Original Article

Institution of MECHANICAL ENGINEERS

Journal of MECHANICAL ENGINEERING SCIENCE

Proc IMechE Part C:
J Mechanical Engineering Science
227(12) 2780–2794

DOI: 10.1177/0954406213479846
pic.sagepub.com

SAGE

Reliability prediction through guided tail modeling using support vector machines

Erdem Acar

Abstract

Reliability prediction of highly safe mechanical systems can be performed using classical tail modeling. Classical tail modeling is based on performing a relatively small number of limit-state evaluations through a sampling scheme and then fitting a tail model to the tail part of the data. However, the limit-state calculations that do not belong to the tail part are discarded, so majority of limit-state evaluations are wasted. Guided tail modeling, proposed earlier by the author, can provide a remedy through guidance of the limit-state function calculations toward the tail region. In the original guided tail modeling, the guidance is achieved through a procedure based on threshold estimation using univariate dimension reduction and extended generalized lambda distribution and tail region approximation using univariate dimension reduction. This article proposes a new guided tail modeling technique that utilizes support vector machines. In the proposed method, named guided tail modeling with support vector machines (GTM-SVM), the threshold estimation is still performed using univariate dimension reduction and extended generalized lambda distribution, while the tail region approximation is based on support vector machines. The performance of guided tail modeling with support vector machines is tested with mathematical example problems as well as structural mechanics problems with varying number of variables. GTM-SVM is found to be more accurate than both guided tail modeling and classical tail modeling for low-dimensional problems. For high-dimensional problems, on the other hand, the original guided tail modeling is found to be more accurate than guided tail modeling with support vector machines, which is superior to classical tail modeling.

Keywords

Guided simulations, high reliability, support vector machines, tail modeling

Date received: 10 October 2012; accepted: 1 February 2013

Introduction

The limit-state function of a mechanical system is usually evaluated from computationally expensive analyses (e.g., finite element analysis). The simulation techniques such as Monte Carlo method[1] or its advanced variants (e.g. importance sampling,[2] adaptive importance sampling[3] and directional simulation[4]) require a large number of limit-state evaluations, hence they are not suitable for highly safe mechanical systems. Alternatively, the analytical methods such as first-/second-order reliability methods (FORM/SORM) are computationally efficient, but their accuracy diminishes as the limit-state function becomes nonlinear. In order to overcome the drawbacks of these traditional methods, the techniques based on tail modeling have been successfully used for reliability assessment at high reliability levels.[5–11]

Reliability estimation using tail modeling is based on approximating the tail of the cumulative distribution function (CDF) of the limit-state function. Classical tail modeling methods are based on the following procedure.[10] First, a set of limit-state evaluations through Monte Carlo simulations (MCS) is performed. Then, a proper threshold value of the CDF is selected that specifies the tail part. Finally, the generalized Pareto distribution (GPD) is fitted to the tail part (i.e. the portion above the threshold value). In this procedure, only the tail part of the limit-state function evaluations is used in finding the parameters of the GPD, whereas the rest of the data are discarded. That is, the efforts spent for performing limit-state function evaluations that do not belong to the tail part are wasted.

To reduce the amount of wasted data, guided tail modeling (GTM) technique has been proposed recently.[12] In GTM, the limit-state calculations are guided toward the sampling points that have high chances of yielding limit-state values falling into the

Mechanical Engineering Department, TOBB University of Economics and Technology, Söğütözü, Ankara, Turkey

Corresponding author:
Erdem Acar, Mechanical Engineering Department, TOBB University of Economics and Technology, Söğütözü, Ankara 06560, Turkey.
Email: acar@etu.edu.tr

tail region. The guidance of the limit-state evaluations is achieved through a procedure that is based on approximating the limit-state function and calculating the statistical moments of the limit-state function using the univariate dimension reduction (UDR) method[13] along with distribution fitting using extended generalized lambda distributions (EGLD).[14,15] When the limit-state function is nonlinear, the limit-state function approximation using UDR is ineffective. In this article, a new procedure is proposed that does not require limit-state function approximation. Instead, the regions of the design space that yield limit-state values in the tail region are approximated using support vector machines (SVM). Then, the limit-state function calculations are performed by sampling from those regions.

This article is organized as follows: a brief overview of the classical tail modeling is presented in the following section. Brief details of the GTM procedure are outlined in the next section. The proposed method, GTM-SVM, is explained in the subsequent section. The accuracy of the proposed method is evaluated through mathematical and engineering example problems in the following section. Finally, the last section provides the concluding remarks.

Classical tail modeling

Consider the limit-state function $y(x)$, where x is the vector of random variables. For a large threshold value of y_t (see Figure 1), the region above the threshold (i.e. the tail part) can be approximated by using GPD. The GPD approximates the conditional excess distribution of $F_z(z)$, where $z = y - y_t$, through

$$F_z(z) = \begin{cases} 1 - \left\langle 1 + \dfrac{\xi}{\sigma} z \right\rangle_+^{-\frac{1}{\xi}} & \text{if } \xi \neq 0 \\ 1 - \exp\left(-\dfrac{z}{\sigma}\right) & \text{if } \xi = 0 \end{cases} \quad (1)$$

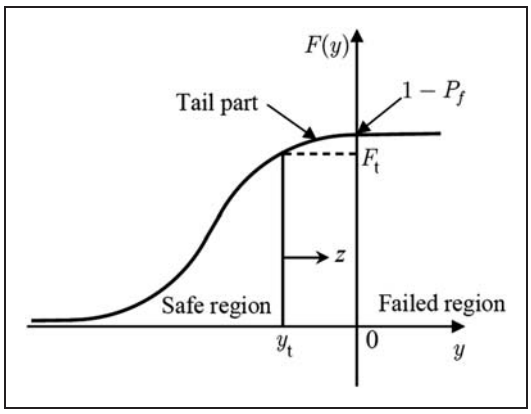

Figure 1. Tail modeling concept.

where $\langle A \rangle_+ = \max(0, A)$, $z \geqslant 0$ and $F_z(z)$ is the GPD with shape and scale parameters ξ and σ, respectively, which need to be determined.

The conditional excess distribution can be related to the cumulative distribution $F(y)$ through

$$F_z(z) = \frac{F(y) - F(y_t)}{1 - F(y_t)} = \frac{F(y) - F_t}{1 - F_t} \quad (2)$$

Then, $F(y)$ above the threshold (i.e. $y \geqslant y_t$) is expressed in terms of the conditional excess distribution, $F_z(z)$, through

$$F(y) = F_t + (1 - F_t) F_z(y - y_t) \quad (3)$$

Once the cumulative distribution function $F(y)$ is obtained, the probability of failure can be estimated from[10]

$$P_f = 1 - F(y = 0) = (1 - F_t)\left\langle 1 - \frac{\xi}{\sigma} y_t \right\rangle_+^{-\frac{1}{\xi}} \quad (4)$$

Also, the reliability index can be calculated from

$$\beta = \Phi^{-1}(1 - P_f) \quad (5)$$

where Φ is the CDF of a standard normal random variable.

The classical tail modeling (CTM) methods are based on: (i) generating samples of the limit-state function through a sampling procedure, (ii) selecting a proper threshold to specify the tail part and (iii) fitting a tail model to the tail part of the data. Details of these steps can be found in a previous study by Ramu[10] In this article, 500 samples are generated through MCS, and 50 of these samples are used to define the tail part and fit a tail model.

GTM

As noted earlier, CTM uses only the tail part of the data in estimating the GPD parameters, whereas the other data are discarded. GTM can reduce the wasted data using the following simple procedure.[12] First, a large number of potential sampling points are generated via MCS. Next, the approximate value of the limit-state function is computed for all these potential sampling points. Then, the points with approximate limit-state values larger than the threshold y_t are stored. Finally, the actual limit-state function calculations are performed only for these stored points as they have high chances of producing actual limit-state values in the tail region. The GTM procedure is shown in Figure 2. In GTM, the approximate model for the limit-state function is constructed through an additive decomposition technique used in the UDR method. The threshold value y_t is estimated via an efficient distribution fitting technique that blends

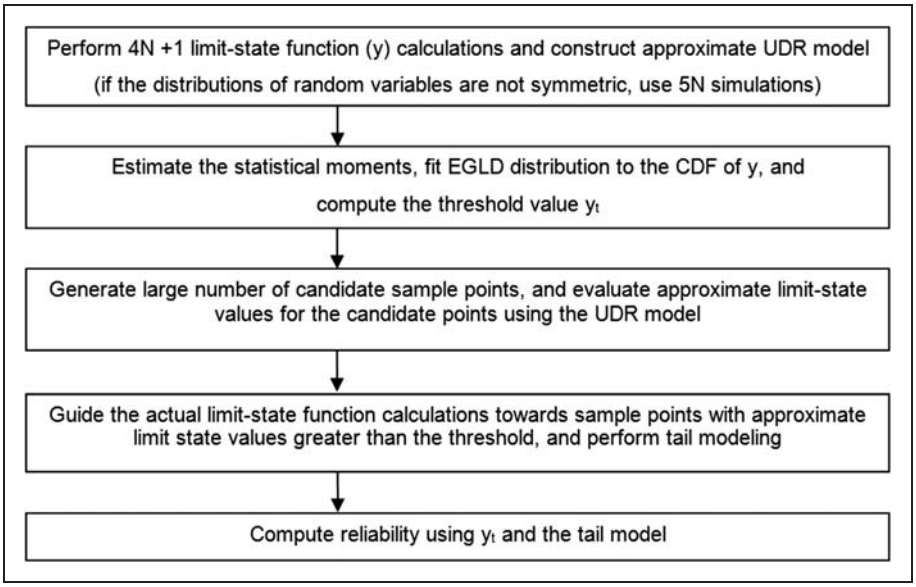

Figure 2. Flowchart for GTM.
GTM: guided tail modeling.

UDR and EGLD. Details of UDR and EGLD are provided below.

Limit-state function approximation using UDR method

As part of the UDR method, an additive decomposition technique is used such that a multi-dimensional limit-state function $y(\mathbf{X})$ is approximated using multiple unidimensional functions as[13]

$$\hat{y}(\mathbf{X}) = \sum_{j=1}^{N} y_j^u(X_j) - (N-1)\, y_0 \tag{6}$$

where each term in the summation, y_j^u, is a unidimensional function that depends on the jth random variable, X_j.

$$y_j^u(X_j) = y(\mu_1, \ldots, \mu_{j-1}, X_j, \mu_{j+1}, \ldots, \mu_N) \tag{7}$$

where y_0 is the value of $y(\mathbf{X})$ calculated at the mean values of all the random variables, μ_j, $j = 1, \ldots, N$.

$$y_0 = y(\mu_1, \ldots, \mu_N) \tag{8}$$

For the unidimensional functions, $y_j^u(X_j)$, metamodels can be constructed using a small number of simulations. A quadratic polynomial in one dimension has three coefficients, hence five sampling points may provide a good approximation for y_j^u. For highly nonlinear functions, however, the number of sampling points may need to be increased. The locations of the sampling points can be determined by using the moment-based quadrature points proposed by Rahman and Xu.[13]

Distribution fitting via UDR method and EGLD

The statistical moments of the limit-state function can be calculated efficiently using the UDR method. After the first four moments of the limit-state function are calculated, these moments can be matched with the moments of an EGLD, so that the distribution parameters of the fitted EGLD are assessed. More information on the EGLD and finding its distribution parameters from the statistical moments can be found in Refs.[14,15]

After the distribution parameters of the EGLD are found, the threshold value y_t can be easily estimated from the inverse CDF of the fitted EGLD via

$$y_t = F_{\mathrm{EGLD}}^{-1}(F_t) \tag{9}$$

where F_{EGLD} is the CDF of the fitted EGLD, and F_t is the selected threshold CDF value. Details of UDR and EGLD can be found in Refs.[13–16]

The proposed method, GTM-SVM

As noted earlier, if the limit-state function is nonlinear, its approximation using UDR is ineffective. In this article, a new procedure is proposed based on support vector machines (SVMs). Instead of approximating the limit-state function as in the case of GTM, the regions of the variable space that yield limit-state values in the tail part are approximated using SVM. Then, the actual limit-state function calculations are performed by sampling from the tail-associated regions of the variable space. The overall procedure of GTM with SVMs is depicted in Figure 3.

SVMs, introduced by Vapnik[17], are based on statistical learning theory. SVMs are widely used for data classification and pattern recognition. SVMs are

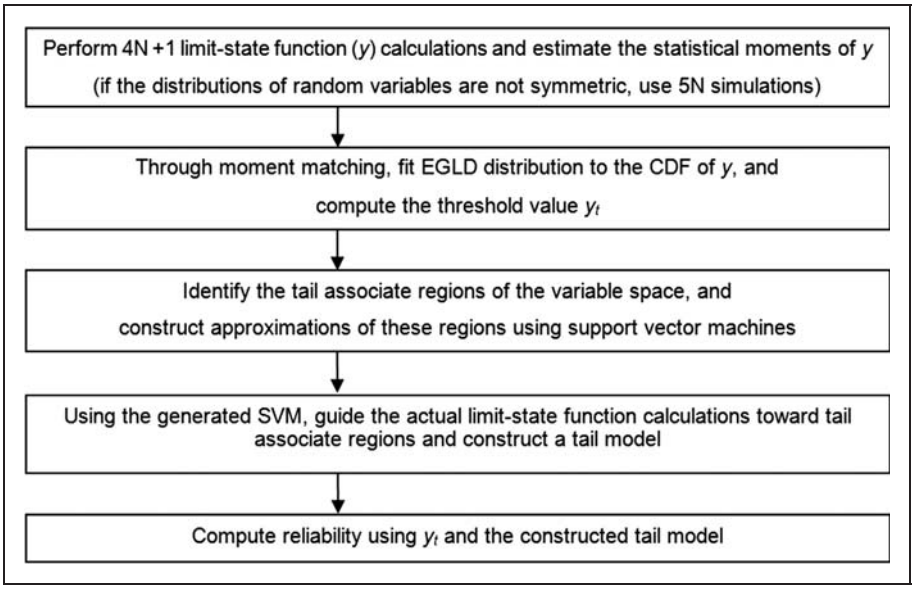

Figure 3. Flowchart for GTM-SVM.
GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

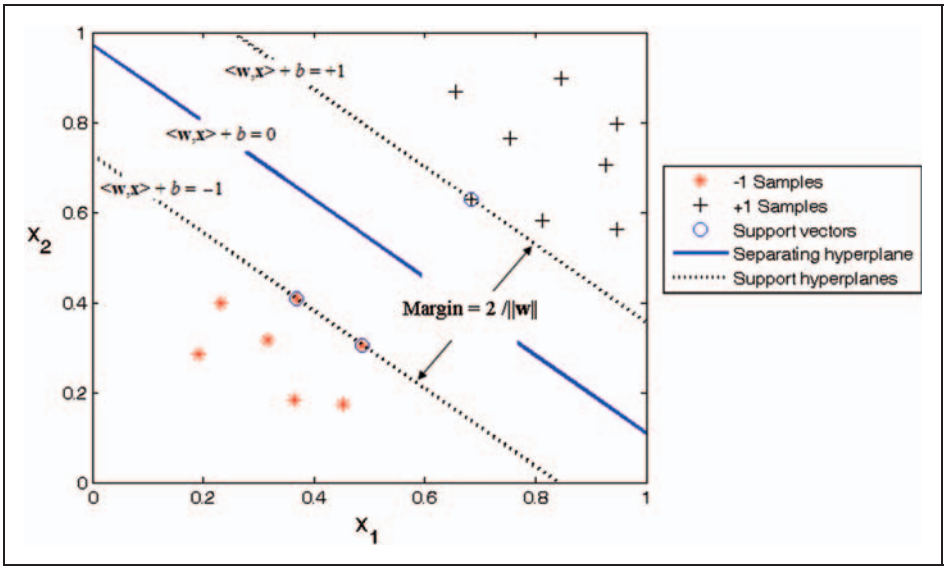

Figure 4. SVM classification example for a linearly separable data set.

originally developed for classification and then extended to regression. When used for classification, the basic idea is to separate the data into classes such that the margin between the classes is maximized.

Consider a data set that can be separated into two classes (see Figure 4). There exists two 'support hyperplanes' separating the data into two classes. Each support hyperplane passes through at least one data point (called 'support vector') of the corresponding class. These support hyperplanes are parallel to each other and are separated by a margin. The basic idea in constructing the support hyperplanes is to maximize the margin between them. The hyperplane that lies halfway between two support hyperplanes is called the 'separating hyperplane'.

If the data classes have values of -1 and $+1$, the separating hyperplane can be defined through

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = 0 \tag{10}$$

where $\mathbf{w}$ is the vector of hyperplane coefficients, $\mathbf{x}$ is the vector of data points, b is the bias and $\langle \mathbf{w}, \mathbf{x} \rangle$ indicates the scalar product of $\mathbf{w}$ and $\mathbf{x}$ vectors. Similarly, the support hyperplanes can be defined through

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = -1 \quad \text{and} \quad \langle \mathbf{w}, \mathbf{x} \rangle + b = +1 \tag{11}$$

The margin between two support hyperplanes is $2/\|\mathbf{w}\|$. Therefore, the hyperplanes can be formed (i.e. the vector of hyperplane coefficients $\mathbf{w}$ and the

bias b can be found) by solving the following optimization problem:

$$\min \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{s.t. } y_i(\langle \mathbf{w}, \mathbf{x}\rangle + b) \geqslant 1 \tag{12}$$

where the constraint reflects the fact that no sample can lie between the support hyperplanes.

The classes can be linearly or nonlinearly separable. If the classes are linearly separable, a hyperplane separating the data into classes can be found in the original variable space. For the case of nonlinearly separable classes, on the other hand, the original variable space is projected to a feature space using a nonlinear Kernel function. In the feature space, the classification problem is similar to that of the linearly separable case. Widely used kernel functions include polynomials, Gaussian functions, Fourier series and splines. In this article, the Gaussian kernel is used following the recommendations of Refs.[18,19] It must be noted that the optimal selection of the kernel function is an active research subject and scope of a future work.

In this article, SVMs are used to separate the data into two classes: (i) the data that belong to the tail region and (ii) the data that do not belong to the tail region. The MATLAB code developed by Gunn[18] is used to develop and implement the SVM models. As noted earlier, Gaussian kernel is used and the kernel width parameter is selected as to minimize the leave-one-out cross-validation misclassification rate. More detailed information on SVMs can be found in Refs.[17–21]

Example problems

To evaluate the performance of the proposed method, GTM-SVM, mathematical example problems as well as structural mechanics problems are used. First, a two-dimensional (or two-variable) mathematical example problem is used to illustrate the approach. Then, example problems with varying dimensions are used to evaluate the performance of GTM-SVM.

An illustrative example (Goldstein–Price function, a two-variable problem)

To illustrate the proposed approach, the well-known Goldstein–Price function is used to define a limit-state function

$$Y = y_{\text{gold}}(x_1, x_2) - y_{\text{crit}} \tag{13}$$

where the Goldstein–Price function is

$$\begin{aligned} y_{\text{gold}}(x_1, x_2) &= \left[1 + (x_1 + x_2 + 1)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right] \\ &\times \left[30 + (2x_1 - 3x_2)^2 \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right] \end{aligned} \tag{14}$$

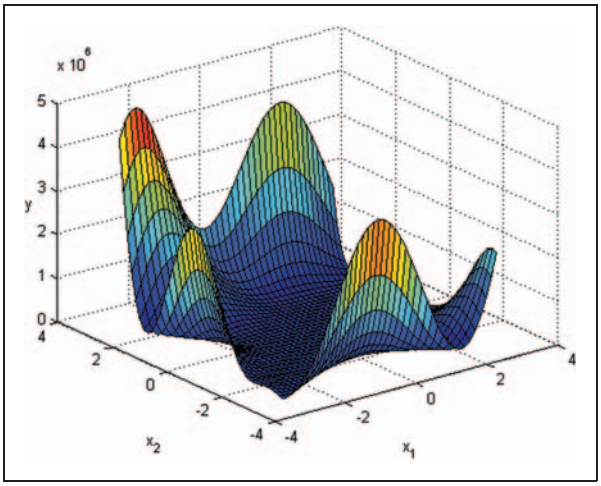

Figure 5. The variation of the Goldstein–Price function over $x_1, x_2 \in [-3, 3]$.

In the Goldstein–Price problem, the variables x_1 and x_2 are taken as random variables following standard normal distributions. Figure 5 shows the behavior of the Goldstein–Price function when the random variables take values $x_1, x_2 \in [-3, 3]$ (i.e. within the range of ± 3 SD away from the mean values). The value of y_{crit} in equation (13) is varied to adjust the reliability level. For instance, $y_{\text{crit}} = 1 \times 10^7$ corresponds to reliability index of 3.25, whereas the use of $y_{\text{crit}} = 8 \times 10^7$ corresponds to reliability index of 4.25. It must be noted here that we model the upper tail, so the positive value of the limit-state function denotes failure, as opposed to the conventional limit-state function setting where the negative value of the limit-state function designates failure.

CTM. The number of simulations is limited to $N = 500$ and $N_t = 50$ (i.e. $F_t = 0.90$) simulations are used to model the tail. In CTM, the first step is to generate N samples of the input random variables from the given distribution types, and perform N limit-state function evaluations. Then, the computed limit-state function values are sorted in ascending order. The last N_t samples in the sorted list are used to fit a tail model. Finally, the fitted model is used to estimate the reliability. For the Goldstein–Price problem, the maximum likelihood method is used to obtain the parameters of GPD, and then the reliability index is computed.

The above procedure is applied to the Goldstein–Price problem. The value of y_{crit} in equation (13) is varied to adjust the reliability level as shown in Table 1. To evaluate the accuracy of CTM, MCS with 10^8 sampling points are performed to provide a basis. To reduce the effect of random sampling, the

whole procedure is repeated 1000 times with different samples, and then the mean absolute error (MAE) is computed. Table 1 shows that the MAE error increases as the reliability level increases.

GTM. GTM is now applied to the Goldstein–Price problem. In GTM, the first step is to compute the statistical moments of the limit-state function. As both x_1 and x_2 follow standard normal distribution (which is a symmetric distribution), $N_{\text{udr}} = 4 \times 2 + 1 = 9$ limit-state function (y) calculations are performed for UDR. Note here that N_{udr} is the number of limit-state function calculations used in UDR. The results corresponding to $y_{\text{crit}} = 1 \times 10^7$ are listed in Table 2.

UDR technique is used to compute the first four statistical moments of the limit-state function. Then, EGLD is fitted to the CDF of y and the threshold value y_t is computed. This prediction is compared with the one calculated through MCS values with 10^8 sampling points. The results presented in Table 3 show that the prediction is acceptable.

Table 1. Accuracy of CTM for reliability index prediction of the Goldstein–Price problem.

y_{crit}	MCS (10^8 simulations)	MAE (%)
1×10^7	3.25	9.3
3×10^7	3.75	12.1
8×10^7	4.25	12.4

MCS: Monte Carlo simulations; MAE: mean absolute error.

Table 2. $4n + 1$ limit-state function evaluations used in UDR.

Simulation no.	x_1	x_2	$Y/10^6$
1	0	0	−9.9994
2	−2.334	0	−9.5754
3	0	−2.334	−9.6783
4	2.334	0	−9.9933
5	0	2.334	−9.5824
6	−0.742	0	−9.9996
7	0	−0.742	−9.9999
8	0.742	0	−9.9990
9	0	0.742	−9.9881

UDR: univariate dimension reduction.

Table 3. Comparison of threshold predictions computed through UDR + EGLD and MCS.

	Threshold, y_t
UDR	-9.76×10^6
MCS	-9.88×10^6

UDR: univariate dimension reduction; EGLD: extended generalized lambda distribution; MCS: Monte Carlo simulation.

The third step of GTM is to fit unidimensional metamodels (i.e. approximate models) using $4 + 1 = 5$ points corresponding to each random variable. Here, polynomial response surface (PRS) approximations are used as metamodels to relate the random variables to the unidimensional functions. The unidimensional PRS for x_1 is constructed using simulation numbers 1–5. The constructed PRS is shown in Figure 6(a). Similarly, the unidimensional PRS for x_2 is constructed using simulation numbers 1 and 6–9. The constructed PRS is depicted in Figure 6(b).

In the fourth step, first a set of candidate sampling points is generated and the corresponding limit-state function values are approximated using the PRSs constructed in the previous step. Then, the candidate points with approximate limit-state function values greater than y_t are stored. Finally, a number of the stored points are selected to be used for tail modeling. For the Goldstein–Price problem, we first generate 10,000 candidate points and calculate the approximate limit-state values for these points. Then, we store 1262 of these points that have approximate limit-state function values greater than y_t. We limit the total number of actual limit-state calculations to $N = 500$. Therefore, we randomly select $N - N_{\text{udr}} = 500 - 9 = 491$ points out of those stored 1262 points. Then, we perform actual limit-state function evaluations for the selected 491 points. We found that only 215 points have the actual limit-state function values greater than y_t. Using these 215 points, a tail modeling is constructed. Maximum likelihood method is used to obtain parameters of GPD.

In the fifth step, the constructed tail model in the previous step is used to calculate the reliability. The accuracy of GTM is evaluated similar to CTM. To reduce the effect of random sampling, the whole procedure is repeated 1000 times with different samples and then the MAE value is computed (Table 4, column 3). The CTM errors in the second column of Table 4 are copied from Table 1. It is seen that the GTM predictions are superior to CTM predictions.

GTM-SVM. The first and the second steps of the GTM-SVM are the same as those of the GTM. The calculation of the statistical moments, fitting probability distribution using EGLD and the computation of the threshold value of the limit-state function, y_t is discussed in the previous section.

In the third step, the tail-associated regions of the variable space are identified and SVMs are constructed to approximate these regions. $N_{\text{svm}} = 100$ training points are generated by constructing a uniform grid over the design space. Note here that N_{svm} is the number of limit-state function calculations used in SVM. The generated 100 points are combined with the nine UDR points to form the overall training data set of 109 points. Figure 7 shows the comparison of the SVM approximation of the tail-associated regions

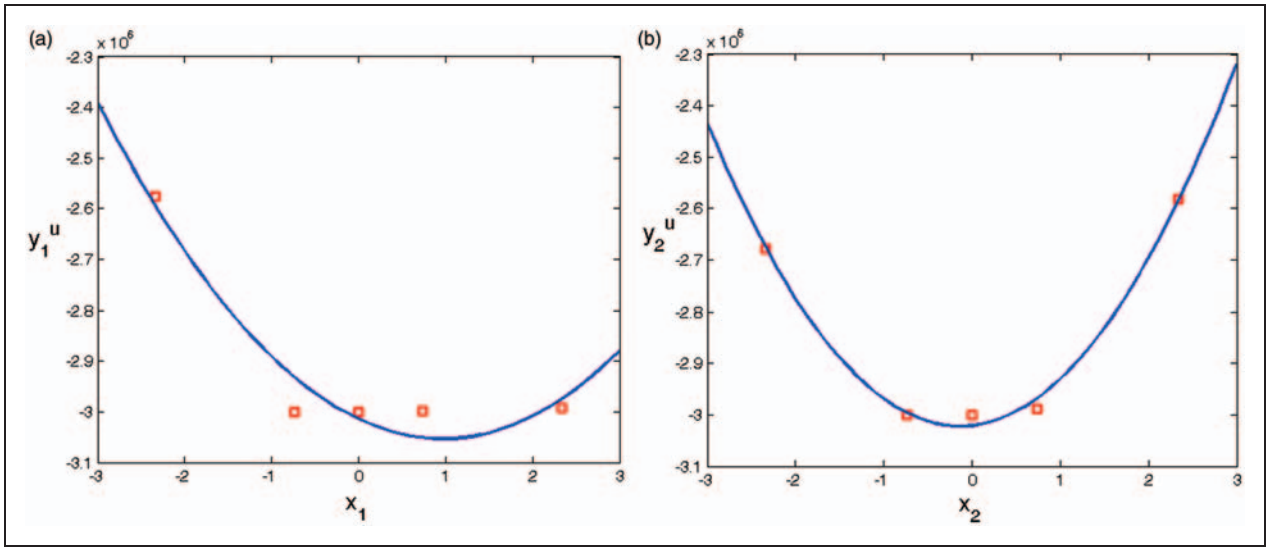

Figure 6. Polynomial response surface approximations generated for the unidimensional functions.

Table 4. Accuracy of GTM for reliability index prediction of the Goldstein–Price problem.

y_{crit}	MAE_CTM (%)	MAE_GTM (%)
1×10^7	9.3	5.0
3×10^7	12.1	6.1
8×10^7	12.4	7.9

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling.

to the actual ones. Figure 7 also shows that 23 out of 109 training points are used as support vectors.

The fourth step is to guide the actual limit-state function calculations by using the generated SVM and perform tail modeling. We first generate 10,000 candidate points and use SVM to classify the ones that belong to the tail region. We find that SVM predicts 981 of these points fall into the tail region. As the total number of actual limit-state function calculations is limited to $N = 500$, we randomly select $N - N_{udr} - N_{svm} = 500 - 9 - 100 = 391$ points from the stored 981 points and perform actual limit-state function evaluations for the selected points. We found that 340 points have the actual limit-state function values greater than y_t. Using these 340 points, tail modeling is performed and the reliability index is estimated. Table 5 shows that the GTM-SVM predictions are better than GTM predictions.

Tuned vibration absorber (a two-variable problem)

The tuned vibration absorber problem is a damped single degree-of-freedom system with dynamic vibration absorber (see Figure 8(a)). This example is taken from a study by Ramu.[10] The original system is externally excited by a harmonic force and the vibration of the system is reduced by the absorber. The amplitude of the vibration depends on the following system parameters: (i) $R = m/M$, the mass ratio of the absorber to the original system, (ii) ζ, the damping ratio of the original system, (iii) $\beta_1 = \omega_{n1}/\omega$, the ratio of the natural frequency of the original system to the excitation frequency and (iv) $\beta_2 = \omega_{n2}/\omega$, the ratio of the natural frequency of the absorber to the excitation frequency.

The limit-state function for this problem can be expressed as

$$Y = y(\beta_1, \beta_2) - y_{crit} \tag{15}$$

where $y(\beta_1, \beta_2)$ is the amplitude of the system normalized by the amplitude of the quasi-static response of the system and this normalized amplitude can be calculated from

$$y(\beta_1, \beta_2) = \frac{\left|1 - \left(\frac{1}{\beta_2}\right)^2\right|}{\sqrt{\left\{\begin{array}{l}\left[1 - R\left(\frac{1}{\beta_1}\right)^2 - \left(\frac{1}{\beta_1}\right)^2 - \left(\frac{1}{\beta_2}\right)^2 + \left(\frac{1}{\beta_1\beta_2}\right)^2\right]^2 \\ \quad + 4\zeta^2\left[\frac{1}{\beta_1} - \frac{1}{\beta_1\beta_2^2}\right]^2\end{array}\right\}}} \tag{16}$$

The random variables of the problem are β_1 and β_2, and they follow normal distribution with mean value of 1 and SD of 0.025. R and ζ are taken as deterministic variables possessing the following values: $R = 0.01$, $\zeta = 0.01$. The normalized amplitude of the original system is plotted in Figure 8(b). The value of y_{crit} in equation (15) is adjusted to obtain various values of reliability indices as listed in Table 6.

In CTM, $N = 500$ and $N_t = 50$ simulations are used to model the tail. The MAE values are provided in the second column of Table 7.

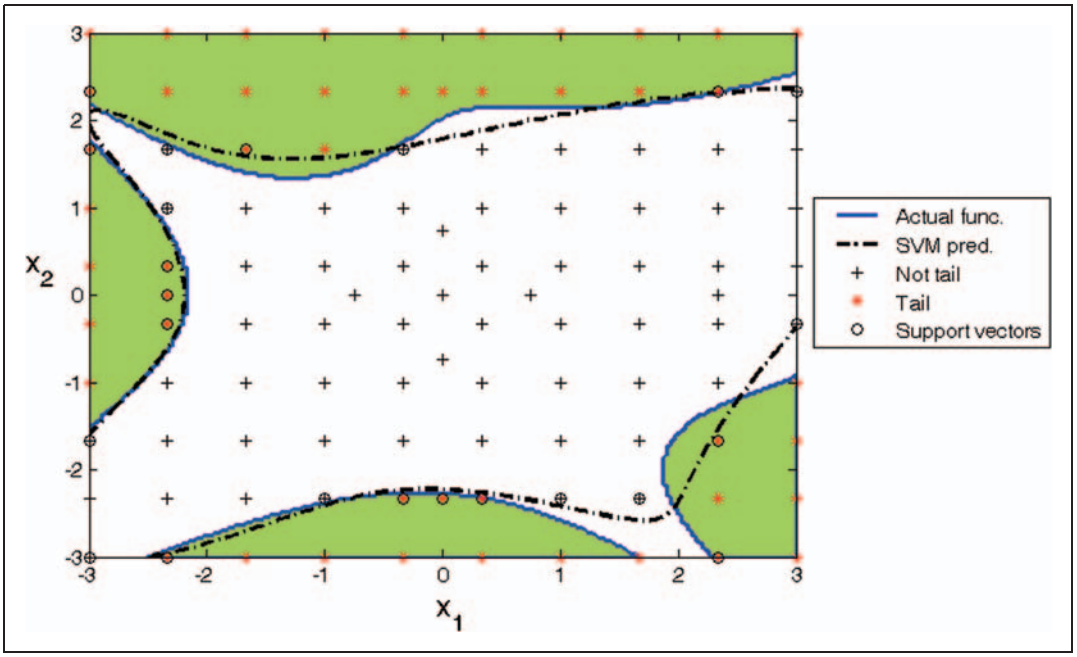

Figure 7. Approximating the tail associated regions of the variable space of Goldstein–Price function using SVM. Red asterisks indicate the points that belong to the tail region and the points that do not belong to the tail region are shown in black '+' sign. The green areas bordered by the blue lines show the actual regions of design space that yield limit-state values in the tail region, whereas the black line is the SVM prediction of these regions.

Table 5. Accuracy of GTM-SVM for reliability index prediction of the Goldstein–Price problem.

y_{crit}	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
1×10^7	9.3	5.0	4.0
3×10^7	12.1	6.1	4.1
8×10^7	12.4	7.9	5.0

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

In GTM, $N_{udr} = 6 \times 2 + 1 = 13$ limit-state evaluations are performed for UDR and the first four statistical moments are calculated. Then, EGLD is used to compute the threshold y_t. Next, the unidimensional metamodels are generated using PRS. Then, 10,000 candidate points are generated, the approximate limit-state values are obtained via UDR and it is found that 802 points are predicted to belong to tail region. Next, we randomly select $N - N_{udr} = 487$ points from the stored points, perform actual limit-state function evaluations and find that 310 points have the actual limit-state function values greater than y_t. Using these 310 points, tail model is constructed and the reliability index is computed. The MAE values are provided in the third column of Table 7. It is seen that GTM predictions are better than CTM predictions.

In GTM-SVM, the UDR points and threshold estimation of GTM are utilized. In addition to the 13 UDR points, $N_{svm} = 100$ more training points are generated by constructing a uniform grid over the design space (see Figure 9). As shown in Figure 9, 23 out of 113 training points are used as support vectors. We first generate 10,000 candidate points, use SVM to classify that 563 of these points fall into the tail region. Then, we randomly select $N - N_{udr} - N_{svm} = 387$ points from the stored points, perform actual limit-state function evaluations and find that 357 points have the actual limit-state function values greater than y_t. Using these 357 points, tail model is constructed and the reliability index is estimated. The MAE values are provided in the fourth column of Table 7. We see that GTM-SVM predictions are better than GTM predictions, which are better than CTM predictions.

Short column design (a three-variable problem)

A short column subjected to a normal force P and biaxial bending moments M_x and M_y is considered (see Figure 10). The cross-section is rectangular with dimensions b and h. Using the elastic–plastic constitutive law, the limit-state function can be written as

$$y(b, h, \sigma_Y) = 1 - \frac{4M_x}{bh^2\sigma_Y} - \frac{4M_y}{b^2h\sigma_Y} - \frac{P^2}{(bh\sigma_Y)^2} \qquad (17)$$

where the negative value of this function designates failure. The loading is assumed to be deterministic with the following values: $P = 140\,\text{kN}$, $M_x = M_y = 14\,\text{kN*m}$. The cross-section dimensions and the yield stress are taken as random variables and their statistical properties are provided in Table 8.

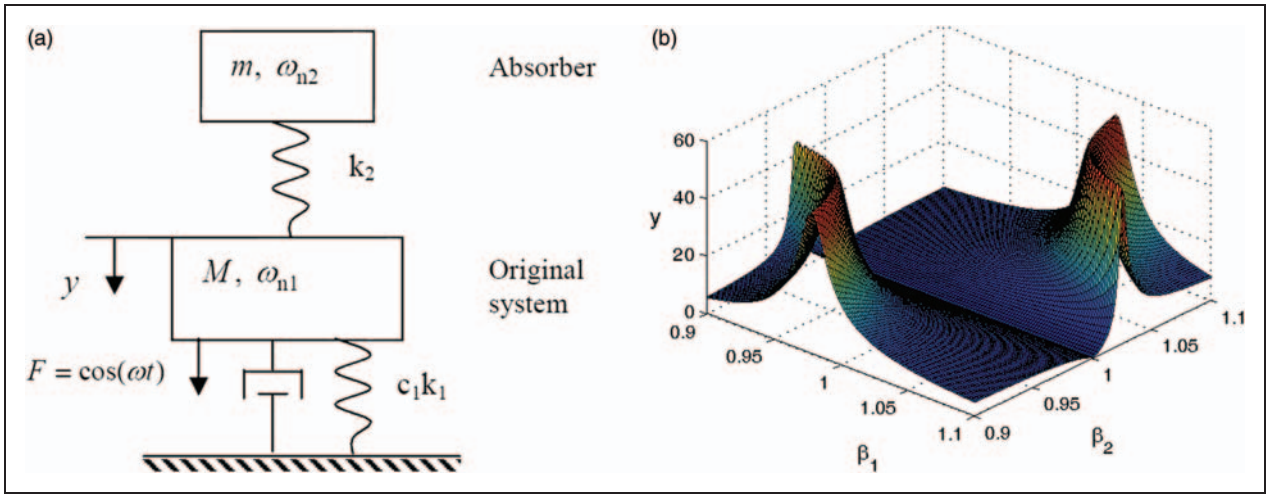

Figure 8. (a) Tuned vibration absorber and (b) the normalized amplitude of the vibration absorber.

Table 6. Various reliability levels for the tuned vibration absorber problem.

y_{crit}	Relativity index
27	2.29
48	3.03
53	3.86

Note: The reliability index values are computed through MCS with 10^8 samples.

Table 7. Accuracy of GTM-SVM for reliability index prediction of the tuned vibration absorber problem.

y_{crit}	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
27	6.5	7.5	6.2
48	8.6	3.6	2.6
53	16.9	10.4	6.8

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

To obtain varying reliability levels, the limit-state function is formulated as

$$Y = -y(\beta_1, \beta_2) - y_{\text{crit}} \tag{18}$$

where y_{crit} term in equation (18) is adjusted to obtain various values of reliability indices as listed in Table 9.

In CTM, $N = 500$, and $N_t = 50$ simulations are used to model the tail. The MAE values are provided in the second column of Table 10.

In GTM, $N_{\text{udr}} = 4 \times 3 + 1 = 13$ limit-state evaluations are performed for UDR and the first four statistical moments are calculated. Then, EGLD is used to compute the threshold y_t. Next, the unidimensional metamodels are generated using PRS. Then, 10,000 candidate points are generated, the approximate limit-state values are obtained via UDR and it is found that 1585 points are predicted to belong to tail region. Next, we randomly select $N - N_{\text{udr}} = 487$ points from the stored points, perform actual limit-state function evaluations and find that 263 points have the actual limit-state function values greater than y_t. Using these 263 points, tail model is constructed and the reliability index is computed. The MAE values are provided in the third column of Table 10. It is seen that GTM predictions are better than CTM predictions.

In GTM-SVM, the UDR points and threshold estimation of GTM are utilized. In addition to the 13 UDR points, $N_{\text{svm}} = 4^3 = 64$ more training points are generated by constructing a uniform grid over the design space (four divisions in each dimension). Using the $64 + 13 = 77$ training points, SVM is constructed. The graphical depiction of the SVM is not provided in the article. We first generate 10,000 candidate points, use SVM to classify that 589 of these points fall into the tail region. Then, we randomly select $N - N_{\text{udr}} - N_{\text{svm}} = 423$ points from the stored points, perform actual limit-state function evaluations and find that 363 points have the actual limit-state function values greater than y_t. Using these 363 points, tail model is constructed and the reliability index is estimated. The MAE values are provided in the fourth column of Table 10. We see that GTM-SVM predictions are better than GTM predictions, which are better than CTM predictions.

A four-variable highly nonlinear problem

The number of variables is now increased to four, and a four-variable highly nonlinear problem is considered. The limit-state function has the following form

$$Y = \frac{1}{x_1 x_2} - \log\left(\left|\frac{1}{x_3 x_4}\right|\right) - y_{\text{crit}} \tag{19}$$

The variables x_1–x_4 are assumed to be random with the statistical properties given in Table 11. The y_{crit}

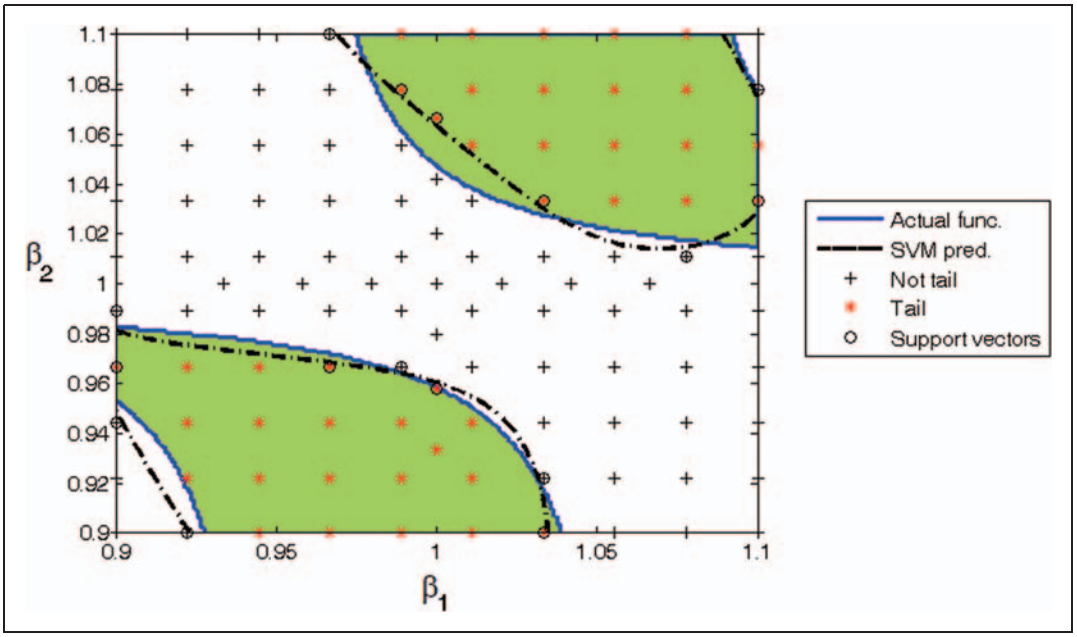

Figure 9. Support vector machines constructed for the tuned vibration absorber.

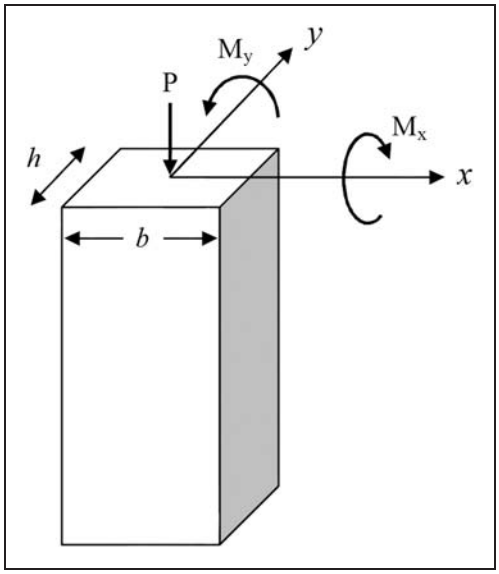

Figure 10. Short column subjected to a normal force and biaxial bending moments.

Table 8. The random variables of the short column design problem.

Random variable	Distribution	Mean	SD
b (m)	Normal	0.4	0.1
h (m)	Normal	0.4	0.1
σ_Y (MPa)	Normal	40	4

term in equation (19) is adjusted to obtain various values of reliability indices as listed in Table 12.

In CTM, $N = 500$, and $N_t = 50$ simulations are used to model the tail. The MAE values are provided in the second column of Table 13.

Table 9. Various reliability levels for the short column design problem.

y_{crit}	Relativity index
0	2.76
5	3.27
100	3.66

Note: The reliability index values are computed through MCS with 10^8 samples.

Table 10. Accuracy of GTM-SVM for reliability index prediction for the short column design problem.

y_{crit}	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
0	9.8	4.1	3.8
5	18.6	6.1	5.1
100	33.3	17.7	14.1

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

Table 11. The random variables of the four-variable highly nonlinear problem.

Random variable	Distribution	Mean	SD
x_1 and x_2	Normal	2	0.65
x_3 and x_4	Normal	10	3

In GTM, $N_{udr} = 4 \times 4 + 1 = 17$ limit-state evaluations are performed for UDR and the first four statistical moments are calculated. Then, EGLD is used to compute the threshold y_t. Next, the unidimensional

metamodels are generated using PRS. Then, 10,000 candidate points are generated, the approximate limit-state values are obtained via UDR and it is found that 1344 points are predicted to belong to tail region. Next, we randomly select $N-N_{\text{udr}}=483$ points from the stored points, perform actual limit-state function evaluations and find that 300 points have the actual limit-state function values greater than y_t. Using these 300 points, tail model is constructed and the reliability index is computed. The MAE values are provided in the third column of Table 13. It is seen that GTM predictions are better than CTM predictions.

In GTM-SVM, the UDR points and threshold estimation of GTM are utilized. In addition to the 17 UDR points, $N_{\text{svm}}=40$ more training points are generated through random sampling. Using the $40+17=57$ training points, SVM is constructed. The graphical depiction of the SVM is not provided in the article. We first generate 10,000 candidate points and use SVM to classify that 934 of these points fall into the tail region. Then, we randomly select $N-N_{\text{udr}}-N_{\text{svm}}=443$ points from the stored points, perform actual limit-state function evaluations and find that 301 points have the actual limit-state function values greater than y_t. Using these 301 points, tail model is constructed and the reliability index is estimated. The MAE values are provided in the fourth column of Table 13. We see that GTM-SVM predictions are slightly worse than GTM predictions, whereas GTM–SVM predictions are better than CTM predictions.

Table 12. Various reliability levels for the four-variable highly nonlinear problem.

y_{crit}	Relativity index
8	2.94
16	3.68
64	4.62

Note: The reliability index values are computed through MCS with 10^8 samples.

Table 13. Accuracy of GTM-SVM for reliability index prediction for the four-variable highly nonlinear problem.

y_{crit}	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
8	11.0	3.3	3.4
16	17.2	9.4	10.2
64	26.8	15.8	17.3

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

Propped cantilever beam problem (a seven-variable problem)

In this example, a propped cantilever beam (see Figure 11) under triangular distributed load is examined. The limit-state function for this problem is formulated as the difference between the maximum allowable deflection v_{crit} and the maximum deflection of the beam $v_{\max}$ due to the applied triangular distributed load as given by

$$Y = v_{\max} - v_{\text{crit}} \tag{20}$$

The deflection of the beam (positive downwards) at any location x can be found from

$$v(x) = \frac{q_0 x^2}{120LEI}\left(4L^3 - 8L^2x + 5Lx^2 - x^3\right); \quad I = \frac{b_f d^3 - (b_f - t_w)(d - 2t_f)^3}{12} \tag{21}$$

It can be found that the maximum deflection occurs at $x=0.5528\,L$ (i.e. $v_{\max}=v(0.5528\,L)$). In this problem, there are seven random variables following normal distribution with mean and SD values provided in Table 14. The maximum allowable deflection v_{crit} is altered within 4 –5 mm to attain reliability index values as listed in Table 15.

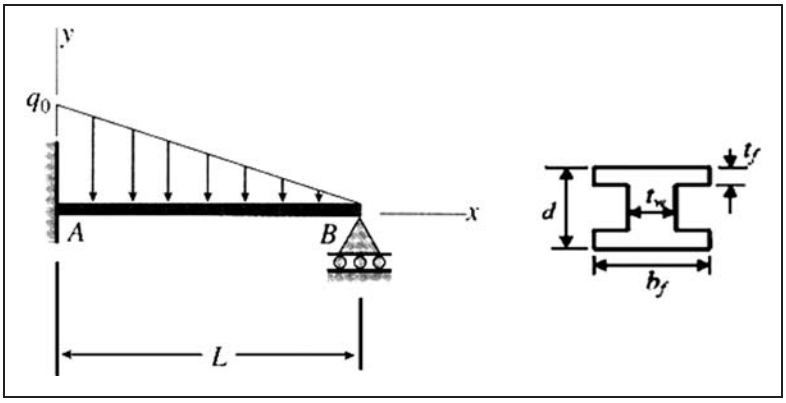

Figure 11. The cross-section and triangular distributed loading on the propped cantilever beam.

Table 14. Mean and SD values of the random variables for the propped cantilever beam problem.

Random variable	Mean (SD)
q_0 (kN/m)	20 (2)
L (m)	6 (0.3)
E (GPa)	210 (10)
d (cm)	25 (0.5)
b_f (cm)	25 (0.5)
t_w (cm)	2 (0.2)
t_f (cm)	2 (0.2)

Note: All random variables follow normal distribution.

Table 15. Various reliability levels for the propped cantilever beam problem.

v_{crit} (mm)	Relativity index
4.0	2.98
4.5	3.50
5.0	3.97

Note: The reliability index values are computed through MCS with 10^8 samples.

Table 16. Accuracy of GTM-SVM for reliability index prediction for the propped cantilever beam problem.

v_{crit} (mm)	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
4.0	11.4	3.6	5.2
4.5	14.8	7.2	8.1
5.0	18.6	13.2	13.3

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.

In CTM, $N = 500$ and $N_t = 50$ simulations are used to model the tail. The MAE values are provided in the second column of Table 16.

In GTM, $N_{udr} = 4 \times 7 + 1 = 29$ limit-state evaluations are performed for UDR and the first four statistical moments are calculated. Then, EGLD is used to compute the threshold y_t. Next, the unidimensional metamodels are generated using PRS. Then, 10,000 candidate points are generated, the approximate limit-state values are obtained via UDR and it is found that 1386 points are predicted to belong to the tail region. Next, we randomly select $N - N_{udr} = 471$ points from the stored points, perform actual limit-state function evaluations and find that 292 points have the actual limit-state function values greater than y_t. Using these 292 points, tail model is constructed and the reliability index is computed. The MAE values are provided in the third column of Table 16. It is seen that GTM predictions are better than CTM predictions.

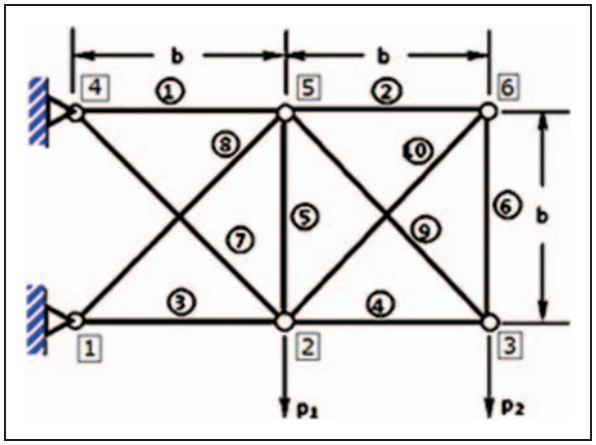

Figure 12. Loading and boundary conditions for the ten-bar truss problem.

In GTM-SVM, the UDR points and threshold estimation of GTM are utilized. In addition to the 29 UDR points, $N_{svm} = 41$ more training points are generated by constructing a uniform grid over the design space (four divisions in each dimension). Using the $41 + 29 = 70$ training points (10 times the number of variables), SVM is constructed. The graphical depiction of the SVM is not provided in the article. We first generate 10,000 candidate points, use SVM to classify that 1898 of these points fall into the tail region. Then, we randomly select $N - N_{udr} - N_{svm} = 431$ points from the stored points, perform actual limit-state function evaluations, and find that 190 points have the actual limit-state function values greater than y_t. Using these 190 points, tail model is constructed and the reliability index is estimated. The MAE values are provided in the fourth column of Table 16. We see that GTM-SVM predictions are slightly worse than GTM predictions, whereas GTM-SVM predictions are better than CTM predictions.

Ten-bar truss problem (a 10-variable problem)

The ten-bar truss structure in Figure 12 is made of aluminum with a weight density of $\rho = 2768\,\text{kg/m}^3$ ($\rho = 0.1\,\text{lb/in.}^3$), an elasticity modulus of $E = 70\,\text{GPa}$ ($E = 10^4\,\text{ksi}$) and the bay length of $b = 9.144\,\text{m}$ ($b = 360\,\text{in}$). The joints 2 and 3 are subjected to vertical loads $P_1 = P_2 = 444.8\,\text{kN}$ ($P_1 = P_2 = 100\,\text{kips}$). The design requirement is that the maximum deflection at node 3 should not exceed a critical value. The limit-state function for this problem can be written as

$$Y = v_3(\mathbf{A}) - v_{crit} \tag{22}$$

where $\mathbf{A}$ is vector of the cross-section areas of the truss members. All cross-section areas are taken as random variables following normal distribution with a mean value of $16.13\,\text{cm}^2$ ($2.5\,\text{in.}^2$) and the SD of $3.226\,\text{cm}^2$ ($0.5\,\text{in.}^2$). To calculate deflection at node 3, first the member forces are calculated as explained

Table 17. Various reliability level values for the ten-bar truss problem.

v_{crit} (cm)	Relativity index
55.88	2.58
63.50	3.24
76.20	3.88

Table 18. Accuracy of GTM-SVM for reliability index prediction for the ten-bar truss problem.

v_{crit} (cm)	MAE_CTM (%)	MAE_GTM (%)	MAE_GTM-SVM (%)
55.88	7.8	2.2	3.3
63.50	15.0	4.8	6.9
76.20	21.2	15.4	17.8

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine.
Note: The number of training points including the UDR points is 100.

by Kumar et al.,[22] then the deflection can be calculated using energy methods, such as the method of virtual forces. The value of y_{crit} in equation (22) is adjusted to obtain various values of reliability indices as listed in Table 17.

In CTM, $N = 500$, and $N_t = 50$ simulations are used to model the tail. The MAE values are provided in the second column of Table 18.

In GTM, $N_{udr} = 4 \times 10 + 1 = 41$ limit-state evaluations are performed for UDR and the first four statistical moments are calculated. Then, EGLD is used to compute the threshold y_t. Next, the unidimensional metamodels are generated using PRS. Then, 10,000 candidate points are generated, the approximate limit-state values are obtained via UDR and it is found that 1137 points are predicted to belong to the tail region. Next, we randomly select $N - N_{udr} = 459$ points from the stored points, perform actual limit-state function evaluations and find that 395 points have the actual limit-state function values greater than y_t. Using these 395 points, tail model is constructed and the reliability index is computed. The MAE values are provided in the third column of Table 18. It is seen that GTM predictions are better than CTM predictions.

In GTM-SVM, the UDR points and threshold estimation of GTM are utilized. In addition to the 41 UDR points, $N_{svm} = 59$ more training points are generated through random sampling. Using the $59 + 41 = 100$ training points (10 times the number of variables), SVM is constructed. The graphical depiction of the SVM is not provided in the article. We first generate 10,000 candidate points, use SVM to classify that 1640 of these points fall into the tail region. Then, we randomly select $N - N_{udr} - N_{svm} = 400$ points from the stored points, perform actual limit-state function evaluations and find that 222 points have the actual limit-state function values greater than y_t. Using these 222 points, tail model is constructed and the reliability index is estimated. The MAE values are provided in the fourth column of Table 18. We see that GTM-SVM predictions are worse than GTM predictions, whereas GTM-SVM predictions are better than CTM predictions.

Summary of results

The mean absolute errors obtained for all example problems are summarized in this section. The MAE values are normalized with the MAE of CTM and the results are presented in Table 19. It is seen that for two- and three-variable problems, the performance of GTM-SVM is better than both GTM and CTM. For four-variable problems, the performance of GTM-SVM is slightly worse than GTM, but better than CTM. For 7- and 10-variable problems, GTM-SVM is definitely worse than GTM, but better than CTM. Overall, it is observed that the performance of GTM-SVM reduces as the number of variables in the problem increases.

Table 19 also shows for most example problems that the ratio of the errors of the GTM and the error of the CTM increases as the reliability level increases. It basically indicates that the effectiveness of the GTM over the CTM reduces as the reliability level increases.

For low-dimension problems, one may argue that an accurate response surface model (RSM) can be constructed with $N = 500$ simulations and it can be used within an MCS procedure to get more accurate reliability predictions. Here, Kriging models are used as the RSMs. Table 20 presents the comparison of reliability prediction accuracies of GTM-SVM and the use of RSM within MCS for low-dimension problems. For Goldstein–Price function, the use of RSM within MCS leads to better reliability predictions than the proposed method. For tuned vibration absorber problem and the short column design problem, the use of RSM within MCS leads to better reliability predictions at relatively smaller reliabilities, whereas the proposed method performs better at relatively higher reliabilities. Amplification of errors in reliability estimations due to small errors in RSMs at high reliabilities was also reported in the literature by Ramu et al.[23]

Concluding remarks

A new GTM, a technique that utilizes SVMs, to reduce the amount of discarded data was proposed in this article. The proposed method, GTM-SVM, is based on guiding the limit-state function calculations toward the tail region using SVM. The guidance was

Table 19. MAEs for all example problems normalized with the corresponding MAE of CTM.

Example problem	n_{var}	Relativity index MCS(10^8)	NMAE CTM	NMAE GTM	NMAE GTM-SVM
Goldstein–Price	2	3.25	1.0	0.54	0.43
		3.75	1.0	0.50	0.34
		4.25	1.0	0.64	0.40
Tuned vibration absorber	2	2.29	1.0	1.15	0.95
		3.03	1.0	0.42	0.30
		3.86	1.0	0.62	0.40
Short column design	3	2.76	1.0	0.42	0.39
		3.27	1.0	0.33	0.27
		3.66	1.0	0.53	0.42
A four-variable highly nonlinear problem	4	2.94	1.0	0.30	0.31
		3.68	1.0	0.55	0.59
		4.62	1.0	0.59	0.65
Propped cantilever beam	7	2.98	1.0	0.32	0.46
		3.50	1.0	0.49	0.55
		3.97	1.0	0.71	0.72
Ten-bar truss	10	2.58	1.0	0.28	0.42
		3.24	1.0	0.32	0.46
		3.88	1.0	0.73	0.84

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine; NMAE: mean absolute error normalized with the mean absolute error of CTM.

Table 20. Reliability prediction accuracies of GTM-SVM and the use of response surface model within Monte Carlo framework for low dimension problems.

Example problem	n_{var}	Relativity index MCS(10^8)	% MAE* GTM_SVM	% MAE RSM
Goldstein–Price	2	3.25	4.0	0.1
		3.75	4.1	0.2
		4.25	5.0	0.6
Tuned vibration absorber	2	2.29	6.2	0.7
		3.03	2.1	2.2
		3.86	6.8	7.3
Short column design	3	2.76	3.8	0.6
		3.27	5.1	6.1
		3.66	14.1	25.5

MAE: mean absolute error; GTM: guided tail modeling; CTM: classical tail modeling; SVM: support vector machine; RSM: response surface model.

achieved through a procedure based on threshold estimation using univariate dimension reduction and extended generalized lambda distribution and tail region approximation using SVM.

The performance of GTM-SVM was tested with mathematical as well as structural mechanics example problems with varying number of variables ranging from 2 to 10. The performance of GTM-SVM is compared with the performance of the original GTM and CTM. From the results obtained from these examples, the following were observed:

- For two- and three-variable problems, the performance of GTM-SVM was better than GTM as well as CTM. For the four-variable problem, the performance of GTM-SVM was slightly worse than GTM, but better than CTM. For 7- and 10-variable problems, GTM-SVM was definitely worse than GTM, but better than CTM. Based on these findings, we can conclude that the GTM-SVM is more suitable for low-dimensional problems.
- It was found that the ratio of the errors of the GTM and the GTM-SVM to the error of CTM increased as the reliability level increased. That is, the effectiveness of the GTM over the CTM was reduced as the reliability level increased.
- For low-dimension problems, constructing an accurate RSM and using it within an MCS procedure provided a good alternative to the proposed method. It was found that this alternative approach performed better for relatively smaller reliabilities, whereas the proposed method performed better at relatively higher reliabilities.

An important conclusion of this article is that the original GTM is suitable for high-dimension problems (typical for real-life engineering problems), while the GTM-SVM is more suitable for low-dimension problems with two or three variables (typical for idealized engineering problems).

Funding

This work was supported by The Scientific and Technological Research Council of Turkey TÜBİTAK [grant number MAG-109M537].

Conflict of interest

None declared.

References

1. Liu JS. *Monte Carlo strategies in scientific computing*. New York: Springer-Verlag, 2001.
2. Melchers RE. Importance sampling in structural systems. *Struct Saf* 1989; 6: 3–10.
3. Wu YT. Computational methods for efficient structural reliability and reliability sensitivity analysis. *AIAA J* 1994; 32: 1717–1723.
4. Nie J and Ellingwood BR. Directional methods for structural reliability analysis. *Structural Saf* 2000; 22: 233–249.
5. Boos D. Using extreme value theory to estimate large percentiles. *Technometrics* 1984; 26: 33–39.
6. Hasofer A. Non-parametric estimation of failure probabilities. In: F Casciati and B Roberts (eds) *Mathematical models for structural reliability*. Boca Raton, FL: CRC Press, pp.195–226.
7. Caers J and Maes M. Identifying tails, bounds, and endpoints of random variables. *Struct Saf* 1998; 20: 1–23.
8. Kim NH, Ramu P and Queipo NV. Tail modeling in reliability-based design optimization for highly safe structural systems. In: *47th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference*, Newport, RI, 2006, paper no. 1825, Newport, RI: AIAA.
9. Ramu P, Kim NH, Haftka RT, et al. System reliability analysis and optimization using tail modeling. In: *11th AIAA/ISSMO multidisciplinary analysis and optimization conference*, Portsmouth, VA, 2006, paper no. 7012, Portsmouth, VA: AIAA.
10. Ramu P. *Multiple tail models including inverse measures for structural design under uncertainties*. PhD Thesis, University of Florida, Gainesville, FL, 2007.
11. Mourelatos ZP, Song J and Nikolaidis E. Reliability estimation of large-scale dynamic systems by using re-analysis and tail modeling. In: *SAE world congress & exhibition*, Detroit, MI, April 2009, Paper no. 2009-01-0200.
12. Acar E. Guided tail modeling for efficient and accurate reliability estimation of highly safe mechanical systems. *Proc IMechE, Part C, J Mechanical Engineering Science* 2011; 225: 1237–1251.
13. Rahman S and Xu H. A univariate dimension-reduction method for multi-dimensional integration in stochastic mechanics. *Probab Eng Mech* 2004; 19: 393–408.
14. Karian ZE, Dudewicz EJ and McDonald P. The extended generalized lambda distribution system for fitting distributions to data: history, completion of theory, tables, applications, the ''final word'' on moment fits. *Commun Stat Simul* 1996; 25: 611–642.
15. Fournier B, Rupin N, Bigerelle M, et al. Estimating the parameters of a generalized lambda distribution. *Comput Stat Data Anal* 2007; 51: 2813–2835.
16. Acar E, Rais-Rohani M and Eamon C. Reliability estimation using univariate dimension reduction and extended generalized lambda distribution. *Int J Reliab Saf* 2010; 24: 166–187.
17. Vapnik V. *The nature of statistical learning theory*. New York: Springer-Verlag, 2000.
18. Gunn SR. Support vector machines for classification and regression. Technical Report, University of Southampton, UK, 1997.
19. Wang WJ, Xu ZB, Lu WZ, et al. Determination of the spread parameter in the Gaussian kernel for classification and regression. *Neurocomputing* 2003; 55: 643–663.
20. Cristianini N and Shawe-Taylor J. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge, UK: Cambridge University Press.
21. Abe S. *Support vector machines for pattern classification*. London: Springer.
22. Kumar S, Pippy RJ, Acar E, et al. Approximate probabilistic optimization using exact-capacity-approximate-response-distribution (ECARD). *Struct Multidiscip O* 2009; 38: 613–626.
23. Ramu P, Kim NH and Haftka RT. *Error amplification in failure probability estimates of small errors in surrogates*. In: *SAE 2007 world congress*, Detroit, MI, April 16–19, Paper No. 2007-01-0549.