

Cross entropy-based importance sampling using Gaussian densities revisited

Sebastian Geyer^{a,*}, Iason Papaioannou^a, Daniel Straub^a

^a*Engineering Risk Analysis Group, Technische Universität München, Arcisstraße 21, 80333 München*

Abstract

The computation of the probability of a rare (failure) event is a common task in structural reliability analysis. In most applications, the numerical model defining the rare event is nonlinear and the resulting failure domain often multimodal. One strategy for estimating the probability of failure in this context is the importance sampling method. The efficiency of importance sampling depends on the choice of the importance sampling density. A near-optimal sampling density can be found through application of the cross entropy method. The cross entropy method is an adaptive sampling approach that determines the sampling density through minimizing the Kullback-Leibler divergence between the theoretically optimal importance sampling density and a chosen parametric family of distributions. In this paper, we investigate the suitability of the multivariate normal distribution and the Gaussian mixture model as importance sampling densities within the cross entropy method. Moreover, we compare the performance of the cross entropy method to sequential importance sampling, another recently proposed adaptive sampling approach, which uses the Gaussian mixture distribution as a proposal distribution within a Markov Chain Monte Carlo algorithm. For the parameter updating of the Gaussian mixture within the cross entropy method, we propose a modified version of the expectation-maximization algorithm that works with weighted samples. To estimate the number of distributions in the mixture, the density-based spatial clustering of applications with noise (DBSCAN) algorithm is adapted to the use of weighted samples. We compare the performance of the different methods in several examples, including component reliability problems, system reliability problems and reliability in varying dimensions. The results show that the cross entropy method using a single Gaussian outperforms the cross entropy method using Gaussian mixture and that both distribution types are not suitable for high dimensional reliability problems.

This is an electronic reprint of the original article published by Elsevier in:
Structural Safety, 2019, Vol.76, pp. 15–27
URL: <http://www.sciencedirect.com/science/article/pii/S0167473017303429>
DOI: <https://doi.org/10.1016/j.strusafe.2018.07.001> (please cite accordingly).
This reprint differs from the original in layout, pagination and typographic details.

Keywords: reliability analysis, simulation, importance sampling, cross entropy method, Gaussian mixture, expectation maximization, sequential importance sampling

1. Introduction

In structural reliability, the goal is to assess the effects of uncertain input variables on the performance of an engineering system. The main objective is to evaluate the probability of unsatisfactory performance of the system, the probability of failure P_F . Let \mathbf{X} be a random vector with outcome space $\mathbf{x} \in \mathbb{R}^n$ that collects all uncertain input variables. The performance of a system can be assessed by means of the limit

*Corresponding author

Email address: s.geyer@tum.de (Sebastian Geyer)

state function, denoted by $g(\mathbf{x})$. If the limit state function gives a value smaller or equal to zero the system fails, while otherwise the system has satisfactory performance. Hence, the event defining failure of the system is given by

$$F = \{\mathbf{x} \in \mathbb{R}^n : g(\mathbf{x}) \leq 0\} \quad (1)$$

The probability of failure is defined through the following integral:

$$P_F = \int_{\mathbb{R}^n} I(\mathbf{x}) \cdot f(\mathbf{x}) \, d\mathbf{x} \quad (2)$$

In this definition, $f(\mathbf{x})$ is the joint probability density function (PDF) of \mathbf{X} and $I(\mathbf{x})$ is an indicator function defined as

$$I(\mathbf{x}) = \begin{cases} 1 & \text{if } g(\mathbf{x}) \leq 0 \\ 0 & \text{else} \end{cases} \quad (3)$$

The function $g(\mathbf{x})$ depends on the outcome of an engineering model, and hence the probability of failure cannot be evaluated analytically. Therefore, the failure probability needs to be approximated by means of an efficient method that minimizes the number of model evaluations [1]. Several methods have been developed, including approximation methods such as the first/second order reliability method (FORM/SORM) [2, 3] and sampling-based methods such as the Monte Carlo simulation (MCS) and its adaptive variants (e.g. [4–10]). The main advantage of sampling-based strategies is their robustness with respect to the complexity of the limit state function. Furthermore, they are asymptotically exact and, in contrast to approximation methods, information about the quality of the obtained result is more easily available.

Sampling methods are based on MCS; however, MCS is especially inefficient in estimating small failure probabilities, which is typically the goal in reliability assessment of engineering systems. The variance of the MCS estimate is inversely proportional to the target failure probability, which leads to prohibitively large sample sizes for obtaining sufficiently accurate estimates. The aim of advanced sampling-based methods is to enhance the efficiency of MCS through reducing the required sample size while keeping the variance of the estimate low.

A standard variant of MCS is the importance sampling (IS) method. IS aims at decreasing the variance of the MCS probability estimate by sampling from an alternative sampling density, the so-called IS density $h(\mathbf{x})$. Eq. (2) can be modified as follows:

$$P_F = \int_{\mathbb{R}^n} \frac{I(\mathbf{x}) \cdot f(\mathbf{x})}{h(\mathbf{x})} \cdot h(\mathbf{x}) \, d\mathbf{x} \quad (4)$$

Provided that the support of $h(\mathbf{x})$ contains the failure domain, this modification does not alter the value of the integral. The IS estimate of P_F is given as follows:

$$\hat{P}_F = \frac{1}{n_s} \sum_{i=1}^{n_s} I(\mathbf{x}_i) \cdot \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)}, \quad (5)$$

wherein the samples $\{\mathbf{x}_i, i = 1, \dots, n_s\}$ are distributed according to $h(\mathbf{x})$. The efficiency of IS depends on the choice of the IS density. A near-optimal IS density can be found through application of the cross entropy (CE) method [11]. The CE method is an adaptive sampling approach that determines the sampling density through minimizing the Kullback-Leibler (KL) divergence between the theoretically optimal IS density and a chosen parametric family of probability distributions. Typically, the multivariate normal distribution is chosen as parametric distribution family, while recently the Gaussian mixture (GM) has been proposed for handling multimodal failure domains [7, 12]. In this study we investigate the performance of these two distribution types within the CE method and propose a new updating scheme for estimating the parameters of the GM model through a modified expectation-maximization (EM) algorithm. Moreover, another recently proposed approach, called sequential importance sampling (SIS) [6], which also makes use of the Gaussian mixture, is discussed and its performance is compared to the one of the CE method.

The structure of the paper is as follows. First, the CE method is described and a general algorithm for its implementation is presented. Afterwards, the multivariate Gaussian and Gaussian mixture distribution types are introduced and their integration in the CE method is described; for integrating the Gaussian mixture model, a novel EM algorithm is proposed. Subsequently, a brief overview of SIS is given. The performance of the different methods is demonstrated by means of numerical examples. Finally, the results are summarized and the conclusions are presented.

2. Cross entropy method

For an efficient and reliable estimation of the failure probability with IS, the sampling density needs to be chosen carefully. In fact, there exists an optimal IS density, whose PDF is [13]:

$$p^*(\mathbf{x}) = \frac{I(\mathbf{x}) \cdot f(\mathbf{x})}{\int_{\mathbb{R}^n} I(\mathbf{x}) \cdot f(\mathbf{x}) d\mathbf{x}} \quad (6)$$

Eq. (6) is the density of the random variables censored at the failure domain; its normalizing constant is the target probability of failure. The IS density of Eq. (6) leads to a variance of the probability estimate of zero. That is, a single sample of this density would lead to the exact value of the probability of failure. However, this density is not applicable in practice, as it requires a priori knowledge of the failure domain and the target failure probability.

Even though Eq. (6) cannot be used directly, one can use samples at the failure domain to identify a near-optimal IS density through fitting a distribution model. The CE method identifies the parameters of a chosen distribution model through minimizing the KL divergence between the sought sampling density and the optimal IS density of Eq. (6). The KL divergence is a measure of the difference between two PDFs, and is defined as [14]

$$D(p_1(\mathbf{x}), p_2(\mathbf{x})) = E_{p_1} \left[\ln \left(\frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \right) \right] = \int_{\mathbb{R}^n} p_1(\mathbf{x}) \cdot \ln \left(\frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \right) d\mathbf{x} , \quad (7)$$

where $E_{p_1} [\cdot]$ denotes the expectation with respect to $p_1(\mathbf{x})$. Eq. (7) is also known as the relative cross entropy [15]. In the following the terms cross entropy and KL divergence are used interchangeably. The expression in Eq. (7) can be divided into two parts:

$$D(p_1(\mathbf{x}), p_2(\mathbf{x})) = \int_{\mathbb{R}^n} p_1(\mathbf{x}) \cdot \ln(p_1(\mathbf{x})) d\mathbf{x} - \int_{\mathbb{R}^n} p_1(\mathbf{x}) \cdot \ln(p_2(\mathbf{x})) d\mathbf{x} \quad (8)$$

Replacing $p_1(\mathbf{x})$ with the optimal (but unknown) IS density $p^*(\mathbf{x})$ and $p_2(\mathbf{x})$ with a parametric IS density $h(\mathbf{x}; \mathbf{v})$, where \mathbf{v} is the parameter vector, Eq. (8) can be rewritten to describe the cross entropy between these two PDFs:

$$D(p^*(\mathbf{x}), h(\mathbf{x}; \mathbf{v})) = \int_{\mathbb{R}^n} p^*(\mathbf{x}) \cdot \ln(p^*(\mathbf{x})) d\mathbf{x} - \int_{\mathbb{R}^n} p^*(\mathbf{x}) \cdot \ln(h(\mathbf{x}; \mathbf{v})) d\mathbf{x} \quad (9)$$

As the parametric IS density appears in the second term only and $p^*(\mathbf{x})$ is invariant, a minimum of the cross entropy can be found by minimizing only the second part. Substituting Eq. (6) for $p^*(\mathbf{x})$ in Eq. (9) results in the following optimization problem [7]:

$$\arg \min_{\mathbf{v}} D(p^*(\mathbf{x}), h(\mathbf{x}; \mathbf{v})) = \arg \max_{\mathbf{v}} \int_{\mathbb{R}^n} I(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \ln(h(\mathbf{x}; \mathbf{v})) d\mathbf{x} \quad (10)$$

The IS density $h(\mathbf{x}; \mathbf{v})$ found by the minimization of $D(p^*(\mathbf{x}), h(\mathbf{x}; \mathbf{v}))$ is termed *near-optimal IS density*. For the efficient evaluation of Eq. (10), an alternative sampling density $h(\mathbf{x}; \mathbf{w})$ is defined, which is of the

same type as the near-optimal sampling density $h(\mathbf{x}; \mathbf{v})$. The optimization program is modified accordingly:

$$\begin{aligned}
& \arg \min_{\mathbf{v}} D(p^*(\mathbf{x}), h(\mathbf{x}; \mathbf{v})) \\
&= \arg \max_{\mathbf{v}} \int_{\mathbb{R}^n} I(\mathbf{x}) \cdot \ln(h(\mathbf{x}; \mathbf{v})) \cdot W(\mathbf{x}; \mathbf{w}) \cdot h(\mathbf{x}; \mathbf{w}) \, d\mathbf{x} \\
&= \arg \max_{\mathbf{v}} E_{\mathbf{w}} [I(\mathbf{x}) \cdot \ln(h(\mathbf{x}; \mathbf{v})) \cdot W(\mathbf{x}; \mathbf{w})]
\end{aligned} \tag{11}$$

In this expression, $E_{\mathbf{w}} [\cdot]$ denotes the mathematical expectation with respect to the alternative sampling density $h(\mathbf{x}; \mathbf{w})$, while $W(\mathbf{x}; \mathbf{w})$ is the likelihood ratio of the original sampling PDF to the alternative sampling PDF:

$$W(\mathbf{x}; \mathbf{w}) = \frac{f(\mathbf{x})}{h(\mathbf{x}; \mathbf{w})} \tag{12}$$

The expectation in Eq. (11) is approximated via IS. With \mathbf{x}_i referring to the i -th sample drawn from the density $h(\mathbf{x}; \mathbf{w})$, $i = 1, \dots, n_s$, the solution to Eq. (11) is approximated by

$$\begin{aligned}
& \arg \min_{\mathbf{v}} D(p^*(\mathbf{x}), h(\mathbf{x}; \mathbf{v})) \\
&\approx \arg \max_{\mathbf{v}} \frac{1}{n_s} \cdot \sum_{i=1}^{n_s} I(\mathbf{x}_i) \cdot \ln(h(\mathbf{x}_i; \mathbf{v})) \cdot W(\mathbf{x}_i; \mathbf{w})
\end{aligned} \tag{13}$$

If Eq. (13) has a global maximum, which is typically the case in structural reliability [4], this maximum can be found by taking the gradient with respect to \mathbf{v} and setting the result to zero.

According to Eq. (13), obtaining a reliable estimate of the KL divergence requires that a substantial number of samples from $h(\mathbf{x}; \mathbf{w})$ fall in the failure domain. The CE method solves this problem through introducing a series of intermediate failure domains that gradually approach the target failure domain. In this way, the CE optimization problem is solved for the optimal IS density of each intermediate failure domain using samples from the fitted parametric density obtained at the previous step. The l -th intermediate failure domain and corresponding indicator function are defined as follows:

$$F^{(l)} = \{\mathbf{x} \in \mathbb{R}^n : g(\mathbf{x}) \leq \xi^{(l)}\}, \quad \xi^{(l)} \geq 0 \tag{14}$$

$$I_{\xi^{(l)}}(\mathbf{x}) = \begin{cases} 1 & \text{if } g(\mathbf{x}) \leq \xi^{(l)} \\ 0 & \text{else} \end{cases} \tag{15}$$

The intermediate failure domain is defined such that $\rho \cdot n_s$ samples are located in this domain, where ρ is a chosen value in the interval $]0, 1[$. The corresponding $\xi^{(l)}$ is calculated as the ρ -quantile of the limit state function values of the samples. Thus, the optimization problem of Eq. (13) can be rewritten for the l -th step as

$$\begin{aligned}
& \arg \min_{\mathbf{v}} D(p^{*(l)}(\mathbf{x}), h(\mathbf{x}; \mathbf{v})) \\
&\approx \arg \max_{\mathbf{v}} \frac{1}{n_s} \cdot \sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot \ln(h(\mathbf{x}_i; \mathbf{v})) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}),
\end{aligned} \tag{16}$$

where $p^{*(l)}(\mathbf{x})$ is the theoretically optimal IS density corresponding to the l -th intermediate failure domain. The solution for Eq. (16) then provides a density which approximates this intermediate failure domain. In the next step, n_s samples are drawn from this density and the procedure is repeated until at least $\rho \cdot n_s$ samples fall in the actual failure domain. Starting from an initial sampling density, it is possible to obtain

a near-optimal IS density within a number of steps of sampling and parameter updating. The general CE algorithm for structural reliability applications can be summarized as follows [11]:

1. Initializing:
 - Set $l = 1$.
 - Choose a distribution family and initial parameters $\mathbf{w}^{(l)}$ for the sampling density $h(\mathbf{x}; \mathbf{w}^{(l)})$.
 - Select sample size n_s and sample quantile ρ .
2. Sampling:
 - Generate n_s random samples of $h(\mathbf{x}; \mathbf{w}^{(l)})$.
 - Evaluate the limit state function $g(\mathbf{x})$ for the samples.
 - Determine the ρ -quantile of the limit state function values denoted by $\xi^{(l)}$.
 - If $\xi^{(l)} \leq 0$, set $\xi^{(l)} = 0$ and proceed with step 4.
3. Parameter updating:
 - Find the solution $\hat{\mathbf{v}}^{(l)}$ to the optimization program of Eq. (16).
 - Set $\mathbf{w}^{(l+1)} = \hat{\mathbf{v}}^{(l)}$.
 - Set $l = l + 1$ and return to step 2.
4. Estimating the probability of failure:

$$\hat{P}_F = \frac{1}{n_s} \cdot \sum_{i=1}^{n_s} I(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}) \quad (17)$$

3. CE method with single Gaussian densities

The classical choice for the distribution family within the CE method is the multivariate normal distribution, referred to as single Gaussian (SG) density. It is fully described by its first two moments, the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$. Hence, the unknown parameter vector in the optimization problem of the CE method is $\mathbf{v} = [\boldsymbol{\mu}; \boldsymbol{\Sigma}]$. As the SG belongs to the exponential family, closed-form updating rules can be derived from Eq. (16) [14].

3.1. Updating rules for SG

Inserting the SG density into Eq. (16), taking the gradient of the objective function with respect to \mathbf{v} and setting this to zero, one gets:

$$\frac{1}{n_s} \cdot \sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot \nabla_{\mathbf{v}} \ln(\mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}) = \mathbf{0} \quad (18)$$

Here, $\mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ describes the SG PDF with parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. From Eq. (18), one obtains the following parameter updates:

$$\boldsymbol{\mu}^{(l+1)} = \frac{\sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}) \cdot \mathbf{x}_i}{\sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)})} \quad (19)$$

$$\boldsymbol{\Sigma}^{(l+1)} = \frac{\sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}) \cdot (\mathbf{x}_i - \boldsymbol{\mu}^{(l+1)}) \cdot (\mathbf{x}_i - \boldsymbol{\mu}^{(l+1)})^{\top}}{\sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)})} \quad (20)$$

3.2. Initialization

The implemented algorithm follows the general CE algorithm scheme presented in Section 2. A point of special interest when using an adaptive IS algorithm is the choice of the initial sampling parameters. One possibility would be to perform a FORM analysis to identify the most likely failure point and center the sampling density at this point, e.g. [16]. However, this approach concentrates on the most important failure domain while other domains of interest may be neglected. Moreover, it requires the solution of an optimization problem, which could potentially increase the computational cost significantly. For this reason, we choose the initial sampling density as the original density of the random variables, i.e. $h(\mathbf{x}; \mathbf{w}^{(1)}) = f(\mathbf{x})$. In this case, the first step of the CE algorithm becomes equivalent to performing a MCS on a system with increased failure probability.

4. CE method with Gaussian mixtures

This section introduces the Gaussian mixture (GM) as parametric model within the CE method, which was recently proposed for handling multimodal failure domains [7]. The PDF of the GM model is defined as the sum of a number of Gaussian PDFs, each of them multiplied by a weighting factor:

$$h_{\text{GM}}(\mathbf{x}; \mathbf{v}) = \sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (21)$$

The GM distribution has valid cumulative distribution function and PDF, provided that $\sum_{k=1}^K \pi_k = 1$. The unknown parameters of the CE optimization problem in this case are $\mathbf{v} = [\pi_k; \boldsymbol{\mu}_k; \boldsymbol{\Sigma}_k, k = 1, \dots, K]$. This optimization problem is solved by substituting the GM PDF into Eq. (16), taking the gradient of the objective function with respect to \mathbf{v} and setting it to zero:

$$\frac{1}{n_s} \cdot \sum_{i=1}^{n_s} I_{\xi^{(l)}}(\mathbf{x}_i) \cdot W(\mathbf{x}_i; \mathbf{w}^{(l)}) \cdot \nabla_{\mathbf{v}} \ln \left(\sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) = \mathbf{0} \quad (22)$$

The gradients of the logarithm of the GM PDF in Eq. (22) with respect to the different components of \mathbf{v} are:

$$\frac{\partial \ln(h_{\text{GM}}(\mathbf{x}_i; \mathbf{v}))}{\partial \pi_k} = \gamma_{i,k} \cdot \frac{1}{\pi_k}, \quad k = 1, \dots, K \quad (23)$$

$$\frac{\partial \ln(h_{\text{GM}}(\mathbf{x}_i; \mathbf{v}))}{\partial \boldsymbol{\mu}_k} = \gamma_{i,k} \cdot \boldsymbol{\Sigma}_k^{-1} \cdot (\mathbf{x} - \boldsymbol{\mu}_k), \quad k = 1, \dots, K \quad (24)$$

$$\frac{\partial \ln(h_{\text{GM}}(\mathbf{x}_i; \mathbf{v}))}{\partial \boldsymbol{\Sigma}_k} = \gamma_{i,k} \cdot \frac{1}{2} (\boldsymbol{\Sigma}_k^{-1} \cdot (\mathbf{x} - \boldsymbol{\mu}_k) \cdot (\mathbf{x} - \boldsymbol{\mu}_k)^\top \cdot \boldsymbol{\Sigma}_k^{-1} - \boldsymbol{\Sigma}_k^{-1}), \quad k = 1, \dots, K \quad (25)$$

Herein, $\gamma_{i,k}$ is the probability of sample i to belong to distribution k and is defined as follows:

$$\gamma_{i,k} = \frac{\pi_k \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k=1}^K \pi_k \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \quad (26)$$

It holds that $\sum_{k=1}^K \gamma_{i,k} = 1$. The term $\gamma_{i,k}$ appears in each of the three partial derivatives. As the parameters for which Eq. (22) is to be solved are part of this term, closed-form updating rules cannot be obtained. The term equals 1 for $K = 1$, which verifies the results from the previous section, where closed-form updating rules were derived for a single Gaussian distribution. An approximation of $\gamma_{i,k}$ in step l can be found based

on the sampling density of the previous step, i.e. [7]

$$\gamma_{i,k}^{(l)} \approx \frac{\pi_k^{(l-1)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(l-1)}, \boldsymbol{\Sigma}_k^{(l-1)})}{\sum_{k=1}^K \pi_k^{(l-1)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(l-1)}, \boldsymbol{\Sigma}_k^{(l-1)})} \quad (27)$$

If K does not change throughout the sampling steps of the CE algorithm, this approximation may be sufficient. However, in order to keep the algorithm flexible and insensitive to different initial parameters, an efficient procedure to obtain the maximum likelihood estimates (MLE) of the parameters is derived in the following for the parameter update of the GM distribution.

4.1. Maximum likelihood estimation of distribution parameters

The available samples are denoted as $\hat{\mathbf{x}}$ within this section, with their corresponding vector of importance weights denoted by \mathbf{W} . The parameter vector to be fitted is denoted by \mathbf{v} . In general, the MLE estimate of \mathbf{v} is defined as follows:

$$\hat{\mathbf{v}}_{\text{MLE}} = \arg \max_{\mathbf{v}} h_{\mathbf{W}}(\hat{\mathbf{x}}|\mathbf{v}) \quad (28)$$

As the samples \mathbf{x}_i , $i = 1, \dots, n_s$ are independent, the joint likelihood $h_{\mathbf{W}}(\hat{\mathbf{x}}|\mathbf{v})$ can be defined as the product of the likelihoods of the samples raised to the power of the corresponding weight $W(\mathbf{x}_i)$, which implies $W(\mathbf{x}_i)$ occurrences of \mathbf{x}_i , $i = 1, \dots, n_s$:

$$h_{\mathbf{W}}(\hat{\mathbf{x}}|\mathbf{v}) = \prod_{i=1}^{n_s} h(\mathbf{x}_i|\mathbf{v})^{W(\mathbf{x}_i)} \quad (29)$$

Since it is often easier to maximize the log-likelihood rather than the likelihood, the MLE estimate can be obtained by solving the following optimization program:

$$\hat{\mathbf{v}}_{\text{MLE}} = \arg \max_{\mathbf{v}} \ln(h_{\mathbf{W}}(\hat{\mathbf{x}}|\mathbf{v})) \quad (30)$$

This does not change the solution due to the monotonicity of the logarithm. The logarithm of the joint likelihood is given by

$$\ln(h_{\mathbf{W}}(\hat{\mathbf{x}}|\mathbf{v})) = \sum_{i=1}^{n_s} W(\mathbf{x}_i) \cdot \ln(h(\mathbf{x}_i|\mathbf{v})) \quad (31)$$

Comparing Eqs. (30) and (31) with Eq. (13) shows that obtaining the MLE estimate of the parameters of a parametric density with pre-selected samples according to the indicator function used in the CE method is equivalent to solving the CE optimization problem. This equivalence has also been discussed in [4].

An efficient way to compute maximum likelihood estimates of parameters of mixture distributions is the expectation-maximization (EM) algorithm [17]. Originally, the EM algorithm was developed for unweighted samples. By including the weights of the samples in the maximum likelihood estimation, it is possible to adapt the algorithm such that it can be used to fit an IS density within the CE method. In the following, a general EM updating scheme for weighted samples is derived and adapted to fitting the GM distribution.

To obtain the MLE estimate for a mixture distribution directly, not only $\hat{\mathbf{x}}$ and the corresponding weights \mathbf{W} are required, but in addition the assignment of the samples to the distributions in the mixture, e.g. as defined in Eq. (26) [18]. If only $\hat{\mathbf{x}}$ and \mathbf{W} are available, the EM method treats this assignment as missing data and uses an iterative procedure to approximate the MLE estimate. In each step of the iteration, a guess for the missing data is used to maximize the expected log-likelihood of the full data set \mathbf{Z} , consisting of the available and the missing data.

For a given value of the parameter vector $\hat{\mathbf{v}}$, the expected log-likelihood (or Q -function) is defined as [18]:

$$Q(\mathbf{v}|\hat{\mathbf{v}}) = E_{\mathbf{Z}|\hat{\mathbf{x}},\hat{\mathbf{v}}} [\ln(h_{\mathbf{W}}(\mathbf{z}|\mathbf{v}))] = \int_{\mathbf{Z}(\hat{\mathbf{x}})} \ln(h_{\mathbf{W}}(\mathbf{z}|\mathbf{v})) \cdot h_{\mathbf{W}}(\mathbf{z}|\hat{\mathbf{x}},\hat{\mathbf{v}}) \, d\mathbf{z}, \quad (32)$$

where $\mathcal{Z}(\hat{\mathbf{x}})$ denotes the support of \mathcal{Z} conditioned on $\hat{\mathbf{x}}$. Based on an initial guess for the missing data, the EM algorithm proceeds with two steps. The maximization step (M-step) provides a new parameter vector by maximizing the expected log-likelihood of \mathcal{Z} . Afterwards, in the expectation step (E-step), the expected log-likelihood is calculated following Eq. (32), based on the updated parameter vector. This procedure is repeated until convergence is achieved.

In the next section we discuss application of the EM algorithm for obtaining the parameters of the GM model in the CE method with $n_\rho = \rho \cdot n_s$ available samples.

4.2. EM algorithm for the GM model

For each of the K distributions in the GM three quantities have to be estimated; namely the mean vector $\boldsymbol{\mu}_k$, covariance matrix $\boldsymbol{\Sigma}_k$ and weighting factor π_k . As mentioned above, the missing data describes the assignment of the samples to the distributions in the mixture. In step m of the EM algorithm, this assignment can be approximated by a probabilistic assignment of the samples to the distributions using the parameter estimates $[\pi_k^{(m)}; \boldsymbol{\mu}_k^{(m)}; \boldsymbol{\Sigma}_k^{(m)}, k = 1, \dots, K]$. In this way, an $n_\rho \times K$ matrix is obtained with entries calculated as follows:

$$\gamma_{i,k}^{(m)} = \frac{\pi_k^{(m)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(m)}, \boldsymbol{\Sigma}_k^{(m)})}{\sum_{k=1}^K \pi_k^{(m)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(m)}, \boldsymbol{\Sigma}_k^{(m)})}, \quad \begin{array}{l} i = 1, \dots, n_\rho \\ k = 1, \dots, K \end{array} \quad (33)$$

Inserting Eq. (33) into Eq. (32) the following expression for the Q -function in step m can be derived [18]:

$$Q(\mathbf{v}|\hat{\mathbf{v}}^{(m)}) = \sum_{i=1}^{n_\rho} \sum_{k=1}^K W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot \ln(\pi_k \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)), \quad (34)$$

which completes the E-step of the algorithm. The logarithm in Eq. (34) reads as follows:

$$\begin{aligned} & \ln(\pi_k \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)) \\ &= \ln(\pi_k) - \frac{1}{2} \ln(|\boldsymbol{\Sigma}_k|) - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^\top \cdot \boldsymbol{\Sigma}_k^{-1} \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k) + C \end{aligned} \quad (35)$$

The constant C can be dropped, as the equation is part of an optimization program. Defining

$$n_k^{(m)} = \sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \quad (36)$$

allows to rewrite Eq. (34) as

$$\begin{aligned} Q(\mathbf{v}|\hat{\mathbf{v}}^{(m)}) &= \sum_{k=1}^K n_k^{(m)} \cdot \left(\ln(\pi_k) - \frac{1}{2} \ln(|\boldsymbol{\Sigma}_k|) \right) + \\ & - \frac{1}{2} \sum_{i=1}^{n_\rho} \sum_{k=1}^K W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k)^\top \cdot \boldsymbol{\Sigma}_k^{-1} \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k) \end{aligned} \quad (37)$$

The optimization program, which is to be solved within the M-step of the algorithm, can be expressed as follows:

$$\begin{aligned} & \text{maximize}_{\mathbf{v}} Q\left(\mathbf{v}|\hat{\mathbf{v}}^{(m)}\right) \\ & \text{subject to } \sum_{k=1}^K \pi_k = 1, \quad \pi_k \geq 0, \quad k = 1, \dots, K, \\ & \quad \quad \quad \Sigma_k \succ 0, \quad k = 1, \dots, K, \end{aligned} \quad (38)$$

where the latter condition ensures that the covariance matrix is positive definite. By means of the method of Lagrange multipliers the following updating rules for each of the K distributions in the mixture can be found [18]:

$$\pi_k^{(m+1)} = \frac{n_k^{(m)}}{\sum_{k=1}^K n_k^{(m)}} \quad (39)$$

$$\boldsymbol{\mu}_k^{(m+1)} = \frac{\sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot \mathbf{x}_i}{n_k^{(m)}} \quad (40)$$

$$\Sigma_k^{(m+1)} = \frac{\sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k^{(m+1)}) \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k^{(m+1)})^\top}{n_k^{(m)}}, \quad k = 1, \dots, K \quad (41)$$

With these updating rules, the M-step of the algorithm can be performed, providing a new parameter guess for the subsequent E-step.

The EM algorithm to fit a GM model based on weighted samples can be summarized as follows:

1. Initialization

- Set $m = 1$.
- Choose number of distributions in the mixture K and initial values for $\gamma_{i,k}^{(m)}$, $i = 1, \dots, n_\rho$, $k = 1, \dots, K$.
- Set $L^{(m)} = -\infty$ for convergence check.

2. M-step

- Calculate

$$n_k^{(m)} = \sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)}, \quad k = 1, \dots, K$$

- Update distribution weights:

$$\pi_k^{(m+1)} = \frac{n_k^{(m)}}{\sum_{k=1}^K n_k^{(m)}}, \quad k = 1, \dots, K$$

- Update mean vectors:

$$\boldsymbol{\mu}_k^{(m+1)} = \frac{\sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot \mathbf{x}_i}{n_k^{(m)}}, \quad k = 1, \dots, K$$

- Update covariance matrices:

$$\boldsymbol{\Sigma}_k^{(m+1)} = \frac{\sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \gamma_{i,k}^{(m)} \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k^{(m+1)}) \cdot (\mathbf{x}_i - \boldsymbol{\mu}_k^{(m+1)})^\top}{n_k^{(m)}}, \quad k = 1, \dots, K$$

3. E-step

- Compute new estimate for the probabilistic assignment:

$$\gamma_{i,k}^{(m+1)} = \frac{\pi_k^{(m+1)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(m+1)}, \boldsymbol{\Sigma}_k^{(m+1)})}{\sum_{k=1}^K \pi_k^{(m+1)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(m+1)}, \boldsymbol{\Sigma}_k^{(m+1)})}, \quad \begin{matrix} i = 1, \dots, n_\rho \\ k = 1, \dots, K \end{matrix}$$

- Calculate the log-likelihood:

$$L^{(m+1)} = \frac{1}{\sum_{i=1}^{n_\rho} W(\mathbf{x}_i)} \cdot \sum_{i=1}^{n_\rho} W(\mathbf{x}_i) \cdot \ln \left(\sum_{k=1}^K \pi_k^{(m+1)} \cdot \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k^{(m+1)}, \boldsymbol{\Sigma}_k^{(m+1)}) \right)$$

4. Check convergence

- If $|L^{(m+1)} - L^{(m)}| > \epsilon \cdot |L^{(m+1)}|$, set $m = m + 1$ and return to step 2.
- Otherwise convergence is reached. Exit the algorithm.

It is mentioned that the one-step parameter update of the GM model, as proposed in [7] and further applied in [12], is equivalent to a single step of the EM algorithm with an initial guess for the missing data, estimated based on the sampling density of the previous step in the CE method. Hence, the parameter updating scheme of [7] may result in suboptimal parameter estimates.

The EM procedure does not require additional evaluations of the model and hence its contribution to the computational effort is negligible.

4.3. Choice of the number of Gaussians in the GM model

The number of Gaussians in the mixture directly influences the performance of the algorithm. Choosing this number too small will result in neglecting failure domains with less importance, while choosing this number too large gives inaccurate estimates of the parameters for small sample sizes. Subsequent steps of the CE method with poor choices of K amplify these effects, as the samples are drawn from the GM distribution approximated in the previous step.

Several methods exist to obtain a guess for the number of distributions in a mixture model. One such method is the k-means algorithm [19] and the further developed k-means++ algorithm [20]. They can cluster available data and identify the centroids. The quality of the clustering can be measured, e.g. by the so-called silhouette value. However, neither k-means nor k-means++ is able to account for the sample weights. The method proposed in [21], based on the so-called minimum message length (MML) criterion, accounts for the weights of the samples; however, evaluation of the MML criterion is not straightforward and has to be adapted to the particular mixture model.

In this study, we identify the number of distributions in the GM model with the density-based spatial clustering of applications with noise (DBSCAN) algorithm [22]. In a nutshell, this algorithm detects the neighbors of every data point set and groups the data into three sets: 1) core points, which have a minimum number of points in their direct neighborhood; 2) border points, which are in the neighborhood of a core point but do not have enough points in their neighborhood to be identified as a core point; and 3) noise, which means that the data point is not near to other points and thus is neglected. A set of data points is identified as a cluster if it consists of a minimum number of points that do not exceed a maximum distance to the next neighbor. Originally, the DBSCAN algorithm was developed for unweighted samples. Here, we modify the algorithm to work with weighted samples. We do so through modifying the criterion used to identify a cluster. Instead of counting the number of points, we evaluate the weighted sum of the points in

a cluster, whereby each weight is the corresponding IS weight of the sample. This method is able to detect clusters of arbitrary shape and identify outliers, marking them as noise. An initial guess for the missing data can then be obtained through assigning each sample to the corresponding cluster. Based on this initial pseudo-E-step, the alternating procedure of the EM algorithm can be conducted until the change in the log-likelihood is sufficiently small.

To reduce the risk of neglecting failure domains with small probability mass, not only the n_ρ samples with the smallest limit state function values are taken into account. Instead, an adaptive sample range is chosen for the DBSCAN algorithm. By choosing the samples between the 0.05- and the 0.25-quantiles of the limit state function values in the first step, the samples are selected according to the shape of the failure domain and the effect of possible outliers is reduced. In the subsequent steps, the samples in the 0.2-quantile of the limit state function values are selected, since it is assumed that the effect of outliers is reduced due to the distribution fitting by the EM algorithm and the applied IS weights.

4.4. Initialization

The CE algorithm outlined in Section 2 is directly applicable to the case with the GM distribution. For the parameter update, the EM algorithm is applied with n_ρ samples located at the intermediate failure domain. The choice of the initial sampling density is an important question. One possibility is to identify the initial sampling density through a pre-sampling step. Kurtz and Song [7] propose to run the CE method with a SG density employing a small sample size and thereby locate the radius of a hypersphere that intersects the failure domain. The mean vectors of the GM initial sampling density are uniformly sampled on the surface of this hypersphere, while the distribution weights are chosen as uniform weights and the covariance matrices as identity matrices [7]. However, this method requires additional computational effort and the number of distributions in the mixture has to be chosen a priori. Moreover, as discussed in [12], the sampled mean vectors from this procedure might lead to a biased estimate, as they might cluster near one of the important regions and not capture all of the failure domains. In [12], it is suggested to obtain the mean vectors through Latin hypercube sampling from the whole outcome space. Although this method is superior to the previous one regarding the computational effort, as it does not require additional evaluations of the limit state function, it still requires an initial choice of the numbers of distributions in the mixture. Here, we choose the initial sampling density as the original density of the random variables. This results in an unweighted initial sampling step and does not require to define the number of distributions in the mixture a priori, which is estimated by the DBSCAN algorithm as discussed in Section 4.3.

5. SIS with Gaussian mixture

SIS is an adaptive sampling method that was recently proposed for efficient estimation of the failure probability of engineering systems [6]. In SIS, it is proposed to make use of a GM model that is fitted by the EM algorithm for weighted samples, as presented in Section 4.2. Hence, a direct comparison of the performance of SIS to the CE method can be drawn within this study. In this section, a brief introduction to the principle of SIS is given, containing the most important ideas and their incorporation in the SIS algorithm. Note that in [6] the method is presented for application in an equivalent standard normal space. Here, we present the method for general application, i.e. without the need to transform the random variable space.

The basic idea of SIS is to adaptively sample a sequence of distributions that gradually approximate the optimal IS density $p^*(\mathbf{x})$ of Eq. (6). An approximation of $p^*(\mathbf{x})$ can be obtained through inserting the following smooth approximation of the indicator function in Eq. (6):

$$I(\mathbf{x}) \approx \Phi\left(\frac{-g(\mathbf{x})}{\sigma_L}\right) \quad (42)$$

This approximation becomes more accurate for a decreasing value of σ_L . Accordingly, a sequence of densities $h_l(\mathbf{x})$, $l = 1, \dots, L$ is defined, as

$$h_l(\mathbf{x}) = \frac{1}{P_l} \cdot \Phi\left(\frac{-g(\mathbf{x})}{\sigma_l}\right) \cdot f(\mathbf{x}) = \frac{\eta_l(\mathbf{x})}{P_l} \quad (43)$$

with $\infty = \sigma_1 > \sigma_2 > \dots > \sigma_L$. The normalizing constant P_l of density h_l can be expressed as [6]:

$$P_l = \int \eta_l(\mathbf{x}) \, d\mathbf{x} = P_{l-1} \int \frac{\eta_l(\mathbf{x})}{\eta_{l-1}(\mathbf{x})} \cdot h_{l-1}(\mathbf{x}) \, d\mathbf{x} = P_{l-1} \cdot \mathbb{E}_{h_{l-1}}[w_l(\mathbf{x})] \quad (44)$$

where $w_l(\mathbf{x}) = \frac{\eta_l(\mathbf{x})}{\eta_{l-1}(\mathbf{x})}$. Hence, an estimate of the ratio of subsequent normalizing constants can be derived through IS with samples distributed according to h_{l-1} as

$$\hat{S}_l = \frac{\hat{P}_l}{\hat{P}_{l-1}} = \frac{1}{n_s} \sum_{i=1}^{n_s} w_l(\mathbf{x}_i) \quad (45)$$

The parameters σ_l , $l = 2, \dots, L$ are chosen adaptively to ensure that two subsequent distributions do not vary significantly [6]. Sampling from the sequence of Eq. (43) can be performed through a resample-move scheme starting from samples from $h_1(\mathbf{x}) = f(\mathbf{x})$. In each step, samples from $h_{l-1}(\mathbf{x})$ are weighted according to the next distribution in the sequence through employing the likelihood ratio $w_l(\mathbf{x})$. The resulting weighted samples are resampled to obtain unweighted samples from $h_l(\mathbf{x})$ and moved to regions of high probability density of $h_l(\mathbf{x})$ through Markov chain Monte Carlo (MCMC) sampling. To achieve the latter it is proposed in [6] to apply an independent Metropolis-Hastings (MH) algorithm, which is demonstrated to perform well in low to moderate dimensions. The probability of failure is estimated with the following expression:

$$\hat{P}_F = \frac{\hat{P}_L}{n_s} \sum_{i=1}^{n_s} I(\mathbf{x}_i) \cdot \frac{f(\mathbf{x}_i)}{\eta_L(\mathbf{x}_i)}, \quad (46)$$

where

$$\hat{P}_L = \prod_{l=1}^L \hat{S}_l \quad (47)$$

For further details on the implementation of the method the reader is referred to [6].

The proposal distribution in the independent MH algorithm used in the MCMC step of the SIS algorithm is chosen in [6] as the GM model fitted to the samples from $h_{l-1}(\mathbf{x})$ weighted by $w_l(\mathbf{x})$. The fitting is performed by application of the EM algorithm for weighted samples as described in Section 4.2. However, in contrast to the CE method, which only employs $n_\rho = \rho \cdot n_s$ samples, SIS accounts for all n_s available samples in the parameter update, weighted according to the next distribution in the sequence. In [6], the number of densities in the mixture is chosen a priori. To increase the flexibility and to make the results comparable to the ones of the CE method, here the number of densities is chosen through application of the DBSCAN algorithm. In the first step of the SIS algorithm, the samples featuring weights between the 0.7-quantile and the 0.95-quantile of the importance weight values are selected for the clustering process, while in the following steps $\frac{n_s}{2}$ samples with highest importance weight values are used. Based on the cluster centers, an initial guess of the missing data is obtained.

6. Numerical investigations

We investigate the performance of the CE method with both the SG density and the GM distribution by means of several numerical examples and compare it to the performance of SIS. The effect of varying sample size is analyzed and the performance of the methods in increasing dimensions is assessed. It is noted that in the field of structural reliability the main computational effort lies in the model evaluation. Hence,

the total number of samples, i.e. the number of limit state function evaluations, is chosen as a measure for the computational cost.

All selected examples are problems in the independent standard normal space. Hence, all random variables x_1, \dots, x_n have the standard Gaussian distribution. The first example is a parabolic component reliability problem in two dimensions while the second one is a series system problem with four components. The third example consists of a linear limit state function and is well-suited to demonstrate the effect of increasing dimensions. The final example combines a linear limit state surface with a convex one on opposite sides of the origin. The reference values for the probability of failure for examples 6.1, 6.2 and 6.4 are obtained by MCS with 10^9 samples; for example 6.3, an analytical expression for the failure probability is available.

In the CE method, ρ is set to 0.1, which results in $n_\rho = 0.1 \cdot n_s$ available samples for the parameter update. In the MCMC algorithm within SIS, each Markov chain has length 10, with $\frac{n_s}{10}$ chains being run in each level. We measure the performance of the method in terms of the mean $E[\hat{P}_F]$ and coefficient of variation $\delta_{\hat{P}_F}$ of the estimate for the probability of failure, the average number of sampling steps required to reach the failure domain \bar{l} , the average total number of samples \bar{N}_{tot} (limit state function evaluations) and the average number of distributions in the GM in the final step \bar{k}_{fin} . These performance measures are estimated from 500 independent simulation runs.

Table 1: Results for the concave limit state function with a reference probability of failure of $3.01 \cdot 10^{-3}$. Comparison of CE method using SG, GM and SIS using GM for varying number of samples.

Number of samples per level	$E[\hat{P}_F]$	$\delta_{\hat{P}_F}$	\bar{l}	\bar{N}_{tot}	\bar{k}_{fin}
CE method using SG					
250	$2.99 \cdot 10^{-3}$	0.40	3.20	$0.80 \cdot 10^3$	1
500	$3.01 \cdot 10^{-3}$	0.18	3.09	$1.55 \cdot 10^3$	1
1000	$3.01 \cdot 10^{-3}$	0.11	3.02	$3.02 \cdot 10^3$	1
2000	$3.00 \cdot 10^{-3}$	0.07	3.00	$6.00 \cdot 10^3$	1
4000	$3.01 \cdot 10^{-3}$	0.05	3.00	$12.00 \cdot 10^3$	1
CE method using GM					
250	$1.53 \cdot 10^{-3}$	0.72	4.85	$1.21 \cdot 10^3$	2.55
500	$2.67 \cdot 10^{-3}$	0.33	3.51	$1.76 \cdot 10^3$	2.30
1000	$2.94 \cdot 10^{-3}$	0.10	3.17	$3.17 \cdot 10^3$	2.19
2000	$3.00 \cdot 10^{-3}$	0.04	3.08	$6.15 \cdot 10^3$	2.12
4000	$3.02 \cdot 10^{-3}$	0.04	3.06	$12.23 \cdot 10^3$	2.03
SIS using GM					
250	$1.67 \cdot 10^{-3}$	0.59	4.08	$1.02 \cdot 10^3$	3.09
500	$2.66 \cdot 10^{-3}$	0.21	3.98	$1.99 \cdot 10^3$	2.39
1000	$2.93 \cdot 10^{-3}$	0.12	3.98	$3.98 \cdot 10^3$	2.29
2000	$2.98 \cdot 10^{-3}$	0.07	4.00	$8.00 \cdot 10^3$	2.17
4000	$3.00 \cdot 10^{-3}$	0.05	4.00	$16.00 \cdot 10^3$	2.15

6.1. Concave limit state function

The limit state function of the first example has a parabolic shape, concave to the origin of the standard normal space and has two design points, as shown in Figure 1. It is defined as [23]:

$$g_1(\mathbf{x}) = 5 - x_2 - 0.5(x_1 - 0.1)^2 \quad (48)$$

The reference value for the probability of failure is $3.01 \cdot 10^{-3}$. The simulation results for CE using SG, GM and SIS using a GM are shown in Table 1. While using the SG in the CE method results in a practically unbiased estimate for any sample size, both methods using a GM show a significant negative bias for small sample sizes. In terms of the coefficient of variation of the estimate $\delta_{\hat{P}_F}$, the CE method with the SG is also superior for sample sizes lower than 1000 per level. By increasing the sample size, this performance gap can be closed and all methods provide unbiased estimates with comparable $\delta_{\hat{P}_F}$ of around 0.05 with

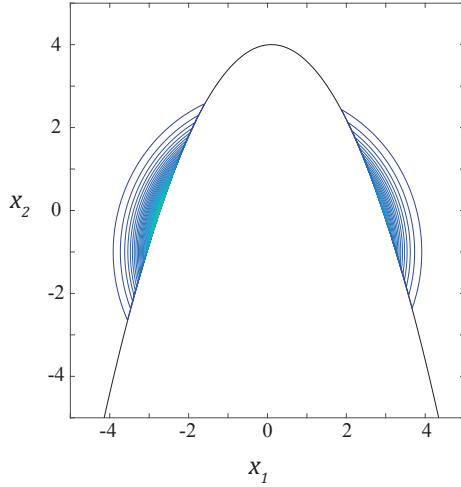


Figure 1: Optimal IS density for the concave limit state function with two design points.

4000 samples per level. For increasing sample sizes, the average number of sampling steps \bar{l} stabilizes for all methods. The CE method converges in three steps if the sample size is sufficiently large, while SIS needs an additional step on average. This results in a larger computational effort, which is illustrated by the total number of samples \bar{N}_{tot} .

For small sample sizes, the DBSCAN algorithm cannot identify the two design points properly, but overestimates the required number of distributions in the mixture on average. The EM algorithm is not able to correct this initial guess and fits the GM model with K distributions as proposed by the DBSCAN algorithm. As the sample size increases, a correct identification of the important regions is achieved for all methods. In addition, the performance of the CE method using GM with different choices of K and a fixed sample size of 1000 samples per level is investigated for this example. For this investigation, the initial value of K is fixed throughout the simulation, i.e. the DBSCAN algorithm is not applied to identify K . The initial guess for the probabilistic assignment of the samples to the distributions in the EM algorithm is obtained by randomly choosing K samples from the available samples at each level and setting them as the initial mean vectors. The results are summarized in Table 2. It appears that the performance of the CE method strongly depends on the choice of K . For an increasing K both the negative bias and $\delta_{\hat{P}_F}$ increase. The reason for this behavior is that as K increases a smaller number of samples is available for estimating the parameters of each distribution. Even in the case where two distributions are chosen, the method performs worse than when the adaptive mixture is used by application of the DBSCAN algorithm; the latter converges faster, which implies a smaller computational effort, and results in a smaller bias and $\delta_{\hat{P}_F}$ of the estimate.

Table 2: Results for the concave limit state function with a reference probability of failure of $3.01 \cdot 10^{-3}$. Comparison of CE method using GM with $n_s = 1000$ per level and the DBSCAN algorithm to CE method using GM with $n_s = 1000$ per level and different specified numbers of distributions in the mixture.

Number of distributions in the mixture	CE method using GM			
	$E[\hat{P}_F]$	$\delta_{\hat{P}_F}$	\bar{l}	\bar{N}_{tot}
DBSCAN	$2.94 \cdot 10^{-3}$	0.10	3.17	$3.17 \cdot 10^3$
2	$2.77 \cdot 10^{-3}$	0.19	4.00	$4.00 \cdot 10^3$
3	$2.70 \cdot 10^{-3}$	0.35	4.07	$4.07 \cdot 10^3$
5	$2.44 \cdot 10^{-3}$	0.43	4.11	$4.11 \cdot 10^3$
10	$1.77 \cdot 10^{-3}$	1.03	4.24	$4.24 \cdot 10^3$

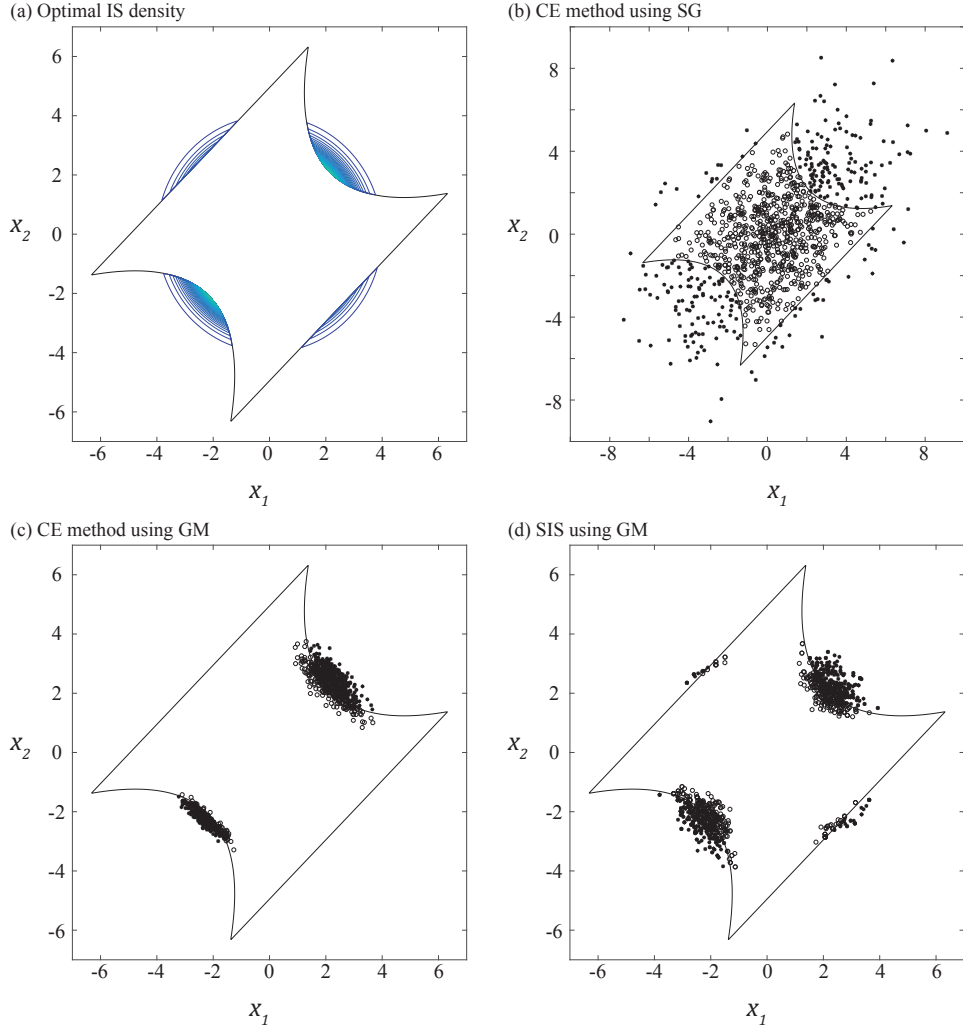


Figure 2: (a) Optimal IS density for series system problem with four distinct failure domains; (b)-(d) Samples of approximate optimal IS density with $n_s = 1000$ obtained from CE method using SG (b), CE method using GM (c) and SIS using GM (d).

6.2. Series system problem

This example consists of a series system reliability problem with four components. The limit state function is defined by [24]:

$$g_2(\mathbf{x}) = \min \left\{ \begin{array}{l} 0.1(x_1 - x_2)^2 - \frac{(x_1 + x_2)}{\sqrt{2}} + 3 \\ 0.1(x_1 - x_2)^2 + \frac{(x_1 + x_2)}{\sqrt{2}} + 3 \\ x_1 - x_2 + \frac{\sqrt{2}}{2} \\ x_2 - x_1 + \frac{\sqrt{2}}{2} \end{array} \right\} \quad (49)$$

Two of the components feature a parabolic limit state surface convex to the origin, while the remaining two components are of the linear type. Altogether, this results in four distinct failure domains.

The reference value for the probability of failure is $2.22 \cdot 10^{-3}$. For the case with $n_s = 1000$, the CE method using SG results in an estimate of $2.17 \cdot 10^{-3}$, producing a small negative bias, while $\delta_{\hat{P}_F}$ is 0.21. Using the GM in the CE method results in a significantly larger negative bias and a larger $\delta_{\hat{P}_F}$; the average estimate

of the probability of failure is $1.45 \cdot 10^{-3}$ and $\delta_{\hat{P}_F}$ is 0.30. SIS results in an average estimate of $1.82 \cdot 10^{-3}$ with $\delta_{\hat{P}_F}$ of 0.16. In Figure 2, the samples of the final step of an exemplary simulation run are illustrated for the three methods. The CE method using SG results in widely spread samples covering not only the failure regions but also the safe domain, while the methods that employ the GM model concentrate on the failure regions. Apparently, the CE method using GM is not able to detect all failure domains, it only covers the two most important ones which results in a strongly biased estimate for the failure probability. In contrast, SIS using GM identifies successfully the failure regions and generates samples in each of the distinct domains.

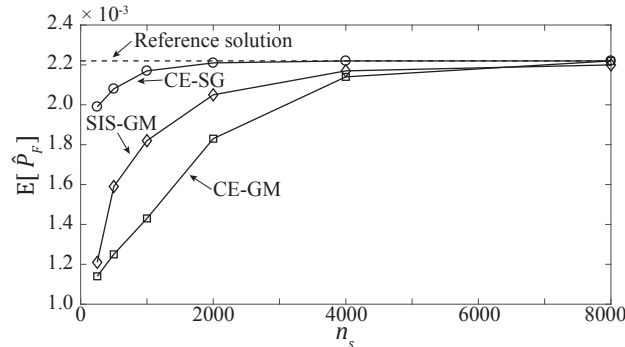


Figure 3: Estimate of the failure probability for series system problem against number of samples per level.

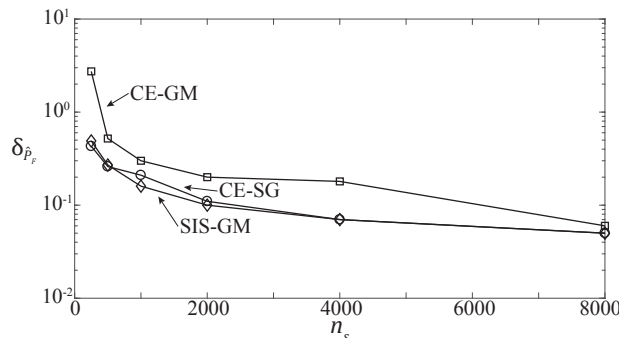


Figure 4: $\delta_{\hat{P}_F}$ for series system problem against number of samples per level.

We now investigate the effect of varying sample size. Figure 3 shows the average estimate for an increasing number of samples per sampling step n_s . It is shown that the CE method using SG provides an unbiased estimate for a relatively low number of samples, whereas, the CE method using GM features a significant negative bias even for relatively large sample sizes. SIS generates a negative bias for small sample sizes but converges to an unbiased estimate faster than the CE method using GM. For large sample sizes all three of the methods provide practically unbiased estimates. In terms of $\delta_{\hat{P}_F}$, which is illustrated in Figure 4, the CE method using SG and SIS have similar performance, while the CE method using GM results in a larger $\delta_{\hat{P}_F}$ for low and moderate sample sizes. Only when using a large number of samples this performance gap can be closed. While the CE method using SG converges within approximately three steps for $n_s \geq 1000$, SIS always converges in four steps if n_s is larger or equal to 500. The CE method using GM needs more steps than SIS to converge for small numbers of samples, but if n_s is chosen as 2000 or larger, it converges faster than SIS. For large n_s it converges as fast as the CE method using SG. Finally, the number of distributions in the GM in the final step of the simulation with respect to the sample size are analyzed. As the limit state function consists of four distinct failure domains, the optimal number is

$k_{fin} = 4$. In both the CE method using GM and SIS using GM, the DBSCAN algorithm underestimates the number of distributions when using a small number of samples and is only able to identify the two parabolic failure domains with higher probability mass. For increasing n_s , the DBSCAN algorithm within SIS is able to identify all four distinct modes, while in the CE method using GM it underestimates this number on average.

6.3. Linear limit state function in varying dimensions

This example is defined by the following limit state function:

$$g_3(\mathbf{x}) = \beta - \frac{1}{\sqrt{n}} \sum_{i=1}^n x_i \quad (50)$$

It is designed to test the effect of increasing dimensions on the performance of a method, as its probability of failure is independent of the number of dimensions and depends only on the parameter β . Moreover, the limit state function remains linear and features one failure domain for any number of dimensions [25]. An analytical solution is available for the probability of failure, given by $P_F = \Phi(-\beta)$. We choose $\beta = 3.5$, which corresponds to a failure probability of $2.33 \cdot 10^{-4}$. The simulation in two dimensions for $n_s = 1000$ samples per level results in strongly biased estimates for the CE method. Using the SG, the average estimate of the probability of failure is $1.52 \cdot 10^{-4}$ with a $\delta_{\hat{P}_F}$ of 1.53. Using the GM results in a mean estimate of $0.64 \cdot 10^{-5}$ with a $\delta_{\hat{P}_F}$ of 1.44. The number of distributions in the GM is estimated as 2.33 on average.

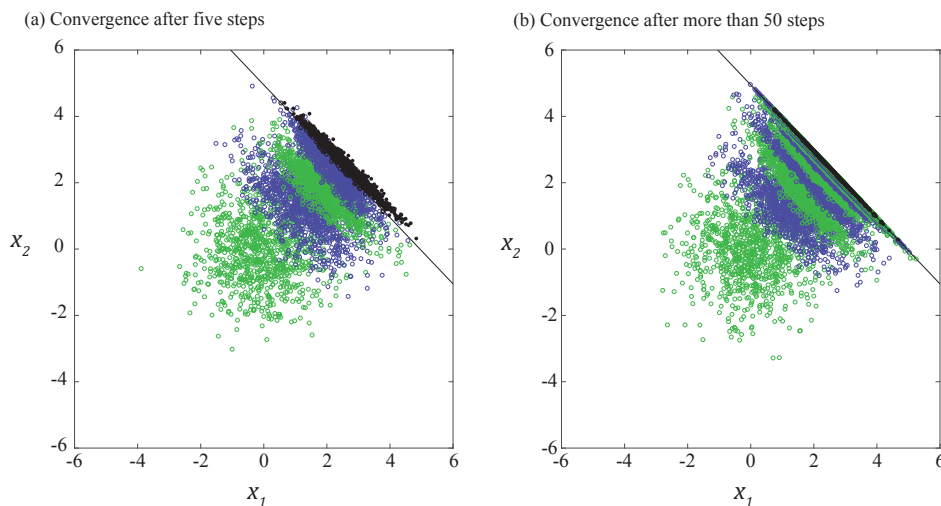


Figure 5: Step-wise samples for the linear limit state function obtained from CE method using SG with $n_s = 1000$ per level. (a) Convergence to the failure domain in five steps; (b) Convergence to the failure domain in more than 50 steps.

We now investigate the number of sampling steps required to reach the failure domain in the CE method. On average, the CE method using SG needs 16.89 steps, while the CE method using GM needs 36.43 steps. In some cases, the algorithm does not converge to the failure domain within the maximum number of steps, while in other cases, the algorithm converges in less than 10 steps. This occurs due to the linear shape of the failure domain. Because the Gaussian model is only able to describe linear dependence, the fitted IS density based on the $\rho \cdot n_s$ available samples at the linear intermediate failure domain results in a high linear dependence between the random variables. Hence, most of the generated samples in the next step are located in a narrow sector around the previous intermediate failure domain. This decreases the step size towards the actual failure domain. In the extreme case, the step size tends towards zero and the algorithm gets stuck at an intermediate level with a sampling density giving almost perfect linear dependence between the random variables. If the samples of the first steps are sufficiently scattered, the resulting sampling

densities feature a smaller linear dependence. In that case, the algorithm is able to advance to the failure domain and give an estimate of the failure probability. Figure 5 (a) illustrates a case with fast convergence to the failure domain, while Figure 5 (b) shows a case with slow convergence to the failure domain. The increasing linear dependence between the samples and the decreasing step-size towards the failure domain are clearly visible. This effect becomes less severe with an increase of the number of samples per level. Running the simulation for the two-dimensional case using SIS with $n_s = 1000$ per level results in a negative bias, as the probability of failure is estimated as $1.60 \cdot 10^{-4}$, yet $\delta_{\hat{P}_F} = 0.54$ is significantly smaller than with the CE method. The average number of steps until convergence is 5.17, while the number of Gaussians in the mixture is 3.27 on average. Apparently, the MCMC algorithm within SIS prevents the algorithm from getting stuck at intermediate failure domains, as it converges in less than 10 steps in every simulation run. To investigate the effect of increasing dimensions, the number of samples per level is increased for the two-dimensional case until all methods result in satisfactory performance in terms of bias and coefficient of variation of the estimate. For $n_s = 8000$ the CE method using SG gives an estimate of $2.26 \cdot 10^{-4}$ for the failure probability in 5.67 steps on average with a corresponding $\delta_{\hat{P}_F}$ of 0.10. The CE method using GM requires 5.58 steps on average to estimate the probability of failure as $2.01 \cdot 10^{-4}$ with a $\delta_{\hat{P}_F}$ of 0.31, while SIS gives an estimate of $2.28 \cdot 10^{-4}$ with a $\delta_{\hat{P}_F}$ of 0.04 within 5.00 steps on average. Hence, $n_s = 8000$ per level is chosen for the analysis of increasing dimensions.

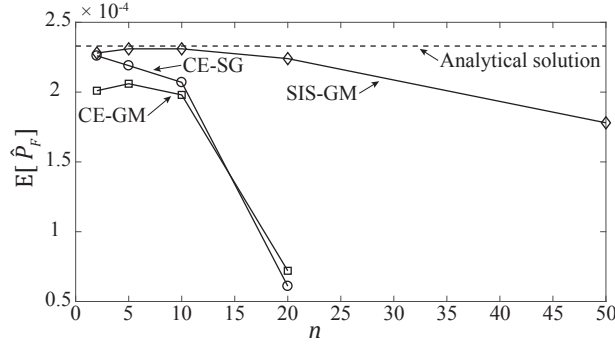


Figure 6: Estimated probability of failure for the linear limit state function plotted against number of random variables obtained with $n_s = 8000$ per level.

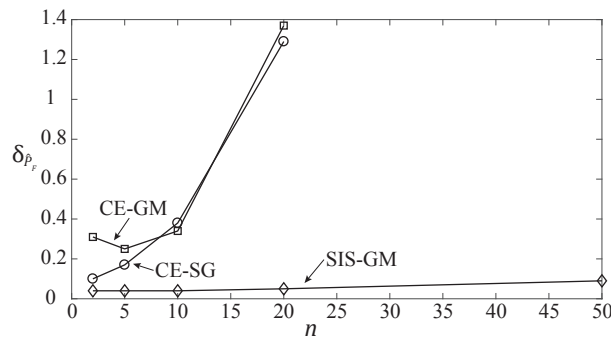


Figure 7: $\delta_{\hat{P}_F}$ for the linear limit state function plotted against number of random variables.

Figure 6 compares the average estimates of the failure probability obtained with the three different methods, while the dashed line marks the analytical solution of the linear limit state function. In Figure 7, $\delta_{\hat{P}_F}$ is compared for varying number of dimensions. The two versions of the CE method show similar behavior, as both feature a rapid increase of the bias and $\delta_{\hat{P}_F}$ for increasing dimensions. It is noted that

for $n \geq 10$ the DBSCAN algorithm always reduces the GM to a single Gaussian density and thus the two methods coincide for those cases. The figures show that the CE method with Gaussian distributions is not able to deal with large numbers of random variables. As soon as n is larger than ten, the negative bias and $\delta_{\hat{P}_F}$ increase dramatically. In the case when $n = 20$, the results are not reliable anymore. SIS results in practically unbiased estimates for up to 20 dimensions, while this bias increases with further increase of the number of random variables. $\delta_{\hat{P}_F}$ remains moderate for up to 50 dimensions, with an increasing tendency. It is mentioned that in [7], a $\delta_{\hat{P}_F}$ of 0.05 is achieved for this problem in 60 dimensions with approximately $4.5 \cdot 10^4$ samples using the GM in the CE method. However, this number does not include the samples required for the pre-sampling step to determine the mean vectors and thus cannot be directly compared to the computational cost reported in this paper.

6.4. Combination of linear and convex limit state function

Finally, a combination of a linear limit state function and a limit state function with convex shape is investigated in a two-dimensional standard normal space. It features two distinct failure domains on opposite sides of the origin. The combined limit state function is defined by [25, 26]:

$$g_4(\mathbf{x}) = \min \left\{ \begin{array}{l} 3.2 + \frac{1}{\sqrt{n}} \sum_{i=1}^n x_i \\ 0.1(x_1 - x_2)^2 - \frac{(x_1 + x_2)}{\sqrt{2}} + 2.5 \end{array} \right\} \quad (51)$$

The reference value for the probability of failure is $4.90 \cdot 10^{-3}$. This example is used for investigating the performance of the methods in system problems with limit state functions of different type. While the convex limit state function features a distinct design point with most of the probability mass in its direct neighborhood, the failure domain of the linear limit state function exhibits probability mass distributed along the linear limit state surface. Furthermore, the convex limit state function makes up more than 85% of the overall probability mass in the failure domain.

The optimal IS density and samples of the approximated near-optimal IS densities obtained with the investigated methods are illustrated in Figure 8. It appears that the CE method using SG is able to capture both failure domains with a small number of samples per level ($n_s = 1000$), whereas, the CE method using GM is more prone to neglecting the linear failure domain and to exclusively approximating the area around the design point of the convex failure domain. By an increase of the sample size this effect is reduced and the algorithm successfully detects both failure domains. Table 3 compares the statistics of the simulation obtained with the CE method and SIS for a varying sample size.

It is shown that the CE method using SG gives nearly unbiased estimates for the failure probability with $n_s = 1000$ and the small negative bias vanishes with an increasing sample size. The CE method using GM needs a larger number of samples to give unbiased results, for $n_s = 1000$ there is a negative bias. This bias can be attributed to the fact that the method neglects the linear failure domain. The convex limit state function alone has a reference value for the probability of failure of $4.21 \cdot 10^{-3}$, which is close to the value estimated by the CE method using GM with 1000 samples per level. By increasing the sample size the bias is decreased until the method gives a nearly unbiased estimate with 16000 samples per level. In terms of $\delta_{\hat{P}_F}$, for small to moderate sample sizes the CE method using SG outperforms the CE method using GM and for large sample sizes the two methods perform approximately the same. For this particular example the CE method using SG always converges in three steps while the CE method using GM needs an additional step, converging in four steps for any number of samples per level. The comparison with SIS shows that SIS gives biased estimates for a small number of samples per level and this bias vanishes with an increasing sample size. $\delta_{\hat{P}_F}$ is smaller than the one obtained with the CE method, while the numerical effort is comparable to the CE method using GM.

The number of Gaussians in the mixture estimated by the DBSCAN algorithm increases for increasing sample size, while in SIS this number slightly decreases for increasing sample size. That is, the CE method with GM overestimates the number of modes of the optimal IS density as the sample size increases. A reason for this can be found in the linear failure domain. Several Gaussians are allocated there by the algorithm, each of them with highly correlated random variables, which results in parallel arranged sample clusters in

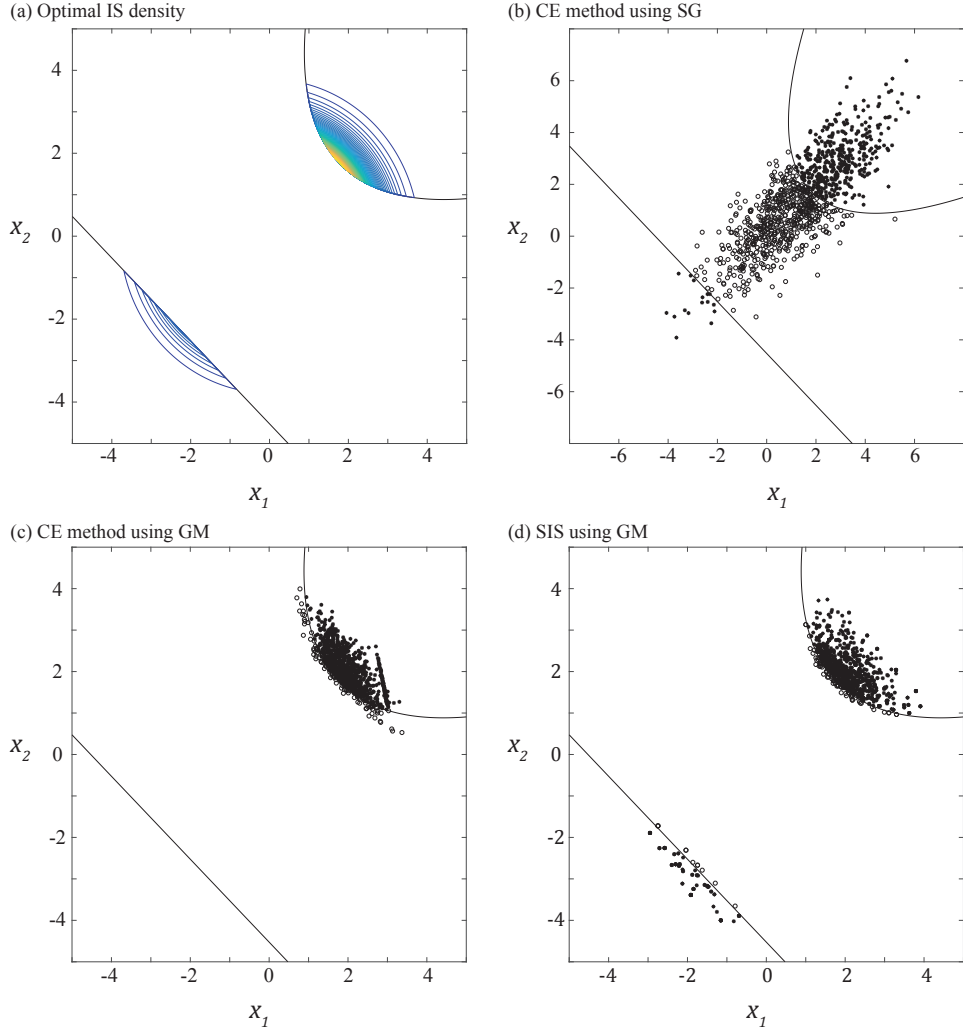


Figure 8: (a) Optimal IS density for combined limit state function with linear and convex limit state surface; (b)-(d) Samples of approximate optimal IS density obtained from CE method using SG with $n_s = 1000$ (b), CE method using GM with $n_s = 1000$ (c) and SIS using GM with $n_s = 1000$ (d).

the failure domain of the linear limit state function. SIS is more robust in this regard due to the MCMC algorithm conducted in each iteration step.

6.5. Accuracy of the investigated methods

This section considers an overall measure for the accuracy of a method, which combines the bias and variability of the estimates. It is defined as the square root of the mean-square error of the estimated probability of failure:

$$err_{\hat{P}_F} = \sqrt{E \left[\left(\hat{P}_F - P_F \right)^2 \right]} = \sqrt{\left(E[\hat{P}_F] - P_F \right)^2 + \left(E[\hat{P}_F] \cdot \delta_{\hat{P}_F} \right)^2} \quad (52)$$

With this measure, it is possible to compare the performances of the investigated methods by plotting the error as a function of the total computational effort, i.e. the average total number of samples, needed to

Table 3: Results for the combination of a linear and a convex limit state function with a reference probability of failure of $4.90 \cdot 10^{-3}$. Comparison of CE method using SG, GM and SIS using GM for varying number of samples.

Number of samples per level	$E[\hat{P}_F]$	$\delta_{\hat{P}_F}$	\bar{l}	\bar{N}_{tot}	\bar{k}_{fin}
CE method using SG					
1000	$4.83 \cdot 10^{-3}$	0.13	3.00	$3.00 \cdot 10^3$	1
2000	$4.89 \cdot 10^{-3}$	0.10	3.00	$6.00 \cdot 10^3$	1
4000	$4.90 \cdot 10^{-3}$	0.07	3.00	$12.00 \cdot 10^3$	1
8000	$4.89 \cdot 10^{-3}$	0.04	3.00	$24.00 \cdot 10^3$	1
16000	$4.90 \cdot 10^{-3}$	0.03	3.00	$48.00 \cdot 10^3$	1
CE method using GM					
1000	$4.19 \cdot 10^{-3}$	0.26	4.00	$4.00 \cdot 10^3$	3.49
2000	$4.52 \cdot 10^{-3}$	0.16	4.00	$8.00 \cdot 10^3$	3.54
4000	$4.74 \cdot 10^{-3}$	0.14	4.00	$16.00 \cdot 10^3$	4.29
8000	$4.80 \cdot 10^{-3}$	0.05	4.00	$32.00 \cdot 10^3$	4.74
16000	$4.85 \cdot 10^{-3}$	0.04	4.00	$64.00 \cdot 10^3$	5.21
SIS using GM					
1000	$4.64 \cdot 10^{-3}$	0.12	3.96	$3.96 \cdot 10^3$	2.96
2000	$4.76 \cdot 10^{-3}$	0.08	3.98	$7.97 \cdot 10^3$	2.61
4000	$4.83 \cdot 10^{-3}$	0.05	4.00	$16.00 \cdot 10^3$	2.52
8000	$4.87 \cdot 10^{-3}$	0.03	4.00	$32.00 \cdot 10^3$	2.47
16000	$4.88 \cdot 10^{-3}$	0.02	4.00	$64.00 \cdot 10^3$	2.42

converge. For the examples analyzed in Sections 6.1 - 6.4, the normalized error is plotted in Figure 9. An interesting effect can be observed for the linear limit state function (Figure 9 (c)). The overall computational effort of the two CE-based methods first decreases with an increasing number of samples per level before eventually increasing again. This can be explained by the fact that with an increasing number of samples per level, the methods need less steps to converge to the failure domain and therefore the total number of limit state function calls can decrease.

7. Discussion

This study investigates the performance of the CE method for structural reliability with two different IS densities, the multivariate normal distribution and the Gaussian mixture distribution. Updating procedures for both distributions within the CE algorithm are discussed. We extend the one-step updating procedure for the GM distribution proposed in [7] to an iterative updating procedure through a new EM algorithm. Furthermore, the DBSCAN algorithm is introduced as a clustering algorithm to determine the number of distributions in the mixture. The initial parameters of the sampling distribution are chosen such that the first CE iteration step is an unweighted MCS. This has the advantage that no knowledge of the failure modes of the investigated problem is required in advance.

Several examples of limit state functions in standard normal space are analyzed to show the performance of the different methods. These include component reliability problems as well as system reliability problems in low and high dimensions. Moreover, the results are compared to SIS, another recently proposed sampling-based method. Within this method, the EM algorithm is used to fit a GM distribution based on weighted samples and the fitted distribution is applied as proposal distribution in an MCMC sampling approach. Hence, a direct comparison of SIS and the CE method can be drawn.

The concave example shows that the CE method using GM performs worse than the CE method using SG. Especially when a small sample size per level is chosen, the CE method using GM features a significant bias, while the CE method using SG gives nearly unbiased estimates. Furthermore, the CE method using SG converges faster to the failure domain, which decreases the computational effort. SIS using GM performs better than the CE method using GM but compared to the CE method using SG the bias and $\delta_{\hat{P}_F}$ for small sample sizes is larger and it requires more steps to converge. In addition, the effect of the number of distributions in the GM is investigated. It appears that a specified number results in larger bias and $\delta_{\hat{P}_F}$

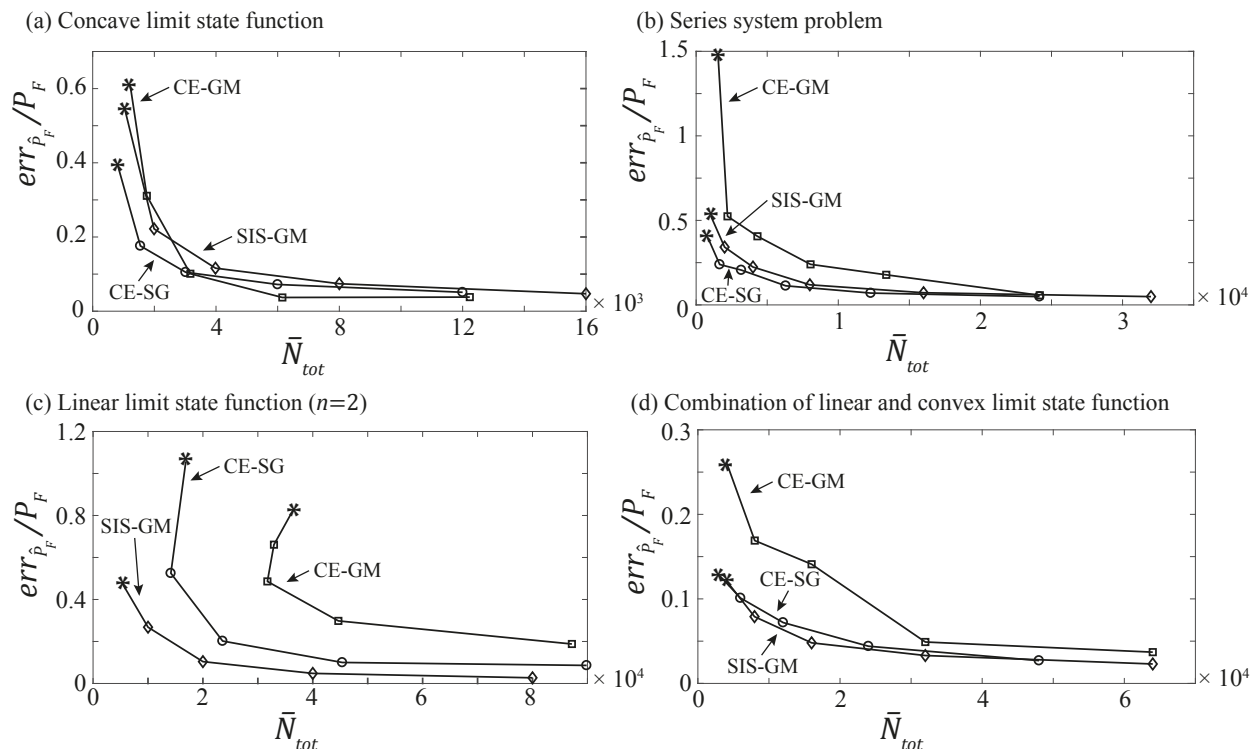


Figure 9: Normalized error of the investigated methods for examples 6.1 - 6.4 plotted against the average of the total number of samples. For each curve, the asterisk at the beginning indicates the smallest choice of n_s per level; along the curves the n_s per level increases.

compared to an adaptive estimation of the number of Gaussians K using the DBSCAN algorithm, even for the case when K is chosen as the number of failure domains of the problem.

The series system problem shows the ability of the SG distribution to describe multimodal failure by widely spread samples across the outcome space. The CE method using GM faces severe problems in identifying all failure domains, which derives from the fact that only a small number of samples is taken into account for the parameter update, whereas, SIS using GM can correctly identify all failure regions and approximate the failure domain with a GM distribution. The reason for that can be found in the MCMC algorithm conducted in every step of the SIS algorithm, which moves the samples to regions of high probability mass of the target distribution.

A linear limit state function with a failure probability independent of the dimension is analyzed to show the performance of the different methods in varying dimensions. A simulation in two dimensions with small number of samples per level shows that the CE method has difficulties in approximating the optimal IS density of the linear failure domain. Depending on the spread of the samples in the initial steps, the algorithm either converges to the failure domain or gets stuck at an intermediate domain. By choosing a large number of samples this behavior can be avoided, which is also apparent in the results obtained in [7]. As the number of random variables increase, both the CE method using SG and the CE method using GM show a rapidly increasing bias and $\delta_{\hat{P}_F}$. A reliable estimate cannot be provided for dimensions larger than 10. On the other hand, SIS is able to provide estimates with acceptable bias and $\delta_{\hat{P}_F}$ for up to 50 random variables.

Finally, a limit state function with two failure domains of different shape is investigated. One of the domains is of convex shape and features a distinct design point while the other one is described by a linear limit state surface. It is shown that the CE method using SG is able to capture both failure domains and provide an unbiased estimate even for a small number of samples per level, while the CE method using GM requires

a larger number of samples to achieve the same accuracy. If the number of samples is chosen too small, the CE algorithm with GM is drawn to the design point of the convex limit state function and thus cannot capture the second failure domain. This problem does not occur when using SIS with a GM distribution. In Figure 9, we plotted the normalized error of the investigated methods, which combines the bias and coefficient of variation, as a function of the computational effort. This comparison shows that the CE method using SG performs more accurate than the CE method using GM for most of the investigated problems or at least equally accurate. SIS using GM gives accurate results for all investigated examples, but is sometimes less accurate than the CE method using SG for small computational efforts. Altogether, the CE method using GM performs poorly if no knowledge on the system behavior is available in advance. Using the original joint PDF of the random variables as the initial sampling density provides flexibility, however in cases with multiple failure regions of varying importance, the samples used to fit the GM model in the first sampling step tend to concentrate in regions of higher importance. In [7], the authors attempt to overcome this issue through running a pre-sampling step to gain information about the failure modes of the system. However, this approach requires additional computational effort and if a small sample size is used in the pre-sampling it might also result in ignoring failure regions [12]. One way to improve the performance of the CE method with GM could be to run a FORM analysis to identify the design points of all failure domains [27]. The distributions in the GM can then be centered at the design points in the first step of the CE algorithm. However, there is no guarantee that the FORM optimization will identify all design points, whereas the additional involved computational effort of gradient-based optimization algorithms that are often used within FORM increases fast with increase of the dimension.

8. Conclusion

This study investigates the suitability of Gaussian densities as importance sampling densities in the cross entropy (CE) method. We discuss that contrary to the the single Gaussian distribution, closed-form updating rules for the parameters cannot be obtained for the Gaussian mixture distribution and the update can be done with an iterative expectation-maximization algorithm. The number of distributions in the mixture is crucial for the performance of the method. By introducing a modified DBSCAN algorithm, an efficient procedure to estimate this number in each step of the CE method is proposed. We show that the CE method using a single Gaussian is superior to the CE method using a Gaussian mixture if no knowledge of the system behavior is available in advance both in terms of the quality of the estimate and the computational effort. We also demonstrate that the CE method using either single Gaussian or Gaussian mixture is not a good choice for linear and high dimensional problems. We compare the performance of the CE method with the one of sequential importance sampling (SIS), which employs the Gaussian mixture model as a proposal distribution in a Markov chain Monte Carlo sampling setting. SIS is able to approximate the optimal importance sampling density of problems with multiple failure domains with the Gaussian mixture model and outperforms the CE method in problems with linear failure domains and moderately high dimensions.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) in the framework of the priority programme SPP 1886 by grant STR 1140/6-1.

References

- [1] M. Lemaire, A. Chateauneuf, J.-C. Mitteau, *Structural Reliability*, Wiley-ISTE, 2009.
- [2] R. Rackwitz, Reliability analysis - a review and some perspectives, *Structural Safety* 23 (4) (2001) 365 – 395.
- [3] A. Der Kiureghian, First- and second-order reliability methods, in: E. Nikolaidis, D. M. Ghiocel, S. Singhal (Eds.), *Engineering design reliability handbook*, CRC Press, 2005, Ch. 14.
- [4] R. Y. Rubinstein, D. P. Kroese, *Simulation and the Monte Carlo method*, 3rd Edition, Springer, 2017.
- [5] S.-K. Au, J. L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, *Probabilistic Engineering Mechanics* 16 (4) (2001) 263 – 277.

- [6] I. Papaioannou, C. Papadimitriou, D. Straub, Sequential importance sampling for structural reliability analysis, *Structural Safety* 62 (2016) 66 – 75.
- [7] N. Kurtz, J. Song, Cross-entropy-based adaptive importance sampling using Gaussian mixture, *Structural Safety* 42 (2013) 35 – 44.
- [8] S. Au, J. Beck, A new adaptive importance sampling scheme for reliability calculations, *Structural Safety* 21 (2) (1999) 135 – 158.
- [9] M. Hohenbichler, R. Rackwitz, Improvement of second-order reliability estimates by importance sampling, *Journal of Engineering Mechanics* 114 (12) (1988) 2195–2199.
- [10] P. Koutsourelakis, H. Pradlwarter, G. Schuëller, Reliability of structures in high dimensions, part i: algorithms and applications, *Probabilistic Engineering Mechanics* 19 (4) (2004) 409 – 417.
- [11] P. T. de Boer, D. P. Kroese, S. Mannor, R. Y. Rubinstein, A tutorial on the cross-entropy method, *Annals of operations research* 134 (1) (2005) 19–67.
- [12] D. Y. Yang, J. Teng, D. M. Frangopol, Cross-entropy-based adaptive importance sampling for time-dependent reliability analysis of deteriorating structures, *Structural Safety* 66 (2017) 38 – 50.
- [13] G. Biondini, An introduction to rare event simulation and importance sampling, in: *Big data analytics*, Vol. 33 of *Handbook of statistics*, Elsevier, 2015, pp. 29 – 68.
- [14] D. P. Kroese, R. Y. Rubinstein, P. W. Glynn, Chapter 2 - The cross-entropy method for estimation, in: *Handbook of Statistics: Machine Learning: Theory and Applications*, Vol. 31 of *Handbook of Statistics*, Elsevier, 2013, pp. 19 – 34.
- [15] J. C. Baez, T. Fritz, A bayesian characterization of relative entropy, *CoRR* abs/1402.3067.
- [16] M. Fujita, R. Rackwitz, Updating first- and second-order reliability estimates by importance sampling, *Structural Eng./ Earthquake Eng.* 5 (1) (1988) 31–37.
- [17] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society. Series B (Methodological)* 39 (1) (1977) 1–38.
- [18] Y. Chen, M. R. Gupta, EM demystified: an expectation-maximization tutorial, Tech. rep., Department of electrical engineering, University of Washington (2010).
- [19] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Volume 1: *Statistics*, University of California Press, Berkeley, Calif., 1967, pp. 281–297.
- [20] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, Technical Report 2006-13, Stanford InfoLab (2006).
- [21] I. D. Gebru, X. Alameda-Pineda, F. Forbes, R. Horaud, EM algorithms for weighted-data clustering with application to audio-visual scene analysis, *CoRR* abs/1509.01509.
- [22] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: *Proceedings of the second international conference on knowledge discovery and data mining*, AAAI Press, 1996, pp. 226–231.
- [23] A. Der Kiureghian, T. Dakessian, Multiple design points in first and second-order reliability, *Structural Safety* 20 (1) (1998) 37 – 49.
- [24] P. H. Waarts, Structural reliability using finite element analysis; an appraisal of DARS: directional adaptive response surface sampling, Ph.D. thesis, Technische Universiteit Delft (2000).
- [25] S. Engelund, R. Rackwitz, A benchmark study on importance sampling techniques in structural reliability, *Structural Safety* 12 (4) (1993) 255 – 276.
- [26] S. Katsuki, D. M. Frangopol, Hyperspace division method for structural reliability, *Journal of Engineering Mechanics* 120 (11) (1994) 2405–2427.
- [27] G. Schuëller, R. Stix, A critical appraisal of methods to determine failure probabilities, *Structural Safety* 4 (4) (1987) 293 – 309.