Structure and Infrastructure Engineering, Vol. 2, Nos. 3–4, September–December 2006, 261–275

Taylor & Francis
Taylor & Francis Group

Enhanced Sequential Optimization and Reliability Assessment method for probabilistic optimization with varying design variance

XIAOLEI YIN and WEI CHEN*

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Rd, Evanston, IL 60208-3111, USA

(*Received 14 May 2005; accepted in revised form 29 November 2005*)

The Sequential Optimization and Reliability Assessment (SORA) method is a single-loop method containing a serial of cycles of decoupled deterministic optimization and reliability assessment for improving the efficiency of probabilistic optimization. However, the original SORA method as well as some other existing single-loop methods do not take into account the effect of varying design variance (changing variance) in design problems. In this paper, to enhance the SORA method, three formulations are proposed in order to improve the efficiency for solving problems with changing variance. These formulations are categorized by the different strategies of Inverse Most Probable Point (IMPP) approximation. Mathematical examples and a speed reducer design problem are utilized to test and compare the effectiveness of the proposed formulations. The insight gained from our study on the applicability of different approaches can be extended and utilized in other probabilistic optimization strategies that require IMPP estimations.

Keywords: Probabilistic optimization; Single-loop method; Most probable point; Changing variance

1. Introduction

Probabilistic design optimization offers an approach for making reliable design decisions with the consideration of uncertainty (Wu and Wang 1996, Sundaresan *et al.* 1995, Carter 1997, Grandhi and Wang 1998, Melchers 1999, Du and Chen 2000, Sopory *et al.* 2004, Huang *et al.* 2005). In the existing applications, probabilistic optimization is accomplished by either the double-loop (DLP) methods (Reddy *et al.* 1994, Wu 1994, Yu *et al.* 1998, Youn *et al.* 2003) or the single-loop (SLP) methods (Chen and Hasselman 1997, Sues and Cesare 2000, Du and Chen 2004, Liang *et al.* 2004). The traditional double-loop method consists of the design optimization loop (outer loop) in the original design variable space, which iteratively performs the reliability analysis (inner loop) until achieving the optimum and meeting the desired probabilistic constraints. Due to the nested structure of outer and inner loops, the double-loop methods are often computationally expensive and not affordable for a variety of practical applications.

Recent years have seen many efforts made towards developing new and efficient probabilistic optimization strategies. The single-loop method is developed to avoid the nested coupling of the outer and inner loops (Chen and Hasselman 1997, Du and Chen 2004, Liang *et al.* 2004). The existing single-loop methods deviate in the way of how optimization and reliability analysis are organized. One type of single-loop methods directly integrate the reliability analysis into a design optimization procedure and solve the integrated problem as a single optimization problem. The other type of single-loop methods decouple the optimization and reliability analysis and solve them sequentially from cycle to cycle. We call the former 'integrated single-loop method' and the later 'decoupled, sequential single-loop method'. An example under the 'integrated single-loop method' is the 'single-loop single vector' approach (Chen and Hasselman 1997) that is employed in a reduced, uncorrelated and normalized space other than the standard normal space. Without conducting the expensive Most Probable Point (MPP) (Hasofer and Lind 1974) search, it uses the

*Corresponding author. Email: weichen@northwestern.edu

Structure and Infrastructure Engineering
ISSN 1573-2479 print/ISSN 1744-8980 online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/15732470600590317

steepest descent direction from the previous iteration to approximate MPP and generates an equivalent deterministic optimization problem. Although this method avoids nested loops, it needs to evaluate all the constraint derivatives to identify the active constraints by using an active constraint searching strategy. Another efficient method under the 'integrated single-loop method' is the approach recently developed by Liang *et al.* (2004), where a probabilistic optimization problem is converted into an equivalent deterministic optimization problem by adding an equality constraint that determines the MPP based on the Karush–Kuhn–Tucker (KKT) optimality condition of the reliability constraints. Although the method greatly improves the efficiency by eliminating the reliability analysis loop, the KKT condition is only a necessary but not sufficient condition. The method may have difficulties in locating the exact MPP when there are multiple, local MPPs.

Examples of the 'decoupled, sequential single-loop methods' include the safety-factor approach (Wu *et al.* 2001) and our recently developed Sequential Optimization and Reliability Assessment (SORA) method (Du and Chen 2004). The SORA method (Du and Chen 2004) decouples the probabilistic optimization problem in the form of a serial of sequential Deterministic Optimization (DO) followed by Reliability Assessments (RA). The SORA method achieves high efficiency by taking the following measures: the use of R-percentile formulation (Inverse Most Probable Point–IMPP concept) to evaluate constraint satisfaction only up to the required reliability level (R) (Tu *et al.* 1999); the use of an efficient and robust IMPP search algorithm for reliability assessment (Du and Chen 2001); and the employment of a shifting vector for estimating IMPP in the DO formulation of a new cycle. Because the IMPP estimation of a new cycle is always based on the exact IMPP from the previous cycle, the method improves the accuracy in IMPP estimation, which is especially useful when there are multiple, local optimal solutions to IMPPs. The safety-factor approach (Wu *et al.* 2001) follows the similar strategy of decoupling optimization and reliability assessment, but approximates the MPP by performing a reliability analysis with the 'shifted' limit-state function. Gu and Yang (2003) compared various probabilistic optimization strategies on a set of benchmarking problems. It shows that the SORA method is one of the most robust and efficient methods.

One limitation of the SORA method and some other existing probabilistic optimization methods is that the existing formulations and solution strategies only take into account the random design variables with constant variances. In practice, instead of fixed variance, design variance often depends on the magnitude of a design variable. For example, the concentricity of the cylindric surface of a shaft needs to be controlled within a certain range if the geometry of the shaft is that of a stepped cylinder. Compared with a small shaft, to manufacture one with larger size in proportion, larger variance of the concentricity is allowed due to the increasing difficulties in keeping the same concentricity when the length of the shaft becomes larger. One efficient way to describe the changing variance is to introduce the *coefficient of variation* (rate of standard deviation to the mean value of a random variable). The changing variance in probabilistic design poses more difficulties in predicting the MPP and the constraint boundary of an equivalent DO formulation as well as the final design solution.

Our objective in this paper is to enhance the SORA method by developing new formulations that take into account the changing variance, while keeping the decoupling, single-loop strategy the same. Three different formulations are examined in this work, and compared to the use of the original SORA method for problems with changing variance. Observations and recommendations are made based on comparative studies using several illustrative examples.

The paper is organized as follows. Section 2 of this paper provides a brief review of the original SORA method as well as the concept of IMPP. In Section 3, four formulations that employ different strategies to consider the effect of changing variance are presented. They are (1) Shifting Vector Formulation (Approach 0 or the original SORA method); (2) Approximated $\boldsymbol{u}_{\mathrm{IMPP}}$ Formulation (Approach 1); (3) Direct Linear Estimation Formulation (Approach 2); and (4) Quasi First Order Approximation of $\boldsymbol{x}_{\mathrm{IMPP}}$ Formulation (Approach 3). The efficiencies and optimal results of these methods are compared in Section 4 through illustrative examples. Section 5 provides the conclusion.

2. Review of the SORA method

2.1 *Background of probabilistic optimization*

A typical probabilistic design model (Du 2002) is defined as

$$\begin{array}{ll} \text{Find} & \boldsymbol{d}, \boldsymbol{\mu}_x \\ \text{Minimize} & f(\boldsymbol{d}, \boldsymbol{x}, \boldsymbol{p}) \\ \text{Subject to} & P_i\{g_i(\boldsymbol{d}, \boldsymbol{x}, \boldsymbol{p}) \geq 0\} \geq R_i, \quad i = 1, 2, \ldots, m \end{array} \tag{1}$$

where $\boldsymbol{d}$ vector stands for deterministic design variables; $\boldsymbol{x}$ vector is composed of all the random design variables X_i, with variable distributions depending on the changing mean locations, $\boldsymbol{\mu}_x$, of design variables $\boldsymbol{x}$ during optimization process; $\boldsymbol{p}$ vector is for random parameters with fixed distributions, e.g. the mean and standard deviation of $\boldsymbol{p}$ are fixed and cannot be changed during optimization process; and m is the number of reliability constraints. Different from deterministic design, probabilistic design focuses on maintaining design feasibility for constraints at desired probabilistic levels while achieving the optimum of the objective function. Thus at the optimal point, the probability of constraint satisfaction, P_i for $g_i \geq 0$, should be no less than the desired reliability level R_i, which is referred to as reliability assessment. The g_i functions are the so-called limit state functions (Hasofer and Lind 1974).

One efficient way for reliability assessment is to employ the Most Probable Point (MPP) approach (Hasofer and Lind 1974). Integration of the reliability problem into the design optimization problem via the MPP optimality conditions was first proposed by Rackwitz and Fiessler (1978). With the MPP approach, all random quantities $\boldsymbol{x}$ and $\boldsymbol{p}$ are transformed into $\boldsymbol{u}$ in a standardized normal space, called $\boldsymbol{u}$-space. MPP is formally defined as a point located on the limit state function hypersurface with the minimum distance to the origin point in $\boldsymbol{u}$-space. The corresponding minimum distance is the safety index β indicating the reliability level of the limit state function as shown in figure 1. When the First Order Reliability Method (FORM) is used, reliability R is approximated as the standardized normal Cumulative Distribution Function (CDF) $\Phi(\beta)$. Higher order estimations, such as Second Order Reliability Method (SORM) (Fiessler *et al.* 1979, Breitung 1984, Der Kiureghian *et al.* 1987), could be used if a more precise estimation of reliability is needed.

Determining the MPP point on the limitation state function, or called the original MPP search, can be implemented by solving the following optimization problem in equation (2).

$$\begin{aligned} &\min_{U} \|\boldsymbol{u}\| \\ &s.t. \quad g(\boldsymbol{u}) = 0 \end{aligned} \tag{2}$$

where $\|\boldsymbol{u}\|_{\min}$ is the safety index β. Inversely, given a desired reliability level R and correspondingly the safety index β, the procedure of finding the corresponding MPP on the limit state function is called the IMPP search. The use of IMPP, in replace of MPP, to assess the feasibility of reliability constraints is motivated by the need for improving the computational efficiency. Nevertheless, most of the existing MPP and IMPP search algorithms have convergence difficulties for non-concave and non-convex problems. The IMPP search algorithm used in the SORA method is an improved method, called Modified Advanced Mean Value (MAMV) method. Its advantages are detailed in Du *et al.* (2004).

At the MPP as well as IMPP, for a normal distribution function, the following relationship holds when transforming random variables $\boldsymbol{x}$ into the normal space $\boldsymbol{u}$:

$$X_{\mathrm{IMPP}i} = \mu_{x_i} + \sigma_{x_i} \cdot U_{\mathrm{IMPP}i} \tag{3}$$

More details for dealing with non-normal random variables are given in section 3.5.

2.2 *The SORA method*

The Sequential Optimization and Reliability Assessment (SORA) method is a decoupled, sequential single-loop method for probabilistic optimization (Du and Chen 2004). As shown in figure 2, SORA improves the computational efficiency of probabilistic optimization by decoupling the reliability assessment from the optimization loop. In each cycle k, the procedure contains two separate parts of Deterministic Optimization (DO) and Reliability Assessment (RA). The DO at cycle k is formulated by including the predicted IMPP estimated based on the exact IMPP verified from the RA in cycle $k-1$, marked as a dashed line in the flowchart. Therefore, the DO formulation in a new cycle is built upon the verified IMPP from the previous cycle. The dashed line only indicates the information flow but not a nested relationship between DO and RA. In each cycle, following DO, the design solution obtained from DO is verified by checking the feasibility of probabilistic constraint in RA. If the feasibility is satisfied, the process will stop after verifying the convergence criterion; otherwise the cycle will be repeated.

Two important measures are taken in converting probabilistic constraints to equivalent deterministic constraints in the DO formulation of SORA: one is related to the use of R percentile formulation to replace the original reliability constraint; the other applies a shifting vector to

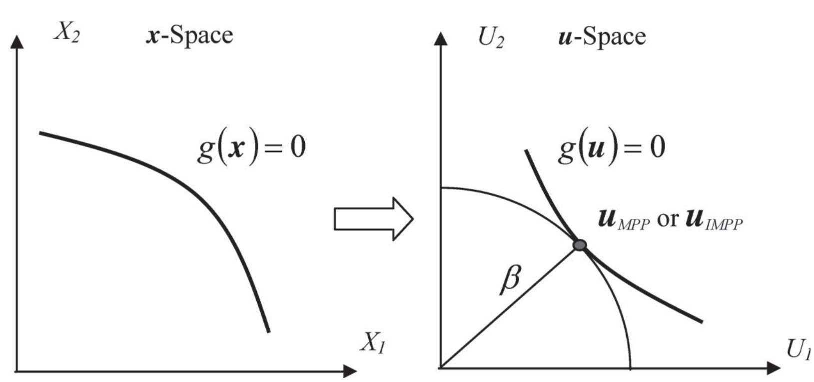

Figure 1. Transformation of input variables and illustration of most probable point (MPP).

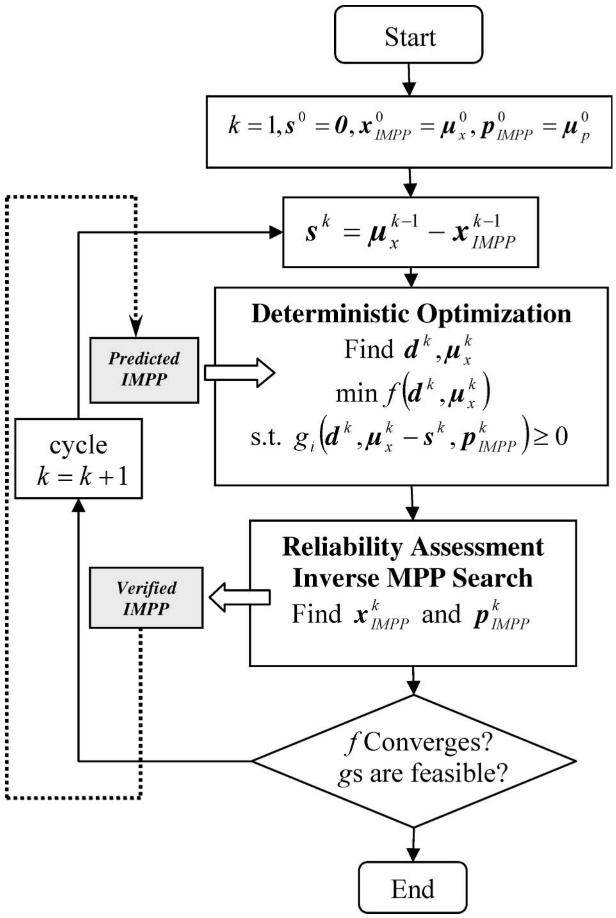

Figure 2. Flowchart of the original SORA method.

refine the feasible region if the design solution from DO is verified to be infeasible from RA.

Given the desired reliability R, the original expression of a constraint g in probabilistic design is $P\{g(\boldsymbol{d}, \boldsymbol{x}, \boldsymbol{p}) \geq 0\} \geq R$. The percentile performance g^R is the value of the limit state function g that makes the integration area under the probabilistic distribution function of g for $g \geq g^R$ exactly equal to the required reliability R. The concept of using percentile performance for feasibility assessment is identical to that used in the Performance Measure Approach (PMA) (Tu *et al.* 1999). The original probabilistic constraint is written in the equivalent R-percentile formulation as

$$g^R \geq 0, \text{ and further } g^R = g(\boldsymbol{d}, \boldsymbol{x}_{\text{IMPP}}, \boldsymbol{p}_{\text{IMPP}}) \geq 0 \tag{4}$$

The use of R-percentile formulation saves the computational effort by evaluating the design feasibility only up to the *desired* reliability level (R), which is often lower than the *actual* reliability level when a probabilistic constraint is feasible. It also converts the original probabilistic formulation to an equivalent deterministic optimization as:

$$\begin{aligned} &\text{Find} \quad \boldsymbol{d}, \boldsymbol{\mu}_x \\ &\text{Minimize} \quad f(\boldsymbol{d}, \boldsymbol{\mu}_x, \boldsymbol{p}) \\ &\text{Subject to} \quad \boldsymbol{g}(\boldsymbol{d}, \boldsymbol{x}_{\text{IMPP}}, \boldsymbol{p}_{\text{IMPP}}) \geq 0 \end{aligned} \tag{5}$$

The model in equation (5) shows that for a critical probabilistic constraint, the IMPPs should sit on the boundary of the deterministic constraint. Since the exact IMPPs of $\boldsymbol{x}$ and $\boldsymbol{p}$ at the design solution $(\boldsymbol{d}, \boldsymbol{\mu}_x)$ of the current DO are not known until the RA is implemented, they are approximated based on the exact IMPPs from the RA in the previous cycle. With the original SORA method, a shifting vector concept is applied and the ith constraint of the DO formulation in cycle $k+1$ becomes

$$g_i(\boldsymbol{d}, \boldsymbol{\mu}_x - \boldsymbol{s}^{k+1}, \boldsymbol{p}^{k+1}_{\text{IMPP}}) \geq 0 \tag{6}$$

where the shifting vector $\boldsymbol{s}^{k+1}$ in cycle $k+1$ is defined as

$$\boldsymbol{s}^{k+1} = \boldsymbol{\mu}^k_x - \boldsymbol{x}^k_{\text{IMPP}} \text{ or } s^{k+1}_i = -\sigma_{x_i} \cdot U^k_{\text{IMPP}i} \tag{7}$$

The idea behind using the shifting vector is illustrated in figure 3 for a problem having two random design variables X_1, X_2 with means being μ_1 and μ_2, respectively. It shows that when the exact IMPP from RA in cycle k, i.e. $\boldsymbol{x}^k_{\text{IMPP}}$, falls to the left side of the DO constraint, the probabilistic constraint is not feasible. To ensure that the IMPP is located on the deterministic constraint boundary, in the next cycle $k+1$, the new boundary marked as the dotted line is moved from the previous deterministic constraint boundary by the shifting vector $\boldsymbol{s}^{k+1}$. We can also interpret the shift concept by relating equations (4) and (6) to the R-percentile formulation of constraints at cycle $k+1$, i.e.

$$g_i(\boldsymbol{d}, \boldsymbol{x}^{k+1}_{\text{IMPP}}, \boldsymbol{p}^{k+1}_{\text{IMPP}}) \geq 0 \tag{8}$$

Relating to equations (6) and (7), we are basically using the following predictions for the IMPP at cycle $k+1$ with the SORA method:

$$\begin{aligned} \boldsymbol{x}^{k+1}_{\text{IMPP}} &\approx \boldsymbol{\mu}^{k+1}_x - \boldsymbol{s}^{k+1} = \boldsymbol{\mu}^{k+1}_x - (\boldsymbol{\mu}^k_x - \boldsymbol{x}^k_{\text{IMPP}}) \\ &= \boldsymbol{\mu}^{k+1}_x - \boldsymbol{\mu}^k_x + \boldsymbol{x}^k_{\text{IMPP}} \end{aligned} \tag{9}$$

$$\boldsymbol{p}^{k+1}_{\text{IMPP}} \approx \boldsymbol{p}^k_{\text{IMPP}} \tag{10}$$

As more mathematical details will be revealed later in section 3.3, equations (9) and (10) provide accurate estimations of IMPPs when the limit state function is linear in the x domain and when the variances of $\boldsymbol{x}$ are constants (variance of $\boldsymbol{p}$ is constant by default). Under these conditions, the SORA method should find the optimal probabilistic solution in cycle 2. Under general conditions, it often takes a few more cycles to reach the final solution while the feasibility of a solution progressively

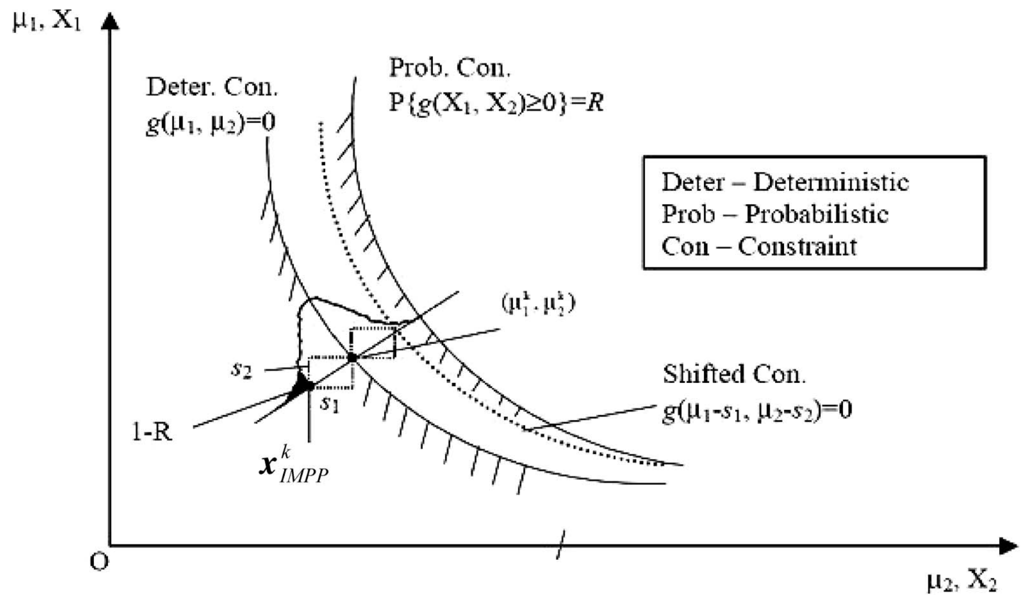

Figure 3. Shifting distance of probabilistic boundaries.

improves and the estimations of IMPPs get closer to the real values.

The SORA method has been successfully applied to structure design applications such as vehicle crashworthiness and other engineering problems (Liu *et al.* 2003, Du and Chen 2004). Compared to the double-loop method, the SORA method is shown to be much more efficient. However, the original SORA method employs a shifting vector that only depends on the information of the previous cycle, but is irrelevant to the design point and design variance in a new cycle. In theory, the method should perform perfectly in the case of linear constraints containing random variables with constant variances. For general conditions, especially in the case that the variance of a design variable is changing with respect to the mean location, improved formulations need to be developed to achieve higher efficiency for convergence.

3. Enhancing SORA for problems with changing variances

In this work, we consider three new formulations to incorporate changing variances into the formulation of SORA method. The original shifting vector formulation in SORA approach is also tested for its efficiency in dealing with changing variance. We call it Approach 0. All new approaches (Approaches $1-3$) follow the same strategy of decoupling the deterministic optimization (DO) and reliability assessment (RA) as in SORA. They deviate in the formulations for predicting the IMPP in a new cycle. The principles of these different approaches are reviewed in section 3. Empirical results are provided in section 4.

3.1 *Approach 0 (original SORA): Shifting Vector Formulation*

Developed directly from the original SORA, this approach uses the same DO formulation as in the original SORA method (equation (6)), but updates the variance right before the RA when verifying the exact IMPP. Assuming that the deviation of a random design variable is a function of its mean value, the core part of Approach 0 is shown in figure 4. In principle, this approach should be able to progressively improve the design solution by shifting the constraint boundary to the feasible region based on the exact IMPP obtained from the most recent cycle. However, because the changing variance increases the computational effort for the IMPP search and the formulation used in DO does not incorporate the changing variance information, this approach is expected to take more cycles to reach the final optimal solution compared to applying the same shifting vector formulation to problems with constant variance.

3.2 *Approach 1: Approximated* $\mathbf{u}_{IMPP}$ *Formulation*

Based on equation (3), the following relationship should hold at cycle $k+1$:

$$X_{\mathrm{IMPP}i}^{k+1} = \mu_{x_i}^{k+1} + \sigma_{x_i}^{k+1} \cdot U_{\mathrm{IMPP}i}^{k+1} \tag{11}$$

Approach 1 approximates the IMPP in a new cycle by replacing $U_{\mathrm{IMPP}i}^{k+1}$ in the above equation by $U_{\mathrm{IMPP}i}^{k}$, i.e.

$$X_{\mathrm{IMPP}i}^{k+1} \approx \mu_{x_i}^{k+1} + \sigma_{x_i}(\mu_{x_i}^{k+1}) \cdot U_{\mathrm{IMPP}i}^{k} \tag{12}$$

At Cycle k+1

1. Define shifting vector $\boldsymbol{s}^{k+1} = \boldsymbol{\mu}_x^k - \boldsymbol{x}_{IMPP}^k$
2. Deterministic Optimization to find $\boldsymbol{d}^{k+1}$ and $\boldsymbol{\mu}_x^{k+1}$

$$\min f\left(\boldsymbol{d}^{k+1}, \boldsymbol{\mu}_x^{k+1}\right)$$
$$\text{s.t. } g_i\left(\boldsymbol{d}^{k+1}, \boldsymbol{\mu}_x^{k+1} - \boldsymbol{s}^{k+1}, \boldsymbol{p}_{IMPP}^k\right) \le 0$$

3. Update deviation as $\sigma_{x_i}^{k+1} = \sigma_{x_i}(\mu_{x_i}^{k+1})$
4. Reliability Assessment to find $\boldsymbol{x}_{IMPP}^{k+1}$ and $\boldsymbol{p}_{IMPP}^{k+1}$

Figure 4. Core of cycle $k+1$ of Approach 0.

The flow chart of Approach 1 is similar to that of Approach 0 in figure 4, except that the shifting vector is now changed to

$$s_i^{k+1} = -\sigma_{x_i}(\mu_{x_i}^{k+1}) \cdot U_{\text{IMPP}i}^k \tag{13}$$

With Approach 1, the variance information is incorporated by updating the standard deviation in equation (13) as the function of the mean value, $\sigma_{x_i}(\mu_{x_i})$, in a new cycle. However, since the $\boldsymbol{u}_{\text{IMPP}}$ information is estimated based on the exact IMPP from the previous cycle, the formulation only provides an approximation. It can be expected that if $\boldsymbol{u}_{\text{IMPP}}^{k+1}$ is not too far away from $\boldsymbol{u}_{\text{IMPP}}^k$, Approach 1 should work well and converge quickly.

3.3 *Approach 2: Direct Linear Estimation Formulation*

The proposed Approach 2 derives the $\boldsymbol{x}_{\text{IMPP}}^{k+1}$ by considering the slope information of the linearized limit state function $g(\boldsymbol{x})$ at the exact IMPP from RA cycle k. Consider a linear constraint

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} a_i X_i + a_0 \tag{14}$$

where a_i are the constraint function coefficients and $\boldsymbol{x} = (X_1, \ldots, X_n)$ is the design variable vector composed of n independent random variables with normal distributions. After transforming $\boldsymbol{x}$ to $\boldsymbol{u} = (U_1, \ldots, U_n)$ in $\boldsymbol{u}$-space using

$$U_i = \frac{X_i - \mu_{x_i}}{\sigma_{x_i}},$$

the limit state function in $\boldsymbol{u}$-space becomes

$$\begin{aligned} g(\boldsymbol{U}) &= \sum_{i=1}^{n} a_i(\sigma_{x_i} U_i + \mu_{x_i}) + a_0 \\ &= \sum_{i=1}^{n} b_i U_i + b_0 \end{aligned} \tag{15}$$

where $b_i = a_i \sigma_{x_i}$ and $b_0 = \sum_{i=1}^{n} a_i \mu_{x_i} + a_0$, $\boldsymbol{b} = [b_1, \ldots, b_n]$. As shown in figure 1, the IMPP is the point where the norm of $\boldsymbol{u}$ reaches its minimum of

$$\beta = \|\boldsymbol{u}\|_{\min} = \|\boldsymbol{u}^*\| = \frac{|b_0|}{\|\boldsymbol{b}\|} \quad \text{when } \boldsymbol{u} = \boldsymbol{u}^* \tag{16}$$

Here β satisfies the equation of $\beta = \Phi^{-1}(R)$; R is the desired reliability for the constraint and $\|\cdot\|$ is the norm of an n-dimensional vector; and $\boldsymbol{u}^*$ is a vector starting from the origin and perpendicular to the hypersurface with the norm β and the direction $\hat{\boldsymbol{b}} = -\boldsymbol{b}/\|\boldsymbol{b}\|$. The IMPP in the $\boldsymbol{u}$-space can be written as

$$U_{\text{IMPP}i} = U_i^* = \|\boldsymbol{u}^*\| \cdot \left(\frac{-\boldsymbol{b}}{\|\boldsymbol{b}\|}\right)_i = \frac{-b_i \beta}{\|\boldsymbol{b}\|} \tag{17}$$

Hence for any IMPP,

$$X_{\text{IMPP}i} = \mu_{x_i} + \sigma_{x_i} U_{\text{IMPP}i} = \mu_{x_i} - \frac{b_i \beta \sigma_{x_i}}{\|\boldsymbol{b}\|} \tag{18}$$

From equation (18), we can see that for random design variables with *constant* variances, if the slope, i.e. $\boldsymbol{a} = [a_1, \ldots, a_n]$, of the linearized limit state functions does not change too much when the IMPP moves from cycle to cycle, the second item in equation (18) is almost a fixed value no matter where $\boldsymbol{\mu}_x$ is. This explains the reason why in the original SORA method, for constant variance, the difference between $\boldsymbol{x}_{\text{IMPP}}$ and $\boldsymbol{\mu}_x$ is treated as a constant vector. When the variance is changing, we use equation (18) to predict IMPP in a new cycle $k+1$ as,

$$X_{\text{IMPP}i}^{k+1} \approx \mu_{x_i}^{k+1} - \frac{b_i \beta \sigma_i(\mu_{x_i}^{k+1})}{\|\boldsymbol{b}\|} = \overline{h}_i(\boldsymbol{\mu}_x^{k+1}) \tag{19}$$

where $\boldsymbol{b}$ is obtained by linearizing the $g(\boldsymbol{x})$ function at the exact IMPP from RA in cycle k. The whole prediction function is represented as $\overline{h}_i(\boldsymbol{\mu}_x)$. The bar above h_i means the prediction is based on linearizing a limit state function.

In Approach 2, the formulation of IMPP estimation may look identical with other existing methods that introduce the linearization of limit state functions. However, in Approach 2, the tangent information of the limit state function is evaluated at the verified (true) IMPP from the previous cycle. Therefore the accuracy in IMPP estimation may be improved. Furthermore, all the tangent information is calculated as a part of IMPP search without additional cost.

Based on its principle, Approach 2 is expected to solve a probabilistic optimization problem with linear constraints in no more than three cycles: the first cycle for the initial deterministic optimization, the second cycle for locating a design solution with IMPP exactly at the boundary of DO, and the last one for convergence check. For nonlinear constraints, it is expected that Approach 2 should be quite efficient if the slope of a limit state function does not change too much when the IMPP moves from cycle to cycle.

3.4 *Approach 3: Quasi First Order Approximation of* $\mathbf{x}_{IMPP}$ *Formulation*

Approach 1 directly utilizes the exact IMPP from the previous cycle, and Approach 2 uses the tangent information of a limit state function at the exact IMPP, to estimate the IMPP in a new cycle. Different from other existing methods for IMPP estimation, Approach 3 employs the Taylor expansion of the IMPP in $\boldsymbol{x}$-space and provides the estimation of IMPP based on both the information of the exact IMPP and the slope of a limit state function from the previous cycle.

For nonlinear functions, suppose $\boldsymbol{x}_{\text{IMPP}}$ is a function of $\boldsymbol{\mu}_x$ as $\boldsymbol{x}_{\text{IMPP}} = h(\boldsymbol{\mu}_x)$, no matter whether the variance is fixed or changing. Here the function h is unknown and cannot be expressed explicitly. In DO formulation of cycle $k+1$, let

$$\begin{aligned}\boldsymbol{x}_{\text{IMPP}}^{k+1} &= h(\boldsymbol{\mu}_x^{k+1}) \approx h(\boldsymbol{\mu}_x^{k}) + \frac{\partial h^k}{\partial \boldsymbol{\mu}_x}(\boldsymbol{\mu}_x^{k+1} - \boldsymbol{\mu}_x^{k}) \\ &= \boldsymbol{x}_{\text{IMPP}}^{k} + \frac{\partial h^k}{\partial \boldsymbol{\mu}_x}(\boldsymbol{\mu}_x^{k+1} - \boldsymbol{\mu}_x^{k})\end{aligned} \tag{20}$$

With our approach, h is approximated in the form of $\overline{h}$, see equation (21). The $\overline{h}$ is defined in equation (19) based on the linearized limit state function. With this approximation, the partial derivatives in equation (20) can be derived from the analytical expression of $\overline{h}$ instead of the unknown function h. We call this method 'Quasi First Order Approximation' because it is not necessary to evaluate the first-order derivatives in equation (20) numerically.

$$\boldsymbol{x}_{\text{IMPP}}^{k+1} \approx \boldsymbol{x}_{\text{IMPP}}^{k} + \frac{\partial \overline{h}^k}{\partial \boldsymbol{\mu}_x}(\boldsymbol{\mu}_x^{k+1} - \boldsymbol{\mu}_x^{k}) \tag{21}$$

With the analytical approach, the ith component of $\overline{h}$ is $\overline{h}_i$ in equation (19). In the case of $\sigma_{x_i} = r_i \mu_{x_i}$, where the coefficients of variation r_i are constants, the partial derivatives can be evaluated as:

$$\frac{\partial \overline{h}_i^k}{\partial \mu_{x_j}} = -1 + \frac{2X_{\text{IMPP}i}^k}{\mu_{x_i}} + \frac{\left(\mu_{x_i} - X_{\text{IMPP}i}^k\right)^3 a_i^2 r_i^2}{(a_i \beta r_i^2)^2 \mu_{x_i}^3}\Bigg|\, \mu_{x_i} = \mu_{x_i}^k \quad \text{for } i = j \tag{22}$$

$$\frac{\partial \overline{h}_i^k}{\partial \mu_{x_j}} = \frac{a_j^2 r_j^2 \mu_{x_j}\left(\mu_{x_i} - X_{\text{IMPP}i}^k\right)^3}{(a_i \beta r_i^2)^2 \mu_{x_i}^4}\Bigg|\, \mu_{x_i} = \mu_{x_i}^k, \mu_{x_j} = \mu_{x_j}^k \quad \text{for } i \neq j \tag{23}$$

Note that if the partial derivative is close to one (i.e. the change of IMPP in $\boldsymbol{x}$-domain is the same as the change of μ_x from cycle k to $k+1$), the above equation of $\boldsymbol{x}_{\text{IMPP}}^{k+1}$ in equation (21) degenerates into the expression used in the original SORA as in equation (9). Similar to Approach 2, $\boldsymbol{b}$ is obtained by linearizing the $g(\boldsymbol{x})$ function at the exact IMPP from RA in cycle k without the need for additional function evaluations. For a random design variable X_i with constant variance, the partial derivatives in equation (22) and (23) are taken as one and zero, respectively. Under such condition, the IMPP estimation formulation is the same as the one used in the original SORA.

Because the Taylor expansion is only a local expansion in a small neighborhood of the expanding point, we expect that the estimation will not be very accurate in the first several cycles when $\boldsymbol{\mu}_x$ may change significantly. The estimation will become more accurate in later cycles when design points do not differ too much.

If we consider problems with constant variances as a special case of using all four approaches presented, we expect that all the methods except Approach 2 should behave the same due to the same estimation formulation of IMPP when taking the variances as constants. Approach 2 differs slightly since its estimation of IMPP is based on linearized constraints. In section 4, all these four approaches are applied in several examples to check their validity and effectiveness.

3.5 *Implementation for different types of distributions*

In the previous sections that introduce the different proposed formulations, we have assumed that $\boldsymbol{x}$ follow normal distributions. Normal distributions are commonly used in engineering applications and a simple expression can be used to describe the linear relationship between the changing variance and the mean, i.e. $\sigma_x = r_x\, \mu_x$, where r_x is named as the *constant coefficient of variation*. For problems with non-normal distributions, the same strategies for predicting an IMPP in a new cycle can be followed, but the

μ_x and σ_x in those formulations need to be updated to μ_x^N and σ_x^N, the descriptors of mean and standard deviation for equivalent normal distributions. Following Rackwitz–Fiessler's two-parameter equivalent normal method (Rosenblatt 1952), the μ_x^N and σ_x^N of an equivalent normal distribution at a point of interest X^* can be expressed as

$$\begin{aligned} \sigma_x^N &= \phi\{\Phi^{-1}[F_x(X^*)]\}/f_x(X^*) \\ \mu_x^N &= X^* - \Phi^{-1}[F_x(X^*)]\sigma_x^N \end{aligned} \tag{24}$$

where ϕ is the Probability Density Function (PDF) of standard normal distribution and F_x, f_x are the CDF and PDF of the non-normal distribution of X, respectively. The same transformation needs to be used for searching the IMPP in the RA that follows the DO in each cycle.

For non-normal distributions, to ensure that the standard deviation of the *whole* distribution is linearly proportional to the mean of the distribution, i.e. $\sigma_x = r_x\mu_x$, special considerations can be taken to properly select the parameters of the non-normal distributions. In Appendix A, the equations for choosing parameters in exponential, uniform, Weibull, and log-normal distributions to achieve the linear relationship between mean and standard deviation are provided.

4. Comparative studies

In this section, two mathematical problems and one design example are used to compare the effectiveness of the three proposed formulations as well as the original SORA approach (Approach 0). The results from the Double-loop method using the R-percentile formulation are considered as the 'exact' solutions for reference. In all examples, we assume that the random variables are normally distributed, and the changing variance is described as $\sigma_{x_i} = r_i\mu_{x_i}$, where r_i is the constant coefficient of variation for X_i.

4.1 *Problem 1 with linear constraints*

We consider only linear constraints in the first mathematical example. The probabilistic optimization model of this example is shown in equation (25), where the objective function is nonlinear and all four constraint functions gs are linear in the original design space. All input variables are random design variables (X_1 to X_6) with normal distributions $N(\mu_i, \sigma_i)$, $i = 1, \ldots, 6$. The desired reliability is 0.99865 for each constraint. For each approach, we examine two cases with different values of coefficients of variation and two other cases with different constant variances.

$$\begin{aligned} &\min \quad f(\boldsymbol{x}) = \frac{\mu_1\mu_2 - \mu_4^2}{\mu_3} - \sqrt{\mu_5}\mu_6^3 \\ &\text{s.t.} \quad P(g_i(\boldsymbol{x}) \geq 0) \geq R_i \quad i = 1, \ldots, 4 \\ &\quad g_1(\boldsymbol{x}) = -X_1 + 3X_2 - 5 \geq 0 \\ &\quad g_2(\boldsymbol{x}) = -X_1 - 2X_3 - X_6 + 10 \geq 0 \\ &\quad g_3(\boldsymbol{x}) = X_1 + 2X_4 - X_5 - 8 \geq 0 \\ &\quad g_4(\boldsymbol{x}) = X_2 - 7X_6 + 2 \geq 0 \\ &\quad 1 \leq \mu_1 \leq 10, \quad 2 \leq \mu_2 \leq 8 \\ &\quad 3 \leq \mu_3 \leq 8, \quad 3 \leq \mu_4 \leq 8 \\ &\quad 1 \leq \mu_5 \leq 6, \quad 0.1 \leq \mu_6 \leq 2 \end{aligned} \tag{25}$$

Table 1 illustrates the efficiency and accuracy of the various approaches considered for changing variances. The numbers of function calls used for both DO and RA are also listed. With a numerical tolerance of 1%, we find that all approaches reach the same optimal solution for both cases. All three proposed approaches (1–3) improve the efficiency greatly compared to the double-loop method. The three proposed approaches also achieve better performance compared to the original SORA method, where the shifting vector used in DO only uses the information of the exact IMPP from the previous cycle but does not reflect the effect of changing variance. It is noted that the magnitude of

Table 1. Comparison for Example 1 ($\sigma_i = 0.02\mu_i$ and $\sigma_i = 0.15\mu_i$).

					Function calls		
	Approaches	x	f	Cycle	Total	DO	RA
$r_i = 0.02$	Original SORA	$x = [1.0000, 8.0000, 3.0000, 8.0000, 6.0000, 1.3236]$	−24.3472	4	185	112	73
	Approach 1			3	149	91	58
	Approach 2			3	149	91	58
	Approach 3			3	149	91	58
	Double loop			N/A	1804	N/A	N/A
$r_i = 0.15$	Original SORA	$x = [1.0000, 3.6479, 3.0000, 8.0000, 1.7444, 0.2603]$	−20.1406	6	388	231	157
	Approach 1			4	224	133	91
	Approach 2			3	192	127	65
	Approach 3			4	224	133	91
	Double loop			N/A	1629	N/A	N/A

coefficients of variations, $\boldsymbol{r}_x$, has some impact on the efficiency of all methods except Approach 2, which is shown to be the most efficient in all cases. This is reasonable because Approach 2 predicts the new IMPP by linearizing the limit state function. Since the original constraints are all linear, Approach 2 requires at most three cycles (including one cycle for convergence check) to find the final optimal solution no matter how large the variance is. On the other hand, Approaches 1 and 3 are just as efficient as Approach 2 when the coefficient of variation is small; but require more cycles when the coefficient becomes larger.

The original SORA method and the enhanced approaches generally require more function calls when the coefficient becomes larger. This may not be the case for the double-loop method as the latter has a different working principle. As shown in table 1, the total number of function calls used by the double-loop method for the case of $r_i = 0.15$ is smaller than the case of $r_i = 0.02$. Part of the reason is that the larger coefficient of variation shrinks the feasible region to a smaller one and therefore expedites the convergence process because the double-loop method seeks for a feasible solution from the very beginning. The proposed SORA methods are different, in which the solution moves from an infeasible region to a feasible one progressively.

The results of two cases with constant variances are presented in table 2. Except the double-loop method, all approaches converge to the same final optimal solution at Cycle 3. Even though the constant variance has different magnitude in two tested cases, because all constraints are linear, both the number of cycles and the number of function calls stay the same, respectively.

Table 2. Comparison for Example 1 (constant variation).

	$\sigma_i = 0.02$				$\sigma_i = 0.15$			
		Function calls				Function calls		
	Cycles	Total	DO	RA	Cycles	Total	DO	RA
Original SORA Approach 1 Approach 2 Approach 3	3	135	84	51	3	135	84	51
Double loop	N/A	1569			N/A	1793		

4.2 *Problem 2 with nonlinear constraints*

Problem 2 is developed from the example introduced in Liang *et al.* (2004) with two random design variables X_1 and X_2. Both of them are normally distributed with means of μ_1, μ_1, and deviations σ_1, σ_1 respectively. The problem has three nonlinear constraints with the same desired reliability of 0.99865. The formulation used by Liang *et al.* (2004) is modified slightly here in the first constraint as shown in equation (26), which helps enlarge the feasible design region to explore the behaviors of our proposed approaches for large changing variances. The results with only two design variables facilitate the graphical illustration to help better understand the proposed approaches. Problem 2 is again examined for two cases with different values of coefficients of variation. The comparison of results from different approaches is provided in table 3.

$$
\begin{aligned}
\min \quad & f(\boldsymbol{x}) = \mu_1 + \mu_2 \\
\text{s.t.} \quad & P(g_i(\boldsymbol{x}) \geq 0) \geq R_i \\
& g_1(\boldsymbol{x}) = X_1^2 + \frac{X_2}{20} - 1 \geq 0 \\
& g_2(\boldsymbol{x}) = \frac{(X_1 + X_2 - 5)^2}{30} + \frac{(X_1 - X_2 - 12)^2}{120} - 1 \geq 0 \\
& g_3(\boldsymbol{x}) = \frac{80}{X_1^2 + 8X_2 + 5} - 1 \geq 0 \\
& 0.01 \leq \mu_i \leq 10, \quad i = 1, 2
\end{aligned}
\tag{26}
$$

Within the numerical tolerance of 0.02% for convergence, all approaches generate the same optimal solutions for constant and changing variances. From the results listed in tables 3 and 4, it is noted that for all approaches, it generally needs more cycles and function calls to reach the final optimal solutions for cases with larger constant variances or larger coefficients of variation. For problems with constant variance, all proposed approaches as well as the original SORA method are more efficient than the double-loop method. While dealing with changing variance, the original

Table 3. Results of Example 2 for constant variances.

	$\sigma_i = 0.02$				$\sigma_i = 0.20$				$\sigma_i = 0.40$			
		Function calls				Function calls				Function calls		
	Cycles	Total	DO	RA	Cycles	Total	DO	RA	Cycles	Total	DO	RA
Original SORA Approach 1 Approach 2 Approach 3	3	87	46	41	3	117	55	62	4	295	100	195
Double loop	N/A		491		N/A		644		N/A		1004	

SORA method is the most sensitive to the coefficient of variation. It requires much more cycles and function calls compared to the three proposed approaches when the coefficient becomes larger. For problems with changing variance, because the variance is location-dependent, the exact IMPP becomes more sensitive to the location of a design, posing challenges for IMPP estimation. Among all approaches, Approach 2 is shown to be the most effective in all cases. Especially when $r_i = 0.20$, Approach 2 requires the least number of cycles, and therefore the least number of function calls for DO and RA, respectively. In figure 5, we graphically illustrate the locations of the predicted (estimated) IMPPs and the exact IMPPs from cycle to cycle in the decoupled single loop process with the formulation used with Approach 2, for $\boldsymbol{r} = [0.2\ 0.2]$. The IMPPs illustrated are only associated with g_2, an active constraint at the optimal solution.

The numbers 1–4 in figure 5 denote the cycle numbers from 1 to 4. X, P and E refer to the design point (obtained from DO), predicted IMPP (using Approach 2) and the exact IMPP (verified by RA), respectively. The feasible region is between the boundaries of constraints 1 and 2. In the first cycle, since no information of IMPP exists, the IMPP is set as the design point. The exact IMPP (E1) verified from the RA shows that E1 deviates from P1. The information of E1 is used to predict the IMPP in the next cycle (P2). It shows that P2 sits exactly on the boundary of equivalent deterministic constraint 2. However, after RA, it is found that the exact IMPP (E2) locates in the infeasible region. In Cycle 3, the information of E2 is used to predict

Table 4. Results of Example 2 for changing variances.

	$r_i = 0.02$				$r_i = 0.10$				$r_i = 0.20$			
		Function calls				Function calls				Function calls		
	Cycles	Total	DO	RA	Cycles	Total	DO	RA	Cycles	Total	DO	RA
Original SORA	6	141	76	65	13	289	151	138	18	795	250	545
Approach 1	3	81	46	35	3	90	52	38	6	369	94	275
Approach 2	3	81	46	35	3	90	52	38	4	268	73	195
Approach 3	3	81	46	35	3	90	52	38	5	329	85	244
Double loop	N/A		491		N/A		626		N/A		794	

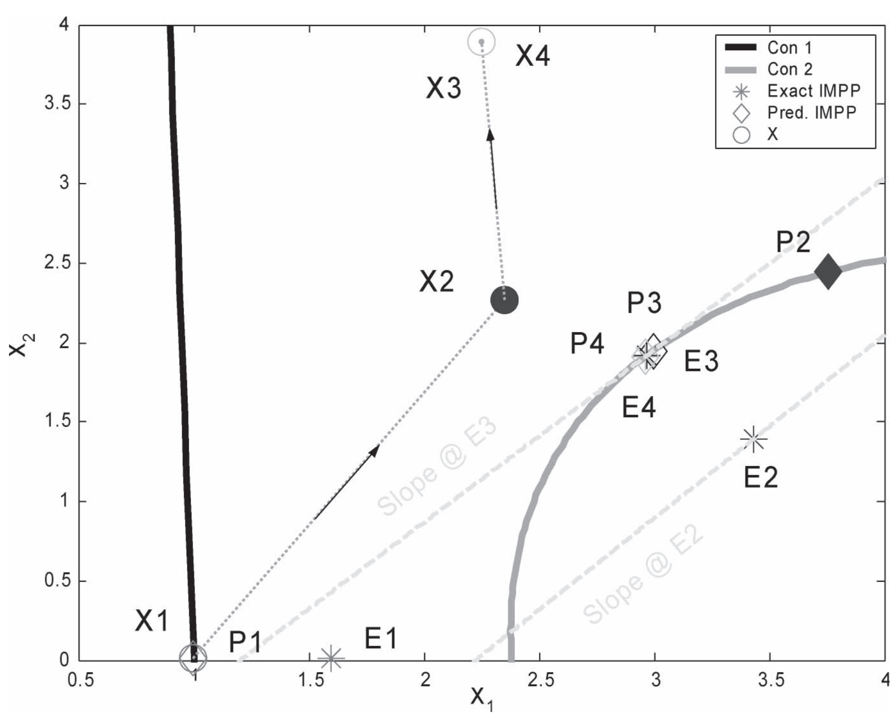

Figure 5. History of predicted and exact IMPPs with Approach 2 (Example 2).

P3. It shows that the predicted P3 is quite close to the exact one (E3), almost overlapping with each other on the constraint boundary. Cycle 3 brings an infeasible solution back to a feasible one. The final cycle (cycle 4) provides the convergence check of the optimal result. The locations of P4 and E4 almost overlap with those of P3 and E3. From this illustration, we note that Approach 2 works quite effectively even though the locations of design point X vary quite a lot and the variance is large from cycle to cycle. From the plot of the slopes of the constraint functions at the exact IMMPs in cycles 2 and 3, i.e. E2 and E3, respectively (dash lines in figure 5), we see that the slopes at these two IMMPs are almost identical even though the design point has moved quite a lot. This matches with the assumption used in Approach 2 for IMPP prediction and results in good estimations.

To compare the effectiveness of the IMPP estimation using different approaches, the discrepancies of the estimated IMPPs and the exact IMPPs are plotted in figure 6. All the discrepancies are described in terms of the distances between the estimated and the exact IMPPs.

Discrepancies of all approaches in the first cycle are the same because the IMPP is set as the design point and the DO produces the same result. All discrepancies go down to zero with the increase of cycles, indicating all approaches are able to converge and to finally match the estimated and the exact IMPPs despite the different cycles needed. Compared to the original SORA approach (Approach 0), Approaches 1, 2 and 3 have significantly improved the efficiency. There is a little oscillation in the curve of Approach 0 before the discrepancy vanishes. That is because the shifting vector used in the original SORA method does not take the change of design point and the effect of changing variance into account. In the first several cycles, because the design points differ a lot compared to the later cycles, the effect of changing variance causes relatively larger discrepancies in prediction. Approaches 2 and 3 provide better estimations of the IMPP compared to Approach 1. Overall, Approach 2 is shown to be the most efficient.

4.3 *A speed reducer design problem*

In this engineering example, the task is to design a simple gear box of a small aircraft engine that can rotate at its most efficient speed. The original design was modeled in Golinski (1970) and then used as an artificial multidisciplinary optimization problem in Boden and Grauer (1995). The probabilistic optimization formulation used in this work is slightly modified from the formulation used in Du (2002). More details on the model formulation and random parameters can be found in Appendix B.

Reliability constraints are considered in the probabilistic formulation for the speed reducer design because of the random quantities involved. Among all the parameters, the teeth module and the number of pinion teeth are the deterministic design variables. There are five random design variables including face width (X_1), shaft-length 1 (X_2), shaft-length 2 (X_3), shaft diameter 1 (X_4), and shaft diameter 2 (X_5). The problem also involves 15 random parameters $P_1 - P_{15}$, including the material properties, the rotation speed, and the engine power. All the random quantities are normally distributed. In this example, the task is to minimize the weight of the speed reducer while subject to 10 probabilistic constraints and one deterministic

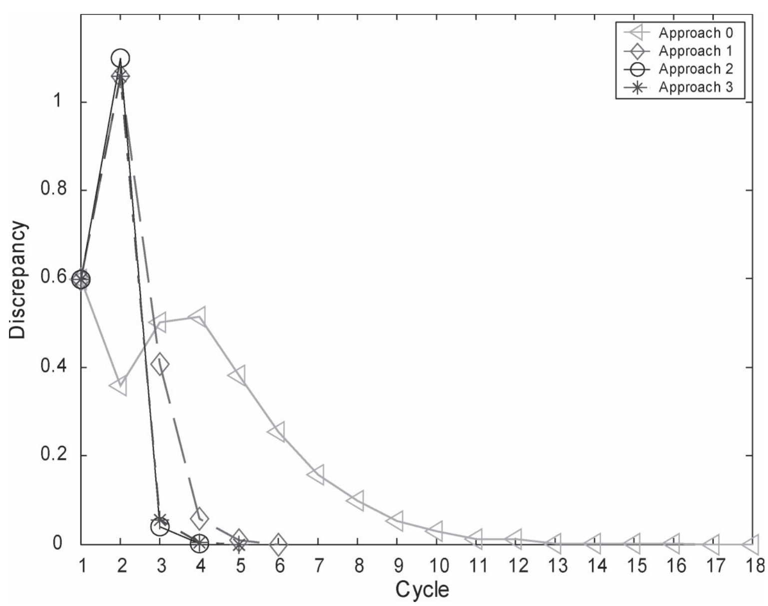

Figure 6. Discrepancy of the estimated and exact IMPPs ($r_i = 0.20$).

constraint. Various approaches are used to solve the problem by assuming that the variances of all five random design variables $\boldsymbol{x}=[X_1\ X_2\ X_3\ X_4\ X_5]$ are changing with the coefficients $\boldsymbol{r}=[0.05\ 0.05\ 0.05\ 0.02\ 0.02]$.

Table 5 lists the comparison of results when setting $R=95\%$ for all probabilistic constraints. Approaches 0, 1, and 3 reach the same optimal solution while Approach 2 provides a slightly conservative result. The double-loop method generates a solution that is in between the proceeding two groups. For this example problem, it is not clear which among the three proposed approaches is superior to the others. However, all the proposed approaches are much more efficient than the double-loop method and the original SORA method (Approach 0), since they take less cycles and function calls.

The reason why Approach 2 provides a conservative solution is further investigated. From table 5, we can see that Approach 2 only requires three cycles to reach the optimal solution. It is found that in the second cycle, the IMPP is overestimated which leads to a conservative result. A part of the problem is that when updating vector $\boldsymbol{b}$ as defined in equations (15) and (16), only the randomness associated with the five random design variables $\boldsymbol{x}$ but not the 15 random design parameters $\boldsymbol{p}$ were incorporated, i.e. it is assumed that $\boldsymbol{p}_{\mathrm{IMPP}}^{k+1} \approx \boldsymbol{p}_{\mathrm{IMPP}}^{k}$. However, since a large portion of the randomness in the problem is associated with the random parameters, the simplification leads to discrepancy in the estimation.

Modification is made by considering all 20 random variables when evaluating $\boldsymbol{b}$, with more information of slope and variance introduced by random parameters $\boldsymbol{p}$. Table 6 provides the optimal solution with the modified Approach 2. It is noted that the new optimal solution from Approach 2 is identical with the ones using Approaches 0, 1, and 3. However, Approach 2 now requires more computational effort as the cycle number goes up to six.

We conclude from this study that when a large number of random parameters exist in a problem, the efficiency of Approach 2 might not be superior to other proposed approaches because the increased function calls are involved in accurately assessing the tangent information of a limit state function.

5. Conclusions

In this paper, to accommodate the effect of varying design variance in probabilistic optimization, the SORA method is enhanced by introducing three different approaches to modify the DO formulation. These methods share the common strategy as the original SORA method in decoupling DO from RA. The three new approaches are distinguished by the different strategies of IMPP estimations in the formulation of the DO. By examining the mathematical principles and through empirical studies, we find that our proposed approaches inherit the high efficiency of the original SORA, while improving the efficiency of solving probabilistic optimization problems with varying design variance, in particular for cases with large coefficients of variation.

Among them, the Approximated $\boldsymbol{u}_{\mathrm{IMPP}}$ Approach (Approach 1) approximates the IMPP by using the previous IMPP in the normalized space as well as updating the mean and variance based on the location of the new design point. Since the IMPP in the $\boldsymbol{u}$-space often changes when the design point changes, this estimation is quite rough. Approach 1 is shown to be the least efficient among the three new approaches. The Direct Linear Estimation Formulation (Approach 2) provides the estimation of the IMPP using the tangent information of a limit state function at the verified IMPP from the previous cycle, in addition to updating the mean and variance information at the new design point. As a whole, Approach 2 is shown to

Table 5. Comparison of approaches for Example 3.

	$\boldsymbol{r}=[0.05\ 0.05\ 0.05\ 0.02\ 0.02]$				
$\sigma_i=r_i\mu_i$	Approach 0	Approach 1	Approach 2	Approach 3	Double loop
d_1	0.7000	0.7000	0.7000	0.7000	0.7000
d_2	17.0000	17.0000	17.0000	17.0000	17.0000
X_1	3.8618	3.8618	4.0308	3.8619	3.7775
X_2	7.0000	7.0000	7.0000	7.0000	7.0000
X_3	7.0000	7.0000	7.0000	7.0000	7.0000
X_4	2.9326	2.9326	2.9409	2.9326	3.1976
X_5	5.0000	5.0000	5.0000	5.0000	5.0000
F	2857.24	2857.24	2926.63	2857.25	2878.97
Cycle	5	3	3	4	N/A
Fun. calls	505	351	359	454	6880
FC DO	216	152	160	210	N/A
FC RA	289	199	199	244	N/A

Fun. call, function calls; FC DO, function calls for DO; FC RA, function calls for RA.

Table 6. Optimal solution using the modified Approach 2.

$\sigma_i = r_i\mu_i$	$\boldsymbol{r} = [0.05\ 0.05\ 0.05\ 0.02\ 0.02]$ Modified Approach 2
d_1	0.7000
d_2	17.0000
X_1	3.8619
X_2	7.0000
X_3	7.0000
X_4	2.9326
X_5	5.0000
F	2857.27
Cycle	6
Fun. call	592
FC DO	258
FC RA	334

Fun. call, function calls; FC DO, function calls for DO; FC RA, function calls for RA.

be the most efficient method in the examples tested, especially when the changing variance is large. Through the empirical study, we find that Approach 2 works effectively because in most applications the slope of the limit state function at IMPP does not change too much even when the design point varies. It should be noted that additional function evaluations may be needed for this approach when the number of random parameters is relatively large and those parameters have to be introduced in assessing the slope of a limit state function. The Quasi First Order Approximation of $\boldsymbol{x}_{\mathrm{IMPP}}$ Formulation (Approach 3) provides the estimation of an IMPP by employing the first-order Taylor expansion at the exact IMPP from the previous cycle. The mean and variance information are also updated at the new design point. Because Taylor expansion is only a local expansion in a small neighborhood of the expanding point, the estimation is shown to be less accurate in the first several cycles (when the design point varies significantly) compared to the later cycles (when solution converges). The accuracy of Approach 3 can be further improved as a topic for future work.

In conclusion, our study further illustrates that the strategy used in SORA for decoupling DO and RA is an effective approach for improving the accuracy in IMPP estimation even in the case of varying variance. All three proposed approaches overcome the limitation of the original SORA method by incorporating the information of changing variance. The insight gained from our study on the applicability of different approaches can be extended and utilized in other probabilistic optimization strategies that require MPP or IMPP estimations.

Acknowledgements

The NSF grant DMI0335880 and the grant from Ford University Research Program are acknowledged.

References

Boden, H. and Grauer, M., OpTiX-II: a software environment for the parallel solution of nonlinear optimization problems. *Ann. Oper. Res.*, 1995, **58**, 129–140.

Breitung, K., Asymptotic approximations for multinormal integrals. *J. Eng. Mech. ASCE*, 1984, **110**, 357–366.

Carter, A.D.S., *Mechanical Reliability and Design* (Wiley: New York), 1997.

Chen, X., Hasselman, T.K., and Neill, D.J., Reliability based structural design optimization for practical applications, in *38th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference and Exhibit and AIAA/ASME/AHS Adaptive Structural Forum*, Kissimmee, Florida, 1997, pp. 2724–2732, AIAA-97-1403.

Der Kiureghian, A., Lin, H.Z., and Hwang, S.J., Second-order reliability approximations. *J. Eng. Mech. ASCE*, 1987, **113**, 1208–1225.

Du, X. and Chen, W., Towards a better understanding of modeling feasibility robustness in engineering design. *J. Mech. Des.*, 2000, **122**, 385–394.

Du, X. and Chen, W., A most probable point based method for uncertainty analysis. *J. Des. Manufact. Automat.*, 2001, **1**, 47–66.

Du, X. and Chen, W., 2004, Sequential Optimization and Reliability Assessment method for efficient probabilistic design. *J. Mech. Des.*, **126**, 225–233.

Du, X., Efficient methods for engineering design under uncertainty. PhD thesis, Department of Mechanical and Industrial Engineering, University of Illinois at Chicago, 2002.

Du, X., Sudjianto, A., and Chen, W., An integrated framework for optimization under uncertainty using inverse reliability strategy. *Trans. ASME*, 2004, **126**, 562–570.

Fiessler, B., Newmann, H.J., and Rackwitz, R., Quadratic limit states in structural reliability. *J. Eng. Mech. ASCE*, 1979, **1095**, 661–676.

Golinski, J., Optimal synthesis problems solved by means of nonlinear programming and random methods. *J. Mechanisms*, 1970, **5**, 287–309.

Grandhi, R.V. and Wang, L.P., Reliability-based structural optimization using improved two-point adaptive nonlinear approximations. *Finite Elem. Anal. Des.*, 1998, **29**, 35–48.

Gu, L. and Yang, R.J., Recent applications on reliability-based optimization of automotive system, in *SAE 2003 World Congress and Exhibition*, Detroit, MI, March 2003, 2003-01-0152.

Hasofer, A.M. and Lind, N.C., Exact and invariant second-moment code format. *J. Eng Mech. Div.*, 1974, **100**(EM1), 111–121.

Huang, H.-Z., Tian, Z.-G., and Zuo, M.J., Intelligent interactive multi-objective optimization method and its application to reliability optimization. *IIE Trans.*, 2005, **37**, 983–993.

Liang, J., Mourelatos, Z.P., and Tu, J., A single-loop method for reliability-based design optimization, in *ASME 2004 Design Engineering Technical Conferences and Computer and Information in Engineering Conference*, Salt Lake City, Utah, 28 September–2 October 2004, DETC 2004-57255.

Liu, H., Chen, W., Sheng, J., and Gea, H.C., Application of the sequential optimization and reliability assessment method to structural design problems, in *ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Chicago, Illinois, 2–6 September 2003, DETC 2003/DAC-48710.

Melchers, R.E., *Structural Reliability Analysis and Prediction* (John Wiley & Sons: Chichester), 1999.

Rackwitz, R. and Fiessler, B., Structural reliability under combined random load sequences. *J. Comput. Struct.*, 1978, **9**, 484–494.

Reddy, M.V., Grandhi, R.V., and Hopkins, D.A., Reliability based structural optimization: a simplified safety index approach. *Comput. Struct.*, 1994, **53**, 1407–1418.

Rosenblatt, M., Remarks on a multivariate transformation. *Ann. Math. Stat.*, **23**, 1952, 470–472.

Sopory, A., Mahadevan, S., Mourelatos, Z.P., and Tu, J., Decoupled and single loop methods for reliability-based optimization and robust design, in *ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Salt Lake City, Utah, 28 September – 2 October 2004, DETC 2004-57404.

Sues, R.H. and Cesare, M., An innovative framework for reliability-based MDO, in *41st AIAA/ASME/ASCE/AHS/ASC SDM Conference*, Atlanta, Georgia, 3 – 6 April 2000, AIAA-2000-1509.

Sundaresan, S., Ishii, K., and Houser, D.R., A robust optimization procedure with variations on design variables and constraints. *Eng. Optimiz.*, 1995, **24**, 101 – 117.

Tu, J., Choi, K.K., and Park, Y.H., A new study on reliability-based design optimization. *J. Mech. Des.*, 1999, **121**, 557 – 564.

Wu, Y.-T. and Wang, W., A new method for efficient reliability-based design optimization, in *Probabilistic Mechanics & Structural Reliability: Proceedings of the 7th Special Conference*, 1996, pp. 274 – 277.

Wu, Y.-T., Computational methods for efficient structural reliability and reliability sensitivity analysis. *AIAA J.*, 1994, **32**, 1717 – 1723.

Wu, Y.-T., Shin, Y., Sues, R., and Cesare, M., Safety-factor based approach for probability-based design optimization, in *42nd AIAA SDM Conference*, Seattle, Washington, 16 – 19 April 2001, AIAA-2001-1522.

Youn, B.D., Choi, K.K., and Park, Y.H., Hybrid analysis method for reliability-based design optimization. *J. Mech. Des.*, 2003, **125**, 221 – 232.

Yu, X., Chang, K.H., and Choi, K.K., Probabilistic structural durability prediction. *AIAA J.*, 1998, **36**, 628 – 637.

Appendix A

The following equations are used to select the parameters of the non-normal distributions, to ensure that the standard deviation of the *whole* distribution is linearly proportional to the mean of the distribution, i.e. $\sigma_x = r_x\mu_x$.

Exponential distribution (λ)

$$\begin{cases} \dfrac{1}{\lambda} = \mu \\ \dfrac{1}{\lambda^2} = \sigma^2 = r^2\mu^2 \end{cases} \Rightarrow \begin{cases} \lambda = \dfrac{1}{\mu} \\ r = \dfrac{1}{\lambda\mu} = 1 \end{cases} \tag{A1}$$

Here the coefficient of variation r must be 1 instead of any other values.

Uniform distribution (a, b)

Given the mean μ and the relation of $\sigma = r\mu$

$$\begin{cases} \dfrac{a+b}{2} = \mu \\ \dfrac{(b-a)^2}{12} = \sigma^2 = r^2\mu^2 \end{cases} \Rightarrow \begin{cases} a = \mu\left(1 - r\sqrt{3}\right) & \text{lower boundary} \\ b = \mu\left(1 + r\sqrt{3}\right) & \text{upper boundary} \end{cases} \tag{A2}$$

Log-normal distribution (a,$\tilde{\sigma}$)

$$\begin{cases} e^{a+\frac{\tilde{\sigma}^2}{2}} = \mu \\ e^{2a+\tilde{\sigma}^2}\left(e^{\tilde{\sigma}^2} - 1\right) = \sigma^2 = r^2\mu^2 \end{cases} \Rightarrow \begin{cases} a = \ln(\mu) - \dfrac{\ln\left(1 + \mathrm{r}^2\right)}{2} \\ \tilde{\sigma} = \sqrt{\ln(1 + r^2)} \end{cases} \tag{A3}$$

Weibull distribution (λ, α)

$$\begin{cases} \Gamma\left(\dfrac{1}{\alpha} + 1\right)\lambda^{-\frac{1}{\alpha}} = \mu \\ \lambda^{-\frac{2}{\alpha}}\left[\Gamma\left(\dfrac{2}{\alpha} + 1\right) - \Gamma\left(\dfrac{1}{\alpha} + 1\right)^2\right] = \sigma^2 = r^2\mu^2 \end{cases} \tag{A4}$$

The values of (λ, α) can be obtained by numerical method.

Appendix B

The means and standard deviations of the random parameters involved in the speed reducer model are listed in table A1.

The computational model of the speed reducer is defined as follows:

$$\begin{aligned} \min \quad & f(\boldsymbol{x}) = 0.7854\mu_1 d_1^2\left(3.3333d_2^2 + 14.9334d_2 - 43.0934\right) \\ & \quad - 1.5079\mu_1\left(\mu_4^2 + \mu_5^2\right) + 7.477\left(\mu_4^3 + \mu_5^3\right) \\ & \quad + 0.7854\left(\mu_2\mu_4^2 + \mu_3\mu_5^2\right) \\ \text{s.t.} \quad & P(g_i(\boldsymbol{x}) \geq 0) \geq R_i, \quad i = 1, \ldots, 10 \\ & g_{11}(\boldsymbol{x}) \geq 0 \\ & g_1(\boldsymbol{x}) = 1 - \frac{P_1}{X_1 d_1^2 d_2} \geq 0 \\ & g_2(\boldsymbol{x}) = 1 - \frac{P_2}{X_1 d_1^2 d_2^2} \geq 0 \\ & g_3(\boldsymbol{x}) = 1 - \frac{P_3 X_2^3}{X_4^4 d_1 d_2} \geq 0 \\ & g_4(\boldsymbol{x}) = 1 - \frac{P_4 X_3^3}{X_5^4 d_1 d_2} \geq 0 \\ & g_5(\boldsymbol{x}) = 1 - \frac{0.5\sqrt{P_6^2 X_2^2/\left(d_1^2 d_1^2\right) + P_7}}{X_4^3 P_5 P_8} \geq 0 \\ & g_6(\boldsymbol{x}) = 1 - \frac{0.5\sqrt{P_6^2 X_3^2/\left(d_1^2 d_1^2\right) + P_9}}{X_5^3 P_{10} P_8} \geq 0 \\ & g_7(\boldsymbol{x}) = 1 - \frac{0.5 P_{11} d_1}{X_1} \geq 0 \\ & g_8(\boldsymbol{x}) = 1 - \frac{X_1}{P_{12} d_1} \geq 0 \\ & g_9(\boldsymbol{x}) = 1 - \frac{0.5(P_{13} X_4 + P_{15})}{X_2} \geq 0 \\ & g_{10}(\boldsymbol{x}) = 1 - \frac{0.5(P_{14} X_5 + P_{15})}{X_3} \geq 0 \\ & g_{11}(\boldsymbol{x}) = 1 - \frac{0.5 d_1 d_2}{40} \geq 0 \\ & 0.7 \leq d_1 \leq 0.8, \quad 17 \leq d_2 \leq 28, \quad 2.6 \leq \mu_1 \leq 4.2, \\ & 7.0 \leq \mu_2 \leq 8.3, \quad 7.0 \leq \mu_3 \leq 8.3, \quad 2.9 \leq \mu_4 \leq 3.95, \\ & 5.0 \leq \mu_5 \leq 6.0 \end{aligned} \tag{A5}$$

Table A1. Information for random parameters in the speed reducer model.

Design parameters	Mean	Standard deviation
P_1	27.0	2.7
P_2	397.5	39.8
P_3	1.93	0.0965
P_4	1.93	0.0965
P_5	1100.0	110.0
P_6	745	74.5
P_7	1.69×10^7	1.69×10^6
P_8	0.1	0.005
P_9	1.575×10^8	1.575×10^7
P_{10}	850.0	34
P_{11}	5.0	0.25
P_{12}	12.0	0.6
P_{13}	1.5	0.75
P_{14}	1.1	0.11
P_{15}	1.9	0.19

Appendix C – nomenclature

CDF	Cumulative Distribution Function
DLP	Double Loop
DO	Deterministic Optimization
FORM	First Order Reliability Method
IMPP	Inverse Most Probable Point
KKT	Karush–Kuhn–Tucker
MAMV	Modified Advanced Mean Value
MPP	Most Probable Point
PDF	Probability Density Function
RA	Reliability Assessment
SLP	Single Loop
SORA	Sequential Optimization and Reliability Assessment
SORM	Second Order Reliability Method
$\boldsymbol{d}$	deterministic design vector
g_i	ith constraint
$\boldsymbol{p}$	random parameter vector
R_i	reliability level for the ith probabilistic constraint
$\boldsymbol{s}$	shifting vector
$\boldsymbol{u}$	independent and standardized normal vector
U_i	ith component of $\boldsymbol{u}$
$\boldsymbol{x}$	random design vector
X_i	ith random design variable
$\boldsymbol{x}_{\mathrm{IMPP}}$	IMPP vector
$X_{\mathrm{IMPP}i}$	ith component of IMPP
$\boldsymbol{u}_{\mathrm{IMPP}}$	IMPP vector in standard normal space
$\mathrm{U}_{\mathrm{IMPP}i}$	ith component of IMPP in standard normal space
β	reliability index
$\boldsymbol{\mu}_{\boldsymbol{x}}$	vector of mean values of random design variables
μ_{x_i}	mean of ith random design variable
σ_{x_i}	deviation of ith random design variable
r_i	coefficient of variation for ith random design variable