Reduction of Random Variables in Structural Reliability Analysis

S. Adhikari and R. S. Langley
Department of Engineering
University of Cambridge
Trumpington Street
Cambridge CB2 1PZ (U.K.)

February 21, 2002

Abstract

The aim of the present report is to propose various methods whereby the number of random variables can be reduced without compromising the accuracy of the reliability calculation.

Contents

1 Introduction

In the reliability analysis of a complex engineering structure a very large number of the system parameters can be considered to be random variables. The difficulty in computing the failure probability increases rapidly with the number of variables, and the aim of the present report is to consider methods whereby the number of variables can be reduced without compromising the accuracy of the reliability calculation. The most common methods of reliability prediction are (see, for example, the books by Thoft-Christensen and Baker, 1982, Augusti *et al.*, 1984, Tichy, 1993, Ditlevsen and Madsen, 1996, Melchers, 1999) (a) FORM (First Order Reliability Method), (b) SORM (Second Order Reliability Method), and (c) asymptotic reliability analysis. In each case the efficiency of the method depends on the efficient computation of the so called *reliability index*, commonly denoted by 'β'. The geometric interpretation of the reliability index is that it is the minimum distance of the failure surface from the origin in a transformed space (referred as the **z**-space) where all the random variables defining the uncertainties of the system are Gaussian, normalized and uncorrelated. The present reduction methods are based on the sensitivity of the failure surface in the **z**-space. If the failure surface is close to linear, then the values of β obtained from these methods are close to the exact value of β obtained using the full set of random variables. However, if the failure surface is significantly nonlinear, the different reduction methods introduce different kind of errors. The nature of these errors are studied using a wide range of numerical examples. It is shown that the values of β obtained using the proposed reduction methods have acceptable accuracy for many large scale structural engineering problems.

2 Approximate Reliability Analysis Methods

Suppose the random variables describing the uncertainties of the structure and loading are considered to form a vector $\mathbf{y} \in \mathbb{R}^n$. The statistics of the system are fully described by the joint probability density function $p(\mathbf{y})$. In general the random variables $\mathbf{y}$ are non-Gaussian. In principle these random variables can be transformed to a set of uncorrelated Gaussian random variables via the Rosenblatt transformation (Rosenblatt, 1952). Further, they can be scaled so that each random varibles has zero mean and unit variance. Suppose these transformed and scaled random variables are $\mathbf{x} \in \mathbb{R}^n$ with a joint probability density function $p(\mathbf{x})$. For a given set of variables $\mathbf{x}$ the structure will either fail under the applied (random) loading or will be safe. The condition of the structure for every $\mathbf{x}$ can be described by a safety margin $g(\mathbf{x})$ so the structure has failed if $g(\mathbf{x}) \leq 0$ and is safe if $g(\mathbf{x}) > 0$. Thus the probability of failure is given by

$$P_f = \int_{g(\mathbf{x})\leq 0} p(\mathbf{x}) d\mathbf{x}. \tag{1}$$

The function $g(\mathbf{x})$ is also known as failure surface or limit-state function. The central theme of a reliability analysis is to evaluate the multidimensional integral (1). For most real-life cases the dimentionality of the integral is large and consequently the exact evaluation of the integral in equation (1) is not possible. For this reason some kind of approximate method is required to evaluate this integral. Using the first-order reliability method (FORM) the probability of failure is given by

$$P_f = \Phi(-\beta) \quad \text{with} \quad \beta = (\mathbf{x}^{*^T}\mathbf{x}^*)^{1/2} \tag{2}$$

where $\mathbf{x}^*$, the 'design point' is the solution of following optimization problem

$$\min \beta = (\mathbf{x}^T\mathbf{x})^{1/2} \quad \text{subject to} \quad g(\mathbf{x}) = 0. \tag{3}$$

Once the reliability index or β is known other more accurate approximate reliability analyses, for example, second-order reliability method (SORM) or asymptotic reliability method, can be performed. Thus the calculation of the design point and the reliability index is very crucial for all approximate reliability methods. Purpose of this article is to propose few methods for efficient calculation of $\mathbf{x}^*$ and β. Proposed methods are based on reduction of random variables.

3 Method 1: Gradient Projection Method

For some point $\mathbf{x}$ in $\mathbb{R}^n$ we can have the first-order Taylor series expansion of $g(\mathbf{x})$ about $\mathbf{x}^*$

$$g(\mathbf{x}) \approx g(\mathbf{x}^*) + (\mathbf{x} - \mathbf{x}^*)^T \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}^*} \tag{4}$$

If $g(\mathbf{x})$ is linear then $\dfrac{\partial g(\mathbf{x})}{\partial \mathbf{x}}$ is independent of $\mathbf{x}$. In this case $\mathbf{x}^*$ will be simply the projection of the origin ($\mathbf{x} = \mathbf{0}$) on $g(\mathbf{x}) = 0$ (see Melchers, 1999, Chapter 3). Also note that the outward normal vector to the hypersurface $g(\mathbf{x}) = 0$ is

$$c_i = \lambda \frac{\partial g(\mathbf{x})}{\partial x_i}|_{\mathbf{x}=\mathbf{0}}. \tag{5}$$

Assume that $\boldsymbol{\nabla} g = \left\{ \dfrac{\partial g(\mathbf{x})}{\partial x_i} \right\} \in \mathbb{R}^n$ is normalized so that $\boldsymbol{\nabla} g^T \boldsymbol{\nabla} g = 1$. Then for linear $g(\mathbf{x})$ it can be shown that

$$\mathbf{x}^* = -\beta \boldsymbol{\nabla} g. \tag{6}$$

Motivated by this, we express $\mathbf{x}$ by

$$\mathbf{x} = v \boldsymbol{\nabla} g \tag{7}$$

where $v \in \mathbb{R}$ is a new random variable. In view of (7), the constrained optimization problem (3) becomes a simple search problem, that is we need to solve for v such that $g(v\boldsymbol{\nabla} g) = 0$ or $g'(v) = 0$. Here

$g'(\bullet) = g(\bullet\nabla g)$ is a (non-linear) function of a *single* variable v. Comparing (6) and (7) it is clear that $v = -\beta$.

This method yields accurate result when $g(\mathbf{x})$ is linear or very close to linear. This is because for linear $g(\mathbf{x})$, ∇g is independent of the choice of $\mathbf{x}$ so that the direction of the outward normal from the failure surface does not change with position along the failure surface. For this reason $\nabla g|_{\mathbf{x}=\mathbf{0}}$ becomes the unit vector along true $\mathbf{x}^*$ and consequently β becomes simply the 'length' of this vector from the origin to the failure surface. Figure 1 shows a linear failure surface in a two dimensional space $\mathbb{R}^2$. The unit vector $\nabla g|_{\mathbf{x}=\mathbf{0}}$ is shown by the arrow. The point $\mathbf{x}^*$ is obtained by projecting this unit vector to the failure surface. The length of this projection is β.

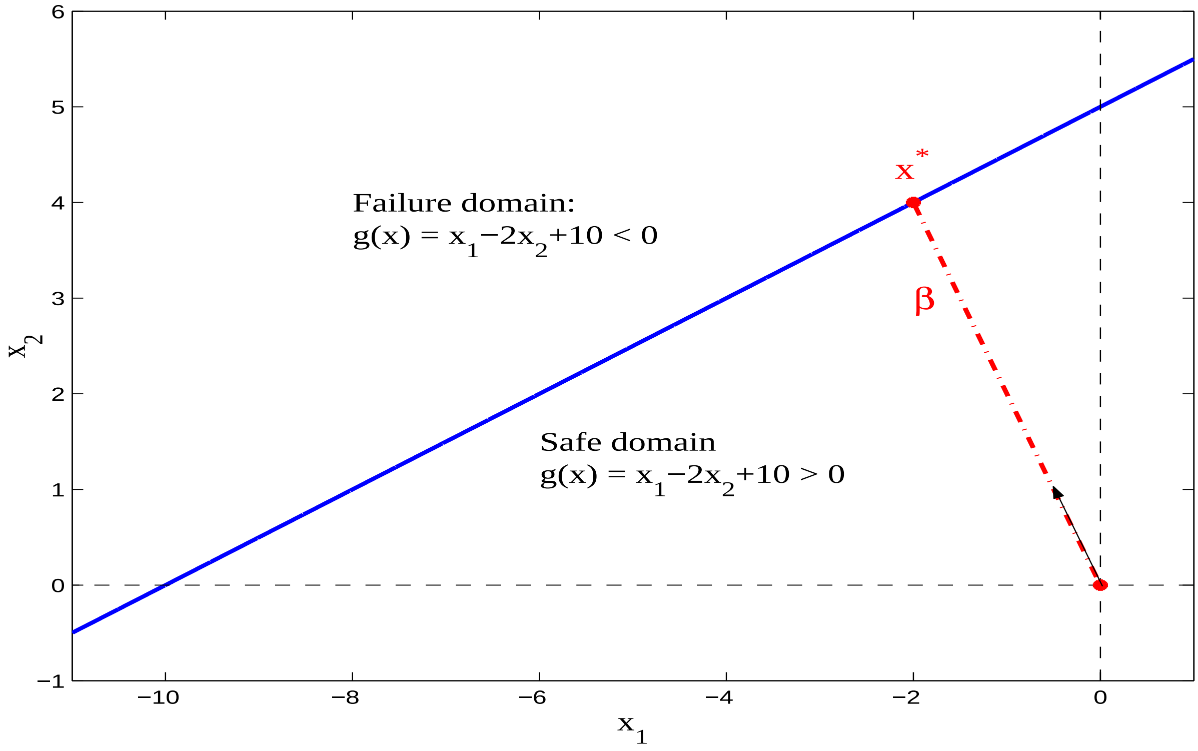

Figure 1: Linear failure surface in $\mathbb{R}^2$: $g(\mathbf{x}) \equiv g(x_1, x_2) = x_1 - 2x_2 + 10$, $\mathbf{x}^* = \{-2, 4\}$ and $\beta = 4.472$.

These simple facts do not hold when $g(\mathbf{x})$ is a nonlinear function. In this case ∇g depends on the choice of $\mathbf{x}$ and the direction of the outward normal from the failure surface changes with position along the failure surface. Therefore $\nabla g|_{\mathbf{x}=\mathbf{0}}$ is in general not the unit vector along true $\mathbf{x}^*$. To solve this problem we propose an iterative method so that the ∇g is updated at each iteration step. We first obtain an initial

$\mathbf{x}^*$, say $\mathbf{x}^*_0$, by projecting $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$ to the failure surface. Next we use this point to obtain a new unit vector $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{x}^*_0}$. Projecting this vector from the origin to the failure surface we obtain the next estimate of $\mathbf{x}^*$, say $\mathbf{x}^*_1$. The method then uses this point to calculate $\boldsymbol{\nabla} g$ and continues until two successive estimates of design points are close enough. In summary, for nonlinear $g(\mathbf{x})$, the iterative procedure can be described as follows:

1. For $k = 0$, select $\mathbf{x}^{(k)} = \mathbf{0}$, a small value of ϵ, say $\epsilon = 0.001$ and a large value of $\beta^{(k)}$, say $\beta^{(k)} = 10$.
2. Calculate $\nabla g_i^{(k)} = \dfrac{\partial g(\mathbf{x})}{\partial x_i}|_{\mathbf{x}=\mathbf{x}^{(k)}}$ for $i = 1, \cdots, n$. Construct the vector $\boldsymbol{\nabla} g^{(k)} = \left\{\nabla g_i^{(k)}\right\} \in \mathbb{R}^n$ and normalize so that $\boldsymbol{\nabla} g^{(k)^T} \boldsymbol{\nabla} g^{(k)} = 1$
3. Solve $g(v \boldsymbol{\nabla} g^{(k)}) = 0$ for v.
4. Increase the index: $k = k + 1$; denote $\beta^{(k)} = -v$ and $\mathbf{x}^{(k)} = v \boldsymbol{\nabla} g^{(k)}$.
5. Denote $\delta\beta = \beta^{(k-1)} - \beta^{(k)}$.
6. (a) If $\delta\beta < 0$ then the iteration is going in the wrong direction. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the best values of these quantities.
   (b) If $\delta\beta < \epsilon$ then the iterative procedure has converged. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the final values of these quantities.
   (c) If $\delta\beta > \epsilon$ then go back to step 2.

It should be noted that the convergence of the proposed iterative method cannot be always guaranteed. It is hoped that if the failure surface is fairly regular and linear then the method would converge. Next, this method is illustrated by a simple example.

3.1 An Example With Two Random Variables

Consider a system with two random variables x_1 and x_2. The non-linear failure surface is given by

$$g(\mathbf{x}) = -\frac{4}{25}(x_1 - 1)^2 - x_2 + 4 = 0. \tag{8}$$

This example is taken from Melchers (1999, page 105). Figure 2 shows the failure surface together with the design vector. The actual design vector and reliability index obtained by Melchers (1999) is given by

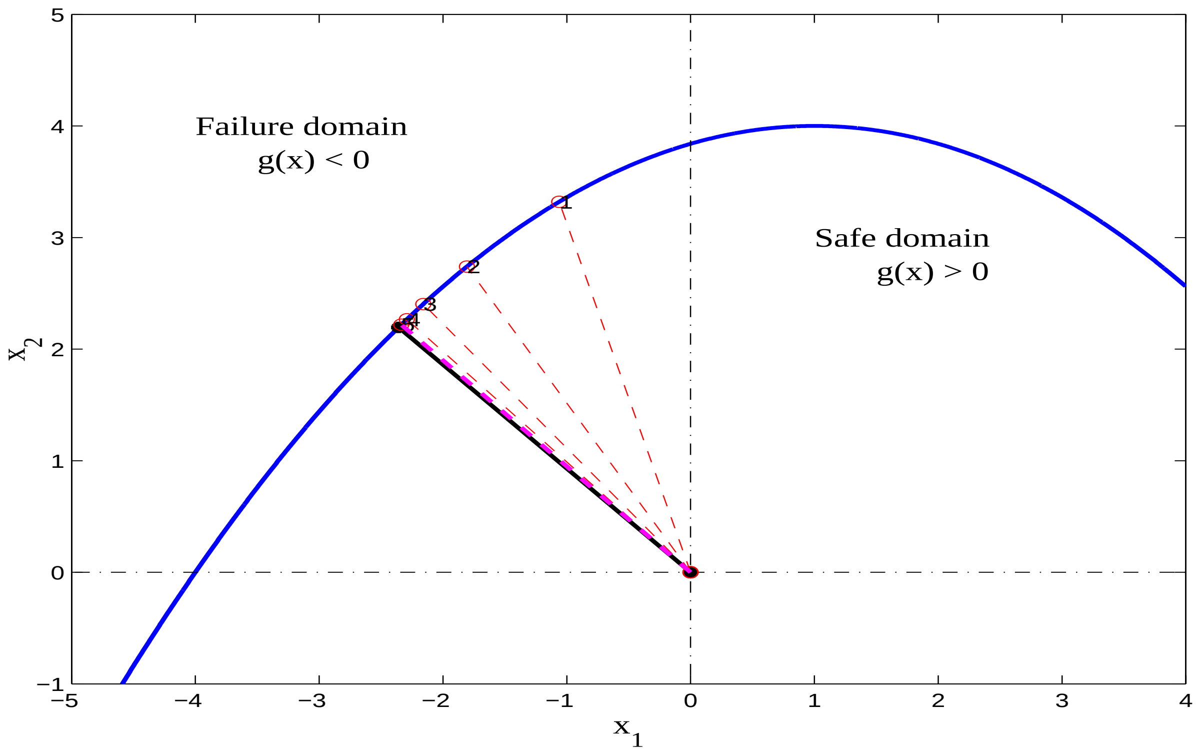

Figure 2: Nonlinear failure surface in two dimensional basic variable space: '—' actual design vector $\mathbf{x}^*$ obtained using FORM with all variables; '−−' design vectors obtained at each iteration step of the proposed iterative method

$$\mathbf{x}^* = \begin{Bmatrix} -2.36 \\ 2.19 \end{Bmatrix} \quad \text{and} \quad \beta = 3.22. \tag{9}$$

To apply the proposed method we need to obtain the gradients of the failure surface:

$$\frac{\partial g(\mathbf{x})}{\partial x_1} = -\frac{8}{25}(x_1 - 1) \tag{10}$$

$$\frac{\partial g(\mathbf{x})}{\partial x_2} = -1. \tag{11}$$

The starting point of the iterative scheme is $\mathbf{x} = \{0, 0\}^T$. Table 1 shows the value of β and $\mathbf{x}^*$ at each iteration step using the proposed method. Note that the after five iterations the results obtained from the proposed method converge to the actual values.

Iteration Number	$\mathbf{x}^*$	β
1	$\{-1.0623, 3.3195\}^T$	3.4854
2	$\{-1.8075, 2.7389\}^T$	3.2815
3	$\{-2.1591, 2.4033\}^T$	3.2307
4	$\{-2.2914, 2.2667\}^T$	3.2231
5	$\{-2.3367, 2.2186\}^T$	3.2222

Table 1: Numerical values of the design vector and reliability index at each iteration step of the proposed iterative method

3.2 An Example With Three Random Variables

Consider a system with three random variables so that $\mathbf{x} = \{x_1, x_2, x_3\}^T$. The non-linear failure surface is given by

$$g(\mathbf{x}) = -\frac{4}{25}(x_1+1)^2 - \frac{(x_2-5/2)^2(x_1-5)}{10} - x_3 + 3; \tag{12}$$

Figure 3 shows the failure surface together with the design vector. The actual design vector and reliability index obtained by using all the three random variables is given by

$$\mathbf{x}^* = \begin{Bmatrix} 2.1286 \\ 1.2895 \\ 1.8547 \end{Bmatrix} \quad \text{and} \quad \beta = 3.1038. \tag{13}$$

To apply the proposed method we need to obtain the gradients of the failure surface:

$$\frac{\partial g(\mathbf{x})}{\partial x_1} = -\frac{8}{25}(x_1+1) - \frac{(x_2-5/2)^2}{10} \tag{14}$$

$$\frac{\partial g(\mathbf{x})}{\partial x_2} = -\frac{(x_2-5/2)(x_1-5)}{5} \tag{15}$$

$$\frac{\partial g(\mathbf{x})}{\partial x_3} = -1. \tag{16}$$

The starting point of the iterative scheme is $\mathbf{x} = \{0, 0, 0\}^T$. Table 2 shows the value of β and $\mathbf{x}^*$ at each iteration step using the proposed method. Note that the after six iterations the results obtained from the proposed method converge to the actual values (see Figure 3).

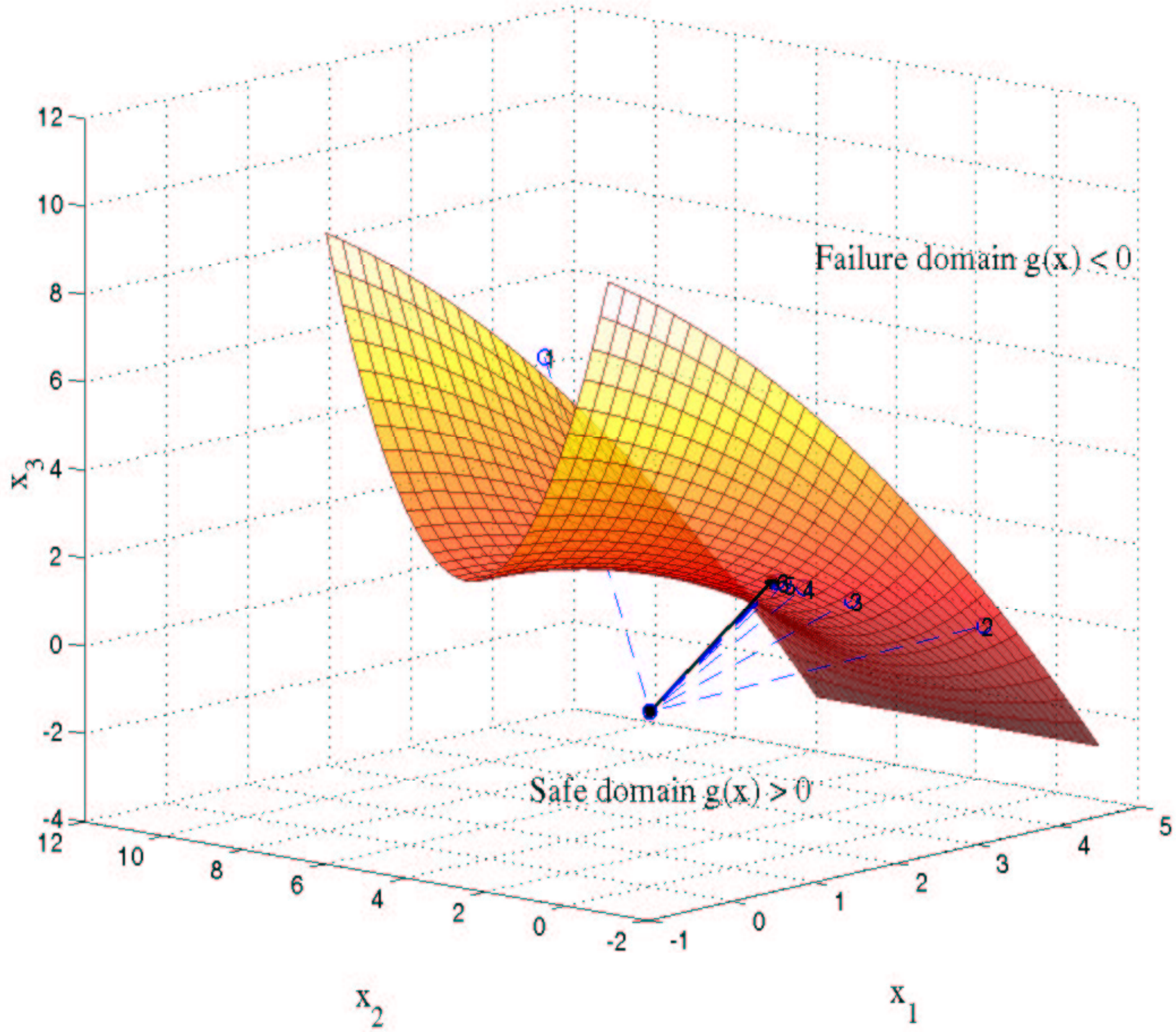

Figure 3: Nonlinear failure surface in three dimensional basic variable space: '—' actual design vector $\mathbf{x}^*$ obtained using FORM with all variables; '−−' design vectors obtained at each iteration step of the proposed iterative method

4 Method 2: Dominant Gradient Method

Here we keep more that one random variable in the analysis. The criteria for selecting the random variables is based on the sensitivity of the failure surface with respect to the random variables. Assume that

$$\boldsymbol{\nabla} g = \left\{ \frac{\partial g(\mathbf{x})}{\partial x_i} |_{\mathbf{x}=\mathbf{0}} \right\} \in \mathbb{R}^n \tag{17}$$

Iteration Number	$\mathbf{x}^*$	β
1	$\mathbf{x}^* = \begin{Bmatrix} 4.2245 \\ 11.1758 \\ 4.4703 \end{Bmatrix}$	12.7565
2	$\mathbf{x}^* = \begin{Bmatrix} 3.8085 \\ -0.5572 \\ 0.4140 \end{Bmatrix}$	3.8713
3	$\mathbf{x}^* = \begin{Bmatrix} 2.8825 \\ 0.8490 \\ 1.1654 \end{Bmatrix}$	3.2230
4	$\mathbf{x}^* = \begin{Bmatrix} 2.4309 \\ 1.1219 \\ 1.6046 \end{Bmatrix}$	3.1213
5	$\mathbf{x}^* = \begin{Bmatrix} 2.2506 \\ 1.2375 \\ 1.7476 \end{Bmatrix}$	3.1066
6	$\mathbf{x}^* = \begin{Bmatrix} 2.1788 \\ 1.2610 \\ 1.8163 \end{Bmatrix}$	3.1042

Table 2: Numerical values of the design vector and reliability index at each iteration step of the proposed iterative method

is normalized so that $\boldsymbol{\nabla} g^T \boldsymbol{\nabla} g = 1$. Suppose only n_d entries of $\boldsymbol{\nabla} g$ has significant non-zero (negative or positive) values while all other entries are close to zero. Consider the index set of these dominant variables is given by I_d. We construct the vector of dominant random variables $\mathbf{v} \in \mathbb{R}^{n_d}$ so that

$$v_j = x_i, \quad \forall j = 1, \cdots, n_d \, \text{and} \quad i \in I_d. \tag{18}$$

This equation can also be represented in a matrix form as

$$\mathbf{v} = \mathbf{D}\mathbf{x} \tag{19}$$

where $\mathbf{D} \in \mathbb{R}^{n_d \times n}$ is such that $\mathbf{D}_{ji} = 1 \, \forall j = 1, \cdots, n_d; i \in I_d$ and zero everywhere else. All other random variables are assumed to be zero, that is $x_i = 0, \forall i \notin I_d$. This implies that these variables assumed to be deterministic so that they do not play any role in the reliability analysis (see the discussion on 'omission sensitivity factors' by Madsen, 1988). Using these reduced set of variables one may obtain the failure surface in the reduced space $g'(\mathbf{v})$. Thus the optimization problem in the reduced space reads

$$\min \beta = (\mathbf{v}^T \mathbf{v})^{1/2} \quad \text{subject to} \quad g'(\mathbf{v}) = 0. \tag{20}$$

The design point in the reduced space, $\mathbf{v}^*$, can be obtained from the solution of (20). From $\mathbf{v}^*$, the design point in the actual space can simply be obtained by substituting $x_i^* = v_j^*, \forall j = 1, \cdots, n_d; i \in I_d$ and $x_i^* = 0, \forall\, i \notin I_d$.

It should be noted that this approach is based on sensitivity of $g(\mathbf{x})$ at $\mathbf{x} = \mathbf{0}$. For linear failure surface $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$ is the unit vector along true $\mathbf{x}^*$ and the dominant random variables selected by above procedure are actually the dominant random variables at the design point. For nonlinear $g(\mathbf{x})$, $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$ is in general not the unit vector along true $\mathbf{x}^*$. Thus the dominant random variables selected from $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$ are not necessarily the dominant random variables at the design point. Ideally the dominant random variables should be selected on the basis of $\boldsymbol{\nabla} g$ evaluated at the design point. Keeping this in mind, we try to update $\boldsymbol{\nabla} g$ in an iterative way so that the dominant random variables are selected on the basis of $\boldsymbol{\nabla} g$ evaluated at points gradually closer to the true design point. First, based on $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$, an initial $\mathbf{v}^*$ is obtained by solving the optimization problem (20). From this initial $\mathbf{v}^*$ the initial design point in the original space, say $\mathbf{x}_0^*$, is obtained. Next we use this point to obtain a new sensitivity vector $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{x}_0^*}$. Based on this we again select the dominant random variables and repeat the procedure to obtain the next estimate of $\mathbf{x}^*$, say $\mathbf{x}_1^*$. The method then uses this point to calculate $\boldsymbol{\nabla} g$ and continues until two successive estimates of design points are close enough. In summary, for nonlinear $g(\mathbf{x})$, the iterative procedure can be described as follows:

1. For $k = 0$, select $\mathbf{x}^{(k)} = \mathbf{0}$, a small value of ϵ, say $\epsilon = 0.001$, a large value of $\beta^{(k)}$, say $\beta^{(k)} = 10$ and also $n_d < n$.
2. Calculate $\nabla g_i^{(k)} = \dfrac{\partial g(\mathbf{x})}{\partial x_i}|_{\mathbf{x}=\mathbf{x}^{(k)}}$. For $i = 1, \cdots, n$ construct the vector $\boldsymbol{\nabla} g^{(k)} = \left\{\nabla g_i^{(k)}\right\} \in \mathbb{R}^n$ and normalize so that $\boldsymbol{\nabla} g^{(k)^T} \boldsymbol{\nabla} g^{(k)} = 1$.
3. Sort $|\nabla g_i^{(k)}|$ to obtain the index set I_d corresponding to the highest n_d values.
4. Set $v_j = x_i^{(k)}, \forall j = 1, \cdots, n_d$, $i \in I_d$ and $x_i^{(k)} = 0, \forall\, i \notin I_d$. Construct $\mathbf{v} = \{v_j\} \in \mathbb{R}^{n_d}$.
5. Using this transformation obtain $g'(\mathbf{v})$ from $g(\mathbf{x})$.
6. Solve the constrained optimization problem: minimize $\beta = (\mathbf{v}^T\mathbf{v})^{1/2}$ subject to $g'(\mathbf{v}) = 0$.
7. Increase the index: $k = k + 1$. Using the solutions from step 6 denote $\beta^{(k)} = \beta$ and $\mathbf{v}^{(k)} = \mathbf{v}$.
8. Obtain $\mathbf{x}^{(k)}$ from the inverse transformation in step 4, that is $x_i^{(k)} = v_j, \forall j = 1, \cdots, n_d$, $i \in I_d$ and $x_i^{(k)} = 0, \forall\, i \notin I_d$.
9. Denote $\delta\beta = \beta^{(k-1)} - \beta^{(k)}$.

10. (a) If $\delta\beta < 0$ then the iteration is going in the wrong direction. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the best values of these quantities.

    (b) If $\delta\beta < \epsilon$ then the iterative procedure has converged. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the final values of these quantities.

    (c) If $\delta\beta > \epsilon$ then go back to step 2.

The initial choice of the dominant random variables, that is, n_d and I_d, can be automated by imposing a selection criteria, for example, by fixing the ratio of $\nabla g_i^{(k)}$ corresponding to the most sensitive random variable and the least sensitive random variable. Note that the index set of the dominant variables I_d may change in different iterations, however we fix n_d for all iterations. Like the method in section 3, the convergence of the proposed iterative method cannot be always guaranteed. In the next section another variant of this approach, which considers the contribution of the neglected variables, is proposed.

5 Method 3: Relative Importance Variable Method

Based on the entries of $\boldsymbol{\nabla} g$ we group the random variables into the 'important' and 'unimportant' random variables. The random variables for which the failure surface is more sensitive are called important variables. Like the approach in section 4, suppose only $n_d < n$ entries of $\boldsymbol{\nabla} g$ with an index set $i \in I_d$ is important. Suppose that the important random variables are casted in the vector $\mathbf{v} \in \mathbb{R}^{n_d}$ such that equation (18) is satisfied. However, unlike the approach in section 4, we do not completely neglect all the 'unimportant' random variables, but consider that they can be represented by a single random variable, say u such that

$$x_i = u \nabla g_i, \quad \forall i \notin I_d. \tag{21}$$

This equation implies that the 'direction' of the unimportant random variables are fixed according to the gradient vector so that u is effectively a scaling parameter in that direction. This method can be viewed as a combination of the methods described in the previous two sections. Now we construct the vector of reduced random variables $\mathbf{z}$ as

$$\mathbf{z} = \begin{Bmatrix} \mathbf{v} \\ u \end{Bmatrix} \in \mathbb{R}^{n_d+1}. \tag{22}$$

Using equations (18) and (21) one may obtain the failure surface in the reduced space $g'(\mathbf{z})$. Thus the optimization problem in the reduced space reads

$$\min \beta = (\mathbf{z}^T \mathbf{z})^{1/2} \quad \text{subject to} \quad g'(\mathbf{z}) = 0. \tag{23}$$

The design point in the reduced space, $\mathbf{z}^* = \begin{Bmatrix} \mathbf{v}^* \\ u^* \end{Bmatrix}$, can be obtained from the solution of (23). From $\mathbf{z}^*$, the design point in the actual space can simply be obtained by substituting $x_i^* = v_j^*, \forall j = 1, \cdots, n_d; i \in I_d$ and $x_i^* = u\nabla g_i, \forall\, i \notin I_d$.

Note that the selection the important and unimportant random variables are based on $\boldsymbol{\nabla} g$ evaluated at $\mathbf{x} = \mathbf{0}$. The important and unimportant random variables should be selected on the basis of $\boldsymbol{\nabla} g$ evaluated at the true design point $\mathbf{x}^*$. Because for nonlinear $g(\mathbf{x})$, $\boldsymbol{\nabla} g|_{\mathbf{x}=\mathbf{0}}$ is in general not the unit vector along true $\mathbf{x}^*$, we update $\boldsymbol{\nabla} g$ by an iterative method similar to the ones described in the previous two sections. In summary this iterative approach can be described as follows:

1. For $k = 0$, select $\mathbf{x}^{(k)} = \mathbf{0}$, a small value of ϵ, say $\epsilon = 0.001$, a large value of $\beta^{(k)}$, say $\beta^{(k)} = 10$ and also $n_d < n$.
2. Calculate $\nabla g_i^{(k)} = \dfrac{\partial g(\mathbf{x})}{\partial x_i}|_{\mathbf{x}=\mathbf{x}^{(k)}}$. For $i = 1, \cdots, n$ construct the vector $\boldsymbol{\nabla} g^{(k)} = \left\{ \nabla g_i^{(k)} \right\} \in \mathbb{R}^n$ and normalize so that $\boldsymbol{\nabla} g^{(k)^T} \boldsymbol{\nabla} g^{(k)} = 1$.
3. Sort $|\nabla g_i^{(k)}|$ to obtain the index set I_d corresponding to the highest n_d values.
4. Set $v_j = x_i^{(k)}, \forall j = 1, \cdots, n_d$, $i \in I_d$ and $x_i^{(k)} = u\nabla g_i^{(k)}, \forall\, i \notin I_d$. Construct $\mathbf{z} = \{v_j, u\} \in \mathbb{R}^{n_d+1}$.
5. Using this transformation obtain $g'(\mathbf{z})$ from $g(\mathbf{x})$.
6. Solve the constrained optimization problem: minimize $\beta = (\mathbf{z}^T\mathbf{z})^{1/2}$ subject to $g'(\mathbf{z}) = 0$.
7. Increase the index: $k = k + 1$. Using the solutions from step 6 denote $\beta^{(k)} = \beta$ and $\mathbf{z}^{(k)} = \mathbf{z}$.
8. Obtain $\mathbf{x}^{(k)}$ from the inverse transformation in step 4.
9. Denote $\delta\beta = \beta^{(k-1)} - \beta^{(k)}$.
10. (a) If $\delta\beta < 0$ then the iteration is going in the wrong direction. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the best values of these quantities.
    (b) If $\delta\beta < \epsilon$ then the iterative procedure has converged. Terminate the iteration procedure and select $\beta = \beta^{(k)}$ and $\mathbf{x}^* = \mathbf{x}^{(k)}$ as the final values of these quantities.
    (c) If $\delta\beta > \epsilon$ then go back to step 2.

Again, like the methods in sections 3 and 4, the convergence of the proposed iterative method cannot be always guaranteed. Beside the three methods proposed here there are several methods that have been used to reduce the number of random variables. Brief description of these methods are given in Appendix B.

6 Numerical Examples

6.1 2D Framed Structure

We consider a 2D framed structure with 3 members. The structure is shown in Figure 4 with element numbering, node numbering and coordinates of the nodes in meters. It is assumed that the axial stiffness (EA) and the bending stiffness (EI) of each member are Gaussian random variables so that there are in total six random variables, $\mathbf{x} \in \mathbb{R}^6$. Further it is also assumed that EA and EI of different members are uncorrelated, that is

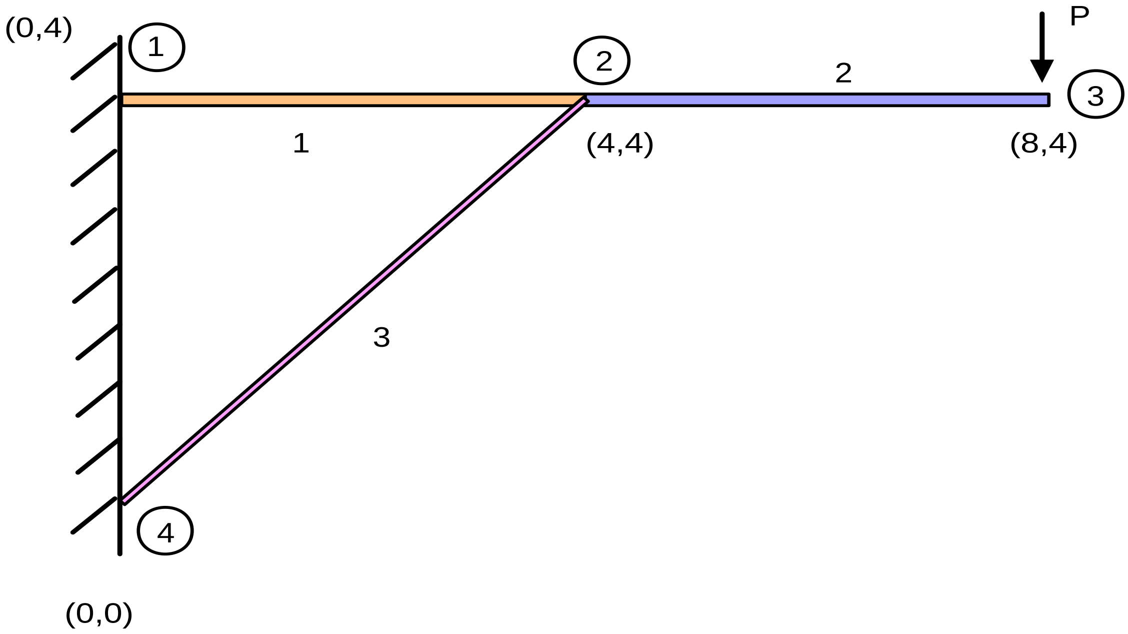

Figure 4: 2D frame with random element properties, $P = 100$KN

$$< EA_i, EA_j >= 0, \forall i \neq j; \quad < EI_i, EI_j >= 0, \forall i \neq j; \quad < EA_i, EI_j >= 0, \forall i, j. \tag{24}$$

Therefore the joint probability density function (pdf) of the random variables are completely characterized by the mean and the standard deviation of the random variables. Table 3 shows the numerical values of the system properties assumes for different members. The standard deviations are expressed as a percentage of the corresponding mean values. The vertical force applied in node 3 is assumed to be 100 KN. The failure condition is given by specifying a maximum allowable vertical displacement at node 3, say d_{max}. Now we construct the failure surface

$$g(\mathbf{x}) = d_{max} - |\delta v_3(\mathbf{x})| \tag{25}$$

Member Id	EA (KN)		EI (KNm^2)	
	Mean	Standard Deviation	Mean	Standard Deviation
1	1.0×10^9	3.0%	2.0×10^4	10.0%
2	5.0×10^9	7.0%	6.0×10^4	5.0%
3	3.0×10^9	10.0%	4.0×10^4	9.0%

Table 3: Element properties of the random 2D frame. The standard deviations are expressed as a percentage of the corresponding mean values.

where the random variable δv_3 is the vertical displacement at node 3. The structure is unsafe when $g(\mathbf{x}) < 0$ that is, when $\delta v_3 > d_{max}$. For numerical calculations it is assumed that $d_{max} = 0.095$m. For more details see the input data for the FORTRAN program is given in Appendix A.1.

Numerical results obtained by using the proposed method is shown in Table 4. The Monte Carlo simulation (MCS) is performed with 30000 samples. The methods in Table 4 are arranged in the order of increasing computational cost. It is clear that the all the proposed methods produces satisfactory agreement with the Monte Carlo simulation (considered as benchmark).

Method	β	P_f
Method 1 (one iteration)	3.619	0.147×10^{-3}
Method 2 with $n_d = 3$ (one iteration)	3.590	0.165×10^{-3}
Method 3 with $n_d = 3$ (one iteration)	3.590	0.165×10^{-3}
Conventional FORM with full set of six variables	3.590	0.165×10^{-3}
MCS with 30,000 samples	–	0.166×10^{-3}

Table 4: Properties of the random frame

6.2 Mutlistoried Portal Frame

We consider a mutlistoried portal frame with 20 members. The details of the structure is shown in Figure 5 with element numbering and node numbering. It is assumed that the axial stiffness (EA) and the bending stiffness (EI) of each member are Gaussian random variables so that there are in total 40 random variables, $\mathbf{x} \in \mathbb{R}^{40}$. Like the previous example it is also assumed that EA and EI of different members are uncorrelated. The joint probability density function (pdf) of the random variables are completely characterized by the mean and the standard deviation of EA and EI of each member . There are three types of elements

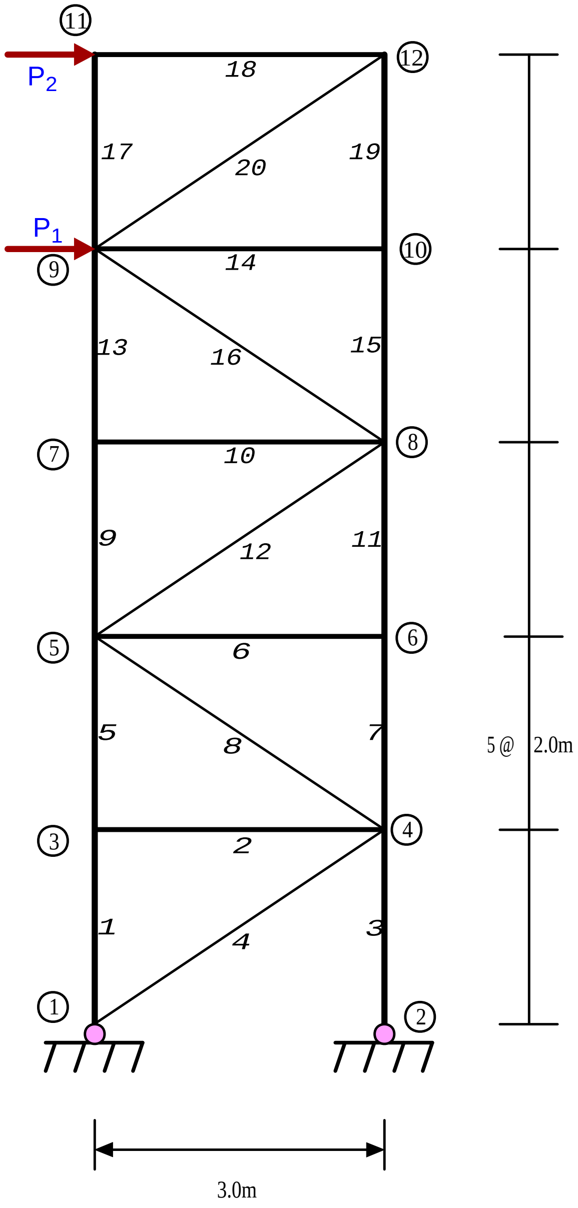

Figure 5: Mutlistoried portal with random element properties, $P_1 = 4.0 \times 10^5$KN and $P_2 = 5.0 \times 10^5$KN

and the numerical values of the properties of each element type is shown in Table 5. It is assumed that the column members are of type 1, the beam members are of type 2 and the bracing members are of type 3. The input data for the `FORTRAN` program is given in the Appendix A.2.

Two horizontal forces $P_1 = 4.0 \times 10^5$KN and $P_2 = 5.0 \times 10^5$KN are applied at nodes 9 and 11 respectively. The failure condition is given by specifying a maximum allowable horizontal displacement at node 9, say

Element Type	EA (KN)		EI (KNm^2)		Element Numbers
	Mean	Standard Deviation	Mean	Standard Deviation	
1	5.0×10^9	7.0%	6.0×10^4	5.0%	1,3,5,7,9,11,13,15,17,19
2	3.0×10^9	3.0%	4.0×10^4	10.0%	2,6,10,14,18
3	1.0×10^9	10.0%	2.0×10^4	9.0%	4,8,12,16,20

Table 5: Element types and associated elements numbers of the random multistoried portal frame. The standard deviations are expressed as a percentage of the corresponding mean values.

d_{max}. We construct the failure surface

$$g(\mathbf{x}) = d_{max} - |\delta h_9(\mathbf{x})| \tag{26}$$

where the random variable δh_9 is the horizontal displacement at node 9. The structure is unsafe when $g(\mathbf{x}) < 0$ that is, when $\delta_3 > d_{max}$. For numerical calculations it is assumed that $d_{max} = 0.16 \times 10^{-2}$m. Numerical results obtained by using the proposed method is shown in Table 6. The Monte Carlo simulation (MCS) is performed with 9000 samples. The methods in Table 6 are arranged in the order of increasing computational cost. It is clear that the all the proposed methods using reduced number of random variables produces same result obtained by conventional FORM with full set of 40 variables. Further, also note that all the approximate reliability estimate methods show satisfactory agreement with the Monte Carlo simulation (considered as benchmark).

Method	β	P_f
Method 1 (one iteration)	2.857	0.214×10^{-2}
Method 2 with $n_d = 4$ (one iteration)	2.857	0.214×10^{-2}
Method 3 with $n_d = 4$ (one iteration)	2.857	0.214×10^{-2}
Conventional FORM with full set of 40 variables	2.857	0.214×10^{-2}
MCS with 9000 samples	–	0.266×10^{-2}

Table 6: Properties of the random frame

7 Conclusions

Methods have been proposed to reduce the number of random variables in structural reliability problems involving a large number of random variables. In total three iterative methods, namely (a) gradient projection method, (b) dominant gradient method, and (c) relative importance variable method, have been proposed. All the three methods are based on the sensitivity vector of the failure surface. Initial numerical results show that there is a possibility to put these methods into real-life problems involving a large number of random variables. Further studies will involve analyzing the efficiency of the proposed methods when applied to problems with highly non-linear failure surfaces, such as in dynamic problems.

References

Augusti, G., Baratta, A., and Casciati, F. (1984), *Probabilistic Methods in Structural Engineering*, Chapman and Hall, London, UK.

Ditlevsen, O. and Madsen, H. O. (1996), *Structural Reliability Methods*, John Wiley and Sons Ltd, Chichester, West Sussex, England.

Madsen, H. O. (1988), “Omission sensitivity factors”, *Structural Safety*, **5**, pp. 35–45.

Melchers, R. E. (1999), *Structural Reliability Analysis and Prediction*, John Wiley and Sons Ltd, Chichester, West Sussex, England, second edition.

Rosenblatt, M. (1952), “Remarks on a multivariate transformation”, *The Annals of Mathematical Statistics*, **23**, pp. 470–472.

Thoft-Christensen, P. and Baker, M. J. (1982), *Structural Reliability Theory and its Applications*, Springer-Verlag, Berlin, Germany.

Tichy, M. (1993), *Applied Methods of Structural Reliability*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

A Input Data File For The FORTRAN Program

A.1 Example in Section 6.1

```
2D Frame Example
Analysis Type (1 = Static, 2 = Dynamic)
1
Main FE Control parameters (nels0,nn0,ndim,np_types)
3  4  2  3
Failure condition (first id: [1,2,3] => [x,y,rot]; second id: node number)
0.095,2,3
The mean property array (ea,ei,rho,alpha,beta)
5.e9 6.e4 3.2 .001 .002
1.e9 2.e4 1.2 .001 .002
3.e9 4.e4 2.2 .001 .002
Specify element type for mean property array "etype"
2 1 3
Randomness associated with mean property (in percentage of the mean value)
07.0 05.0 0.0 10.0 10.0
03.0 10.0 0.0 10.0 10.0
10.0 09.0 0.0 10.0 10.0
Nodal coordinates
0.0 4.0
4.0 4.0
8.0 4.0
0.0 0.0
Nodal connectivity (element no, connecting nodes=2, number of divisions)
1 1 2 1
2 2 3 1
3 2 4 1
Node freedom data
2
1 0 0 1
4 0 0 1
Loaded Nodes
1
3 0.0 -100.0 0.0
Fixed Nodes
0
```

A.2 Example in Section 6.2

```
2D Multistoried Portal Frame Example
Analysis Type (1 = Static, 2 = Dynamic)
1
Main FE Control parameters (nels0,nn0,ndim,np_types)
20  12  2  3
Failure condition (first id: [1,2,3] => [x,y,rot]; second id: node number)
0.16E-02,1,9
The mean property array (ea,ei,rho,alpha,beta)
5.e9 6.e4 3.2 .001 .002
3.e9 4.e4 2.2 .001 .002
1.e9 2.e4 1.2 .001 .002
Specify element type for mean property array "etype"
1 2 1 3 1 2 1 3 1 2 1 3 1 2 1 3 1 2 1 3
Randomness associated with mean property (in percentage of the mean value)
07.0 05.0 0.0 10.0 10.0
03.0 10.0 0.0 10.0 10.0
10.0 09.0 0.0 10.0 10.0
Nodal coordinates
0.0 0.0
0.0 3.0
2.0 0.0
2.0 3.0
4.0 0.0
4.0 3.0
6.0 0.0
6.0 3.0
8.0 0.0
8.0 3.0
10.0 0.0
10.0 3.0
Nodal connectivity (element no, connecting nodes=2, number of divisions)
1 1 3 1
2 3 4 1
3 4 2 1
4 1 4 1
5 3 5 1
6 5 6 1
7 6 4 1
8 5 4 1
9 5 7 1
10 7 8  1
11 8 6 1
12 5 8 1
13 7 9 1
14 9 10 1
```

```
15 10 8 1
16 9 8 1
17 9 11 1
18 11 12 1
19 12 10 1
20 9 12 1
Node freedom data
2
1 0 0 1
2 0 0 1
Loaded Nodes
2
9 4.0e5 0.0 0.0
11 5.0e5 0.0 0.0
Fixed Nodes
0
```

B Random Variable Reduction using Singular Value Decomposition

For some set of points $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m, m \leq n$ in $\mathbb{R}^n$ we can rewrite the first-order Taylor series expansion of $g(\mathbf{x})$ about $\mathbf{x}^*$ in a matrix form as

$$\mathbf{g} \approx \mathbf{S}\mathbf{x} \tag{27}$$

where

$$\mathbf{g} = \begin{Bmatrix} g(\mathbf{x}_1) \\ g(\mathbf{x}_2) \\ \vdots \\ g(\mathbf{x}_m) \end{Bmatrix} \in \mathbb{R}^m \tag{28}$$

and

$$\mathbf{S} = [S_{ij}] \in \mathbb{R}^{m \times n} \quad \text{where} \quad S_{ij} = \frac{\partial g(\mathbf{x})}{\partial x_j} |_{\mathbf{x}=\mathbf{x}_i} \forall j = 1, \cdots, n; i = 1, \cdots, m \tag{29}$$

If all $\mathbf{x}_i, \forall\, i = 1, \cdots, m$ are on failure surface then $\mathbf{g} = \mathbf{0}$. In view of this, we try to minimize the l_2 norm of $\mathbf{g}$ that is

$$||\mathbf{g}||_2 = \mathbf{g}^T\mathbf{g} = \mathbf{x}^T\mathbf{S}^T\mathbf{S}\mathbf{x} = \mathbf{x}^T\mathbf{W}\mathbf{x}, \quad \text{where} \quad \mathbf{W} = \mathbf{S}^T\mathbf{S} \in \mathbb{R}^n. \tag{30}$$

Suppose the matrix of the eigenvectors of $\mathbf{W}$ is $\mathbf{\Phi}$ so that

$$\mathbf{\Phi}^T\mathbf{W}\mathbf{\Phi} = \boldsymbol{\mu} \tag{31}$$

where $\boldsymbol{\mu}$ is a real diagonal matrix as $\mathbf{W}$ is real and symmetric. The rank of $\mathbf{W}$ is $m \leq n$. Suppose $\mathbf{\Psi} \in \mathbb{R}^{n \times m}$ is a partition of $\mathbf{\Phi}$ which contains the eigenvectors corresponding to only non-zero eigenvalues of $\mathbf{W}$. Using this we obtain a transformation

$$\mathbf{v} = \mathbf{\Psi}^T\mathbf{x} \in \mathbb{R}^m. \tag{32}$$

The vector $\mathbf{v}$ is the reduced set of random variables which can be used for reliability assessment. For linear $g(\mathbf{x})$, $\mathbf{W}$ is of rank one and this method reduces to the gradient projection method described in section 3. The set of points $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m$ can be selected in several ways:

- Obtain a single point on $g(\mathbf{x})$ and then choose a set of $(m-1)$ points randomly around it.
- Proceed to $g(\mathbf{x})$ from the origin through the path of steepest decent and take m points closest $g(\mathbf{x})$ along this path.

This approach too can be used in the framework of a iterative scheme. However, in each iteration step singular value decomposition of the matrix $\mathbf{W}$ is required and for this reason this approach is numerically less efficient compared to the methods described in sections 3—5. Beside, the approach depends on the choice of $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m$ and also it is not robust when m is much less than n.