2009-01-0204

# An RBDO Method for Multiple Failure Region Problems using Probabilistic Reanalysis and Approximate Metamodels

**Ramon C. Kuczera, Zissimos P. Mourelatos and Michael Latcha**
Oakland University

**Efstratios Nikolaidis**
University of Toledo



## ABSTRACT

A Reliability-Based Design Optimization (RBDO) method for multiple failure regions is presented. The method uses a Probabilistic Re-Analysis (PRRA) approach in conjunction with an approximate global metamodel with local refinements. The latter serves as an indicator to determine the failure and safe regions. PRRA calculates very efficiently the system reliability of a design by performing a single Monte Carlo (MC) simulation. Although PRRA is based on MC simulation, it calculates "smooth" sensitivity derivatives, allowing therefore, the use of a gradient-based optimizer. An "accurate-on-demand" metamodel is used in the PRRA that allows us to handle problems with multiple disjoint failure regions and potentially multiple most-probable points (MPP). The multiple failure regions are identified by using a clustering technique. A maximin "space-filling" sampling technique is used to construct the metamodel. A vibration absorber example highlights the potential of the proposed method.

## INTRODUCTION

Due to competitive pressures, engineers and scientists are facing an ever-shortening design and validation cycle. The desire for fast development often comes at the cost of reliability and robustness. Early in the design phase it is necessary to make decisions based on incomplete information and with a minimum of physical and virtual tests (FEA). In order to facilitate this decision-making, improved methods are required to assess system reliability and use it in design.

In order to quantify the performance of a system under uncertainty, a method is needed to determine the reliability of a design. By estimating the system reliability at a given design point, an optimization can be performed to determine the design that meets all specifications in terms of probability of failure.

In optimization, an objective is optimized while certain constraints are satisfied. The deterministic optimum though, *does not* necessarily have high reliability. To ensure that the optimum design is also reliable, the optimization formulation must include reliability constraints. Such a formulation is commonly referred as Reliability-Based Design Optimization (RBDO). RBDO approaches focus on maintaining design feasibility (for design constraints) at expected probabilistic levels. They account for inherent input and parameter variations in manufacturing processes, materials, loading, etc.

In RBDO, probability distributions describe the stochastic nature of the design variables and model parameters. Variations are represented by standard deviations (typically assumed to be constant) and a *mean* performance measure is optimized subject to *probabilistic* constraints. RBDO can be a powerful

design tool, since it provides optimum designs in the presence of uncertainty.

Efficiency and accuracy are the two main challenges in RBDO. The reliability, or equivalently the probability of failure, of each constraint must be calculated accurately and efficiently. The commonly used analytical methods (e.g. FORM) are usually efficient but inaccurate, especially for problems with "noisy" limit states, which may exhibit multiple failure regions and potentially multiple most-probable points. Such problems are common in structural dynamics. To alleviate this problem, simulation-based reliability methods may be used. However, they are computationally very expensive and therefore impractical for many engineering problems.

This paper presents a simulation-based RBDO method which is both accurate and efficient, using a PRRA method and an "accurate-on-demand" metamodel which allows us to handle problems with multiple disjoint failure regions, and potentially multiple MPPs. After an introduction to the analytical and simulation-based RBDO methods in section 1.1, we present an overview of the proposed RBDO method in section 2 providing details on the PRRA method in section 2.1, and details on the metamodel construction in section 2.2. Section 3 demonstrates the proposed methodology using a vibration absorber example. Finally, summary and conclusions are presented in section 4.

## 1. ANALYTICAL AND SIMULATION-BASED METHODS IN RBDO

For large-scale systems, the reliability prediction is usually based on efficient computational methods. Both analytical and simulation-based methods are available. The analytical methods are based on the MPP concept. They include the well-known first-order reliability method (FORM) that has been widely used [1], second-order reliability methods (SORM) [2], and multi-point approximation methods [3]. Among the simulation-based methods, the Monte Carlo Simulation (MCS) method is very simple and accurate. However, its computational cost can be prohibitively high. For this reason, more efficient simulation-based techniques have been proposed [4, 5]. Among them, the adaptive importance sampling (AIS) techniques are popular [6,7]. A combination of analytical and simulation-based methods has also been used [8]. The analytical methods are generally simple and efficient, but for complex problems, their accuracy cannot be guaranteed. In simulation-based methods, the accuracy can be controlled but the efficiency is generally not satisfactory.

For system reliability analysis involving multiple failure modes (limit states), the joint failure probability must be taken into account. Due to the difficulty in determining the joint failure probability of more than two failure modes except through MCS, approximations using first-order and second-order bounds have been developed [9, 10].

Simulation-based methods are also used for reliability analysis involving single or multiple limit states. Among MCS-based methods seeking to improve computational efficiency, adaptive importance sampling (AIS) techniques use an importance sampling density function, which is gradually refined to reflect the increasing state of knowledge of the failure domain. The importance sampling methods are divided into direct methods [5, 11], updating methods [12], spherical schemes [11], directional sampling [13], and adaptive schemes [6]. All methods, except the adaptive schemes, require prior knowledge of the failure domain.

## 2. OVERVIEW OF PROPOSED RBDO METHOD

A simulation-based Probabilistic Re-Analysis (PRRA) method is presented in this paper for the design of real-life engineering systems under uncertainty. The method is efficient because it uses a combination of an "accurate-on-demand" global metamodel (surrogate model) with local refinements to assess the reliability of constraints in the RBDO problem. The PRRA method allows the design to change during the optimization process without requiring additional computational effort because it uses a single Monte Carlo simulation.

It is well known that the analytical reliability methods of section 1.1 are computationally efficient, and relatively accurate depending on the application. However, their accuracy deteriorates for problems with highly non-linear and/or "noisy" limit states. Also, they are not appropriate for problems with disjoint failure domains and multiple MPPs. Although simulation-based reliability methods can address the shortcomings of the analytical methods, they are prohibitively expensive for many real-world engineering systems.

The proposed methodology is a simulation-based RBDO method which is computationally efficient and accurate. The high computational effort of most RBDO methods is due to the repeated evaluations of the reliability of each constraint. The PRRA method in this work drastically reduces the effort to estimate the reliability of a constraint. Quite often even a single MC simulation can be very expensive. For this reason, we use an "accurate-on-demand" global metamodel with local refinements to perform the single MC simulation very efficiently. We describe the basics of the PRRA method in section 2.1, and the "accurate-on-demand" metamodeling technique in section 2.2.

### 2.1. ESTIMATION OF FAILURE PROBABILITY IN PRRA

The probability of failure of a system is the integral of the joint PDF $f_{\mathbf{X}}(\mathbf{x})$ times the failure indicator function, over the range of the random variables,

$$p_f = \int_{\mathbf{X}} I(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \quad (1)$$

In Importance Sampling (IS), sample values of the random variables are generated using a sampling PDF

$f_{\mathbf{X}}^{S}(\mathbf{x})$ that yields many failures with high probability of occurrence, instead of the true PDF $f_{\mathbf{X}}(\mathbf{x})$. An estimation of the probability of failure in this case is

$$p_f = \int_{\mathbf{X}} I(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{X}}^{S}(\mathbf{x})} f_{\mathbf{X}}^{S}(\mathbf{x}) d\mathbf{x} \quad (2)$$

where $I(\mathbf{x})$ is the failure indicator function. An unbiased estimator of the probability of failure is,

$$\hat{p}_f = \frac{1}{N}\sum_{i=1}^{N} I(\mathbf{x}_i) \frac{f_{\mathbf{X}}(\mathbf{x}_i)}{f_{\mathbf{X}}^{S}(\mathbf{x}_i)} = \frac{1}{N}\sum_{i=1}^{N_f} \frac{f_{\mathbf{X}}(\mathbf{x}_i)}{f_{\mathbf{X}}^{S}(\mathbf{x}_i)} \quad (3)$$

where $\mathbf{x}_i$, $i$=1,…, $N$ are the sample values of the random design variables generated from the sampling PDF $f_{\mathbf{X}}^{S}(\mathbf{x})$, and $N_f$ is the number of failures. The sum of the right hand side is only for those replications in which the system fails.

Assume that random variables **X** and **Y** are statistically independent, $f_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y}) = f_{\mathbf{X}}(\mathbf{x}) f_{\mathbf{Y}}(\mathbf{y})$, where $f_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$ is the joint PDF of **X** and **Y**, and $f_{\mathbf{X}}(\mathbf{x})$ and $f_{\mathbf{Y}}(\mathbf{y})$ are the marginal PDFs of these variables. A designer can control the mean value of **X** but not **Y.** The key idea of PRRA is that it is sufficient to calculate the failure indicator function of a system only for a single sample of values, {$\boldsymbol{x}_i$ , $\boldsymbol{y}_i$ $i$ = 1,.., $N$} of random variables **X** and **Y** in order to estimate the system failure probability for other PDFs. Indeed, if one generates a sample from $f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$ and calculates and saves the sample values that caused system failure {$\boldsymbol{x}_i$ , $\boldsymbol{y}_i$ $i$ = 1,.., $N_f$}, then one can reuse these sample values to estimate the probability of failure $p_f$ for another PDF, $f_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$,

$$\hat{p}_f = \frac{1}{N}\sum_{i=1}^{N_f} \frac{f_{\mathbf{X}}(\mathbf{x}_i) f_{\mathbf{Y}}(\mathbf{y}_i)}{f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x}_i,\mathbf{y}_i)}. \quad (4)$$

Eq. (4) holds for any combination of PDFs in the numerator and denominator such that the support (set of all values for which a function is non zero) of $f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$ contains the support of $f_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$.

Eq. (4) allows us to estimate the failure probabilities of many designs very efficiently because it only requires calculation of ratios of PDFs; it does not require calculation of the failure indicator function. This reduces the cost of system reliability analysis by several orders of magnitude because the calculation of the failure indicator function dominates the cost of simulation. The probability of system failure as a function of the mean values of the design variables $\boldsymbol{\mu}_{\mathbf{X}}$ is,

$$\hat{p}_f(\boldsymbol{\mu}_{\mathbf{X}}) = \frac{1}{N}\sum_{i=1}^{N_f} \frac{f_{\mathbf{X}}(\mathbf{x}_i,\boldsymbol{\mu}_{\mathbf{X}}) f_{\mathbf{Y}}(\mathbf{y}_i)}{f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x}_i,\mathbf{y}_i)} \quad (5)$$

where $f_{\mathbf{X}}(\mathbf{x},\boldsymbol{\mu}_{\mathbf{X}})$ is the PDF of **X** given the mean value of the vector of the random variables **X** as a function of the mean values of these variables $\boldsymbol{\mu}_{\mathbf{X}}$.

An unbiased estimator of the standard deviation of the system failure probability is,

$$\sigma_{\hat{p}_f} = \frac{1}{\sqrt{N(N-1)}} \sqrt{\sum_{i=1}^{N_f} \left( \frac{f_{\mathbf{X}}(\mathbf{x}_i,\boldsymbol{\mu}_{\mathbf{X}}) f_{\mathbf{Y}}(\mathbf{y}_i)}{f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x}_i,\mathbf{y}_i)} \right)^2 - N \cdot \hat{p}_f{}^2} \,. \quad (6)$$

The 1-$\alpha$ level confidence interval of the system failure probability is,

$$\hat{p}_f \pm t_{N-1,1-\frac{\alpha}{2}} \cdot \sigma_{\hat{p}_f} \quad (7)$$

where $t_{N-1,1-\frac{\alpha}{2}}$ is the point that has below its probability 1-$\alpha$/2 for Student's $t$-distribution with ($k$-1) DOF. This interval covers the true system probability of failure with probability 1-$\alpha$.

An algorithm is described below for efficient estimation of the failure probability for many different mean values of the random design variables.

1. Select a sampling PDF $f^{S}{}_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y})$ for the random variables.
2. Generate a sample of values of variables **X** and **Y**.
3. Calculate the system failure indicator function, $I(\boldsymbol{x}_i, \boldsymbol{y}_i)$, for the sample values in step 2. Select the subset of sample values that caused system failure.
4. Estimate the system failure probability using Eq. (5). The standard deviation of this probability and a confidence interval can be also calculated using Eqs (6) and (7).

The inputs to PRRA are the sampling PDF, the sample of values that caused failure and the PDF for which the failure probability is to be estimated. The output consists of the probability of failure and its confidence bounds if needed. PRRA is non-intrusive because it does not require modifications of the computer codes used for deterministic analysis and reliability analysis through MCS.

A two-dimensional example is used here to demonstrate graphically how the PRRA can be used in design. Although it is a low-dimensional problem, it is very challenging because of the highly non-linear limit state around the MPP. As such, it has been used in previous studies [14, 15]. The non-linear limit state function is defined as $g(x_1,x_2) = x_1^3 + x_2^3 - 180$, where $x_1$ and $x_2$ are independent, normally distributed random variables with standard deviation 5. A design optimization algorithm shifts the mean values of the

variables in order to minimize an objective while satisfying all constraints with a given probability of failure (see section 2). The objective of this example is to estimate efficiently the probability of failure as a function of the mean values of these variables and to determine those designs whose failure probability is no greater than some upper limit.

First, we generate 100,000sample points assuming that random variables $x_1$ and $x_2$ have mean values 10 and 9.5, respectively. Figure 1 shows the limit state as a green line and the generated MCS sample points as blue dots. Then, sample points are selected among the MCS points by using the maximin algorithm in section 2.2. This algorithm has the advantage that it places points both on the boundary and on the interior of the domain of the random variables. The limit state function is evaluated at each point, and a metamodel is constructed. The metamodel is then refined (see section 2.4) in order to identify all "failure" points.

When we shift the mean values of random variables $x_1$ and $x_2$, we change the probability of failure (constraint violation). PRRA (Eq. 5) is used to estimate the probability of failure for many combinations of the mean values of these random variables by using information from a single MCS. Figure 1 shows combinations of the mean values of the random variables (corresponding to the red dots) for which the system probability of failure is 0.01.

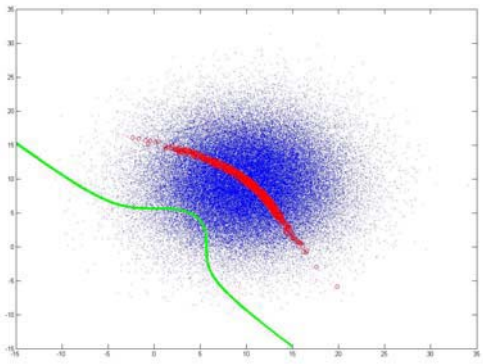

**Figure 1**. Identification of points with 0.01 probability of failure

## 2.2 MAXIMIN SAMPLING METHOD

We use an easy to implement maximin method in which the addition of new samples preserves by default, the "space-filling" properties. This is a very important property which allows us to construct "converged" metamodels with a small number of samples. Convergence is practically achieved if the reconstructed metamodel after a few more samples are added, gives similar predictions with the previous version. Similar predictions are judged based only on the sign, and not the estimated actual value, of the limit state.

Consider a design $D_n = \{\mathbf{x}_1, \mathbf{x}_2, \cdots \mathbf{x}_n\}$ which is composed of a collection of *k*-dimensional samples $\mathbf{x}_i \in \Re^k, i = 1,...,n$. A few judiciously selected samples from $D_n$ define the maximin distance design *D* where, $D \subseteq D_n$. Let $d(\mathbf{u},\mathbf{v})$ be the Eucledian distance between samples **u** and **v**, where $\mathbf{u},\mathbf{v} \in D$. Design *D* simply maximizes the minimum inter-site distance $\min_{\mathbf{u},\mathbf{v}\in D} d(\mathbf{u},\mathbf{v})$, i.e.

$$\max_{D} \min_{\mathbf{u},\mathbf{v}\in D} d(\mathbf{u},\mathbf{v}). \quad (8)$$

In this paper, the maximin sampling technique of Eq. (8) is implemented in a straightforward way. Assume that a design $D_n$ of *n*, *k*-dimensional samples is available, consisting of *n* MC samples. Each of the *k* components of the *n* MC samples is normalized between zero and one in order to account for potentially different units. If *x* denotes the $k^{th}$ component, its normalized value is $x^* = (x - x^L)/(x^H - x^L)$, where $x^L$ and $x^L$ are the low and high values respectively, among all *n* samples.

Point $\mathbf{x}_1^* = \{\bar{x}_1^* \cdots \bar{x}_i^* \cdots \bar{x}_k^*\}$, where $\bar{x}_i^*, i = 1,...,k$ is the mean $x_i^*$ coordinate of all *n* normalized points, is first selected as a "seed" point. Then the distance of all *n* normalized points from $\mathbf{x}_1^*$ is calculated and the point $\mathbf{x}_2^*$ with the largest distance is selected. Subsequently, the distances of the remaining (*n*-2) samples from both $\mathbf{x}_1^*$ and $\mathbf{x}_2^*$ are calculated and the point $\mathbf{x}_3^*$ with the maximum minimum distance is selected. The process is repeated *m* times in order to create the maximin distance design *D* with *m* sample points. It should be noted that the method first places sample points on the boundary of the domain. As more points are created, they are placed in the interior of the domain providing therefore, a "space-filling" design. A very useful property of the method is that the addition of new sample points preserves the "space-filling" properties of the new design. This is essential in order to keep the number of function evaluations low.

## 2.3 AN "ACCURATE-ON-DEMAND" METAMODEL FOR RELIABILITY ESTIMATION

The method addresses problems with multiple MPPs with disjoint failure domains [14,15]. It can also asses for system reliability assessment in problems with multiple limit states. It can easily handle implicit, nonlinear limit-state functions, with correlated or non-correlated random variables, which are described by any probabilistic distribution. The method is based on an "accurate-on-demand" global metamodel of the limit states with local refinements, which can serve as an

indicator to determine the "failure" and "safe" regions. High accuracy of the metamodel is not needed away from the limit states because it is simply used as a "failure" indicator. However, improved accuracy is necessary close to the limit states where sample points have a "low" absolute value. In this work, points with "low" limit-state absolute value define a transition region between the safe and failure regions. All samples in the transition region are identified, and more accurate *local* metamodels of each actual limit state are generated. The local metamodels are used to determine if the transition region points are in the safe or failure domains.

If a limit state has multiple failure regions (that may be even disjoint), or if multiple limit states exist due to a system reliability assessment, the transition region points may be grouped in clusters. A clustering technique is used to identify these groupings so local metamodels can be constructed for each group. This is essential for handling multiple MPP problems that often exist in a variety of engineering applications, such as problems involving vibratory systems. Recently, some work has been reported for multiple MPP problems [16,17].

Although various metamodeling techniques can be successfully used in this work, we have chosen the Cross-Validated Moving Least Squares (CVMLS) [18] method. To construct a metamodel, a "space-filling" sampling algorithm is needed. Commonly used algorithms such as Latin Hypercube (LH) and Optimal Symmetric LH (OSLH) sampling [19] among others, place most of the samples in the interior of the domain. To avoid this problem, the maximin [20] "space-filling" sampling technique is used.

In simulation-based reliability methods, it is the sign of the limit state function, that is important, not the actual value. We predict the sign of the function in this study by constructing approximate metamodels that can efficiently identify the safe (high value) and failure (low value) regions. A similar idea has been reported in [21] using an indicator response surface and in [16] using discriminative sampling.

In the following we describe the algorithm for constructing a metamodel for the case where the designer can control the mean values of all random variables, **X**. The entire sampling region is divided into a safe region, a failure region and a transition region between the safe and failure regions. The safe and failure regions are easily identified using an approximate global metamodel $\tilde{g}_G(\mathbf{x})$ which is also used to identify the transition region. The transition region includes all sample points ***x***, such that $\tilde{g}_G(\mathbf{x}) \in (h_l, h_u)$ where $h_u$ and $h_l$ are the upper and lower bound values respectively, estimated using the global metamodel.

All sample points in the transition region are used in a clustering algorithm [22] that determines if there are distinct transition regions associated with either disjoint or different limit states. Importance sampling is then used to generate additional Monte Carlo sample points within each cluster. The sampling PDF is the original PDF truncated over the transition region in order to reduce the variance of the probability of failure estimation [23, 24].

An efficient "space-filling" sampling is subsequently, used based on a maximin distance algorithm, which efficiently selects a few sample points with "space-filling" properties. The maximin sampling method selects points from the additional MC samples. The selected points are used to construct a local metamodel $\tilde{g}_L(\mathbf{x})$ for each cluster of transition region points. Good accuracy is ensured by a series of refinements using a "space-filling" algorithm that places samples very close to the limit state within a transition region domain and by constructing metamodels of small, local regions.

## 2.4 METAMODEL ALGORITHM

The construction of an approximate global metamodel is described first (steps 1 through 7). Subsequently, a clustering technique is used which identifies potentially disjoint failure domains. The construction of local metamodels, including a series of refinements, is explained in steps 12 through 17.

1. Draw *N* sample values according from the distribution of the random variables. Eq. (9) estimates *N* by keeping the error in the estimated probability of failure less than $\varepsilon$ % with confidence level $\alpha$. In this work, we use $\varepsilon = 10\,\%$, $\alpha = 0.05$ and $\rho = 0.05$.

$$\varepsilon\% = 100\Phi^{-1}(1-\frac{\rho}{2})\sqrt{\frac{(1-P_f^T)}{NP_f^T}} \qquad (9)$$

2. Using the maximin method, *m* sample values are selected from the available *N* values of step 1. Although *m* is kept small, it is at least equal to *n*+1 where *n* is the number of variables.
3. Calculate the limit state value for all *m* points of step 2 and construct an approximate global metamodel $\tilde{g}_G(\mathbf{x})$. A different global metamodel is constructed for each limit state. The choice of the metamodeling technique may slightly affect the efficiency of the proposed method. It is not however, vital to the success of the method. Note that $\tilde{g}_G(\mathbf{x})$ covers only the "cloud" area of the MC samples from step 1.
4. Use $\tilde{g}_G(\mathbf{x})$ to predict the value of each limit state for the *N* MC samples.
5. Identify the upper and lower limits, $h_u$ and $h_l$ respectively, of the transition region. The *N* values of step 4 are sorted in decreasing order. If $\max \tilde{g}_G$ and $\min \tilde{g}_G$ represent the maximum and minimum values, $h_u$ and $h_l$ are equal to $h_u = \eta(\max \tilde{g}_G)$ and $h_l = \eta(\min \tilde{g}_G)$, where $\eta$ (called the transition zone width parameter) is a predetermined small percentage. The transition region includes all MC samples ***x*** for which $h_l \le \tilde{g}_G(\mathbf{x}) \le h_u$. Their number is denoted by *NT*. It should be emphasized that *NT*

includes the transition points from all limit states. Generally a transition zone is initially selected using a value of 10% for $\eta$.

6. Define the safe region and the failure region. The former includes all MC samples ***x*** so that $\tilde{g}_G(\mathbf{x}) > h_u$, and the latter includes all MC samples with $\tilde{g}_G(\mathbf{x}) < h_l$. The number of points in the failure region is $n_G$. For the points in the safe and failure regions, the indicator function is assigned a value of 0 and 1, respectively.
7. Add *p* more points to the *m* points of step 1, and go to step 3. Repeat steps 3 to 6 until convergence is achieved. We have convergence, if the number of failure region points from step 6, does not change more than ten percent between two successive iterations. This "crude" convergence provides a compromise between accuracy and efficiency for the approximate global metamodels. It also defines the transition zone with enough accuracy for the next steps.

At this point, we assume that the sample points in the identified safe and failure regions have indeed a positive and negative value, respectively. However due to the potentially low accuracy of the approximate global metamodel from step 7, we are not certain about the positive or negative sign of the transition points. For this reason, local metamodels of the transition region points are developed which are expected to be more accurate because of their smaller domains. A clustering technique determines the number of local metamodels.

If a limit state has multiple failure regions, or if multiple limit states exist due to a system reliability assessment, the *NT* transition points from the "converged" global metamodel of step 7 may form clusters. A clustering technique described below is then used to identify potential clusters.

8. Using all points in the transition region, clustering groups them in $k$ clusters. Because we do not know the number of existing clusters *a priori*, we start with a relatively large number of clusters. Depending on how many points are grouped in each of the $k$ clusters, we easily determine the number of existing clusters $k_c$ where $k_c < k$. Subsequently, clustering is repeated with only $k_c$ clusters.
9. Once the number of clusters has been identified, we can adjust the width of the transition zone to cover as much area of interest as possible. The transition zone width parameter $\eta$ is steadily increased until the clustering is no longer effective. The maximum value $n_{\max}$ of the transition zone width parameter $\eta$ (maximum value at which clustering is still effective), is used to define the final transition zone. Because for certain problems with multiple disjoint failure regions, the clusters might become very large in size before they become unrecognizable, we limit the size of the transition zone such that it contains no more than 10% of the total number of sample points. These two guidelines ensure that the local metamodels are small enough for increased accuracy, and simultaneously include most of the failure points.
10. For each cluster, determine the sampling domain for each of the random variables. By examining the maximum and minimum value of each random variable within the cluster, we can determine whether each variable is spanning the entire range of values as defined by its PDF, or a portion such as the left or right tail of the PDF. This defines a rectangular "sampling domain" (see dotted domain in Figure 8).
11. Generate *S* sample points for each cluster using importance sampling, so that the generated points are in the rectangular "sampling domain" around the cluster. Because the number of clusters is $k$, each group of sample points is denoted by *S(k).*

The *S(k)* points for each cluster are within the domain of the global metamodel which can be used to predict their limit state value. However, the prediction may be inaccurate because the global metamodel is approximate. For this reason, we further refine the global metamodel locally for each cluster. To this end, we select additional maximin points in the rectangular "sampling domain," and use them to refine the global metamodel. However, the "sampling domain" around a cluster may be much larger in size than the cluster itself. In order to only have points close to the cluster, we use Principal Component Analysis (PCA) [25] to identify a smaller rectangular domain within the "sampling domain" (see "Cluster 1 PCA" domain in Figure 8). A selected maximin point from the *S(k)* points is then kept only if it is within the smaller PCA domain.

12. For the $k^{th}$ cluster, select *m* points using maximin from the *S(k)* points, according to step 11.
13. Calculate the limit state value for all *m* points of step 12 and construct a local metamodel $\tilde{g}_L(\mathbf{x})$. A different local metamodel is constructed for each cluster (or limit state). These local metamodels cover the PCA domain as described in step 11.
14. A "converged" local metamodel is built for each limit state using the iterative process of steps 3 through 7. Points are successively selected from the *S(k)* points of step 11, using the maximin approach. At each iteration, the current version of the local metamodel is used to evaluate all *S(k)* points, and identify the number *mm(k)* of the failed points; i.e points ***x*** with $\tilde{g}_L(\mathbf{x}) < 0$. Convergence is achieved if *mm(k)* does not change more than 10% from the previous iteration. A "converged" local metamodel is achieved with $n_L(k)$ points.
15. The converged local metamodel from step 14 is used to identify $NT_L(k)$ points ***x*** from the group of the *S(k)* points, so that $\alpha(\max \tilde{g}_L) \le \tilde{g}_L(\mathbf{x}) \le \alpha(\min \tilde{g}_L)$, $\alpha$ =0.1. These points

are located very close to the limit state having therefore, a small absolute value.

16. At this point, a series of refinements for each local metamodel is performed. The maximin approach is used to identify *m* out of the $NT_L(k)$ points of step 15 which are added to the existing $n_L(k)$ points of step 14, and the local metamodel is updated. Again, all *S(k)* points are evaluated using the updated local metamodels and a new number $mm(k)$ of failed points is identified. The process is repeated by adding *m* more points and updating the local metamodels, until the number $mm(k)$ is converged within a relative error of 1% from the previous iteration.
17. Repeat steps 12 through 16 for each of the $k$ clusters.

## 3. A VIBRATION ABSORBER EXAMPLE

### 3.1 PROBLEM DESCRIPTION

A tuned damper system is shown in Figure 2. It consists of the original system and a vibration absorber. For simplicity, the original system has a single degree of freedom, and is subject to a harmonic excitation $f(t) = F\cos(\omega \cdot t)$. The absorber is attached to the original system in order to reduce its vibration amplitude.

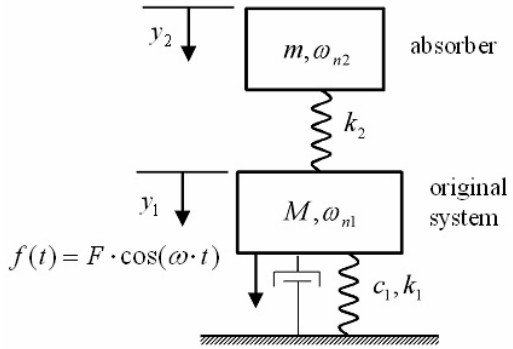


**Figure 2**. Tuned vibration absorber

The amplitude $X_1$ of the original system is a function of four parameters. In this example, we normalized it by the amplitude of its static response $F/k_1$ as follows

$$y_1 = \frac{X_1}{F/k_1} = \frac{\left|1-(\frac{1}{\beta_2})^2\right|}{\sqrt{\left[1-R(\frac{1}{\beta_1})^2-(\frac{1}{\beta_1})^2-(\frac{1}{\beta_2})^2+\frac{1}{\beta_1^2\beta_2^2}\right]^2+4\varsigma^2\left[(\frac{1}{\beta_1})-\frac{1}{\beta_1\beta_2^2}\right]^2}} \quad (10)$$

In Eq. (10), $R = m/M$ is the ratio of the absorber mass to the original system mass, $\varsigma$ is the damping ratio of the original system, and $\beta_1 = \omega_{n_1}/\omega$ and $\beta_2 = \omega_{n_2}/\omega$ are the ratios of the natural frequencies $\omega_{n_1}$ and $\omega_{n_2}$ of the original system and vibration absorber respectively, to the excitation frequency $\omega$. It is assumed that the absorber does not provide additional damping to the overall system (see Figure 2). For illustration, *R* and $\varsigma$ are treated as deterministic variables with values *R*=0.01 and $\varsigma$=0.01, respectively. Only $\beta_1$ and $\beta_2$ are random variables which are assumed normally distributed with mean 1.0 and standard deviation 0.025.

The absorber is added to "absorb" the vibratory energy when the original system is at resonance or close to resonance. In this case, the absorber motion becomes large, and the motion of the original system reduces considerably. In order for the absorber action to be effective, the absorber must be "tuned." Tuning means that the natural frequency of the absorber, and the natural frequency of the system are approximately equal to the excitation frequency $\omega$; i.e. $\omega_{n_1} \approx \omega_{n_2} \approx \omega$, or equivalently, $\beta_1 \approx \beta_2 \approx 1$. In this case, the vibratory amplitude of the original system is almost zero, and the amplitude of the absorber,

$$y_2 = \frac{X_2}{F/k_1} = \frac{1}{\sqrt{\left[1-R(\frac{1}{\beta_1})^2-(\frac{1}{\beta_1})^2-(\frac{1}{\beta_2})^2+\frac{1}{\beta_1^2\beta_2^2}\right]^2+4\varsigma^2\left[(\frac{1}{\beta_1})-\frac{1}{\beta_1\beta_2^2}\right]^2}} \quad (11)$$

is very high. Displacement $y_2$ is restricted below a maximum allowable value for durability.

Figure 3 shows contours of $y_1$ and $y_2$ as a function of $\beta_1$ and $\beta_2$. There is a trade-off between effective action of the absorber (small $y_1$ and simultaneously, large $y_2$), and durability of the absorber (small $y_2$).

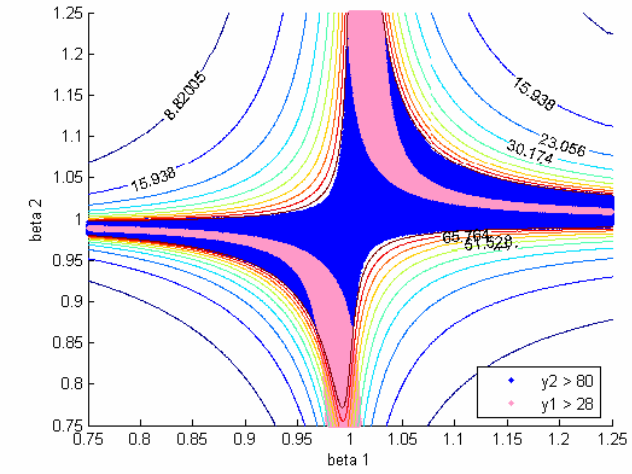


**Figure 3**. Displacement contours of original system and vibration absorber

The objectives in this example are 1) to maximize the effectiveness of the absorber by

maximizing $y_2$ without exceeding a limit $y_2 \le 80$ for durability, and 2) to reduce the risk of the normalized amplitude $y_1(\beta_1,\beta_2)$ exceeding 28, in the presence of uncertainty in the natural frequencies of the original system and the absorber. At the same time, we want $\beta_1 \le 0.96$ in order to ensure an effective absorber action at "high" excitation frequencies $\omega \ge \omega_{n_1}/0.96$. To achieve all this, the following RBDO problem is solved

$$\max_{\mu_{\beta_1},\mu_{\beta_2}} \; y_2\left(\mu_{\beta_1},\mu_{\beta_2}\right) \qquad (12)$$

$$\text{s.t.} \quad P\left(y_1\left(\beta_1,\beta_2\right)\le 28\right) \ge R = 1 - p_f^t,$$

$$\beta_1 \le 0.96$$

$$y_2\left(\beta_1,\beta_2\right) \le 80$$

where $\left(\mu_{\beta_1},\mu_{\beta_2}\right)$ are the means of the two random design variables $\left(\beta_1,\beta_2\right)$, and $R = 1 - p_f^t$ is the target reliability. The target probability of failure $p_f^t$ is approximated using the target reliability index $\beta^t$, and the standard normal cumulative distribution function $\Phi$, as $p_f^t = \Phi(-\beta^t)$.

### 3.2 IMPLEMENTATION OF PRRA

The PRRA method calculates the probability $P\left(y_1\left(\beta_1,\beta_2\right)\le 28\right)$ in the RBDO problem of Eq. (11), using Eq. (5). The sampling PDF is assumed equal to $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2) = f^S\left(\beta_1\right)f^S\left(\beta_2\right)$ where $f^S\left(\beta_1\right)$ and $f^S\left(\beta_2\right)$ are normal distributions with mean 1 and standard deviation equal to 0.05 which is twice the actual standard deviation of $\beta_1$ and $\beta_2$. The sampling PDF can be therefore, viewed as an "inflated" actual distribution so that the support of $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2)$ contains the support of the true distribution $f_{\beta_1,\beta_2}(\beta_1,\beta_2) = f\left(\beta_1\right)f\left(\beta_2\right)$ (see section 2.1).

The "MCS Hull" in Figure 4 is a convex hull enclosing all generated sample points from $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2)$. The dotted domain encloses all sample points of the "MCS Hull" which are within $\pm 3\sigma$ of $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2)$ where $\sigma$ is the standard deviation of the sampling distribution. The solid domains show representative $\pm 3\sigma$ clouds of the actual PDF $f_{\beta_1,\beta_2}(\beta_1,\beta_2)$ for different mean values of $\beta_1$ and $\beta_2$. We assumed that the support of $f_{\beta_1,\beta_2}(\beta_1,\beta_2)$, centered at any point within the dotted domain, is enclosed by the support of $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2)$.

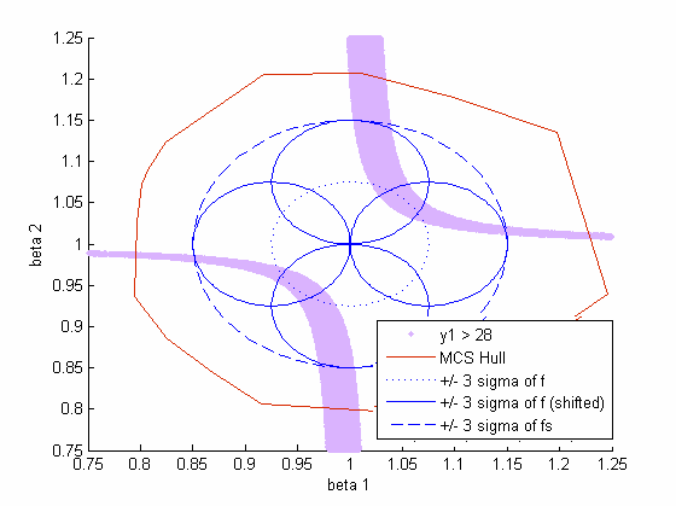


**Figure 4**. Demonstration of sampling PDF and actual PDFs at different designs

According to the PRRA method, it is sufficient to calculate the value of the failure indicator function for only a single sample of values, $\{(\beta_{1_i},\beta_{2_i})$, $i$ = 1,.., $N\}$ of random variables $(\beta_1,\beta_2)$ in order to estimate the system failure probability for other PDFs. We therefore, generate a sample from $f^S{}_{\beta_1,\beta_2}(\beta_1,\beta_2)$, and calculate and save the coordinates of the $N_f$ sample values that caused failure. We then reuse them to estimate the probability of failure $\hat{p}_f$ for another PDF $f_{\beta_1,\beta_2}(\beta_1,\beta_2,\mu_{\beta_1},\mu_{\beta_2})$ by reweighing the values of the failure indicator function, as

$$\hat{p}_f(\mu_{\beta_1},\mu_{\beta_2}) = \frac{1}{N}\sum_{i=1}^{N_f} \frac{f_{\beta_1,\beta_2}\left(\beta_1,\beta_2,\mu_{\beta_1},\mu_{\beta_2}\right)}{f^S_{\beta_1,\beta_2}\left(\beta_1,\beta_2\right)} \qquad (13)$$

The PRRA method allows us to recalculate the system failure probability using only a single sample of values. The computational effort however, even for a single sample, may be very high. For this reason, we build a global metamodel with local refinements according to section 2.2, and use it to calculate the *N* values of $y_1$ for the sample $\{(\beta_{1_i},\beta_{2_i})$, $i$ = 1,.., $N\}$. The following section provides details.

### 3.3 "ACCURATE-ON-DEMAND" METAMODEL

One hundred thousand MC sample points (*N*=100,000) are generated using the $\beta_1 \sim N(1,0.025)$

and $\beta_2 \sim N(1,0.025)$ distributions (step 1 of section 2.2.1). Each point is represented by a cross in Figure 5.

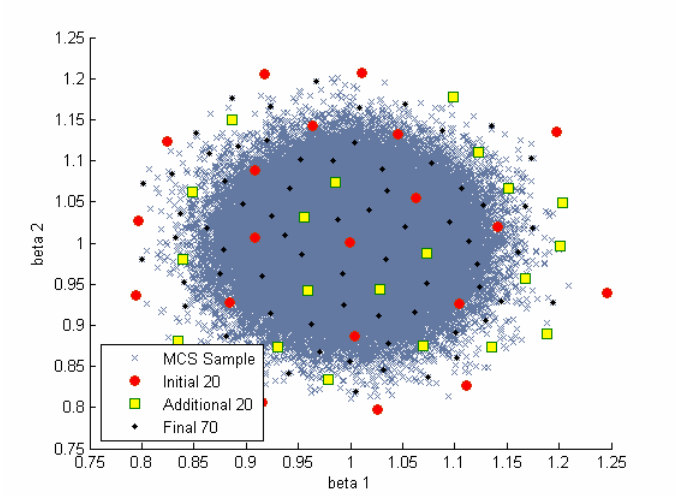


**Figure 5**. Sample points for vibration absorber

A global metamodel is first generated using the iterative process of steps 1 through 7. CVMLS metamodels are created progressively with *m*=5 (step 2) until convergence is achieved. The maximin method selects the points from the 100,000 MC sample points. The selected points for *m*=20, 40, and 110 are indicated in Figure 5 with large circles, squares and small circles, respectively. As points are added, we increasingly "space fill" the design space. The intent is to use a minimum number of points, and therefore function evaluations, without substantial loss of accuracy. A global metamodel is generated and used to predict the limit state value for all 100,000 MC sample points (step 4). The safe, failure and transition regions are identified according to steps 5 and 6. Table 1 summarizes the number of points in each region for different number of selected points *m*.

**Table 1.** Global metamodel convergence for vibration absorber

| Number of Samples | Number of Safe Points | Number of Failure Points | Number of Transition Points |
|---|---|---|---|
| 20 | 100,000 | 0 | 0 |
| 25 | 98,113 | 39 | 1,848 |
| 30 | 99,992 | 1 | 7 |
| 35 | 99,986 | 4 | 10 |
| 40 | 99,996 | 1 | 3 |
| 45 | 97,703 | 464 | 1,833 |
| 50 | 98,700 | 123 | 1,177 |
| 55 | 87,604 | 6,692 | 5,704 |
| 60 | 89,284 | 5,120 | 5,596 |
| 65 | 90,932 | 4,213 | 4,855 |
| 70 | 90,124 | 4,886 | 4,990 |
| 75 | 93,736 | 2,722 | 3,542 |
| 80 | 91,771 | 4,025 | 4,205 |
| 85 | 91,056 | 4,225 | 4,519 |
| 90 | 90,481 | 4,744 | 4,775 |
| 95 | 91,676 | 4,098 | 4,226 |
| 100 | 89,897 | 5,490 | 4,613 |
| 105 | 88,921 | 6,280 | 4,799 |
| 110 | 88,753 | 6,450 | 4,797 |

According to step 7, convergence of the global metamodel is achieved with *m*=110. The 6,450 failure points are within 10% of the 6,280 failure points of the previous iteration (*m*=105). Therefore, $n_G$ is equal to 6,450 (see step 6), and *NT* is equal to 4,797 (see step 5). Both values are listed in the last row of Table 1. For the definition of the transition region, we have used $\eta = 0.10$, as described in step 5 of the algorithm.

At this point, an *approximate* global metamodel has been obtained with "accuracy-on-demand." Although it can not accurately predict all successes and failures among the 100,000 MC sample points, it is accurate enough to locate the safe, unsafe and transition zones.

Using the converged global metamodel, we calculate the limit state at all *N*=100,000 sample points, and partition them into safe, unsafe and transition points. Subsequently, a cluster analysis is performed using the *NT* = 4797 transition points and 6450 predicted failure points. We initially use the large number of 10 clusters in order to make sure we identify all existing clusters. Figure 6 shows the points of the two dominant clusters. For clarity, we also show the disjoint failure zone, and the maximin points which were used to build the global metamodel.

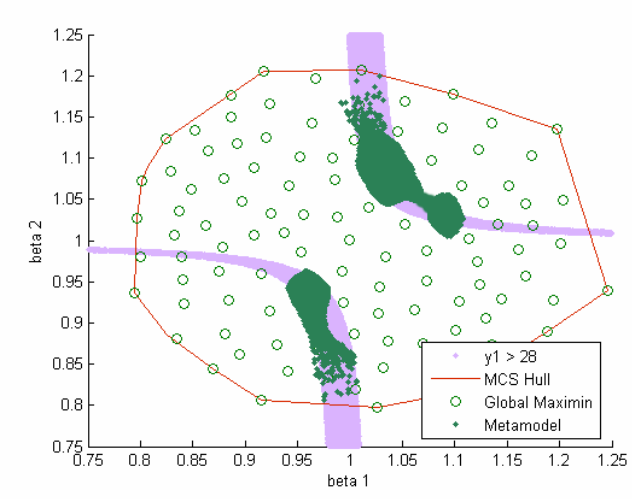


**Figure 6**. Identified clusters with *n* = 10%

It is clear that there are only two clusters containing the majority of the transition and predicted failure points. The iterative process described in step 9 is now used to steadily increase the transition zone width parameter $\eta$ until clustering is no longer effective, or until we reach the maximum number of points allowed in the transition zone (15% of the cluster sample points). Starting from $\eta = 10\%$ we increase $\eta$ by one percent at each iteration, until we have included 15% of the transition zone points. This occurs for $\eta = n_{\max} = 31.46\%$. Figure 7 shows the identified transition zone and failure zone points.

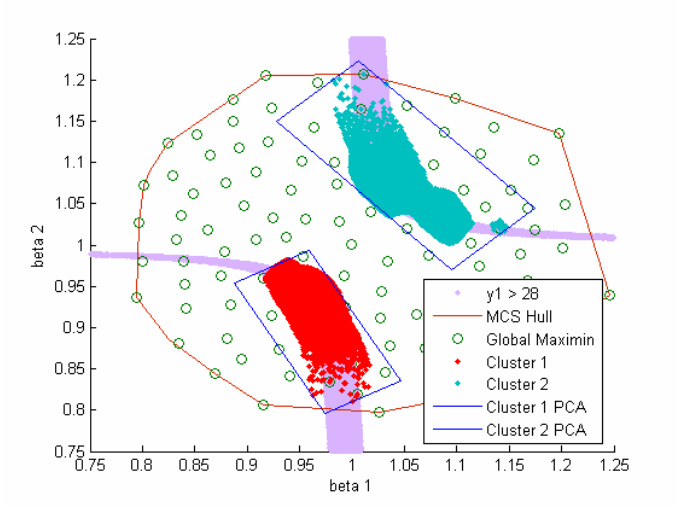


**Figure 7**. Clusters with *n* = 31.46%

Using the two clusters of Figure 7, we determine a rectangular importance sampling domain for each cluster (see step 11). For the first cluster, $\beta_1$ varies between 0.8877 and 1.0465 and $\beta_2$ varies between 0.7960 and 0.9937. The corresponding CDF range for $\beta_1$ is 0.0123 to 0.8237 indicating that we should sample from the middle region with $0.0123 < s < 0.8237$. Similarly the corresponding CDF range for $\beta_2$ is 0.7959 and 0.9937. Therefore, $\beta_2$ is sampled from the left tail with $s < 0.4501$. Figure 8 shows the importance sampling domain with the dotted line.

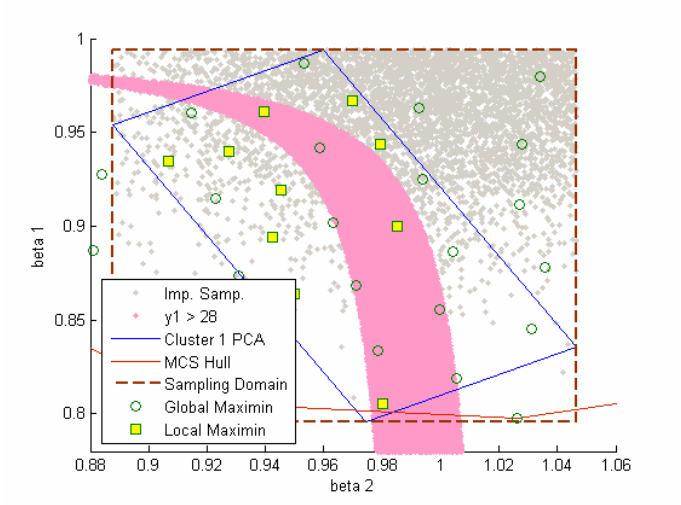


**Figure 8**. Importance sampling and PCA domains for first cluster

As mentioned in step 11, additional MC sample points are generated within the importance sampling domain according to the defined sampling PDFs. *S*=100,000 sample points are created, which are shown in Figure 8 with grey dots. A local metamodel is then built according to steps 12 to 14. The maximin algorithm is used to successively select *m*=5 points (starting from 5 points) from the *S*=100,000 sample points within the importance sampling domain, which are also within the PCA domain of Figure 8, and a local metamodel is built using the CVMLS algorithm. A converged local metamodel is achieved with $n_L$=10 points using the 10% stopping criterion of step 14. Convergence details of the local metamodel are shown in Table 2.

**Table 2.** Local metamodel convergence for vibration absorber (1st cluster)

| Number of Samples | Total Number of Samples in Metamodel | Number of Safe Points | Number of Failure Points |
|---|---|---|---|
| 5 | 115 | 90,546 | 5,457 |
| 10 | 120 | 89,121 | 5,386 |

This converged local metamodel is further refined by selecting sample points near the limit state. Fifteen additional groups of *mm*=5 samples are selected resulting in a change of $\left(\frac{11{,}443-11{,}460}{11{,}443}\right)\cdot 100 = 0.15\%$ in the number of failure points (see Table 3), which is less than the 1% threshold required for convergence of the refined local metamodel (see step 16). Figure 9 shows with red triangles, the additional 75 sample points (15 groups of 5 points each) of Table 3. They are all located close to the limit state, covering also the failure domain since the latter is slim. It also shows with orange dots, all failure points as evaluated by the refined local metamodel. We observe that the orange dot domain overlaps with the actual failure domain $y_2 > 28$, indicating that the refined local metamodel is accurate.

**Table 3.** Refinement of local metamodel for vibration absorber (1st cluster)

| Number of Samples | Total Number of Samples in Metamodel | Number of Safe Points | Number of Failure Points |
|---|---|---|---|
| 15 | 125 | 87,131 | 8,537 |
| 20 | 130 | 85,376 | 11,079 |
| 25 | 135 | 86,028 | 10,063 |
| 30 | 140 | 84,786 | 11,416 |
| 35 | 145 | 85,560 | 9,785 |
| 40 | 150 | 85,703 | 9,686 |
| 45 | 155 | 85,674 | 9,248 |
| 50 | 160 | 85,114 | 10,397 |
| 55 | 165 | 85,639 | 9,220 |
| 60 | 170 | 85,381 | 10,644 |
| 65 | 175 | 85,794 | 10,200 |
| 70 | 180 | 84,996 | 11,489 |
| 75 | 185 | 84,999 | 11,620 |
| 80 | 190 | 84,916 | 11,443 |
| 85 | 195 | 84,951 | 11,460 |

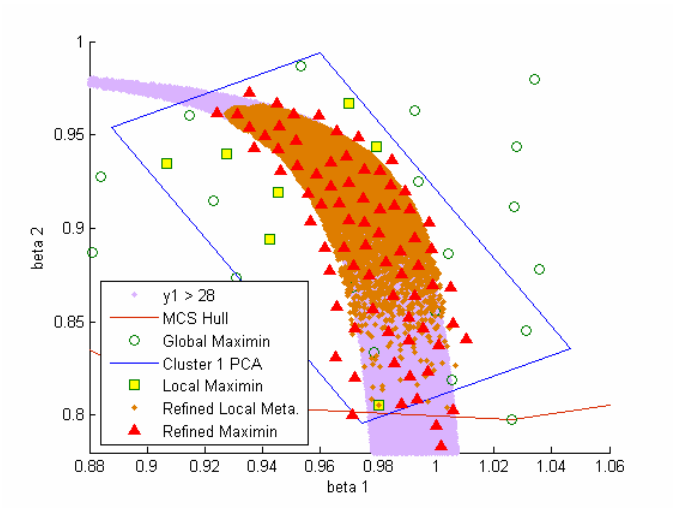


**Figure 9**. Refined local metamodel for first cluster

The same procedure is also followed for the second cluster. A local metamodel is built and refined for improved accuracy. The results are shown in Tables 4 and 5.

**Table 4.** Local metamodel convergence for vibration absorber (2nd cluster)

| Number of Samples | Total Number of Samples in Metamodel | Number of Safe Points | Number of Failure Points |
|---|---|---|---|
| 5 | 115 | 92,684 | 5,122 |
| 10 | 120 | 94,120 | 4,255 |
| 15 | 125 | 94,937 | 3,685 |
| 20 | 130 | 95,422 | 3,258 |
| 25 | 135 | 92,458 | 4,916 |
| 30 | 140 | 92,538 | 4,867 |

**Table 5.** Refinement of local metamodel for vibration absorber (2nd cluster)

| Number of Samples | Total Number of Samples in Metamodel | Number of Safe Points | Number of Failure Points |
|---|---|---|---|
| 35 | 145 | 91,483 | 6,567 |
| 40 | 150 | 91,226 | 6,910 |
| 45 | 155 | 90,581 | 7,697 |
| 50 | 160 | 90,575 | 7,651 |

According to the information in Tables 1 through 5, the number of function evaluations to build the metamodel is 245 (110 for the global metamodel, and 85 and 50 for the refined local metamodels of the first and second clusters, respectively).

## 3.4 SOLUTION OF THE RBDO PROBLEM

Figure 10 shows the optimum solution of the RBDO problem of Eq. (11) for different values of the target probability of failure $p_f^t$, or equivalently the target reliability index $\beta^t$. First, the PRRA method was used to identify all points in the dotted domain of Figure 4 which satisfy the probabilistic constraint $P(y_1(\beta_1,\beta_2)\le 28)\ge R=1-\Phi(-\beta^t)$ with $\beta^t\ge 3$. For that, the above probability was calculated using Eq. (12) for a set of "space-filling" $(\mu_{\beta_1},\mu_{\beta_2})$ points within the dotted domain of Figure 4, and the data was used to create a metamodel of the failure probability. Note that the failure probability metamodel is different from the metamodel of section 3.3. The failure probability metamodel was then used to estimate the failure probability for a very large number of $(\mu_{\beta_1},\mu_{\beta_2})$ points. Figure 10a shows the domain within which $P(y_1(\beta_1,\beta_2)\le 28)\ge 1-\Phi(-3)$, and Figure 10b provides a zoomed-in version.

As we maximize $y_2$, the probabilistic constraint becomes active, resulting in the optimum solution which is indicated by the red point for different values of $p_f^t$. For $p_f^t$ = 0.00135 ($\beta^t=3$), the optimum solution is $(\mu_{\beta_1},\mu_{\beta_2})$ = (0.9511, 1.0365), and for $p_f^t$ = 0.00115, 0.001, and 0.0009, the optimum solution is (0.9490, 1.0378), (0.9478, 1.0394), and (0.9466, 1.0402), respectively. The iso-lines in Figure 10b indicate the value of $y_2$.

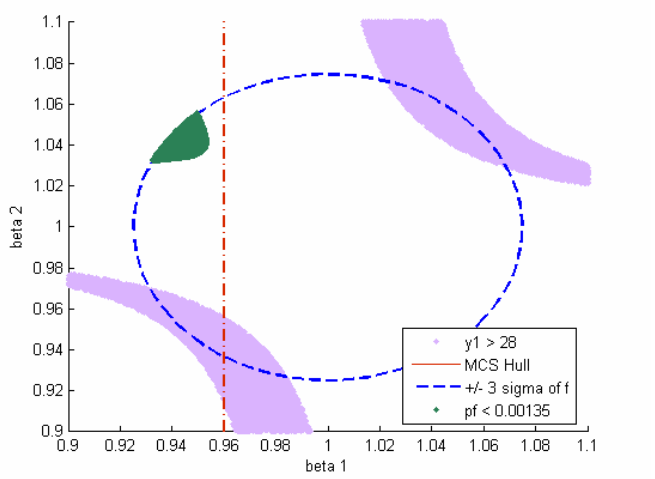


**Figure 10a**. Optimum solutions for $p_f$ < 0.00135

To validate the calculated optimum designs, a MC simulation was performed with one million sample points at the $(\mu_{\beta_1},\mu_{\beta_2})$ = (0.9511, 1.0365) design. The MC simulation resulted in a probability of failure of 0.002114 instead of 0.00135 as our method indicated (see previous paragraph). The reason for the discrepancy is that our approximate global metamodel did not identify the narrow portion of the failure domain corresponding to the 1st cluster. As shown in Fig. 7, the rectangular importance sampling domain of the 1st cluster does not include the narrow portion of the failure domain. The fidelity of the used maximin points to build the global metamodel was not fine enough to capture the narrow failure domain. An improvement of the current methodology is underway to address this issue.

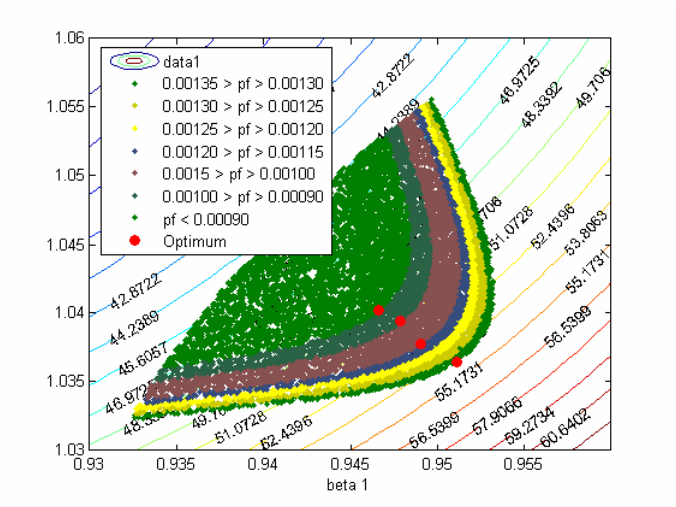


**Figure 10b**. Optimum solutions for $p_f < 0.00135$ (zoomed-in)

## 4. SUMMARY AND CONCLUSIONS

An RBDO methodology has been presented using a Probabilistic Re-analysis (PRRA) method, and an "accurate-on-demand" metamodel with local refinements. The combination of the PRRA method and the metamodel improves computational efficiency with good accuracy. The PRRA calculates the optimum design with only a single Monte Carlo simulation, and the metamodel reduces the computational effort for the single Monte Carlo simulation. The metamodel is capable of handling problems with multiple disjoint failure regions and multiple most-probable points. It uses an approximate global metamodel which is refined locally for improved accuracy. The local domains are identified using clustering and a Principal Component Analysis (PCA). The overall methodology was demonstrated using a vibration absorber RBDO example, where the failure domain is disjoint with multiple most-probable points. The optimal design was obtained for different levels of system probability of failure, using only 245 function evaluations.

## REFERENCES


1. Rackwitz, R., and Fiessler, B., "Structural Reliability under Combined Random Load Sequences," *Computers and Structures*, 9(5), 484-494, 1978.
2. Breitung, K., "Asymptotic Approximations for Multinormal Integrals," *ASCE Journal of Engineering Mechanics*, 110, 357-366, 1984.
3. Der Kiureghian, A., and T. Dakessian, "Multiple Design Points in First and Second-Order Reliability," *Structural Safety*, 20(1), 37-49, 1998.
4. Bucher, C. G., "Adaptive Sampling – An Iterative Fast Mote Carlo Procedure," *Structural Safety*, 5, 119-126, 1988.
5. Melchers, R. E., "Improved Importance Sampling Methods for Structural System Reliability Calculation," *Proceedings of International Conference on Structural Safety and Reliability (ICOSSAR)*, 1185-1192, 1989.
6. Karamchandani, A., Bjerager, P., and Cornell, C. A., "Adaptive Importance Sampling," *5th International Conference on Structural Safety and Reliability*, (Eds. Ang, A. H.-S., Shinozuka, M. and Schuëller, G. I.), ASCE, 855-862, New York, 1989.
7. Mahadevan, S., and Raghothamachar, P., "Adaptive Simulation for System Reliability Analysis of Large Structures," *Computers and Structures*, 77, 725-734, 2000.
8. Zou, T., Mahadevan, S., Mourelatos, Z. P., and Meernik, P. R., "Reliability Analysis of Automotive Body-Door Subsystem," *Reliability Engineering and System Safety*, 78(3), 315-324, 2002.
9. Cornell, C. A., "Bounds on the Reliability of Structural Systems", *Journal of Structural Division,* 93 ST1, 171-200, 1967.
10. Ditlevsen, O., "Narrow Reliability Bounds for Structural Systems", *Journal of Structural Mechanics*, 7(4), 453-472, 1979.
11. Schueller, G. I., and R. Stix., "A Critical Appraisal of Methods to Determine Failure Probabilities" *Structural Safety,* 4(4),293-309
12. Fu, G., and Moses, F., "Multimodal Simulation Method for System Reliability Analysis", *Journal of Engineering Mechanics*, 119(6), 1173-1179, 1993.
13. Ditlevsen, O., and Bjerager, P., "Plastic Reliability Analysis by Directional Simulation", *DCAMM Report*, 353, Technical University of Denmark, Lyngby, Denmark, June 1987.
14. Mourelatos, Z. P., Kuczera, R., and Latcha, M., "An Efficient Monte Carlo Reliability Analysis using Global and Local Metamodels," *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA-2006-7087, Portsmouth, VA, September 6-8, 2006.
15. Kuczera, R., Mourelatos, Z. P., and Latcha, M., "A Monte Carlo Reliability Assessment for Multiple Failure Region Problems using Approximate Metamodels," *Proceedings of ASME Design Engineering Technical Conferences*, Paper# DETC2007-34957, 2007.
16. Shan, S., and Wang, G. G., "Failure Surface Frontier for Reliability Assessment on Expensive Performance Function," *ASME Journal of Mechanical Design*, 128, 1227-1235, 2006.
17. Pitchumani, R., and Mawardi, A., "SAMS: Stochastic Analysis with Minimal Sampling – A fast Algorithm for Analysis and Design under Uncertainty," *ASME Journal of Mechanical Design*, 127, 558-571, 2005.
18. Tu, J., and Jones, D. R., "Variable Screening in Metamodel Design by Cross-Validated Moving Least Squares Method," *Proceedings of 44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, AIAA-2003-1669, Norfolk, VA, April 7-10, 2003.
19. Ye, K. Q., Li, W., and A. Sudjianto, "Algorithmic Construction of Optimal Symmetric Latin Hypercube

Designs," *Journal of Statistical Planning and Inference*, 90, 145-159, 2000.
20. Johnson, M. E., Moore, L. M., and Ylvisaker, D., "Minimax and Maximin Distance Design," *Journal of Statistical Planning and Inference*, 26, 131-148, 1990.
21. Zou, T., Mourelatos, Z. P., Mahadevan, S., and Tu, J., "Indicator Response Surface for Simulation-Based Reliability Analysis," *ASME Journal of Mechanical Design*, 130(7), 2008.
22. Johnson, S. C., "Hierarchical Clustering Schemes," *Psychometrika*, 2, 241-254, 1967.
23. Melchers, R. E., *Structural Reliability Analysis and Prediction, 2nd Edition*, John Wiley & Sons, Chichester, U.K., 1999.
24. Farizal, and Nikolaidis, E., "Assessment of Imprecise Reliability using Efficient Probabilistic Reanalysis," *SAE World Congress*, Detroit, MI, Paper# 2007-01-0552, 2007.
25. Jolliffe, I. T., *Principal Component Analysis*, Springer-Verlag, New York, 1986.