2, we present the optimization path for Adam over the contour of f . It can be seen that Adam converges to a biased optimum close to the real optimum. For this example, the difference between the objective function in the true optimum and the biased one found is small, as can be observed in Table 2.

Table 2: Results for Example 1.

	b^*	h^*	$P(g_1 \geq 1)$	$P(g_2 \geq 1)$	$f(b^*, h^*)$
True	1.982	13.715	1.99×10^{-4}	1.05×10^{-4}	27.262
Found	1.964	14.145	7.24×10^{-5}	8.44×10^{-5}	27.827

The distance of $\mathbf{d}^{(k)}$ to the optimum $\mathbf{d}^*$ versus iteration is presented in 3. It can be observed that the optimization gets close to the real optimum and then proceeds to converge to the biased one, as can be seen in Figure 2.

For the 10^7 gradients used in 2×10^6 iterations, 1.7×10^6 extra gradients were needed to tune the Chernoff parameter, less than 15% of the total optimization cost. Still, if Monte Carlo sampling was to be used to evaluate the probabilities of failure, we would not be able to use gradient methods, requiring less efficient methods for optimization. A simplex method with a Monte Carlo sample of size 10^4 would require 3×10^4 evaluations of the model per iteration, what means that it would only be able to perform 390 iterations for the same computational cost. Moreover, convergence of the simplex method to local optimum is only guaranteed if SAA is used, what includes a bias in the optimum found.

6 Conclusion

This paper proposed an initial study of the use of the Chernoff bound on the probability of failure to perform risk optimization. Since we are able to derive

(a) f (b) $\hat{f}$

Fig. 1: Contour of f and $\hat{f}$ with respect to b and h for the cantilever beam in Section 5.1. The optimum in each case is marked with a black $\times$.

Fig. 2: Optimization path presented over the contour of f . The true optimum is marked with a black $\times$ while the solution found is denoted with a black circle.

Fig. 3: Distance to the optimum per iteration.

gradients of the Chernoff bound with respect to the design variables, stochastic gradient methods can be used to perform optimization. How distant is the approximated optimum from the true one depends on how tight is the Chernoff bound; in other words, how well is the tuning parameter set. To tighten the Chernoff bound, we proposed a heuristic procedure to control the value of the tuning parameter during optimization.

Testing the methodology on a numerical example, a cantilever beam, we observed that the stochastic gradient method converged to a solution close to the true one. Moreover, it converged with as little as 5 model evaluations per iteration, which is much less than using Sample Average Approximation together with classical deterministic optimization methods. The drawback of our approach lies in setting the Chernoff parameter; however, the heuristic that we proposed needed less than 15% of the optimization cost. In the example studied, the advantages of using Adam with the gradient of the Chernoff bound far surpassed the extra cost of tuning the parameter.

For future research, we suggest testing different methods for tuning the constant t ; a discussion on the calibration of t will be presented in a future work to be published by the authors along with complexity analyses. Other suggestions are the use of different optimization algorithms like the Nesterov accelerated gradient descent and the use of importance sampling to reduce the variance in gradient estimates and thus improve convergence, e.g., with information from the first-order reliability method. Finally, the performance of the proposed method in high-dimensional problems must be studied.

References

1. R. H. Lopez, A. T. Beck, Rbdo methods based on form: a review, *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 34 (4) (2012) 506–514. doi:10.1590/S1678-58782012000400012.
2. A. T. Beck, W. J. Gomes, A comparison of deterministic, reliability-based and risk-based structural optimization under uncertainty, *Probabilist Eng Mech* 28 (2012) 18 – 29.
3. W. J. S. Gomes, A. T. Beck, Global structural optimization considering expected consequences of failure and using ann surrogates, *Computers and Structures* 126 (2013) 56–68. doi:10.1016/j.compstruc.2012.10.013.
4. A. T. Beck, W. J. S. Gomes, R. H. Lopez, L. F. F. Miguel, A comparison between robust and risk-based optimization under uncertainty, *Structural and Multidisciplinary Optimization* 52 (3) (2015) 479–492. doi:10.1007/s00158-015-1253-9.
5. W. J. S. Gomes, A. T. Beck, The design space root finding method for efficient risk optimization by simulation, *Probabilistic Engineering Mechanics* 44 (2016) 99–110. doi:10.1016/j.probenmech.2015.09.019.
6. A. Torii, R. Lopez, L. Miguel, A gradient-based polynomial chaos approach for risk and reliability-based design optimization, *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 39 (7) (2017) 2905–2915.
7. A. J. Torii, R. H. Lopez, A. T. Beck, L. F. F. Miguel, A performance measure approach for risk optimization, *Structural and Multidisciplinary Optimization* 60 (3) (2019) 927–947.

8. H. Robbins, S. Monro, A stochastic approximation method, *The annals of mathematical statistics* (1951) 400–407.
9. A. G. Carlon, B. M. Dia, L. F. Espath, R. H. Lopez, R. Tempone, Nesterov-aided stochastic gradient methods using Laplace approximation for Bayesian design optimization, *arXiv preprint arXiv:1807.00653* (2018).
10. A. G. Carlon, R. H. Lopez, L. F. d. R. Espath, L. F. F. Miguel, A. T. Beck, A stochastic gradient approach for the reliability maximization of passively controlled structures, *Engineering Structures* 186 (2019) 1–12.
11. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
12. R. E. Melchers, A. T. Beck, *Structural reliability analysis and prediction*, John Wiley & Sons, 2018.
13. E. Lehman, T. Leighton, A. R. Meyer, *Mathematics for computer science*, Tech. rep., Technical report, 2006. Lecture notes (2010).
14. J. M. Gere, *Mechanics of Materials* 6th, 2004.
15. S. P. Timoshenko, J. M. Gere, *Theory of elastic stability*, Courier Corporation, 2009.