To appear in *Probabilistic Engineering Mechanics*
June 2015

MCMC algorithms for Subset Simulation

Iason Papaioannou*, Wolfgang Betz, Kilian Zwirglmaier, Daniel Straub

Engineering Risk Analysis Group, Technische Universität München, 80290 Munich, Germany

*Corresponding author. Email: iason.papaioannou@tum.de

Abstract

Subset Simulation is an adaptive simulation method that efficiently solves structural reliability problems with many random variables. The method requires sampling from conditional distributions, which is achieved through Markov Chain Monte Carlo (MCMC) algorithms. This paper discusses different MCMC algorithms proposed for Subset Simulation and introduces a novel approach for MCMC sampling in the standard normal space. Two variants of the algorithm are proposed: A basic variant, which is simpler than existing algorithms with equal accuracy and efficiency, and a more efficient variant with adaptive scaling. It is demonstrated that the proposed algorithm improves the accuracy of Subset Simulation, without the need for additional model evaluations.

Keywords

MCMC; Subset Simulation; reliability analysis; high dimensions; conditional sampling; adaptive scaling

1. Introduction

Structural reliability analysis is concerned with the evaluation of the probability of failure, defined by the following n-fold integral:

$$P_f = \int_{g(\mathbf{x})\leq 0} f_{\mathbf{X}}(\mathbf{x})\, d\mathbf{x} \tag{1}$$

$\mathbf{X}$ is a random vector of dimension n and models the system variables that are expected to present an uncertain behavior, $f_{\mathbf{X}}(\mathbf{x})$ is the joint probability density function (PDF) of $\mathbf{X}$ and $g(\mathbf{x}) \leq 0$ defines failure of the system. The function $g(\mathbf{x})$ is termed limit-state function and can include one or several distinct failure modes (Ditlevsen & Madsen 1996). $g(\mathbf{x})$ can express any type of system failure, or indeed any event of interest described by a numerical or analytical model.

It is common to transform the random variables $\mathbf{X}$ to a probability space $\mathbf{U}$ consisting of independent standard normal random variables, through a one-to-one mapping $\mathbf{U} = \mathbf{T}(\mathbf{X})$. If the joint PDF $f_{\mathbf{X}}(\mathbf{x})$ is known then the mapping can be defined by the Rosenblatt transformation (Hohenbichler & Rackwitz 1981). However, usually the probabilistic description of $\mathbf{X}$ comes in terms of marginal distributions and correlations. In this case, the joint PDF is commonly modeled by the Nataf distribution (Gaussian copula) and the mapping to the standard normal space can be achieved through a marginal transformation (Der Kiureghian & Liu 1986). The probability of failure can be expressed in the transformed space as

$$P_f = \int_{G(\mathbf{u})\leq 0} \varphi_n(\mathbf{u})\, d\mathbf{u} \tag{2}$$

where $\varphi_n(\mathbf{u}) = \prod_{i=1}^{n} \varphi(u_i)$, $\varphi(.)$ is the standard normal PDF and $G(\mathbf{u}) = g\left(\mathbf{T}^{-1}(\mathbf{u})\right)$ is the limit-state function in the $\mathbf{U}$-space.

The evaluation of the probability of failure is not a trivial task, especially when the performance of the system for a realization of the random variables is obtained through a computationally demanding model evaluation. Existing approaches include

approximation methods such as the first/second order reliability method (FORM/SORM), as well as simulation techniques based on the Monte Carlo method.

FORM/SORM methods are based on first/second order Taylor series approximation of the limit-state surface at the so-called design point in the **U**-space, which is found through the solution of an optimization problem. These methods result in approximations of the probability of failure, which may involve considerable errors especially in problems where the dimension n of the random variable space is large or in problems with highly nonlinear limit-state functions (Rackwitz 2001, Valdebenito et al. 2010).

The Monte Carlo method is a simple and robust technique that is able to handle any model, independent of its complexity. The efficiency of the Monte Carlo method in its standard form does not depend on the dimension of the random variable space. The classical Monte Carlo method estimates P_f in Eq. (2) by generating samples of the random vector $\mathbf{U}$ and taking the sample mean of the indicator function $I(\mathbf{U})$, where $I(\mathbf{u}) = 1$ if $G(\mathbf{u}) \leq 0$ and $I(\mathbf{u}) = 0$ otherwise. The drawback of the crude Monte Carlo method is that its computational demands for assessing small failure probabilities are high – the required number of samples and hence the required model evaluations is inversely proportional to the probability of failure.

The efficiency of the Monte Carlo method can be enhanced by application of variance reduction methods such as importance sampling, typically applying a unimodal importance sampling density in the $\mathbf{U}$-space (Schuëller & Stix 1987, Bucher 1988). However, in problems with a large number of random variables, unimodal importance sampling fails to describe the important region leading to a dramatic increase in the variance of the resulting estimate (Au & Beck 2003a, Katafygiotis & Zuev 2008). Moreover, importance sampling with multimodal sampling densities (Ang et al. 1992, Au & Beck 1999, Kurtz & Song 2013) encounters difficulties in estimating the sought optimal sampling density in nonlinear high-dimensional problems (Schuëller et al. 2004, Katafygiotis & Zuev 2008).

In order to overcome the inefficiency of the Monte Carlo method in estimating small failure probabilities, while maintaining its independency on the number of random

variables, a number of advanced simulation methods have been developed, including Subset Simulation (Au & Beck 2001), Spherical Subset Simulation (Katafygiotis & Cheung 2007), Line Sampling (Hohenbichler & Rackwitz 1988, Koutsourelakis et al. 2004) and Asymptotic Sampling (Bucher 2009). Here, we focus on Subset Simulation. This method expresses P_f as a product of conditional probabilities that are significantly larger than P_f. These conditional probabilities are estimated by application of Markov Chain Monte Carlo (MCMC) sampling. It should be noted that Subset Simulation has been studied in the mathematical literature under the surrogate terms Sequential Monte Carlo (Del Moral et al. 2006, Cérou et al. 2012) and Generalized Splitting (Botev & Kroese 2012). These papers provide theoretical results on the asymptotic mean and variance as well as the asymptotic normality of the probability estimate.

The efficiency and accuracy of Subset Simulation depends on the ability of the applied MCMC algorithm to estimate accurately the conditional probabilities with a minimum number of samples. The variances of the estimates of the conditional probabilities depend on the correlation of the Markov chains simulated by the MCMC algorithm. A low correlation of the MCMC samples implies a small variance of the respective probability estimate. Originally, Au & Beck (2001) proposed a modified version of the Metropolis-Hastings (M-H) sampler that is based on a component-wise sample generation to avoid the small acceptance rate of the original M-H sampler in high dimensions. Recently, a number of MCMC algorithms that aim at a further improvement of the acceptance rate of the component-wise M-H have been proposed (Miao & Ghosn 2011, Santoso et al. 2011, Zuev & Katafygiotis 2011, Zuev et al. 2012). This paper reviews these algorithms and proposes a new approach for MCMC in the **U**-space. In addition, an adaptive variant of the new method is proposed, which enhances the performance of Subset Simulation without the need for additional model evaluations.

The structure of this paper is as follows. In Section 2, Subset Simulation is described. Estimation of the conditional probabilities with MCMC sampling is discussed in Section 3. Section 3.1 gives the principles of MCMC for Subset Simulation; Section 3.2 reviews existing MCMC algorithms proposed for Subset Simulation; Section 3.3 discusses the proposed algorithm and Section 3.4 presents its adaptive variant. Section 4 is devoted to

numerical evaluations of the performance of the methods. The paper closes with the conclusions in Section 5.

2. Subset Simulation

Subset Simulation is an adaptive Monte Carlo method proposed by Au & Beck (2001) for the estimation of small failure probabilities in high dimensional problems. Let F denote the failure event or any rare event of interest, defined as $F = \{\mathbf{u} \in \mathbb{R}^n : G(\mathbf{u}) \leq 0\}$ in the **U**-space. The idea behind Subset Simulation is to express the event F as the intersection of M intermediate events:

$$F = \bigcap_{j=1}^{M} F_j \tag{3}$$

The intermediate events are nested, i.e. $F_1 \supset F_2 \supset \cdots \supset F_M$, and $F_M = F$ is the failure event. The probability of failure is estimated as a product of conditional probabilities:

$$P_f = \Pr(F) = \Pr\left(\bigcap_{j=1}^{M} F_j\right) = \prod_{j=1}^{M} \Pr(F_j | F_{j-1}) \tag{4}$$

where F_0 is the certain event. The intermediate failure events are selected such that the conditional probabilities $\Pr(F_j|F_{j-1})$ are large. In this way, the original problem of evaluating a small failure probability reduces to a sequence of M intermediate problems that correspond to the estimation of larger conditional probabilities.

The probability $\Pr(F_1|F_0) = \Pr(F_1)$ is computed by application of crude Monte Carlo through simulating independent and identically distributed (i.i.d.) samples from $\varphi_n(\mathbf{u})$. For estimating the probabilities $\{\Pr(F_j|F_{j-1}) : j = 2, \dots, M\}$, one needs to generate samples from the conditional PDFs $\{\varphi_n(\mathbf{u}|F_{j-1}) : j = 2, \dots, M\}$, where:

$$\varphi_n(\mathbf{u}|F_{j-1}) = \frac{\varphi_n(\mathbf{u}) I_{F_{j-1}}(\mathbf{u})}{\Pr(F_{j-1})} \tag{5}$$

wherein $I_{F_{j-1}}(\mathbf{u})$ is the indicator function of F_{j-1}. Generation of i.i.d. samples from $\varphi_n(\mathbf{u}|F_{j-1})$ can be achieved by application of the acceptance-rejection method (Flury 1990). However, this approach is inefficient, especially as the event F_{j-1} approaches the

actual failure event, since the acceptance probability of the samples is proportional to $\Pr(F_{j-1})$. Alternatively, MCMC techniques can be applied for sampling $\varphi_n(\mathbf{u}|F_{j-1})$. MCMC methods produce samples of a target distribution, by constructing a Markov chain that has the target distribution as its stationary distribution (Tierney 1994). If the simulated Markov chain has reached its stationary state, the derived samples will be identically distributed according to $\varphi_n(\mathbf{u}|F_{j-1})$, but they will not be independent. In fact, as will be discussed in Section 2.1, the efficiency in estimating the conditional probabilities $\Pr(F_j|F_{j-1})$ depends on the correlation of the MCMC samples.

The intermediate failure events $\{F_j: j = 1, \dots, M-1\}$ are defined as follows. Because the failure event is $F = \{\mathbf{u} \in \mathbb{R}^n: G(\mathbf{u}) \leq 0\}$, the intermediate failure events F_j can be defined as $F_j = \{\mathbf{u} \in \mathbb{R}^n: G(\mathbf{u}) \leq c_j\}$, where $c_1 > c_2 \dots > c_M = 0$. The values of $\{c_j: j = 1, \dots, M-1\}$ can be chosen adaptively, so that the estimates of the conditional probabilities $\{\Pr(F_j|F_{j-1}): j = 1, \dots, M-1\}$ correspond to a chosen value p_0. This is achieved by successively sampling each conditional PDF $\varphi_n(\mathbf{u}|F_{j-1})$, producing N samples $\{\mathbf{u}_{j-1}^{(k)}: k = 1, \dots, N\}$ of $\varphi_n(\mathbf{u}|F_{j-1})$, and setting c_j equal to the p_0-percentile of the samples $\{G(\mathbf{u}_{j-1}^{(k)}): k = 1, \dots, N\}$. The samples of $\varphi_n(\mathbf{u}|F_{j-1})$ are generated with MCMC, using as seeds the $N_s = p_0 N$ samples $\{\mathbf{u}_{j-1}^{(k)}: k = 1, \dots, N_s\}$ that fell in F_j at subset level $j-1$. This procedure is repeated until the p_0-percentile becomes negative. At this level, the actual failure event $F_M = F$ is reached, for which $c_M = 0$. One can then obtain an estimate of the failure probability as:

$$P_f \approx \hat{P}_f = p_0^{M-1} \hat{P}_M \tag{6}$$

$\hat{P}_M$ is the estimate of the conditional probability $\Pr(F_M|F_{M-1})$ and is given by:

$$\hat{P}_M = \frac{1}{N} \sum_{k=1}^{N} I_F\left(\mathbf{u}_{M-1}^{(k)}\right) \tag{7}$$

where $\{\mathbf{u}_{M-1}^{(k)}: k = 1, \dots, N\}$ are samples from $\varphi_n(\mathbf{u}|F_{M-1})$. The value p_0 of the intermediate probabilities and the number of samples N per subset level are chosen by the analyst. Au & Beck (2001) suggested to use $p_0 = 0.1$, while Zuev et al. (2012) showed

that a choice of $p_0 \in [0.1,0.3]$ leads to similar efficiency. N should be selected large enough to give an accurate estimate of p_0. Moreover, it is assumed that p_0 and N are chosen such that $p_0 N$ and $1/p_0$ are positive integer numbers. The Subset Simulation algorithm can be summarized as follows.

Subset Simulation algorithm (Au & Beck 2001)

Define: N (number of samples in each intermediate step), p_0 (probability of intermediate subsets).

Initial run

1. Generate N i.i.d. samples $\{\mathbf{u}_0^{(k)}: k = 1, \dots, N\}$ from $\varphi_n(\mathbf{u})$
2. Order the samples $\{\mathbf{u}_0^{(k)}: k = 1, \dots, N\}$ in increasing order of magnitude of their limit-state values $\{G(\mathbf{u}_0^{(k)}): k = 1, \dots, N\}$. Find c_1 as the p_0-percentile of the samples $\{G(\mathbf{u}_0^{(k)}): k = 1, \dots, N\}$. Set $F_1 = \{\mathbf{u} \in \mathbb{R}^n: G(\mathbf{u}) \leq c_1\}$
3. $j = 1$

Iterations

4. Repeat while $c_j > 0$
    a. Generate N samples $\{\mathbf{u}_j^{(k)}: k = 1, \dots, N\}$ from $\varphi_n(\mathbf{u}|F_j)$ starting from the N_S samples $\{\mathbf{u}_{j-1}^{(k)}: k = 1, \dots, N_s\}$ for which $\mathbf{u}_{j-1}^{(k)} \in F_j$, where $N_s = p_0 N$:
    Repeat for $k = 1, \dots, N_s$
        Starting from $\mathbf{u}_j^{((k-1)/p_0+1)} = \mathbf{u}_{j-1}^{(k)}$, generate $1/p_0 - 1$ states $\{\mathbf{u}_j^{((k-1)/p_0+t)}: t = 2, \dots, 1/p_0\}$ of a Markov chain with stationary PDF $\varphi_n(\mathbf{u}|F_j)$ applying MCMC sampling
    b. Set $F_{j+1} = \{\mathbf{u} \in \mathbb{R}^n: G(\mathbf{u}) \leq c_{j+1}\}$, wherein c_{j+1} is the p_0-percentile of the samples $\{G(\mathbf{u}_j^{(k)}): k = 1, \dots, N\}$
    c. $j = j + 1$

Estimation of the probability of failure

5. Identify the number N_f of samples $\{\mathbf{u}_{j-1}^{(k)}: k = 1, \dots, N_f\}$ for which $\mathbf{u}_{j-1}^{(k)} \in F$
6. $\hat{P}_f = p_0^{j-1} \frac{N_f}{N}$

2.1 Statistics of the Subset Simulation estimator

The estimator $\hat{P}_f$ in Eq. (6) is biased for a finite N with bias of order $O(N^{-1})$, due to the correlation between the estimates of the conditional probabilities (Au & Beck 2001). The correlation between the estimates is attributed to the fact that the samples conditional on F_{j-1} at subset $j-1$ are used as seeds for the MCMC sampling at subset j. Cérou et al. (2012) showed that a bias of $O(N^{-1})$ is present even in the case where the conditional probabilities are estimated with i.i.d. samples, due to the adaptive estimation of the intermediate failure events. However, the bias is negligible compared to the coefficient of variation of the probability estimate, in both cases.

The coefficient of variation δ_1 of the Monte Carlo estimate $\hat{P}_1$ of the probability $\Pr(F_1)$ is given by the following well-known expression:

$$\delta_1 = \sqrt{\frac{1-P_1}{NP_1}} \tag{8}$$

The coefficients of variation δ_j of the estimates $\hat{P}_j$ of the conditional probabilities $\Pr(F_j|F_{j-1})$, $j = 2, \dots, M$, are obtained as follows (Au & Beck 2001):

$$\delta_j = \sqrt{\frac{1-P_j}{NP_j}\left(1+\gamma_j\right)} \tag{9}$$

where

$$\gamma_j = 2\sum_{k=1}^{N/N_s-1}\left(1-\frac{kN_s}{N}\right)\rho_j(k) \tag{10}$$

$N_s = p_0 N$ is the number of seeds of the MCMC sampling at subset level j, $N/N_s = 1/p_0$ is the length of each chain and $\rho_j(k)$ is the average k-lag auto-correlation coefficient of the stationary sequences $\{I_{F_j}\left(\mathbf{u}_{j-1}^{((l-1)/p_0+t)}\right): t = 1, \dots, N/N_f\}$, $l = 1, \dots, N_s$. $\rho_j(k)$ can be estimated from the samples (Au & Beck 2001). The estimator of Eq. (9) assumes that the different chains are uncorrelated through the indicator function, i.e. possible

dependence between the different seeds is neglected. Comparing Eq. (8) with Eq. (9), one can observe that the coefficient of variation of the estimates of the conditional probabilities is larger than the one of the Monte Carlo estimate with i.i.d. samples. This is due to the correlation of the Markov chain samples, expressed by the factor $\gamma_j > 0$. The efficiency of Subset Simulation decreases as γ_j increases and hence the chain correlation increases.

Applying the first-order Taylor series expansion of Eq. (6), one can derive the following first-order estimate of the square of the coefficient of variation of $\hat{P}_f$:

$$\delta_{\hat{P}_f}^2 \approx \sum_{i=1}^{M}\sum_{j=1}^{M} \delta_i \delta_j \rho_{ij} \tag{11}$$

where ρ_{ij} is the correlation between the estimates $\hat{P}_i$ and $\hat{P}_j$. Further assuming independence of the estimates of the conditional probabilities, we obtain:

$$\delta_{\hat{P}_f}^2 \approx \sum_{j=1}^{M} \delta_j^2 \tag{12}$$

The above is reported to provide an adequate estimate of the coefficient of variation of $\hat{P}_f$ in several applications of Subset Simulation (Au & Beck 2001, Au & Beck 2003b). However, it should be noted that in most cases Eq. (12) tends to underestimate the true coefficient of variation of $\hat{P}_f$.

3. MCMC for Subset Simulation

3.1 Principles of MCMC

As discussed in Section 2, Subset Simulation applies MCMC sampling to sample from each conditional PDF $\varphi_n(\mathbf{u}|F_j)$. Here we summarize the basic principle of MCMC for sampling from $\varphi_n(\mathbf{u}|F_j)$, starting with a brief introduction of the basic theory of Markov chains. For a more detailed discussion, the reader is referred to (Tierney 1994, Besag et al. 1995).

Let us define a stationary discrete-time vector random process $\{\mathbf{U}_t, t \in \mathbb{N}\}$ with marginal PDF $\varphi_n(\mathbf{u}|F_j)$, which possesses the Markov property:

$$\Pr\left(\mathbf{U}_{n+1} \in A | \bigcap\nolimits_{t \le n} \mathbf{U}_t = \mathbf{u}_t\right) = \Pr(\mathbf{U}_{n+1} \in A | \mathbf{U}_n = \mathbf{u}_n) \tag{13}$$

where A is any event in the outcome space of $\mathbf{U}_t$. That is, the conditional distribution of $\mathbf{U}_{n+1}$ given $\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_n$ depends only on $\mathbf{U}_n$. The process $\{\mathbf{U}_t, t \in \mathbb{N}\}$ is called a stationary Markov chain. The conditional density $p(\mathbf{v}|\mathbf{u})$ that defines the transition between two subsequent states $\mathbf{U}_n$ and $\mathbf{U}_{n+1}$ is called transition PDF. Since the process is stationary, the transition PDF must also be stationary, i.e. it does not depend on t. The joint PDF of a stationary Markov chain can be completely defined by its marginal distribution and its stationary transition distribution. The transition PDF $p(\mathbf{v}|\mathbf{u})$ satisfies:

$$\varphi_n(\mathbf{v}|F_j) = \int_{\mathbf{u} \in \mathbb{R}^n} p(\mathbf{v}|\mathbf{u}) \varphi_n(\mathbf{u}|F_j)\, d\mathbf{u} \tag{14}$$

Equation (14) ensures that $\varphi_n(\mathbf{u}|F_j)$ is the stationary (or invariant) distribution of the Markov process $\mathbf{U}_t$. It is easy to see that Eq. (14) will always be satisfied if the following so-called reversibility condition is fulfilled:

$$p(\mathbf{v}|\mathbf{u}) \varphi_n(\mathbf{u}|F_j) = p(\mathbf{u}|\mathbf{v}) \varphi_n(\mathbf{v}|F_j) \tag{15}$$

MCMC methods produce samples of a distribution by simulating states of a stationary Markov process whose marginal distribution is the desired distribution. This can be achieved by simulating every new state of the process from a transition PDF $p(\mathbf{v}|\mathbf{u})$ that satisfies the reversibility condition of Eq. (15). Starting from a state that may or may not be distributed according to the target distribution, the Markov chain will asymptotically converge to the target (stationary) distribution, provided that the chain is aperiodic and irreducible (e.g. see Tierney 1994, Rubinstein & Kroese 2007). It is relatively easy to ensure satisfaction of the latter conditions for most MCMC algorithms; one needs to choose a transition PDF which assigns non-zero probability of remaining at the same state (aperiodicity) and non-zero probability of entering any set in the state space in a finite number of steps (irreducibility). The transient period that is required until the Markov chain approximately reaches its stationary state is termed burn-in period.

Moreover, the generated samples will be correlated according to the correlation of the Markov process which will depend on the particular choice of $p(\mathbf{v}|\mathbf{u})$.

In the context of Subset Simulation, MCMC sampling is applied at subset $j+1$ to sample $\varphi_n(\cdot\,|F_j)$ through simulating states of Markov chains using as starting points (or 'seeds') the samples $\{\mathbf{u}_j^{(k)}: k = 1, \dots, N_s\}$ that fell in F_j at subset j. Since all seeds $\mathbf{u}_j^{(k)}$ are distributed according to $\varphi_n(\cdot\,|F_j)$, the chains have already reached their stationary states at the beginning and no burn-in period is necessary. All states of the Markov chains will be distributed according to the target distribution $\varphi_n(\cdot\,|F_j)$. This property of MCMC sampling in the context of Subset Simulation is termed perfect sampling and has been discussed in (Au et al. 2011, Zuev et al. 2012).

3.1.1 Metropolis-Hastings algorithm

The Metropolis-Hastings (M-H) algorithm (Metropolis et al. 1953, Hastings 1970) is the most widely used MCMC method for sampling from distributions that are difficult to sample from directly. The M-H algorithm for sampling from $\varphi_n(\mathbf{u}|F_j)$ uses a transition PDF $p(\mathbf{v}|\mathbf{u})$ defined as follows:

$$p(\mathbf{v}|\mathbf{u}) = a(\mathbf{u},\mathbf{v})q(\mathbf{v}|\mathbf{u}) + \big(1 - r(\mathbf{u})\big)\delta_{\mathbf{u}}(\mathbf{v}) \tag{16}$$

where $q(\mathbf{u}|\mathbf{v})$ is called proposal PDF, $\delta_{\mathbf{u}}(\mathbf{v})$ is the Dirac mass at $\mathbf{u}$, $a(\mathbf{u},\mathbf{v})$ is defined as:

$$a(\mathbf{u},\mathbf{v}) = \min\left\{1, \frac{\varphi_n(\mathbf{v}|F_j)\; q(\mathbf{u}|\mathbf{v})}{\varphi_n(\mathbf{u}|F_j)\; q(\mathbf{v}|\mathbf{u})}\right\} \tag{17}$$

and

$$r(\mathbf{u}) = \int_{\mathbf{v}\in\mathbb{R}^n} a(\mathbf{u},\mathbf{v})q(\mathbf{v}|\mathbf{u})\, d\mathbf{v}. \tag{18}$$

To generate a sample of the new state $\mathbf{U}_{n+1}$ conditional on the current state $\mathbf{U}_n = \mathbf{u}$, a candidate state $\mathbf{v}$ is generated from the proposal PDF $q(\cdot\,|\mathbf{u})$. The candidate is accepted with probability $a(\mathbf{u},\mathbf{v})$ and the chain moves to $\mathbf{U}_{n+1} = \mathbf{v}$. Otherwise, the candidate state is rejected and the chain remains at $\mathbf{U}_{n+1} = \mathbf{u}$. According to Eq. (16), the probability that the Markov chain remains in its current state is $1 - r(\mathbf{u})$.

It can be shown that the transition PDF of Eq. (16) satisfies the reversibility condition independent of the choice of the proposal PDF (Hastings 1970, Tierney 1994). If the proposal PDF has the symmetry property, i.e. $q(\mathbf{v}|\mathbf{u}) = q(\mathbf{u}|\mathbf{v})$, the algorithm reduces to the original Metropolis sampler (Metropolis et al. 1953). Inserting Eq. (5) into Eq. (17), and noting that $I_{F_j}(\mathbf{u}) = 1$, since the current state of the chain already follows the target distribution, one obtains:

$$\begin{aligned} a(\mathbf{u},\mathbf{v}) &= \min\left\{1, \frac{\varphi_n(\mathbf{v})\, q(\mathbf{u}|\mathbf{v})}{\varphi_n(\mathbf{u})\, q(\mathbf{v}|\mathbf{u})} I_{F_j}(\mathbf{v})\right\} \\ &= \min\left\{1, \frac{\varphi_n(\mathbf{v})\, q(\mathbf{u}|\mathbf{v})}{\varphi_n(\mathbf{u})\, q(\mathbf{v}|\mathbf{u})}\right\} I_{F_j}(\mathbf{v}) = \tilde{a}(\mathbf{u},\mathbf{v}) I_{F_j}(\mathbf{v}) \end{aligned} \tag{19}$$

The above shows that the acceptance probability of the M-H algorithm for sampling from $\varphi_n(\mathbf{u}|F_j)$ in the context of Subset Simulation can be expressed as a product of the acceptance probability for sampling from $\varphi_n(\mathbf{u})$ and the indicator function of F_j. Therefore, the M-H algorithm can be applied in two steps; first, a sample of $\varphi_n(\cdot)$ is generated by application of the M-H sampler with proposal PDF $q(\cdot\,|\mathbf{u})$ and acceptance probability $\tilde{a}(\mathbf{u},\mathbf{v})$; second, the sample is accepted if it lies in F_j, otherwise the chain remains in its current state. Let the current state of the Markov chain be $\mathbf{u}_0$. The transition from the state $\mathbf{u}_0$ to the next state $\mathbf{u}_1$ of the M-H algorithm for sampling from $\varphi_n(\mathbf{u}|F_j)$ is as follows:

M-H algorithm for sampling from $\varphi_n(\mathbf{u}|F_j)$

1. Generate candidate sample $\mathbf{v}$ from $\varphi_n(\cdot)$
   a. Generate a pre-candidate $\boldsymbol{\xi}$ by sampling from the PDF $q(\cdot\,|\mathbf{u}_0)$
   b. Accept or reject $\boldsymbol{\xi}$

$$\mathbf{v} = \begin{cases} \boldsymbol{\xi}, & \text{with prob. } \tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) \\ \mathbf{u}_0, & \text{with prob. } 1 - \tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) \end{cases}$$

where

$$\tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) = \min\left\{1, \frac{\varphi_n(\boldsymbol{\xi})\, q(\mathbf{u}_0|\boldsymbol{\xi})}{\varphi_n(\mathbf{u}_0)\, q(\boldsymbol{\xi}|\mathbf{u}_0)}\right\}$$

2. Accept or reject $\mathbf{v}$

$$\mathbf{u}_1 = \begin{cases} \mathbf{v}, & \mathbf{v} \in F_j \\ \mathbf{u}_0, & \mathbf{v} \notin F_j \end{cases}$$

3.1.2 Acceptance rate of the M-H in high dimensions

As discussed in (Au & Beck 2001, Schuëller et al. 2004, Katafygiotis & Zuev 2008), the M-H algorithm becomes inefficient for high dimensional problems. This is due to the fact that the probability that the pre-candidate is rejected in step 1 increases rapidly with increasing number of random variables n. This will lead to many repeated samples and hence to an increased correlation of the Markov chain. To illustrate this, consider the case where a pre-candidate state $\boldsymbol{\xi}$ is generated by a random walk with proposal PDF $q(\boldsymbol{\xi}|\mathbf{u}_0) = \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0) = \prod_{i=1}^{n} \varphi(\xi_i - u_{0i})$. That is, the proposal PDF is chosen as the n-dimensional independent standard normal PDF centered at the current state $\mathbf{u}_0$. We can then compute the mean acceptance rate of the pre-candidate as follows:

$$\begin{aligned} \tilde{r}(\mathbf{u}_0) &= \mathrm{E}_{\boldsymbol{\xi}}[\tilde{a}(\mathbf{u}_0, \boldsymbol{\xi})] = \int_{\mathbb{R}^n} \min\left\{1, \frac{\varphi_n(\boldsymbol{\xi})}{\varphi_n(\mathbf{u}_0)}\right\} \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0) d\boldsymbol{\xi} \\ &= \int_{|\boldsymbol{\xi}| \le |\mathbf{u}_0|} \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0) d\boldsymbol{\xi} + \int_{|\boldsymbol{\xi}| > |\mathbf{u}_0|} \frac{\varphi_n(\boldsymbol{\xi})}{\varphi_n(\mathbf{u}_0)} \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0) d\boldsymbol{\xi} \\ &= \int_{|\boldsymbol{\xi}| \le |\mathbf{u}_0|} \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0) d\boldsymbol{\xi} + \frac{1}{2^{\frac{n}{2}}} \exp\left(\frac{|\mathbf{u}_0|^2}{4}\right) \int_{|\boldsymbol{\xi}| > |\mathbf{u}_0|} \varphi_n\left(\boldsymbol{\xi} - \frac{1}{2}\mathbf{u}_0; \frac{1}{2}\mathbf{I}\right) d\boldsymbol{\xi} \\ &= P(|\mathbf{u}_0|^2; n; |\mathbf{u}_0|^2) + \frac{1}{2^{\frac{n}{2}}} \exp\left(\frac{|\mathbf{u}_0|^2}{4}\right) \left[1 - P\left(2|\mathbf{u}_0|^2; n; \frac{|\mathbf{u}_0|^2}{2}\right)\right] \end{aligned} \tag{20}$$

where $P(x; n; \lambda)$ is the CDF of the noncentral chi-squared distribution with n degrees of freedom and noncentrality parameter λ. $P(x; n; \lambda)$ can be evaluated through the following expression:

$$P(x; n; \lambda) = \frac{e^{-(x+\lambda)/2} x^{n/2-1}}{2^{n/2}} \sum_{j=0}^{\infty} \frac{(\lambda x)^j}{2^{2j} j!\, \Gamma\left(j + \frac{n}{2}\right)} \tag{21}$$

In Figure 1, the mean acceptance rate of the pre-candidate state of the M-H algorithm is plotted against the dimension of the random variable space n for different values of the distance of the current state from the origin $|\mathbf{u}_0|$. The figure illustrates the fast decay of the acceptance rate with increase of n, implying a fast increase of the probability of repeated candidates.

The performance of the M-H algorithm in sampling from high dimensional distributions can be improved through application of an adaptive proposal PDF that is continuously modified to match a target acceptance rate (Andrieu & Thoms 2008). However, for most M-H variants the optimal acceptance rate in high dimensions is remarkably low. For example, in the random walk M-H with proposal PDF $q(\boldsymbol{\xi}|\mathbf{u}_0) = \varphi_n(\boldsymbol{\xi} - \mathbf{u}_0; \sigma^2\mathbf{I})$, the optimal parameter σ for sampling from a multivariate distribution with i.i.d. components is found for a mean acceptance rate of 0.234 in the limit case where $n \to \infty$ (Roberts et al. 1997). This result is approximately valid for dimensions as low as 6 (Gelman et al. 1996). For dimension $n = 1$, the optimal acceptance rate of the random walk M-H is 0.44. This result will become relevant later.

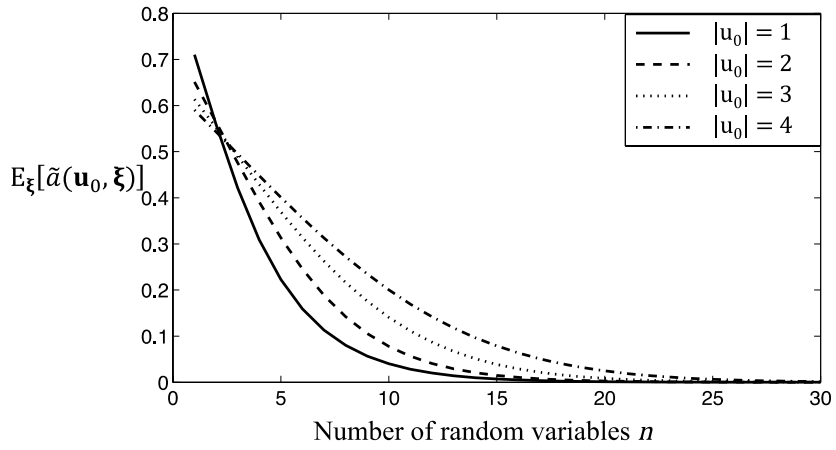


Figure 1: Mean acceptance rate of the original M-H algorithm applied to sampling from the independent standard normal distribution, as a function of the number of random variables n.

3.1.3 Measures of efficiency of the M-H

To further understand the performance of the M-H algorithm in high dimensions, it is useful to discuss criteria by which optimality can be measured. Optimality is usually defined in terms of some diffusion velocity measure that expresses how fast the Markov chain converges to ergodic moment estimates (Roberts et al. 1997). Consider for example the sample mean $\overline{U}$ of a one-dimensional Markov chain U_t. In (Gelman et al. 1996), efficiency is measured by the factor with which the asymptotic variance of $\overline{U}$ needs to be multiplied to obtain the same variance as with independent sampling. This measure is an increasing function of the reciprocal of the integral of the autocorrelation function of U_t, implying that maximizing the efficiency is equivalent to minimizing the correlation of the chain. When sampling from a distribution with n i.i.d. components, the same measure can be applied to an arbitrary component of the Markov chain. This efficiency measure has its asymptotic optimal value at $0.331/n$ in the independent random walk M-H (Gelman et al. 1996), indicating that the efficiency of the algorithm decreases fast with increasing n.

In the context of Subset Simulation, the M-H algorithm is applied to estimate each conditional probability $\Pr(F_j|F_{j-1})$, $j = 2, \ldots, M$. With this in mind, a measure of efficiency of the M-H can be defined as the factor by which the variance of the estimate $\hat{P}_j$ of $\Pr(F_j|F_{j-1})$ needs to be multiplied to obtain the same variance under independent sampling. Using the asymptotic estimate of the variance of $\hat{P}_j$ discussed in Section 2.1, we define the following efficiency measure:

$$eff_\gamma = \left(1 + \gamma_j\right)^{-1} \tag{22}$$

where γ_j is given in Eq. (10). The factors γ_j were also used in (Zuev et al. 2012) to obtain optimal acceptance rates of the component-wise M-H, as will be discussed in Section 3.4.

3.2 Existing approaches based on the M-H sampler

This section discusses approaches proposed for MCMC sampling for estimation of the conditional probabilities for Subset Simulation based on the M-H algorithm. The methods discussed here aim at maintaining the acceptance rate of the pre-candidate state of the

chain generated during the M-H transition at acceptable levels even for high dimensional problems.

3.2.1 Component-wise M-H

The component-wise or modified M-H algorithm was proposed by Au & Beck (2001) for sampling from high dimensional conditional distributions in the context of Subset Simulation. The method differs from the original M-H algorithm in the generation of the candidate state. That is, instead of using a n-dimensional proposal PDF, each coordinate ξ_i of the pre-candidate $\boldsymbol{\xi}$ is generated from a one-dimensional proposal PDF $q_i(\cdot\,|u_{0i})$ that depends on the i-th coordinate u_{0i} of the current state. This step is equivalent to the single component M-H algorithm or M-H within Gibbs algorithm (Haario et al. 2005) for sampling from independent multivariate distributions. The method as described in (Au & Beck 2001) requires that the random variable space be independent, however independence is achieved by the transformation of the original random variable space to the **U**-space. It is noted however that the principle of the method can also be applied to multivariate conditional distributions for which a full probabilistic description is available without the need for a transformation to an independent space (Haario et al. 2005). The algorithm is summarized as follows:

Component-wise M-H algorithm for sampling from $\varphi_n(\mathbf{u}|F_j)$ (Au & Beck 2001)

1. Generate candidate sample $\mathbf{v} = (v_1, \cdots, v_n)$ from $\varphi_n(\cdot)$. For each $i = 1, \dots, n$
   a. Generate a pre-candidate ξ_i by sampling from the PDF $q_i(\cdot\,|u_{0i})$
   b. Accept or reject ξ_i

$$v_i = \begin{cases} \xi_i, & \text{with prob. } \tilde{a}_i(u_{0i}, \xi_i) \\ u_{0i}, & \text{with prob. } 1 - \tilde{a}_i(u_{0i}, \xi_i) \end{cases}$$

   where

$$\tilde{a}_i(u_{0i}, \xi_i) = \min\left\{1, \frac{\varphi(\xi_i)\, q_i(u_{0i}|\xi_i)}{\varphi(u_{0i})\, q_i(\xi_i|u_{0i})}\right\}$$

2. Accept or reject **v**

$$\mathbf{u}_1 = \begin{cases} \mathbf{v}, & \mathbf{v} \in F_j \\ \mathbf{u}_0, & \mathbf{v} \notin F_j \end{cases}$$

Due to the independence of the random vector **U**, the component-wise M-H algorithm satisfies the reversibility condition independent of the choice of the one-dimensional proposal PDF – see (Au & Beck 2001, Zuev et al. 2012) for the corresponding proof. Au & Beck (2001) report that the performance of the algorithm is insensitive to the choice of the proposal distribution and suggest to choose $q_i(\cdot\,|u_{0i})$ as the uniform PDF centered at u_{0i} with width of 2. Alternative choices of the spread of the proposal PDF based on the statistics of the samples conditional on F_j are discussed in (Au et al. 2010, Miao & Ghosn 2011). Au et al. (2010) suggested accounting for the relative influence of each parameter by choosing the variance of the proposal PDF as the sample variances of the components of the seeds at each subset level. Miao & Ghosn (2011) proposed to further scale these sample variances taking as scaling factor 2.38^2, which results in the optimal variance of the proposal PDF of the M-H algorithm for target distributions with i.i.d. components (Gelman et al. 1996). However, as discussed in (Zuev et al. 2012) and in Section 3.4 of this paper, the optimal scaling factor for sampling the conditional normal distribution differs considerably from the scaling factor that optimizes the efficiency for sampling high dimensional target distributions.

It is noted that the probability of repeated candidates in each dimension of the component-wise M-H algorithm is non-zero and depends on the one-dimensional proposal PDF. However, since each component moves independently, the probability of having repeated candidates simultaneously in all components decreases geometrically with increasing number of random variables n. Hence, the method is suitable for application to high-dimensional problems.

3.2.2 M-H with repeated generation of pre-candidate states

A different approach for overcoming the low acceptance rate of the original M-H algorithm was proposed by Santoso et al. (2011). In this method, the candidate state is generated through a repeated generation of pre-candidate samples until acceptance of the

pre-candidate is achieved. Hence, the algorithm avoids the generation of repeated candidates by ensuring that the pre-candidate is always accepted, independent of the dimension of the random variable space. The resulting update of the Markov chain is as follows:

M-H algorithm with repeated generation for sampling from $\varphi_n(\mathbf{u}|F_j)$ (Santoso et al. 2011)

1. Generate candidate sample $\mathbf{v}$ from $\varphi_n(\cdot)$
    a. Generate a pre-candidate $\boldsymbol{\xi}$ by sampling from the PDF $q(\cdot\,|\mathbf{u}_0)$
    b. Accept or reject $\boldsymbol{\xi}$

$$\mathbf{v} = \begin{cases} \boldsymbol{\xi}, & \text{with prob. } \tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) \\ \mathbf{u}_0, & \text{with prob. } 1 - \tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) \end{cases}$$

where

$$\tilde{a}(\mathbf{u}_0, \boldsymbol{\xi}) = \min\left\{1, \frac{\varphi_n(\boldsymbol{\xi})\, q(\mathbf{u}_0|\boldsymbol{\xi})}{\varphi_n(\mathbf{u}_0)\, q(\boldsymbol{\xi}|\mathbf{u}_0)}\right\}$$

    c. If $\boldsymbol{\xi}$ is rejected go to a.
2. Accept or reject $\mathbf{v}$

$$\mathbf{u}_1 = \begin{cases} \mathbf{v}, & \mathbf{v} \in F_j \\ \mathbf{u}_0, & \mathbf{v} \notin F_j \end{cases}$$

This approach is inspired by the M-H with delayed rejection (Tierney & Mira 1999), however in the latter method the acceptance probability of the pre-candidate sample is updated in each re-generation in order to guarantee the satisfaction of the reversibility condition. The method of Santoso et al. (2011) does not allow for an analytical expression of the transition PDF, hence satisfaction of the reversibility condition cannot be verified analytically. Santoso et al. (2011) evaluated the transition PDF numerically for a one-dimensional truncated normal distribution using a uniform proposal PDF and showed that the reversibility condition is approximately satisfied. However, because of the approximate nature, the stationary distribution generated with this algorithm will differ from the target distribution. This is illustrated in Figure 2, where the CDF of the

one-dimensional truncated normal distribution with different normalizing constants is compared to the empirical CDF from 10^4 samples using a uniform proposal PDF with width of 2.

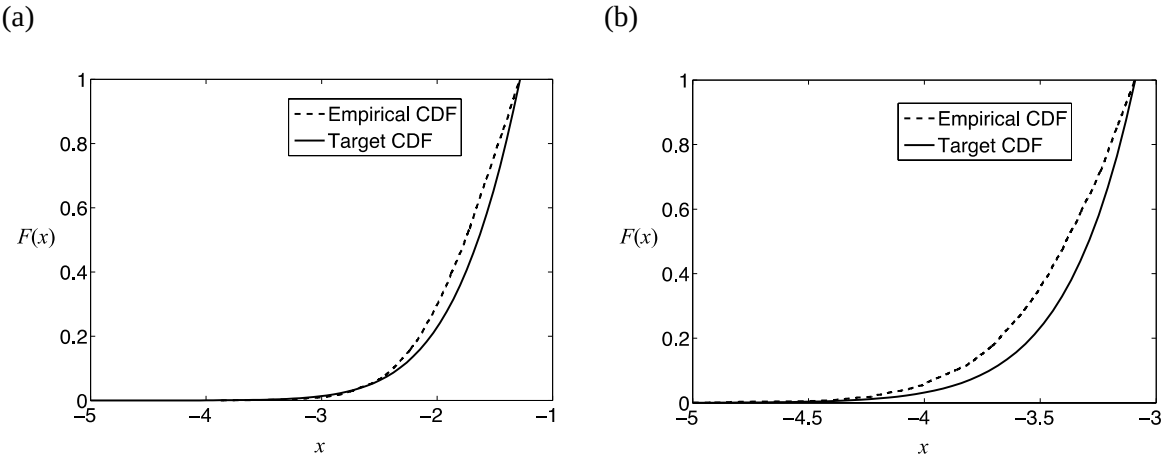


Figure 2: Empirical CDF of the M-H with repeated generation of pre-candidate states against target CDF of the Markov chain for the one-dimensional truncated normal distribution with probability normalizing constant (a) 10^{-1} and (b) 10^{-3}.

3.2.3 Component-wise M-H with delayed rejection of the candidate state

As discussed earlier, a repeated generation of pre-candidate states requires that the acceptance probability is adapted to account for the fact that earlier samples were rejected. This procedure is called delayed rejection and was developed by Tierney & Mira (1999) for application to Bayesian statistics. Miao & Ghosn (2011) applied this approach in combination with the component-wise M-H algorithm, leading to the following updating procedure with the two proposal PDFs q_{1i} and q_{2i}:

M-H algorithm with delayed rejection of the candidate for sampling from $\varphi_n(\mathbf{u}|F_j)$ (Miao & Ghosn 2011)

1. Generate candidate sample $\mathbf{v}$ from $\varphi_n(\cdot)$. For each $i = 1, \dots, n$
    a. Generate a pre-candidate ξ_{1i} by sampling from the PDF $q_{1i}(\cdot\,|u_{0i})$

b. Accept or reject ξ_{1i}

$$v_i = \begin{cases} \xi_{1i}, & \text{with prob. } \tilde{a}_{1i}(u_{0i}, \xi_{1i}) \\ u_{0i}, & \text{with prob. } 1 - \tilde{a}_{1i}(u_{0i}, \xi_{1i}) \end{cases}$$

where

$$\tilde{a}_{1i}(u_{0i}, \xi_{1i}) = \min\left\{1, \frac{\varphi(\xi_{1i})\, q_{1i}(u_{0i}|\xi_{1i})}{\varphi(u_{0i})\, q_{1i}(\xi_{1i}|u_{0i})}\right\}$$

c. If ξ_{1i} was rejected, generate ξ_{2i} by sampling from the PDF $q_{2i}(\cdot\,|u_{0i}, \xi_{1i}\,)$

d. Accept or reject ξ_{2i}

$$v_i = \begin{cases} \xi_{2i}, & \text{with prob. } \tilde{a}_{2i}(u_{0i}, \xi_{1i}, \xi_{2i}) \\ u_{0i}, & \text{with prob. } 1 - \tilde{a}_{2i}(u_{0i}, \xi_{1i}, \xi_{2i}) \end{cases}$$

where

$$\tilde{a}_{2i}(u_{0i}, \xi_{1i}, \xi_{2i}) = \min\left\{1, \frac{\varphi(\xi_{2i}) q_{1i}(\xi_{1i}|\xi_{2i}) q_{2i}(u_{0i}|\xi_{2i}, \xi_{1i}\,)[1 - \tilde{a}_{1i}(\xi_{2i}, \xi_{1i})]}{\varphi(u_{0i}) q_{1i}(\xi_{1i}|u_{0i}) q_{2i}(\xi_{2i}|u_{0i}, \xi_{1i}\,)[1 - \tilde{a}_{1i}(u_{0i}, \xi_{1i})]}\right\}$$

2. Accept or reject $\mathbf{v}$

$$\mathbf{u}_1 = \begin{cases} \mathbf{v}, & \mathbf{v} \in F_j \\ \mathbf{u}_0, & \mathbf{v} \notin F_j \end{cases}$$

The algorithm allows for the second proposal PDF to depend not only on the current state of the chain but also on the rejected pre-candidate. It can be shown that the method satisfies the reversibility condition independent of the choice of the two proposal PDFs (Tierney & Mira 1999). The method will reduce the chain correlation, since fewer repeated pre-candidates will occur, however its benefit over the component-wise M-H algorithm is limited to low-dimensional problems. For high-dimensional problems, the acceptance rate of the component-wise M-H algorithm is high already for the first pre-candidate.

In addition to the delayed rejection, the algorithm proposed in (Miao & Ghosn 2011) incorporates adaptive regeneration steps in conjunction with the component-wise M-H for the generation of the candidate state. Regeneration techniques (Mykland et al. 1996, Gilks et al. 1998) are based on restarting the Markov chain at random stopping times and have been developed to overcome the burn-in problem of M-H samplers as well as to provide means for applying independent variance estimates of quantities of interest. However, as pointed out in (Au et al. 2011) in response to the approach proposed in (Miao & Ghosn 2011), MCMC sampling in the context of Subset Simulation does not suffer a burn-in problem, since the seeds of each Markov chain follow the target distribution by construction. Therefore, the regeneration steps proposed in (Miao & Ghosn 2011) are not further discussed here.

In the algorithm described above, the concept of delayed rejection is applied to the generation of the candidate state, i.e. before the limit-state function is evaluated to check whether the sample lies in F_j. The same concept can also be applied at the second step of M-H algorithms. For example, assuming that a repeated sample is obtained through the original M-H algorithm, delayed rejection would yield a new candidate sample whose acceptance probability would be evaluated by appropriately adjusting Eq. (17) to keep the target distribution unchanged. Zuev & Katafygiotis (2011) combined this concept with the component-wise generation and proposed an algorithm for delayed rejection of the sample obtained after acceptance/rejection of the candidate state that is suitable for application to high-dimensional problems. In their approach, if the candidate state is rejected, the accepted components of the pre-candidate state are re-sampled from a different one-dimensional proposal PDF and accepted or rejected with a suitable acceptance probability. The method reduces the chain correlation and hence decreases the variance of probability estimate, however it requires additional limit-state function evaluations as compared to the approaches discussed in this paper and is not further examined here.

3.3 Conditional sampling in **U***-space*

In the previous section, existing algorithms were discussed that attempt to obtain high acceptance rates of the pre-candidate state of the M-H sampler. In this section, we introduce a new MCMC algorithm for sampling from $\varphi_n(\cdot\,|F_j)$, which yields candidates $\mathbf{v}$ that always differ from their current state.

In the first step of the M-H sampler for Subset Simulation, a candidate $\mathbf{v}$ is sampled from the independent joint Gaussian PDF $\varphi_n(\cdot)$, conditional on the previous sample $\mathbf{u}_0$. We impose that $\mathbf{v}$ and $\mathbf{u}_0$ are jointly Gaussian with component-wise cross-correlation coefficient ρ_i. That is, $\mathbf{v}$ and $\mathbf{u}_0$ are Gaussian with zero mean vectors, their covariance matrices are equal to the unit diagonal matrix $\mathbf{I}$ and their cross-covariance matrix $\mathbf{R}$ is a diagonal matrix with ith diagonal term equal to ρ_i. Hence, the conditional PDF of $\mathbf{v}$ given $\mathbf{u}_0$ will be the multivariate normal distribution with mean vector $\mathbf{R}\mathbf{u}_0$ and covariance matrix $\mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}}$, which is the n-dimensional independent normal distribution whose ith component has mean value $\rho_i u_{0i}$ and standard deviation $(1-\rho_i^2)^{1/2}$. It is possible to directly sample from this distribution, thus avoiding the generation of repeated candidates through rejection of pre-candidate states. This leads to the following updating scheme:

Conditional sampling for sampling from $\varphi_n(\mathbf{u}|F_j)$

1. Generate candidate sample $\mathbf{v} = (v_1, \cdots, v_n)$ from $\varphi_n(\cdot)$.

   For each $i = 1, \ldots, n$, generate v_i from the normal distribution with mean $\rho_i u_{0i}$ and standard deviation $\sqrt{1-\rho_i^2}$

2. Accept or reject $\mathbf{v}$

$$\mathbf{u}_1 = \begin{cases} \mathbf{v}, & \mathbf{v} \in F_j \\ \mathbf{u}_0, & \mathbf{v} \notin F_j \end{cases}$$

It is noted that the above algorithm assumes that each component v_i of the candidate state is independent of the components u_{0j} of the actual state for all $i \neq j$. The algorithm can be generalized for arbitrary cross-correlation matrices **R** that model possible dependence between any of the components of the actual and candidate states. In this case, the algorithm needs to be modified such that the candidate state **v** is generated from the n-dimensional distribution $\varphi_n(\mathbf{v} - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})$, which denotes the multivariate standard Gaussian distribution with argument $\mathbf{v} - \mathbf{R}\mathbf{u}_0$ and correlation matrix $\mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}}$. The transition PDF defining the transition from the state $\mathbf{u}_0$ to $\mathbf{u}_1$ by means of this algorithm satisfies the reversibility condition of Eq. (15). Therefore, the stationary distribution of the transition will be the conditional normal $\varphi_n(\cdot\,|F_j)$. A detailed proof of the above is given in Appendix A.

Since we eventually sample from the conditional normal distribution $\varphi_n(\cdot\,|F_j)$, a small correlation between the components of the actual and the candidate state does not imply a small correlation of the final samples. This is due to the fact that a small ρ_i will lead to many rejected samples in the second step, which will lead to a large correlation among the resulting samples. On the other hand, a ρ_i close to one will increase the acceptance rate but will lead to a larger correlation of the newly generated samples. Section 3.4 comments on the optimal choice of the parameters ρ_i.

Besides its simplicity, the advantage of this approach lies with the fact that the candidate state is always accepted, without compromising the stationary distribution of the chain. In fact, the algorithm can be understood as a special case of the M-H sampler with proposal distribution obtained by conditioning the joint normal distribution consisting of two independent standard normal random vectors with component-wise cross correlation coefficients ρ_i. Evaluating the acceptance probability $\tilde{\alpha}(\mathbf{u}_0, \mathbf{v})$ in Eq. (19), one sees that both the numerator and denominator in the fraction equal the joint PDF of $\mathbf{u}_0$ and **v**, and consequently $\tilde{\alpha}(\mathbf{u}_0, \mathbf{v})$ equals unity. Therefore, the algorithm can be perceived as an M-H sampler for sampling from $\varphi_n(\cdot\,|F_j)$ with acceptance probability of the pre-candidate equal to 1. It is noted that this proposal distribution has been used for sampling Gaussian process prior models (e.g. Neal 1998).

3.4 Adaptive MCMC with optimal scaling

The performance of the conditional sampling algorithm presented in Section 3.3 depends on the choice of the correlation parameters ρ_i between each component of the actual and the candidate state. Similarly, the performances of the M-H algorithms discussed in Section 3.2 depend on the choice of the variance of the proposal PDF. A large variance (resp. small ρ_i) will lead to many rejected candidates and a small variance (resp. large ρ_i) to a high correlation between states. The performance of the MCMC algorithms can be enhanced by adaptively adjusting the respective parameter during the simulation, employing intermediate results. One approach is to design an adaptive scheme that minimizes a measure of efficiency such as the one introduced in Eq. (22) (e.g. Pasarica & Gelman 2010). This approach has the disadvantage that the efficiency measure cannot always be computed reliably using only a subset of the MCMC samples. Alternatively, one can attempt suboptimal adaptation through a proxy that is computationally tractable. A proxy that is most often used in practice is the expected acceptance probability of the MCMC algorithm, see (Andrieu & Thoms 2008) and references therein. An advantage of this approach is that theoretical results on optimal acceptance probabilities exist for some classes of target distributions (Roberts et al. 1997, Roberts & Rosenthal 1998).

The acceptance rate of MCMC algorithms for sampling from $\varphi_n(\mathbf{u}|F_j)$ with acceptance rate of the pre-candidate equal to 1 depends only on whether or not the candidate state lies on the failure domain. The latter depends on the limit-state function value of the candidate state. That is, MCMC sampling from $\varphi_n(\mathbf{u}|F_j)$ with acceptance of the pre-candidate equal to 1 is equivalent to sampling a target one-dimensional conditional distribution, i.e. the limit-state function conditional on the domain F_j. It has been shown that the optimal acceptance probability of the M-H algorithm for sampling one-dimensional distributions is in some situations (e.g. normal target distributions) approximately 0.44 (Roberts et al. 1997, Roberts & Rosenthal 2001).

We investigate the effect of the parameter ρ_i of the conditional sampling method on the efficiency of the generated Markov chain. ρ_i is selected identically for all dimensions.

The performance of the method is assessed in terms of the efficiency measure eff_γ, defined in Eq. (22). Figure 3 plots eff_γ against the mean acceptance rate $\hat{a}$, computed as the average number of accepted samples divided by the length of each chain, for the limit-states of Example 1 (linear limit-state) and Example 2 (convex and concave limit-states) introduced later in Section 4. The number of random variables in all examples is chosen equal to 100. The curves are plotted for different subset levels j. The value of the intermediate conditional probability p_0 is set to 0.1 for both examples and the number of samples per level N is set to 1000. Based on the results from this numerical experiment, we conjecture that the result of (Roberts et al. 1997) is approximately optimal for the conditional sampling method for Subset Simulation, i.e. the optimal acceptance rate of the algorithm is close to 0.44. Moreover, this optimal acceptance rate agrees with the results presented in (Zuev et al. 2012), who varied the variance of the proposal PDF of the component-wise M-H algorithm and evaluated the resulting γ_j and the acceptance rate of the algorithm. As discussed earlier, minimizing the factors γ_j corresponds to maximizing the efficiency measure given in Eq. (22). Zuev et al. (2012) report that the factors γ_j are rather flat at the optimal acceptance rate, which lies between 0.3 and 0.5.

Based on the above results, we propose to adjust the parameters ρ_i of the conditional sampling method on the fly such that the acceptance probability remains close to the optimal value of 0.44. At subset $j+1$ of Subset Simulation, one needs to obtain samples of $\varphi_n(\cdot\,|F_j)$ through simulating N_s Markov chains using as seeds the samples $\{\mathbf{u}_j^{(k)}: k = 1, \ldots, N_s\}$ that fell in F_j at subset j. The idea of the adaptive procedure is to perform the simulation in steps. At each step, a fraction N_a of the N_s chains are simulated applying the conditional sampling algorithm with the same parameters ρ_i for all chains. For the simulation of the next N_a chains, the parameters ρ_i are adjusted based on the estimated acceptance probability of the previous N_a chains. Importantly, the seeds for the simulation of each N_a chains are chosen at random (without replacement) from the total N_s seeds, in order to impose a uniform velocity on average over the chain distribution. This is necessary to maintain the asymptotic unbiasedness of the Subset Simulation estimator.

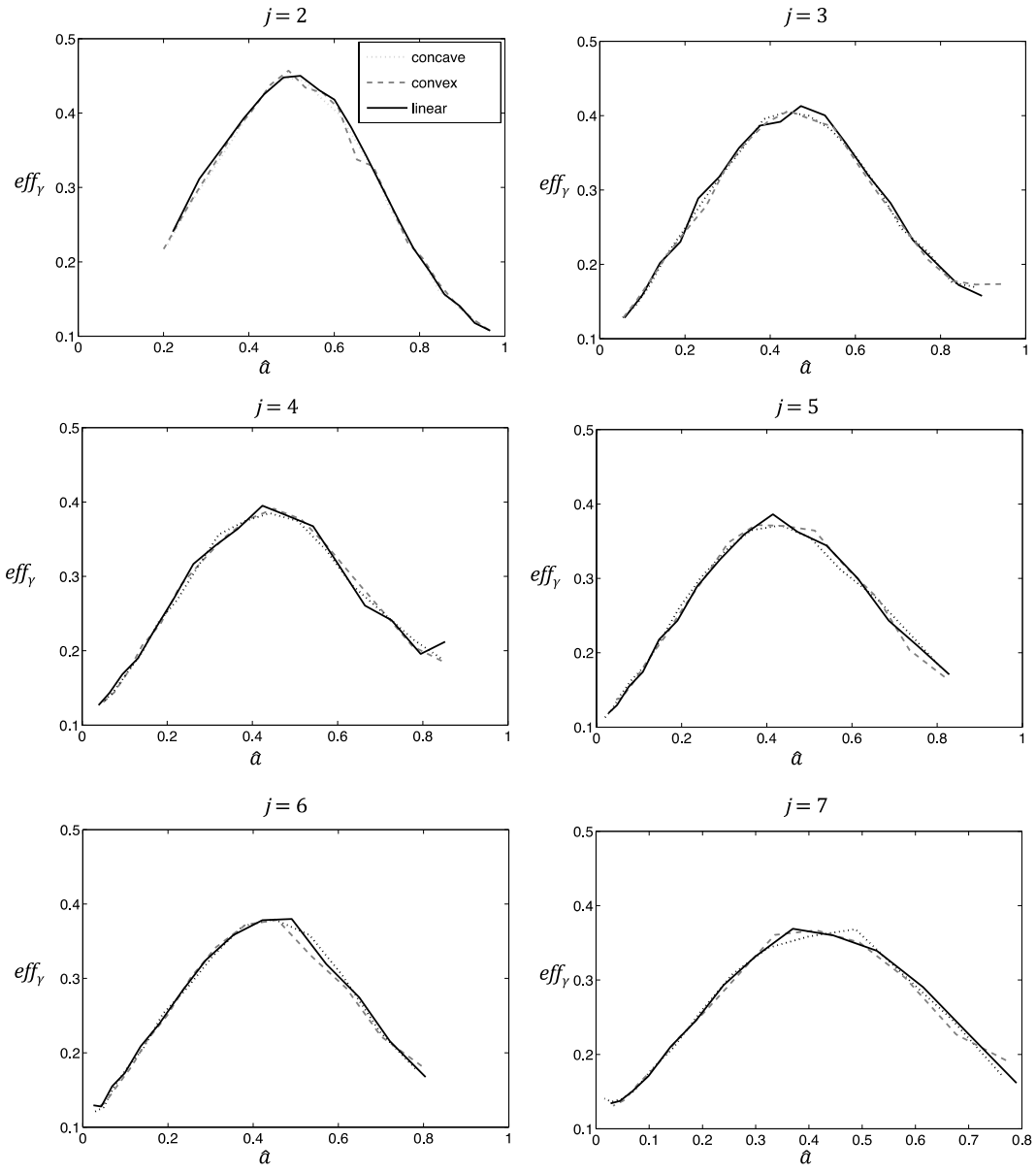


Figure 3: Chain efficiency eff_γ in terms of average acceptance rate $\hat{a}$ of the conditional sampling method, evaluated for the limit states of Example 1(linear), 2a (convex) and 2b (concave) with 100 random variables for subset levels $j = 2, \ldots, 7$

The adaptation is performed by selecting an appropriate standard deviation of the proposal distribution σ_i, which is related to the ρ_i parameter of the conditional sampling algorithm by $\sigma_i = \sqrt{1-\rho_i^2}$. It proceeds as follows: A set of starting values of the standard deviations of the proposal distribution $\sigma_{0i}, i = 1, \dots, n$, are chosen. Moreover, an initial scaling parameter $\lambda_1 \in (0,1)$ is selected. The number of chains N_a after which the proposal distribution will be adapted is selected such that N_s/N_a is a positive integer number. Adaptation is performed for $iter = 1, \dots, N_s/N_a$. At each adaptation step $iter$, the standard deviation of the proposal distribution of each component σ_i is computed through scaling the starting value σ_{0i} by λ_{iter}. However, each σ_i cannot be larger than the standard deviation of the corresponding random variable, which equals 1.0. Hence, σ_i is adapted at each step $iter$ through:

$$\sigma_i = \min(\lambda_{iter}\sigma_{0i}, 1.0) \tag{23}$$

Then N_a seeds are chosen at random from $\{\mathbf{u}_j^{(k)} : k = 1, \dots, N_s\}$ and the conditional sampling algorithm is applied to simulate each corresponding Markov chain with parameters ρ_i chosen as:

$$\rho_i = \sqrt{1-\sigma_i^2} \tag{24}$$

The average acceptance rate of the chains is then evaluated by:

$$\hat{a}_{iter} = \frac{1}{N_a}\sum_{k=1}^{N_a} \widehat{\mathrm{E}}_{\boldsymbol{\xi}}\left[a\left(\mathbf{u}_j^{(k)}\right)\right] \tag{25}$$

where $\widehat{\mathrm{E}}\left[a\left(\mathbf{u}_j^{(k)}\right)\right]$ is the average accepted samples of the chain with seed $\mathbf{u}_j^{(k)}$. The scaling parameter λ_{iter} is then updated using the following recursive relation (Haario et al. 2001):

$$\log \lambda_{iter+1} = \log \lambda_{iter} + \zeta_{iter}[\hat{a}_{iter} - a^*] \tag{26}$$

where a^* is the optimal acceptance rate chosen as 0.44. ζ_{iter} is a positive real number which ensures that the variation of λ_{iter} vanishes. A possible choice is $\zeta_{iter} = iter^{-1/2}$. Alternatively, ζ_{iter} can also be chosen adaptively (Andrieu & Thoms 2008). Eq. (26)

ensures that if the acceptance rate of the chain is smaller than 0.44 then the variance of each one-dimensional conditional normal distribution is decreased (resp. the correlation parameter ρ_i is increased) and if it is larger than 0.44 the variance is increased (resp. ρ_i is decreased).

We now discuss the choice of the starting values σ_{0i} of the standard deviations of the conditional sampling algorithm. One possible choice is a constant σ_{0i} for all dimensions, e.g. $\sigma_{0i} = 1.0$, $i = 1, \dots, n$, which results in the same correlation parameter ρ_i for all random variables. This choice is appropriate for problems with a large number of random variables with approximately identical influence on the limit-state function. Alternatively, the values σ_{0i} can be chosen as the sample variances of the components of the seeds at each subset level (Au et al. 2010). It is reminded that, at subset level $j + 1$, the Markov chains are simulated using as seeds the samples $\{\mathbf{u}_j^{(k)}: k = 1, \dots, N_s\}$ that fell in F_{j+1} at subset j. The sample mean $\hat{\mu}_i$ and sample standard deviation $\hat{\sigma}_i$ of each component i of the seeds are computed through:

$$\hat{\mu}_i = \frac{1}{N_s} \sum_{k=1}^{N_s} u_{ji}^{(k)} \tag{27}$$

and

$$\hat{\sigma}_i^2 = \frac{1}{N_s - 1} \sum_{k=1}^{N_s} \left(u_{ji}^{(k)} - \hat{\mu}_i \right)^2 \tag{28}$$

Random variables with small $\hat{\sigma}_i$ have large influence on the limit-state function at the current threshold level; conversely the influence of random variables with large $\hat{\sigma}_i$ is negligible. Choosing the starting values of the adaptive conditional sampling algorithm as $\sigma_{0i} = \hat{\sigma}_i$, $i = 1, \dots, n$, we ensure that variables with large influence will move locally with a high correlation parameter ρ_i, while variables with negligible influence will move with $\rho_i \approx 0$. This allows accounting for the relative influence of the random variables; at the same time, the adaptive scaling of Eq. (23) makes sure that the acceptance rate is kept close to the optimal value of 0.44. However, in cases where it is known that all random

variables have similar influence on the limit-state function, using the sample standard deviations as starting values of the adaptive algorithm might introduce noise that can undermine the performance of the algorithm. In such cases, a constant choice of the parameters in each dimension is preferable.

The adaptive conditional sampling algorithm is summarized as follows:

Adaptive conditional sampling for generating N samples from $\varphi_n(\mathbf{u}|F_j)$ *at subset step* $j+1$ *of the Subset Simulation*

Define: N_a (number of chains to consider for adaptation) and λ_1 (initial scaling parameter).

Initialize

1. Choose the starting values of the standard deviations σ_{0i}. Proceed with either step 1a. or 1b.
    a. Set $\sigma_{0i} = 1, i = 1, \dots, n$
    b. Compute the sample means $\hat{\mu}_i$ and sample standard deviations $\hat{\sigma}_i$ of each component i of the samples $\{\mathbf{u}_{j-1}^{(k)} : k = 1, \dots, N_s\}$ that fell in F_j at subset j, applying Eqs. (26) and (27). Set $\sigma_{0i} = \hat{\sigma}_i, i = 1, \dots, n$
2. Permute randomly the seeds $\{\mathbf{u}_{j-1}^{(k)} : k = 1, \dots, N_s\}$.

Iterations

3. Repeat for $iter = 1, \dots, N_s/N_a$
    a. Compute the correlation parameters ρ_i of the conditional sampling method through Eqs. (22) and (23).
    b. Repeat for $k = (iter-1)N_a + 1, \dots, iter \cdot N_a$

        Starting from $\mathbf{u}_j^{((k-1)/p_0+1)} = \mathbf{u}_{j-1}^{(k)}$, generate $1/p_0 - 1$ states $\{\mathbf{u}_j^{((k-1)/p_0+t)} : t = 2, \dots, 1/p_0\}$ of a Markov chain with stationary PDF $\varphi_n(\mathbf{u}|F_j)$ applying the conditional sampling algorithm with ρ_i as computed in a.
    c. Evaluate the average acceptance rate $\hat{a}_{iter}$ of the last N_a chains through Eq. (25).

d. Compute the new scaling parameter λ_{iter+1} through Eq. (26).

The parameter N_a defines the frequency of the updating of the scaling parameter. Experience has shown that a choice of $N_a = p_a N_s$ where $p_a \in [0.1,0.2]$, gives good results. The initial value of the scaling parameter $\lambda_1 = 0.6$ is a good choice for the first subset level. For each of the subsequent levels, the initial value of the scaling parameter should be chosen as the final value that was obtained in the previous level. It is noted that the random choice of the seeds (step 2 above) is an essential part of the algorithm, since it ensures that the obtained probability estimate remains asymptotically unbiased.

The adaptive conditional sampling algorithm can be further extended to account for the correlation of the samples at each intermediate failure domain. To this end, step 1b can be modified such that the covariance matrix $\boldsymbol{\Sigma}_0$ is set equal to the sample covariance matrix $\widehat{\boldsymbol{\Sigma}}$ of the samples that fell in F_j at subset j. Then, in step 3a the covariance of the proposal distribution is evaluated as $\boldsymbol{\Sigma} = \lambda_{iter}\boldsymbol{\Sigma}_0$, with λ_{iter} updated through Eq. (26). Samples from $\varphi_n(\mathbf{u}|F_j)$ can then be generated in step 3b through application of the conditional sampling algorithm with candidate state generated from $\varphi_n(\mathbf{v} - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})$. The cross-correlation matrix $\mathbf{R}$ needs to be evaluated such that the covariance matrix of the proposal density is equal to $\boldsymbol{\Sigma}$. It is shown in Appendix B that this requirement leads to:

$$\mathbf{R} = \sum_{i=1}^{n} \sqrt{\theta_i}\, \mathbf{c}_i \mathbf{c}_i^{\mathrm{T}} \tag{29}$$

where $\{\theta_i, \mathbf{c}_i\}$ are the eigenvalues and corresponding eigenvectors of the matrix $\mathbf{I} - \boldsymbol{\Sigma}$. It is only possible to apply Eq. (29) if the matrix $\mathbf{I} - \boldsymbol{\Sigma}$ is positive-semi definite. If this requirement does not hold, this matrix can be replaced with its "nearest" positive-semi definite matrix (e.g. Hayes & Hill 1981).

It is noted that this modification of the adaptive conditional sampling algorithm is mostly applicable to low- or moderate-dimensional problems, for which covariance information with a number of samples in the order of $p_0 N$ can be extracted.

4. Numerical evaluation

This section demonstrates the performance of the MCMC algorithms for Subset Simulation presented with four examples involving many random variables. In the first two examples, we investigate the performance with linear and nonlinear limit-states. The third example is a quadratic limit-state function wherein two random variables have a larger influence on the failure probability. The last example is a numerical limit-state function that depends on the solution of a diffusion equation in a one-dimensional spatial domain by application of the finite element method. The parameters of Subset Simulation are chosen as $N = 1000$ and $p_0 = 0.1$ for all examples.

4.1 Example 1

The first example consists of a limit-state function expressed as a linear function of independent standard normal random variables (Engelund & Rackwitz 1993):

$$g_1(\mathbf{U}) = -\frac{1}{\sqrt{n}}\sum_{i=1}^{n} U_i + \beta \tag{30}$$

where $\mathbf{U}$ is a n-dimensional standard normal random vector. The probability of failure for this limit-state function is $\Phi(-\beta)$ independent of the dimension n, where $\Phi(\cdot)$ is the standard normal CDF.

We use this example to examine the influence of the number of random variables on the performance of the MCMC algorithms. Figures 4 and 5 show the relative bias and coefficient of variation of the probability estimate as a function of the number of random variables obtained by Subset Simulation employing the MCMC algorithms reviewed in Section 3.2 as well as the conditional sampling algorithm presented in Section 3.3. The results are plotted for two different target failure probabilities, namely 10^{-2} and 10^{-4}. For the conditional sampling algorithm (CS), the correlation parameter was chosen as 0.8 for all components. For the component-wise M-H (CWM-H) the proposal distribution is chosen as the normal distribution centered at the current state with unit standard deviation. The same distribution is chosen for both proposal distributions for the

component-wise M-H with delayed rejection (CWM-HwDR). In both the CWM-H and CWM-HwDR, the same proposal distributions were chosen for all components. For the M-H with repeated generation (M-HwRG), the proposal distribution was chosen as the n-dimensional independent standard normal distribution centered at the current state. The statistics are computed from 500 independent simulation runs.

In Figure 4, it is shown that M-HwRG produces biased probability estimates and the bias tends to increase with increasing number of random variables and decreasing probability of failure. This is because the M-HwRG satisfies the reversibility condition only approximately and therefore the samples will not follow the target distribution, as was also demonstrated in Figure 2. On the other hand, the rest of the examined MCMC algorithms give essentially unbiased estimates, since they all satisfy the reversibility condition and hence the samples produced follow the target distribution of the Markov chains.

Figure 5 shows that all methods give similar coefficient of variation for dimension larger than 5, while CS and M-HwRG give smaller coefficient of variation than the other methods in low dimensions. It should be noted that the algorithms with acceptance rate of the pre-candidate less than one (CWM-H and CWM-HwDR) require fewer limit-state function evaluations, since for rejected pre-candidates the limit-state function does not need to be computed to check whether the sample lies on the failure domain. Overall, the number of model evaluations for achieving the same target coefficient of variation is similar for all methods and at any dimension n.

(a) $P_f = 10^{-2}$ (b) $P_f = 10^{-4}$

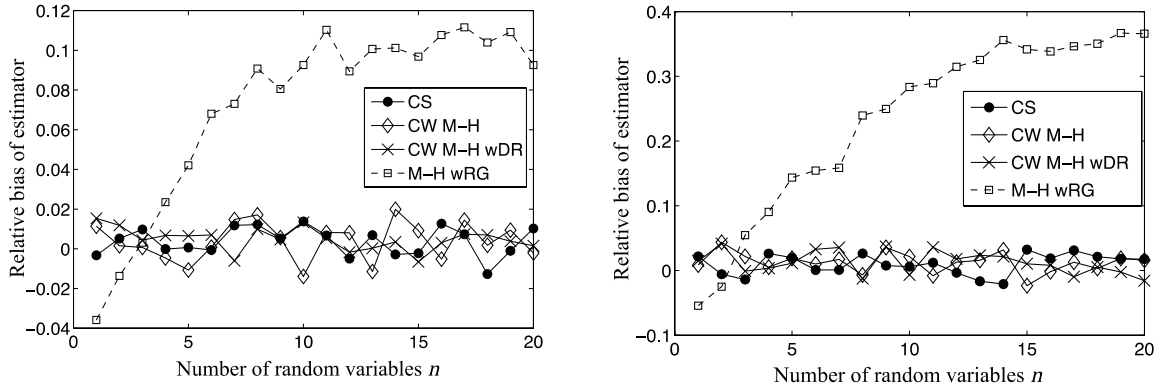


Figure 4: Relative bias of estimator $\hat{P}_f$ from 500 independent simulation runs for Example 1; influence of the number of random variables. (a) $P_f = 10^{-2}$ and (b) $P_f = 10^{-4}$.

(a) $P_f = 10^{-2}$ (b) $P_f = 10^{-4}$

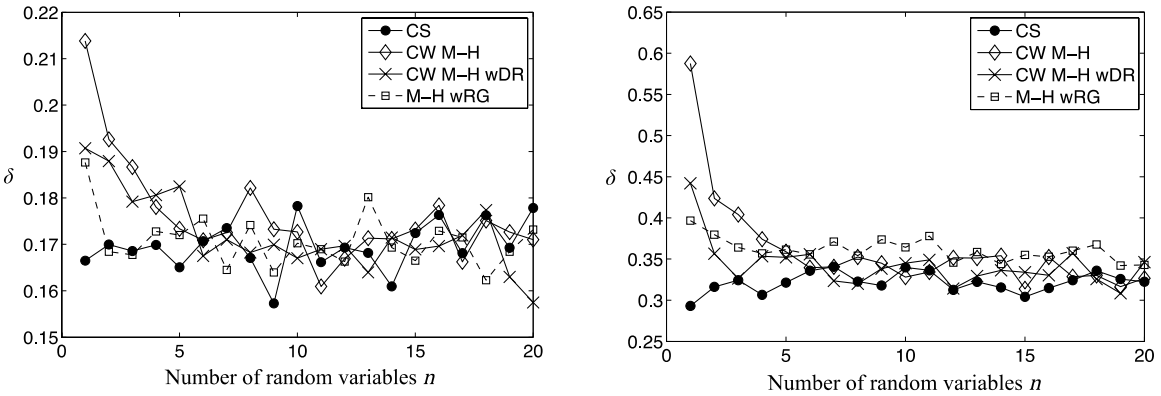


Figure 5: Coefficient of variation δ of the probability estimates evaluated from 500 independent simulation runs for Example 1; influence of the number of random varialbes. (a) $P_f = 10^{-2}$ and (b) $P_f = 10^{-4}$.

Comparing CWM-HwDR with CWM-H, one observes that delayed rejection of the pre-candidate is relevant only for low dimensional problems. For dimensions higher than 5, the two algorithms have similar performance. Since the focus of the paper is high

dimensional problems, CWM-HwDR and M-HwRG (which additionally has a strong bias) are not further examined.

In the following, we fix the number of random variables at $n = 100$ and compare the performance of the CS and CWM-H with the one of the adaptive CS (aCS) algorithm presented in Section 3.4. Figure 6 shows the coefficient of variation of the probability estimate against decreasing target failure probabilities. For aCS we apply a constant $\sigma_{0i} = 1.0$, for all $i = 1, \dots, n$, and the number of chains after which adaptation is performed is chosen as $N_a = 10$. It is shown that the adaptive algorithm performs better than both CS and CWM-H for all values of the target failure probability. The difference is more significant for small failure probabilities; at large probabilities the chosen parameters of CS and CWM-H are close to the optimal ones.

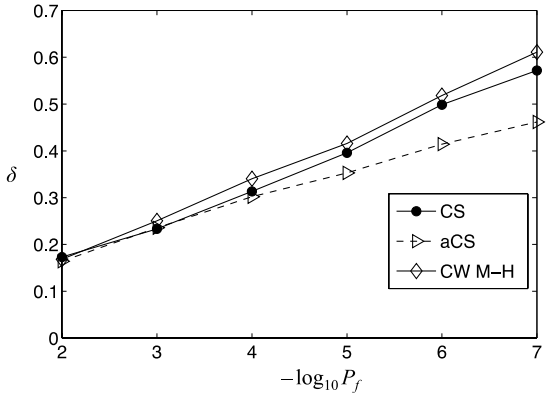


Figure 6: Coefficient of variation δ of the probability estimates evaluated from 500 independent simulation runs plotted against varying target failure probabilities for Example 1 with $n = 100$.

4.2 Example 2

The limit-state function of the second example is a linear function of a vector **X** of independent exponentially distributed random variables with parameter $\lambda = 1$ (Fujita & Rackwitz 1988). We consider two different linear functions of **X**:

$$g_{2a}(\mathbf{X}) = C_a - \sum_{i=1}^{n} X_i \tag{31}$$

and

$$g_{2b}(\mathbf{X}) = -C_b + \sum_{i=1}^{n} X_i \tag{32}$$

The functions are highly nonlinear in **U**-space, due to the marginal transformation of each component of **X**. The function g_{2a} has convex safe domain while the function g_{2b} has concave safe domain in the **U**-space (Engelund & Rackwitz 1993). The probability of failure for g_{2a} is $1 - F_y(C_a)$, where $Y = \sum_{i=1}^{n} X_i$ is a Gamma distributed random variable with shape parameter n and scale parameter $\lambda = 1$. The probability of failure of g_{2b} is $F_y(C_b)$. We choose $n = 100$ for both limit-state functions and vary the thresholds C_a and C_b.

Figure 7 shows the coefficient of variation of the Subset Simulation estimator obtained with CS, aCS and CWM-H for the two nonlinear limit-states of Eqs. (29) and (30) and for varying target failure probabilities. The selected parameters of the methods are the same as in Example 1. It is shown that the adaptive algorithm performs better than the other two for both limit-states and all target failure probabilities. The difference is larger for the concave limit-state, which requires larger scaling of the parameter of the conditional sampling algorithm to achieve the optimal acceptance rate as compared to the convex limit-state. Also, it is noted that the adaptive algorithm has the same performance for both limit-states indicating that the method is not sensitive to the shape of the failure domain.

(a) Convex limit-state (b) Concave limit-state

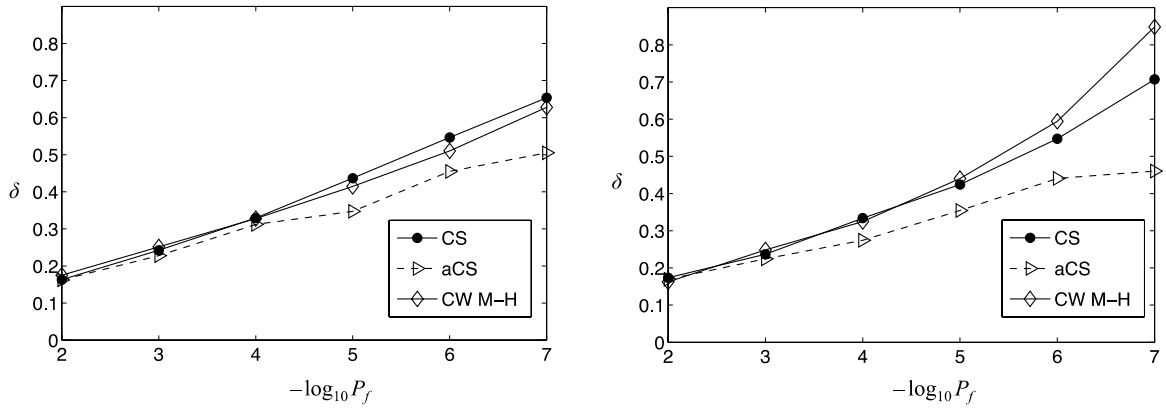


Figure 7: Coefficient of variation δ of the probability estimates evaluated from 500 independent simulation runs plotted against varying target failure probabilities for Example 2 with $n = 100$. (a) convex limit-state function g_{2a} and (b) concave limit-state function g_{2b}.

4.3 Example 3

The third example is constructed by adding a quadratic term to the limit-state of Example 1:

$$g_3(\mathbf{U}) = -\frac{1}{\sqrt{n}}\sum_{i=1}^{n} U_i + \beta + \frac{\kappa}{4}(U_1 - U_2)^2 \tag{33}$$

Here κ is the principle curvature at the design point, i.e. the failure point with largest probability density (Ditlevsen & Madsen 1996). A larger curvature will lead to a smaller failure probability. For a fixed curvature, the probability of failure is the same, independent of n. The exact probability of failure can be expressed as the following function of β and κ:

$$P_f = \int_{x=-\infty}^{-\beta}\int_{v=-\infty}^{\infty} \varphi\left(-x + \frac{1}{2}\kappa v^2\right)\varphi(v)dv\,dx \tag{34}$$

For this example, we again fix the number of random variables at $n = 100$ and vary the target failure probability through varying β for three different choices of the principle curvature, namely $\kappa = 1$, $\kappa = 5$ and $\kappa = 10$. For a small curvature the limit-state

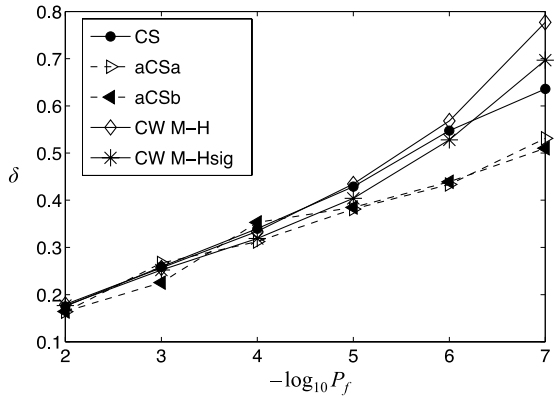
function of Eq. (33) becomes almost linear in U_1 and U_2 and therefore these two random variables have similar influence on the probability of failure than the remaining random variables. Conversely, for a large curvature, U_1 and U_2 dominate the limit-state function and the influence of the remaining random variables becomes negligible.

We study the performance of the following versions of the CS and CWM-H algorithms: The version of CS with a constant correlation parameter chosen as $\rho_i = 0.8, i = 1, \dots, n$; the adaptive CS with constant starting values (aCSa) chosen as $\sigma_{0i} = 1.0, i = 1, \dots, n$; the adaptive CS with starting values taken as the sample standard deviations of the seeds of each subset level (aCSb); the CWM-H with proposal distribution chosen as the normal PDF centered at the current state with unit standard deviation; the CWM-H with the same proposal distribution but with standard deviation of each component taken as the sample standard deviation of the components of the seeds at each subset level (CWM-Hsig), as suggested in (Au et al. 2010). The coefficient of variation obtained for varying failure probabilities and the three selected values of the principle curvature κ are plotted in Figure 8.

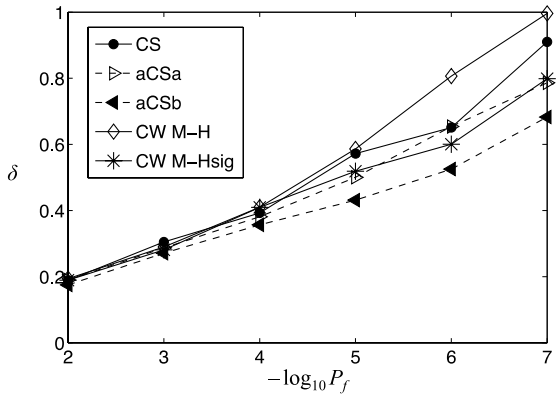
Comparing CS with its adaptive variants, we see that both variants aCSa and aCSb give smaller coefficients of variation at all target failure probabilities and all values of the curvature κ. The adaptive algorithms have similar performance for the case where $\kappa = 1$ for which all random variables have similar influence on the limit-state function. As κ increases, and consequently the relative influence of the first two variables on the limit-state increases, the performance of the adaptive algorithm aCSb that accounts for the relative influence of the random variables becomes better than the one of aCSa that adopts the same starting values for all random variables. The same can be observed when comparing CWM-H with CWM-Hsig: the latter algorithm, which adapts the variance of each component based on the sample variance of the seeds at each subset level, performs better than the former that uses a constant variance. Moreover, the gain in performance of CWM-Hsig as compared to CWM-H is more pronounced in larger curvatures. However,

the adaptive algorithm aCSb performs better than CWM-Hsig, because the former additionally ensures that the acceptance rate is close to the optimal value of 0.44.

(a) $\kappa = 1$



(b) $\kappa = 5$



(c) $\kappa = 10$

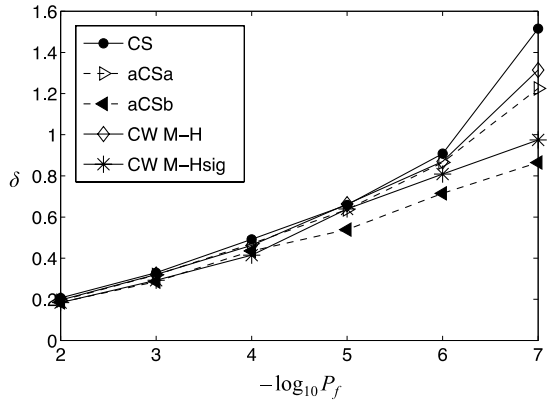


Figure 8: Coefficient of variation δ of the probability estimates evaluated from 500 independent simulation runs plotted against varying target failure probabilities for Example 3 with $n = 100$. (a) $\kappa = 1$ (b) $\kappa = 5$ (c) $\kappa = 10$.

4.4 Example 4

We examine an example whose limit-state function depends on the numerical solution of a partial differential equation. Consider the linear elliptic equation $-\frac{d}{dz}\left(a\frac{dv}{dz}\right) = 1$, $z \in [0,1]$, with boundary conditions $v(0) = 0$ and $\left.\frac{dv}{dz}\right|_{z=1} = 0$. Linear elliptic equations are applied to describe several physical phenomena, such as heat conduction or linear elasticity. Here a is a homogeneous random field with lognormal marginal distribution, mean value $\mu_a = 1$ and standard deviation $\sigma_a = 0.1$. The autocorrelation function of $\ln a$ is $\rho_{\ln a}(z_1, z_2) = \exp\left(-\frac{|z_1 - z_2|}{l}\right)$ with a correlation length $l = 0.01$. The spatial discretization of the boundary value problem is done by 100 piecewise linear finite elements. The natural logarithm of the random field $\ln a$ is represented by its truncated Karhunen-Loève (K-L) expansion (Ghanem & Spanos 1991), which takes the following form:

$$\ln a = \mu_{\ln a} + \sigma_{\ln a} \sum_{i=1}^{n} \sqrt{\theta_i}\chi_i(z)U_i \tag{35}$$

where $\{\theta_i, \chi_i\}$ are the eigenpairs of $\rho_{\ln a}$, which are known analytically for the applied exponential correlation model (Ghanem & Spanos 1991), and $U_i, i = 1, \dots, n$, are independent standard normal random variables. Here we use $n = 200$, which captures 90% of the variability of $\ln a$. The failure event is defined as $v(1)$ exceeding a threshold v_{max}, with corresponding limit-state function $g_4 = v_{max} - v(1)$. The constant v_{max} is chosen to achieve a fixed probability of failure P_f.

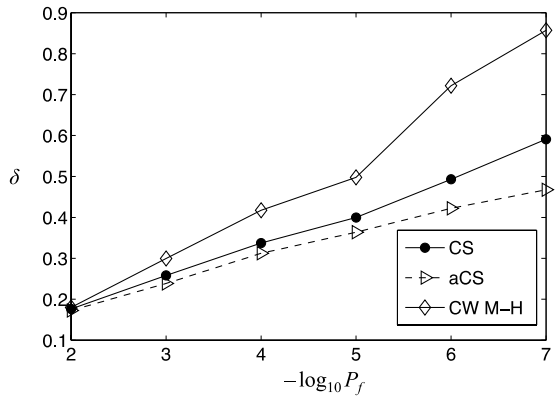


Figure 9: Coefficient of variation δ of the probability estimates evaluated from 500 independent simulation runs plotted against varying target failure probabilities for Example 4.

Figure 9 compares the coefficient of variation of the probability estimate obtained by CS, aCS and CWM-H for varying v_{max} and corresponding target failure probabilities. The correlation parameter for CS and proposal PDF for CWM-H are chosen as in Examples 1 and 2. Due to the small correlation length of the random field, the eigenvalues of $\rho_{\ln a}$ decay slowly and hence a large part of the random variables in the K-L expansion will have considerable influence on the limit-state function. We therefore choose a constant $\sigma_{0i} = 1.0$, for all $i = 1, \ldots, n$, for the aCS method. The figure shows that the aCS performs better than CS and CWM-H. In this example, the particular choice of the proposal distribution of the CWM-H algorithm (normal PDF centered on current state with unit variance) leads to poor performance for small failure probabilities. On the other hand, the chosen correlation parameter of the CS algorithm ($\rho_i = 0.8$) seems to be a good choice for all examples considered.

5. Conclusion

This paper reviewed existing MCMC methods for Subset Simulation and proposed a new method that is based on sampling from a conditional normal distribution. The basic version of the new approach is simpler and performs better than the other methods in low dimensional problems, since it accepts all candidate states of the Markov chain without compromising the target distribution of the chain. In high-dimensional problems, the new method, together with all other algorithms that increase the first level acceptance rate, has a similar performance as the component-wise M-H algorithm, which was originally proposed for Subset Simulation. In addition, an adaptive variant of the conditional sampling method is proposed, which adjusts the correlation parameter of the method on the fly, based on the chain acceptance rate. This adaptive approach also allows accounting for the relative influence of each random variable, based on the statistics of the seeds of the Markov chains at each failure level. Numerical examples demonstrated that the proposed adaptive conditional sampling method significantly improves the performance of the standard conditional sampling and appears to outperform all reviewed existing MCMC methods for Subset Simulation, without increasing the computational cost.

Acknowledgments

This work was supported by the Institute for Advanced Study of the Technische Universität München, funded by the German Excellence Initiative. This support is gratefully aknowledged.

Appendix A

In this Appendix, we show that the algorithm proposed in Section 3.3 results in a Markov chain with stationary distribution $\varphi_n(\cdot\,|F_j)$. Let $\mathbf{u}_0$, $\mathbf{u}_1$ be subsequent states of the chain generated with this algorithm. Following Eq. (16), the transition PDF $p(\mathbf{u}_1|\mathbf{u}_0)$ defining the transition between $\mathbf{u}_0$ and $\mathbf{u}_1$ is given by:

$$p(\mathbf{u}_1|\mathbf{u}_0) = \varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}}) I_{F_j}(\mathbf{u}_1) + \big(1 - r(\mathbf{u}_0)\big)\delta_{\mathbf{u}_0}(\mathbf{u}_1) \tag{36}$$

where $\mathbf{R}$ is the cross-correlation matrix between the actual and the candidate state, $\delta_{\mathbf{u}_0}(\cdot)$ is the Dirac mass at $\mathbf{u}_0$ and $r(\mathbf{u}_0)$ is the acceptance probability of the candidate, evaluated by:

$$r(\mathbf{u}_0) = \int_{\mathbf{v}\in\mathbb{R}^n} \varphi_n(\mathbf{v} - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}}) I_{F_j}(\mathbf{v}) d\mathbf{v} \tag{37}$$

As discussed in Section 3.3, a possible choice of the matrix $\mathbf{R}$ is the diagonal matrix, with ith diagonal entry equal to ρ_i. To show that $\varphi_n(\cdot\,|F_j)$ is the stationary distribution of the chain, it suffices to show that the transition PDF $p(\mathbf{u}_1|\mathbf{u}_0)$ satisfies the reversibility condition of Eq. (15), repeated here for convenience:

$$p(\mathbf{u}_1|\mathbf{u}_0)\varphi_n(\mathbf{u}_0|F_j) = p(\mathbf{u}_0|\mathbf{u}_1)\varphi_n(\mathbf{u}_1|F_j) \tag{38}$$

If the above equation is satisfied and the actual state $\mathbf{u}_0$ follows $\varphi_n(\cdot\,|F_j)$, then the PDF $\pi(\cdot)$ of $\mathbf{u}_1$ is also equal to $\varphi_n(\cdot\,|F_j)$:

$$\begin{aligned}
\pi(\mathbf{u}_1) &= \int_{\mathbf{u}_0\in\mathbb{R}^n} p(\mathbf{u}_1|\mathbf{u}_0)\varphi_n(\mathbf{u}_0|F_j)\, d\mathbf{u}_0 \\
&= \int_{\mathbf{u}_0\in\mathbb{R}^n} p(\mathbf{u}_0|\mathbf{u}_1)\varphi_n(\mathbf{u}_1|F_j)\, d\mathbf{u}_0 \\
&= \varphi_n(\mathbf{u}_1|F_j) \int_{\mathbf{u}_0\in\mathbb{R}^n} p(\mathbf{u}_0|\mathbf{u}_1)\, d\mathbf{u}_0 \\
&= \varphi_n(\mathbf{u}_1|F_j)
\end{aligned} \tag{39}$$

We now need to show that the transition defined by Eq. (36) satisfies the reversibility condition of Eq. (15). Since all Markov chain samples lie in F_j, it is sufficient to consider transition between states in F_j. If the candidate state is rejected and hence the chain remains in its current state, i.e. $\mathbf{u}_1 = \mathbf{u}_0$, then compliance with Eq. (38) is trivial. Hence, one only needs to consider the case where the candidate state is accepted. In this case, transition between $\mathbf{u}_0$ and $\mathbf{u}_1$ is defined by the transition between the current and the

candidate state, given by the density $\varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})$. Therefore, it suffices to show that:

$$\varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_0|F_j) = \varphi_n(\mathbf{u}_0 - \mathbf{R}\mathbf{u}_1; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_1|F_j) \quad (40)$$

Substituting Eq. (5) in the above and using the fact that $\mathbf{u}_0, \mathbf{u}_1 \in F_j$, Eq. (40) can be written as:

$$\varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_0) = \varphi_n(\mathbf{u}_0 - \mathbf{R}\mathbf{u}_1; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_1) \quad (41)$$

To demonstrate the validity of Eq. (41), we introduce the $2n$-dimensional Gaussian random vector $\mathbf{U}$ with joint PDF $\varphi_{2n}(\mathbf{u}\,; \boldsymbol{\Sigma})$. $\mathbf{U}$ has zero mean vector and covariance matrix $\boldsymbol{\Sigma}$ given by:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{R}^{\mathrm{T}} & \mathbf{I} \end{bmatrix} \quad (42)$$

Let $\mathbf{U}_0$, $\mathbf{U}_1$ be n-dimensional random vectors such that $\mathbf{U} = [\mathbf{U}_0 \quad \mathbf{U}_1]^{\mathrm{T}}$. Then both $\mathbf{U}_0$ and $\mathbf{U}_1$ will have marginal PDF $\varphi_n(\cdot)$, while the conditional PDF of $\mathbf{U}_0$ given $\mathbf{U}_1$ will be $\varphi_n(\mathbf{u}_0 - \mathbf{R}\mathbf{u}_1; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})$ and the conditional distribution of $\mathbf{U}_1$ given $\mathbf{U}_0$ will be $\varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})$. Therefore, the joint PDF of $\mathbf{U}$ can be expressed as:

$$\begin{aligned} &\varphi_{2n}(\mathbf{u}\,; \boldsymbol{\Sigma}) = \varphi_n(\mathbf{u}_1 - \mathbf{R}\mathbf{u}_0; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_0) \\ &= \varphi_n(\mathbf{u}_0 - \mathbf{R}\mathbf{u}_1; \mathbf{I} - \mathbf{R}\mathbf{R}^{\mathrm{T}})\varphi_n(\mathbf{u}_1) \end{aligned} \quad (43)$$

The above shows that Eq. (41) is satisfied, which proves that $\varphi_n(\cdot\,|F_j)$ is the stationary distribution of the Markov chain generated by Eq. (36).

Appendix B

In this Appendix, we show that if the covariance of the proposal density of the conditional sampling algorithm is given as $\boldsymbol{\Sigma}$, then the cross-correlation matrix between the actual and the candidate state $\mathbf{R}$ can be evaluated through Eq. (29), provided that the matrix $\mathbf{I} - \boldsymbol{\Sigma}$ is positive-semi definite. Given the covariance matrix of the proposal density, the cross-correlation matrix $\mathbf{R}$ can be evaluated by requiring that

$$\mathbf{R}\mathbf{R}^{\mathrm{T}} = \mathbf{I} - \boldsymbol{\Sigma} \quad (44)$$

The matrix $\mathbf{R}$ is positive-semi definite, being a covariance matrix, and hence adheres the spectral decomposition

$$\mathbf{R} = \sum_{i=1}^{n} \theta_i^{\mathbf{R}} \, \mathbf{c}_i^{\mathbf{R}} \left(\mathbf{c}_i^{\mathbf{R}}\right)^{\mathrm{T}} \tag{45}$$

where $\{\theta_i^{\mathbf{R}}, \mathbf{c}_i^{\mathbf{R}}\}$ are n pairs of non-negative eigenvalues and orthonormal eigenvectors. Noting that the matrix $\mathbf{R}$ is symmetric and hence $\mathbf{R}^{\mathrm{T}} = \mathbf{R}$:

$$\mathbf{R}\mathbf{R}^{\mathrm{T}} = \sum_{i=1}^{n} \sum_{j=1}^{n} \theta_i^{\mathbf{R}} \, \theta_j^{\mathbf{R}} \mathbf{c}_i^{\mathbf{R}} \left(\mathbf{c}_i^{\mathbf{R}}\right)^{\mathrm{T}} \mathbf{c}_j^{\mathbf{R}} \left(\mathbf{c}_j^{\mathbf{R}}\right)^{\mathrm{T}} \tag{46}$$

From the orthonormality of the eigenvectors it follows that $\left(\mathbf{c}_i^{\mathbf{R}}\right)^{\mathrm{T}} \mathbf{c}_j^{\mathbf{R}} = \delta_{ij}$, where δ_{ij} denotes the Kronecker delta. Substituting into Eq. (46), one gets

$$\mathbf{R}\mathbf{R}^{\mathrm{T}} = \sum_{i=1}^{n} \left(\theta_i^{\mathbf{R}}\right)^2 \mathbf{c}_i^{\mathbf{R}} \left(\mathbf{c}_i^{\mathbf{R}}\right)^{\mathrm{T}} \tag{47}$$

From Eqs. (44) and (47) it is seen that if the matrix $\mathbf{I} - \boldsymbol{\Sigma}$ is positive-semi definite, then its eigenpairs $\{\theta_i, \mathbf{c}_i\}$ can be expressed in terms of the eigenpairs of $\mathbf{R}$ as $\theta_i = \left(\theta_i^{\mathbf{R}}\right)^2$ and $\mathbf{c}_i = \mathbf{c}_i^{\mathbf{R}}$. Therefore, applying Eq. (45), we can express $\mathbf{R}$ as:

$$\mathbf{R} = \sum_{i=1}^{n} \sqrt{\theta_i} \, \mathbf{c}_i \mathbf{c}_i^{\mathrm{T}} \tag{48}$$

References

Andrieu, C. & Thoms, J. 2008. A tutorial on adaptive MCMC. Statistics and Computing, 18: 343-373.

Ang, G.L., Ang A.H.-S. & Tang, W.H. 1992. Optimal importance sampling density estimator. Journal of Engineering Mechanics, ASCE, 118(6): 1146-1163.

Au, S.K. & Beck, J.L. 1999. A new adaptive importance sampling scheme. Structural Safety, 21: 135-158.

Au, S.K. & Beck, J.L. 2001. Estimation of small failure probabilities in high dimensions by Subset Simulation. Probabilistic Engineering Mechanics, 16(4): 263-277.

Au, S.K. & Beck, J.L. 2003a. Important sampling in high dimensions. Structural Safety, 25: 139-163.

Au, S.K. & Beck, J.L. 2003b. Subset Simulation and its application to seismic risk based on dynamic analysis. Journal of Engineering Mechanics, ASCE, 129(8): 901-17.

Au, S.K., Cao Z.J. & Wang, Y. 2010. Implementing advanced Monte Carlo simulation under spreadsheet environment. Structural Safety, 32: 281-292.

Au, S.K., Beck, J.L., Zuev, K.M. & Katafygiotis, L.S. 2011. Discussion of paper by F. Miao and M. Ghosn "Modified Subset Simulation method for reliability analysis of structural systems", Structural Safety, 33:251-260, 2011. Structural Safety, 34: 379-380.

Besag, J., Green, P., Higdon, D. & Mengersen, K. 1995. Bayesian computation and stochastic systems. Statistical Science, 10(1): 3-66.

Botev, Z. I. & Kroese, D. P. 2012. Efficient Monte Carlo simulation via the generalized splitting method. Statistics and Computing, 22: 1-16.

Bucher, C. 1988. Adaptive sampling – an iterative fast Monte Carlo procedure. Structural Safety, 5: 119-126.

Bucher, C. 2009. Asymptotic sampling for high-dimensional reliability analysis. Probabilistic Engineering Mechanics, 24: 504-510.

Cérou, F., Del Moral, P., Furon, T. & Guyader, A. 2012. Sequential Monte Carlo for rare event estimation. Statistics and Computing, 22: 795-808.

Del Moral, P., Doucet, A. & Jasra, A. 2006. Sequential Monte Carlo samplers. Journal of the Royal Statistical Society: Series B, Statistical Methodology, 68(3): 411-436.

Der Kiureghian, A. & Liu, P.-L. 1986. Structural reliability under incomplete probability information. Journal of Engineering Mechanics, ASCE, 112(1): 85-104.

Ditlevsen, O. & Madsen, H.O. 1996. Structural Reliability Methods, Chichester, Wiley.

Engelund, S. & Rackwitz, R. 1993. A benchmark study on importance sampling techniques in structural reliability. Structural Safety, 12(4): 255-276.

Flury, B.D. 1990. Acceptance-rejection sampling made easy. SIAM Review, 32(3): 474-476.

Fujita, M. & Rackwitz, R. 1988. Updating first- and second-order estimates by importance sampling. Structural Engineering/Earthquake Engineering, 5: 53-59.

Gelman, A., Roberts, G.O. & Gilks, W.R. 1996. Efficient Metropolis jumping rules. Bayesian Statistics, 5: 599-607.

Ghanem, R. & Spanos, P. 1991. Stochastic Finite Elements: A Spectral Approach, New York, Springer-Verlag.

Gilks, W.R., Roberts, G.O. & Sahu, S.K. 1998. Adaptive Markov chain Monte Carlo through regeneration. Journal of the American Statistical Association, 93: 1045-1054.

Haario, H., Saksman, E. & Tamminen, J. 2001. An adaptive Metropolis algorithm. Bernoulli, 7(2): 223-242.

Haario, H., Saksman, E. & Tamminen, J. 2005. Componentwise adaptation for high dimensional MCMC. Computational Statistics, 20: 265-273.

Hastings, W.K. 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrica, 57(1): 97-109.

Hayes, J.F. & Hill, W.G. 1981. Modification of estimates of parameters in the construction of generic selection indices (‘Bending’). Biometrics, 37(3): 483-493.

Hohenbichler, M. & Rackwitz, R. 1981. Non-normal dependent vectors in structural safety. Journal of Engineering Mechanics, ASCE, 107: 1227-1238.

Hohenbichler, M. & Rackwitz, R. 1988. Improvements of second-order reliability estimates by importance sampling. Journal of Engineering Mechanics, ASCE, 114: 2195-2199.

Katafygiotis, L.S. & Cheung S.H. 2007. Application of the spherical Subset Simulation method and auxiliary domain method on a benchmark reliability study. Structural Safety, 29: 194-207.

Katafygiotis, L.S. & Zuev K.M. 2008. Geometric insight into the challenges of solving high-dimensional reliability problems. Probabilistic Engineering Mechanics, 23: 208-218.

Koutsourelakis, P.S., Pradlwarter, H.J. & Schuëller, G.I. 2004. Reliability of structures in high-dimensions, part I: algorithms and applications. Probabilistic Engineering Mechanics, 19: 409-417.

Kurtz, N. & Song J. 2013. Cross-entropy-based adaptive importance sampling using Gaussian mixture. Structural Safety, 42: 35-44.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. & Teller, E. 1953. Equation of state calculations by fast computing machines. Journal of Chemical Physics, 21(6): 1087-1092.

Miao, F. & Ghosn M. 2011. Modified Subset Simulation method for reliability analysis of structural systems. Structural Safety, 33: 251-260.

Mykland, P., Tierney, L. & Yu B. 1995. Regeneration in Markov chain samplers. Journal of the American Statistical Association, 90: 233-241.

Neal, R.M. 1998. Regression and classification using Gaussian process priors. Bayesian Statistics, 6: 475-501.

Pasarica, C. & Gelman, A. 2010. Adaptively scaling the Metropolis algorithm using expected squared jumped distance. Statistica Sinica, 20: 343-364.

Rackwitz, R. 2001. Reliability analysis – a review and some perspectives. Structural Safety, 23: 365-395.

Roberts, G.O., Gelman, A. & Gilks, W.R. 1997. Weak convergence and optimal scaling of random walk Metropolis algorithms. Annals of Applied Probability, 7(1): 110-120.

Roberts, G.O. & Rosenthal, J.S. 1998. Optimal scaling of discrete approximations to Langevin diffusions. Journal of the Royal Statistical Society: Series B, Statistical Methodology, 60(1): 255-268.

Roberts, G.O. & Rosenthal, J.S. 2001. Optimal scaling for various Metropolis-Hastings algorithms. Statistical Science, 16(4): 351-367.

Rubinstein, R.Y. & Kroese, D.P. 2007. Simulation and the Monte Carlo Method, 2nd ed., Hoboken, John Wiley & Sons.

Schuëller, G.I., Pradlwarter, H.J. & Koutsourelakis, P.S. 2004. A critical appraisal of reliability estimation procedures for high dimensions. Probabilistic Engineering Mechanics, 19(4): 463-474.

Schuëller, G.I. & Stix, R. 1987. A critical appraisal of methods to determine failure probabilities. Structural Safety, 4(4): 239-309.

Santoso, A.M., Phoon, K.K. & Quek, S.T. 2011. Modified Metropolis-Hastings algorithm with reduced chain correlation for efficient Subset Simulation. Probabilistic Engineering Mechanics, 26: 331-341.

Tierney, L. 1994. Markov chains for exploring posterior distributions. The Annals of Statistics, 22(4): 1701-1762.

Tierney, L., & Mira, A. 1999. Some adaptive Monte Carlo methods for Bayesian inference. Statistics in Medicine, 18: 2507-2515.

Valdebenito, M.A., Pradlwarter, H.J. & Schuëller, G.I. 2010. The role of the design point for calculating failure probabilities in view of dimensionality and structural nonlinearities. Structural Safety, 32: 101-111.

Zuev, K.M., Beck, J.L., Au, S.-K. & Katafygiotis, L.S. 2012. Bayesian post-processor and other enhancements of Subset Simulation for estimating failure probabilities in high dimensions. Computers & Structures, 92-93: 283-296.

Zuev, K.M. & Katafygiotis, L.S. 2011. Modified Metropolis-Hastings algorithm with delayed rejection. Probabilistic Engineering Mechanics, 26: 405-412.