Robust Design Optimization by Spline Dimensional Decomposition

Dongjin Lee[a,*], Ramin Jahanbin[b], Sharif Rahman[a]

[a]*Department of Mechanical Engineering, The University of Iowa, Iowa City, IA 52242, U.S.A.*
[b]*Front End Analytics, 800 Boylston Street, Suite 1600, Boston, MA 02199, U.S.A.*

Abstract

This article highlights new spline-empowered computational methods for solving robust design optimization (RDO) problems in complex mechanical systems. The methods are predicated on a spline dimensional decomposition (SDD) of a high-dimensional, discontinuous, or nonsmooth stochastic response for statistical moment analysis, a novel fusion of SDD and score functions for calculating the second-moment sensitivities with respect to the design variables, and standard gradient-based optimization algorithms. New closed-form formulae are derived for the design sensitivities that are concurrently determined along with the moments. The methods depend on how the statistical moment and sensitivity analyses are engaged with an optimization algorithm, engendering direct and multi-point single-step design processes. Numerical results reveal that the proposed methods deliver accurate and computationally efficient optimal solutions to RDO problems, including an industrial-scale shape design of a robotic gripper jaw.

Keywords: RDO, Second-moment analysis, SDD, Orthonormalized B-splines, ANOVA dimensional decomposition, Score functions, Stochastic optimization.

1. Introduction

Robust design optimization (RDO) is an outstanding paragon for designing an engineering system under uncertainty [1]. Whether or not uncertainty stems from loads, material, the manufacturing process, or operational environments, RDO explicitly treats such randomness by propagating input uncertainties to the objective and constraint functions, ultimately leading to insensitive designs [2]. Thus, RDO can yield a higher-quality and more economical design rather than a conservative design optimization employing heuristically derived safety factors. The accomplishments of RDO have been well reported in various real-world applications, such as those found in the design of aerospace, automotive, civil, and electronic structures, systems, or devices [2–11].

More often than not, in engineering design problems, the objective and/or constraint functions are composed of expensive-to-run functions, such as those stemming from finite element analysis (FEA). Hence, numerous studies have been directed towards employing a surrogate model, for instance, Taylor series or perturbation expansion [12], the point estimate method [13], polynomial chaos expansion (PCE) [2], the tensor-product quadrature rule [14], dimension-reduction methods [14], the polynomial dimensional decomposition (PDD) method [9], Kriging or Gaussian process regression [15], and artificial neural network [10]. While the preceding methods, including many others not listed here for brevity, have been successful in mitigating the extensive computational cost over crude Monte Carlo simulation (MCS), these methods mostly presume a globally continuous, smooth output response over the entire domain of input random variables. As a result, such surrogate methods can be easily degraded or are not suitable for dealing with locally significant changes, including discontinuity or nonsmoothness, in stochastic responses of interest. Therefore, one should approach employing smoothly connected piecewise polynomials, referred to as splines, with low expansion orders through smaller subdomains.

Most recently, a spline dimensional decomposition (SDD), composed of dimension-wise, orthonormalized basis splines (B-splines), has been introduced in papers [16, 17]. This method has been proven to be effective and computationally efficient in treating locally pronounced, high-dimensional, highly nonlinear, or nonsmooth responses. Consequently, a low-variate and/or low-degree SDD approximation with an adequate subinterval size is capable of delivering a more accurate estimate of the response second-moment than the higher-order ones of the existing popular methods, such as PCE or PDD. However, the SDD method is limited to forward uncertainty quantification (UQ) analysis. Therefore, this study focuses on solving RDO problems with SDD by developing new methods to address three challenges: (1) how to synchronously determine design sensitivities with statistical moments of output functions for a given design with no additional computational cost, (2) how to avoid repeated calculations of statistical moments and design sensitivities to the extent possible during design iterations, and (3) how to remarkably lessen the number of function evaluations or FEA, in conjunction with standard gradient-based optimization algorithms for problems with large design spaces. Only

⋆Grant sponsor: U.S. National Science Foundation; Grant No. CMMI-1933114.

*Corresponding author.
Email addresses: dongjin-lee@uiowa.edu (Dongjin Lee), raminj@feasol.com (Ramin Jahanbin), rahman@engineering.uiowa.edu (Sharif Rahman)

by addressing all three challenges successfully will the SDD method be further equipped to solve RDO problems in real-life applications.

The ultimate goal of this work is to create a solid theoretical foundation with a robust computational algorithm for UQ analysis and design optimization of complex engineering systems with high-dimensional, discontinuous, or nonsmooth stochastic responses. The novel design method to meet the goal is based on (1) SDD for determining the second-moment statistics of a high-dimensional, discontinuous, or nonsmooth stochastic response; (2) a novel fusion of SDD and score functions for calculating the second-moment sensitivities with respect to design variables; and (3) standard gradient-based optimization algorithms, constructing direct and multi-point single-step design processes. Therefore, the paper is organized as follows. In Section 2, mathematical statements of a general RDO problem are presented. Section 3 provides a brief description of SDD. In Section 4, new closed-form formulae for design sensitivities of statistical moments are disclosed, and in Section 5, new RDO methods are introduced. In Section 6, three numerical examples ranging from simple mathematical functions to an industrial-scale engineering application prove the efficacy of new RDO methods. Necessary future works are discussed in Section 7, and finally, the conclusions are presented in Section 8.

2. Robust design optimization

Let $\mathbb{N}$, $\mathbb{N}_0$, $\mathbb{R}$, and $\mathbb{R}_0^+$ be the sets of positive integer, non-negative integer, real number, and non-negative real number, respectively. Given $N \in \mathbb{N}$, denote by $\mathbb{R}^N$ the N-dimensional vector space of real numbers, and denote by $[a_k, b_k]$ a finite closed interval, where $a_k, b_k \in \mathbb{R}$ and $b_k > a_k$. Then, $\mathbb{A}^N = \times_{k=1}^N [a_k, b_k]$ is a closed bounded domain of $\mathbb{R}^N$.

Consider a measurable space $(\Omega_{\mathbf{d}}, \mathcal{F}_{\mathbf{d}})$, where $\Omega_{\mathbf{d}}$ is a sample space and $\mathcal{F}_{\mathbf{d}}$ is a σ-field on $\Omega_{\mathbf{d}}$. Defined over $(\Omega_{\mathbf{d}}, \mathcal{F}_{\mathbf{d}})$, let $\{\mathbb{P}_{\mathbf{d}} : \mathcal{F}_{\mathbf{d}} \to [0,1]\}$ be a family of probability measures where, given $M \in \mathbb{N}$ and $N \in \mathbb{N}$, $\mathbf{d} = (d_1, \cdots, d_M)^T \in \mathcal{D}$ is an M-dimensional design vector with non-empty closed set $\mathcal{D} \subset \mathbb{R}^M$. Here, $\mathbf{X} := (X_1, \cdots, X_N)^T : (\Omega_{\mathbf{d}}, \mathcal{F}_{\mathbf{d}}) \to (\mathbb{A}^N, \mathcal{B}^N)$ is an $\mathbb{A}^N$-valued input random vector with $\mathcal{B}^N$ representing the Borel σ-field on $\mathbb{A}^N$, specifying the statistical uncertainties or randomness in loads, material properties, or geometry of complex mechanical systems. The probability law of $\mathbf{X}$ is completely defined by a family of the joint probability density functions (PDF) $\left\{f_{\mathbf{X}}(\mathbf{x};\mathbf{d}) : \mathbf{x} \in \mathbb{R}^N,\ \mathbf{d} \in \mathcal{D}\right\}$ that are related to probability measures $\{\mathbb{P}_{\mathbf{d}} : \mathbf{d} \in \mathcal{D}\}$, so that the probability triple $(\Omega_{\mathbf{d}}, \mathcal{F}_{\mathbf{d}}, \mathbb{P}_{\mathbf{d}})$ of $\mathbf{X}$ depends on $\mathbf{d}$. In theory, a design variable d_l can be any distribution parameter or a statistic; however, here, d_l is limited to the mean of random variable X_l. Indeed, the design parameters as mean values are commonly applied in most engineering problems.

2.1. Problem definition

Let $y_a(\mathbf{X}) := y_a(X_1, \ldots, X_N)$, $a = 0, 1, \ldots, K$, represent a collection of $K+1$ real-valued, square-integrable, measurable transformations on $(\Omega_{\mathbf{d}}, \mathcal{F}_{\mathbf{d}})$, describing output functions of a complex system. They are also referred to as response or performance functions in applications. It is assumed that $y_a : (\mathbb{A}^N, \mathcal{B}^N) \to (\mathbb{R}, \mathcal{B})$ is not an explicit function of $\mathbf{d}$, but y_l implicitly depends on $\mathbf{d}$ via the probability law of $\mathbf{X}$. This is not a major limitation, as most, if not all, RDO problems involve means of random variables as design variables. In addition, let $\mathcal{D} = \times_{l=1}^M [d_{l,L}, d_{l,R}]$ be a closed rectangular subdomain of $\mathbb{R}^M$. From a fundamental viewpoint, RDO is performed by minimizing the mean and standard deviation of the performance individually. It usually leads to a bi-objective optimization problem, asking one to

$$\begin{aligned} \min_{\mathbf{d}\in\mathcal{D}\subseteq\mathbb{R}^M} \quad & \{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})],\ \sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}\}, \\ \text{subject to} \quad & \alpha_a \sqrt{\mathrm{var}_{\mathbf{d}}[y_a(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})] \le 0, \\ & a = 1, \ldots, K,\ 1 \le K < \infty \\ & d_{l,L} \le d_l \le d_{l,U},\ l = 1, \ldots, M, \end{aligned}$$

where

$$\mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})] := \int_{\mathbb{A}^N} y_a(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x};\mathbf{d}) d\mathbf{x}$$

is the mean of $y_a(\mathbf{X})$ and

$$\mathrm{var}_{\mathbf{d}}[y_a(\mathbf{x})] := \mathbb{E}_{\mathbf{d}}\left[y_a(\mathbf{X}) - \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})]\right]^2$$

is the variance of $y_a(\mathbf{X})$. Here, $\mathbb{E}_{\mathbf{d}}$ and $\mathrm{var}_{\mathbf{d}}$ are the expectation and variance operators, respectively, with respect to the probability measure $\mathbb{P}_{\mathbf{d}}$ or $f_{\mathbf{X}}(\mathbf{x};\mathbf{d})d\mathbf{x}$; $\alpha_a \in \mathbb{R}_0^+$, $a = 1, \ldots, K$, are non-negative, real-valued constants associated with the probabilities of constraint satisfaction; and $d_{l,L}$ and $d_{l,U}$ are the lower and upper bounds of the lth design variable d_l.

In many practical cases, the bi-objective optimization problem may demand to make optimal decisions in the presence of trade-offs between two clashing single-objective functions $\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]$ and $\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}$. In such a case, there exist an infinite number of optimal solutions, commonly named Pareto optimal solutions, where none of the single-objective function values can be enhanced without undermining the other. To obtain either multiple Pareto optimal solutions or a single solution that satisfies the preference of a decision-maker, the commonly used scalarization approach converts the bi-objective optimization problem into a single-objective optimization problem. As representative scalarization approaches, there exist the weighted-sum approach [18], the ϵ-constraint approach [19], the weighted-Tchebycheff approach [20, 21], goal programming [22], the lexicographic approach [23], and others [24]. In this study, the weighted-sum approach is mainly used, but there is no limit on the choice of scalarization approaches in solving the bi-objective optimization problem.

2.2. Proposed formulations

Two mathematical formulations of RDO – one expressed with respect to the original input random variables and the other described with respect to transformed input random variables

– are presented in the remainder of this section. Although the two formulations are equivalent to each other, having identical optimal solutions, the latter (alternative formulation) is more useful than the former (original formulation) in light of SDD approximations, as will be discussed in the forthcoming sections.

2.2.1. Original formulation

The mathematical formulation for RDO in most engineering problems comprising an objective function $c_0 : \mathcal{D} \to \mathbb{R}$ and constraint functions $c_a : \mathcal{D} \to \mathbb{R}$, where $a = 1, \ldots, K$ and $1 \leq K < \infty$, calls for one to [3, 4, 9]

$$\begin{aligned} \min_{\mathbf{d} \in \mathcal{D} \subseteq \mathbb{R}^M} \quad & c_0(\mathbf{d}) \quad := G\left(\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})],\ \sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}\right), \\ \text{subject to} \quad & c_a(\mathbf{d}) \quad := \alpha_a \sqrt{\mathrm{var}_{\mathbf{d}}[y_a(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})] \leq 0, \\ & \qquad a = 1, \ldots, K, \\ & \qquad d_{l,L} \leq d_l \leq d_{l,U},\ l = 1, \ldots, M, \end{aligned} \tag{1}$$

where $G(\cdot, \cdot)$ is an arbitrary function associated with any scalarization approach mentioned before. As a choice of scalarization, the weighted sum approach gives a certain weight to the distinct single-objective functions, say, $\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]$ and $\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}$, and then transforms them into a single-objective function, as follows:

$$\begin{aligned} & G\left(\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})],\ \sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}\right) \\ & \qquad := w_1 \frac{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]}{\mu_0^*} + w_2 \frac{\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}}{\sigma_0^*}, \end{aligned}$$

where $w_1 \in \mathbb{R}_0^+$ and $w_2 \in \mathbb{R}_0^+$ are two non-negative, real-valued weights such that $w_1 + w_2 = 1$; $\mu_0^* \in \mathbb{R} \setminus \{0\}$ and $\sigma_0^* \in \mathbb{R}_0^+ \setminus \{0\}$ are two non-zero, real-valued scaling factors. In the weighted-sum approach, equal weights are usually chosen, but they can be distinct or biased, depending on the objective set forth by a designer. In contrast, the scaling factors are relatively arbitrary but usually chosen to normalize the objective functions for a better condition to obtain an optimal solution.

2.2.2. Alternative formulation

Since the design variables are the means of part or all of the input random variables, a linear transformation, such as some shifting or scaling of random variables, creates an alternative formulation of RDO. Let $(X_{k_1}, \ldots, X_{k_M})^\intercal$ be an M-dimensional sub-vector of $\mathbf{X} := (X_1, \ldots, X_N)^\intercal$, $1 \leq k_1 \leq \cdots \leq k_M \leq N$, $M \leq N$, such that the means of its components are M design variables, say, $\mathbb{E}_{\mathbf{d}}[X_{i_k}] = d_k$, $k = 1, \ldots, M$.

Shifting. Let $\mathbf{Z} := (Z_1, \ldots, Z_N)^\intercal$ be an N-dimensional vector of transformed random variables obtained by shifting $\mathbf{X}$ as

$$\mathbf{Z} = \mathbf{X} + \mathbf{r}, \tag{2}$$

where $\mathbf{r} := (r_1, \ldots, r_N)^\intercal$ is an N-dimensional vector of deterministic variables. Define $g_k := \mathbb{E}_{\mathbf{d}}[Z_k]$ as the mean of the kth component of $\mathbf{Z}$. Denote by $(Z_{k_1}, \ldots, Z_{k_M})^\intercal$ a subvector of $\mathbf{Z}$, where the k_lth transformed random variable Z_{k_l} corresponds to the k_lth original random variable X_{k_l}. From the shifting transformation, the mean of Z_{k_l} is

$$\mathbb{E}_{\mathbf{d}}[Z_{k_l}] = d_l + r_{k_l} = g_l \tag{3}$$

and the PDF of $\mathbf{Z}$ is

$$f_{\mathbf{Z}}(\mathbf{z}; \mathbf{g}) = |\mathbf{J}| f_{\mathbf{X}}(\mathbf{x}; \mathbf{d}) = f_{\mathbf{X}}(\mathbf{x}; \mathbf{d}) = f_{\mathbf{X}}(\mathbf{z} - \mathbf{r}; \mathbf{d}),$$

supported on $\bar{\mathbb{A}}^N \subseteq \mathbb{R}^N$ (say). Here, the absolute value of the determinant of the Jacobian matrix is $|\mathbf{J}| = |\det[\partial \mathbf{x} / \partial \mathbf{z}]| = 1$ and the M-dimensional vector $\mathbf{g} := (g_1, \ldots, g_M)^\intercal$ has its lth component $g_l = \mathbb{E}_{\mathbf{d}}[Z_{k_l}]$, $l = 1, \ldots, M$.

Scaling. Let $\mathbf{Z} := (Z_1, \ldots, Z_N)^\intercal$ be an N-dimensional vector of transformed random variables obtained by scaling $\mathbf{X}$ as

$$\mathbf{Z} = \mathrm{diag}[r_1, \ldots, r_N]\mathbf{X}, \tag{4}$$

where $\mathbf{r} := (r_1, \ldots, r_N)^\intercal$ is an N-dimensional vector of deterministic variables. Define $g_k := \mathbb{E}_{\mathbf{d}}[Z_k]$ as the mean of the kth component of $\mathbf{Z}$. Denote by $(Z_{k_1}, \ldots, Z_{k_M})^\intercal$ a subvector of $\mathbf{Z}$, where the k_lth transformed random variable Z_{k_l} corresponds to the k_lth original random variable X_{k_l}. From the scaling transformation, the mean of Z_{k_l} is

$$\mathbb{E}_{\mathbf{d}}[Z_{k_l}] = d_l r_{k_l} = g_l \tag{5}$$

and the PDF of $\mathbf{Z}$ is

$$\begin{aligned} f_{\mathbf{Z}}(\mathbf{z}; \mathbf{g}) &= |\mathbf{J}|\, f_{\mathbf{X}}(\mathbf{x}; \mathbf{d}) = \left|\frac{1}{r_1 \ldots r_N}\right| f_{\mathbf{X}}(\mathbf{x}; \mathbf{d}) \\ &= \left|\frac{1}{r_1 \ldots r_N}\right| f_{\mathbf{X}}(\mathrm{diag}[1/r_1, \ldots, 1/r_N]\mathbf{z}; \mathbf{d}), \end{aligned}$$

supported on $\bar{\mathbb{A}}^N \subseteq \mathbb{R}^N$ (say). Here, the absolute value of the determinant of the Jacobian matrix is $|\mathbf{J}| = |\det[\partial \mathbf{x} / \partial \mathbf{z}]| = |1/(r_1 \ldots r_N)|$ and the M-dimensional vector $\mathbf{g} := (g_1, \ldots, g_M)^\intercal$ has its lth component $g_l = \mathbb{E}_{\mathbf{d}}[Z_{k_l}]$, $l = 1, \ldots, M$.

For each $a = 1, 2, \ldots, K$, define $h_a(\mathbf{Z}; \mathbf{r}) := y_a(\mathbf{X})$ to be the generic output function of the transformed random variables $\mathbf{Z}$, where the relation between $\mathbf{Z}$ and $\mathbf{X}$ is obtained by either shifting transformation in (2) or scaling transformation in (4). In both cases, the RDO formulation calls for one to

$$\begin{aligned} \min_{\mathbf{d} \in \mathcal{D} \subseteq \mathbb{R}^M} \quad & c_0(\mathbf{d}) \quad := G\Big(\mathbb{E}_{\mathbf{g}(\mathbf{d})}[h_0(\mathbf{Z}; \mathbf{r})], \\ & \qquad\qquad \sqrt{\mathrm{var}_{\mathbf{g}(\mathbf{d})}[h_0(\mathbf{Z}; \mathbf{r})]}\Big), \\ \text{subject to} \quad & c_a(\mathbf{d}) \quad := \alpha_a \sqrt{\mathrm{var}_{\mathbf{g}(\mathbf{d})}[h_a(\mathbf{Z}; \mathbf{r})]} \\ & \qquad\qquad - \mathbb{E}_{\mathbf{g}(\mathbf{d})}[h_a(\mathbf{Z}; \mathbf{r})] \leq 0, \\ & \qquad a = 1, \ldots, K, \\ & \qquad d_{l,L} \leq d_l \leq d_{l,U},\ l = 1, \ldots, M, \end{aligned} \tag{6}$$

where

$$\mathbb{E}_{\mathbf{g}(\mathbf{d})}[h_a(\mathbf{Z};\mathbf{r})] := \int_{\bar{\mathbb{A}}^N} h_a(\mathbf{z};\mathbf{r}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}) d\mathbf{z}$$

is the mean of $h_a(\mathbf{Z};\mathbf{r})$ and

$$\mathrm{var}_{\mathbf{g}(\mathbf{d})}[h_a(\mathbf{Z};\mathbf{r})] := \mathbb{E}_{\mathbf{g}(\mathbf{d})}\left[h_a(\mathbf{Z};\mathbf{r}) - \mathbb{E}_{\mathbf{g}(\mathbf{d})}[h_a(\mathbf{Z};\mathbf{r})]\right]^2$$

is the variance of $h_a(\mathbf{Z};\mathbf{r})$. Here, $\mathbb{E}_{\mathbf{g}(\mathbf{d})}$ and $\mathrm{var}_{\mathbf{g}(\mathbf{d})}$ are the expectation and variance operators, respectively, with respect to the probability measure $f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})d\mathbf{z}$, which depends on $\mathbf{d}$. For brevity, the subscript "$\mathbf{g}(\mathbf{d})$" of the expectation operator will be denoted by "$\mathbf{g}$" in the remainder of the paper.

The alternative formulation in (6) is only a rephrasing of (1) in terms of the transformed input random variables $\mathbf{Z}$. As a result, the probability measure of $\mathbf{Z}$ can be unchanged during design iterations, avoiding the need to recalculate measure-associated quantities. For the rest of the paper, RDO problems will be described with respect to the alternative formulation. In addition, $\mathbf{X}$ or $\mathbf{Z}$ and y_l or h_l will be referred to, interchangeably, as input random vector and output function, respectively.

2.2.3. Construction of sub-problems

A gradient-based solution to the RDO problem in (6) mandates suitable smoothness in objective and constraint functions. Therefore, both functions are assumed to be differentiable with respect to design variables. More often than not, as these functions are nonlinear, iterative approximations of (6), resulting in a sequence of RDO sub-problems, are required.

Let $q = 1, 2, \ldots, Q$, $Q \in \mathbb{N}$, be a design iteration count describing the qth RDO sub-problem for (6). Given q, let $\mathbf{d}^{\{q\}}$, $\mathbf{g}^{\{q\}}$, and $\mathbf{r}^{\{q\}}$ be the qth iterative versions of $\mathbf{d}$, $\mathbf{g}$, and $\mathbf{r}$, respectively. Then, the qth RDO sub-problem requires one to

$$\begin{aligned}
\min_{\mathbf{d}^{\{q\}} \in \mathcal{D} \subseteq \mathbb{R}^M} \quad & c_0^{\{q\}}(\mathbf{d}^{\{q\}}) := T\Bigg[G\Big(\mathbb{E}_{\mathbf{g}^{\{q\}}}[h_0(\mathbf{Z};\mathbf{r}^{\{q\}})], \\
& \qquad \sqrt{\mathrm{var}_{\mathbf{g}^{\{q\}}}[h_0(\mathbf{Z};\mathbf{r}^{\{q\}})]}\Big)\Bigg], \\
\text{subject to} \quad & c_a^{\{q\}}(\mathbf{d}^{\{q\}}) := T\Big[\alpha_a \sqrt{\mathrm{var}_{\mathbf{g}^{\{q\}}}[h_a(\mathbf{Z};\mathbf{r}^{\{q\}})]} \\
& \qquad - \mathbb{E}_{\mathbf{g}^{\{q\}}}[h_a(\mathbf{Z};\mathbf{r}^{\{q\}})]\Big] \le 0, \\
& a = 1, \ldots, K, \\
& d_{l,L} \le d_l^{\{q\}} \le d_{l,U},\ l = 1, \ldots, M,
\end{aligned} \tag{7}$$

where $c_0^{\{q\}}$ and $c_a^{\{q\}}$ are the qth objective and constraint functions, respectively. They are obtained iteratively from first- or higher-order Taylor series expansions T of c_0 and c_a at $\mathbf{d}_0^{\{q\}} = (d_{1,0}^{\{q\}}, \ldots, d_{M,0}^{\{q\}})^\intercal$. The solution of (7), denoted by $\mathbf{d}_*^{\{q\}} = (d_{1,*}^{\{q\}}, \ldots, d_{M,*}^{\{q\}})$, is conventionally produced using a suitable programming method, such as the well-known sequential linear and quadratic programming methods. In such a case, the qth RDO sub-problem solution $\mathbf{d}_*^{\{q\}}$ is used as the initial design for the $(q+1)$th RDO sub-problem by setting $\mathbf{d}_0^{\{q+1\}} = \mathbf{d}_*^{\{q\}}$. This process is repeated from a chosen initial design $\mathbf{d}_0 = \mathbf{d}_0^{\{1\}}$ until reaching convergence through all $Q \in \mathbb{N}$ iterations, yielding the final optimal design $\mathbf{d}^* = \mathbf{d}_*^{\{Q\}}$. In this paper, the iterations with respect to q are referred to as design iterations.

The RDO problem described so far does not explicitly mention a specific probability measure of $\mathbf{X}$ or $\mathbf{Z}$. This is because the proposed RDO formulation, whether (1) or (6), is applicable for a general input random vector endowed with independent or dependent probability measures. While most works on RDO mandate independent probability measures, a generalized PCE, capable of tackling dependent random variables, has been reported by the authors in a past work [2].

3. Spline dimensional decomposition

The application of SDD in solving the RDO problem posed in this work requires a number of assumptions on random input: (1) all component random variables X_l, Z_l, $l = 1, \ldots, N$, are statistically independent, but not necessarily identical; (2) each random variable X_l or Z_l is defined on a bounded interval, ensuring existence of all moments; and (3) the joint PDFs of $\mathbf{X}$ and $\mathbf{Z}$ are product-type, that is,

$$f_{\mathbf{X}}(\mathbf{x};\mathbf{d}) = \prod_{l=1}^{N} f_{X_l}(x_l;\mathbf{d}) \text{ and } f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}) = \prod_{l=1}^{N} f_{Z_l}(z_l;\mathbf{g}),$$

where $f_{X_l}(x_l;\mathbf{d})$ and $f_{Z_l}(z_l;\mathbf{g})$ are the marginal PDFs of X_l and Z_l, respectively. Therefore, $\mathbf{Z}$ is defined over the probability triple $(\Omega_{\mathbf{g}}, \mathcal{F}_{\mathbf{g}}, \mathbb{P}_{\mathbf{g}})$, depending on $\mathbf{g}$, where $\Omega_{\mathbf{g}}$ is the sample space of $\mathbf{Z}$, $\mathcal{F}_{\mathbf{g}}$ is a σ-algebra on $\Omega_{\mathbf{g}}$, and $\mathbb{P}_{\mathbf{g}}$ is a probability measure.

Denote by $h(\mathbf{Z};\mathbf{r})$ one of the stochastic performance functions $h_a(\mathbf{Z};\mathbf{r})$, $a = 1, \ldots, K$, used in RDO problems (6). The function $h(\mathbf{Z};\mathbf{r})$ is assumed to be in a reasonably large class of random variables, say, the Hilbert space

$$L^2(\Omega_{\mathbf{g}}, \mathcal{F}_{\mathbf{g}}, \mathbb{P}_{\mathbf{g}}) := \left\{h : \Omega_{\mathbf{g}} \to \mathbb{R} : \int_{\Omega_{\mathbf{g}}} h^2(\mathbf{Z};\mathbf{r}) d\mathbb{P}_{\mathbf{g}} < \infty\right\}.$$

This is tantamount to saying that the real-valued function $h(\mathbf{z};\mathbf{r})$ lives in the equivalent Hilbert space

$$\left\{h : \bar{\mathbb{A}}^N \to \mathbb{R} : \int_{\bar{\mathbb{A}}^N} h^2(\mathbf{z};\mathbf{r}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}) d\mathbf{z} < \infty\right\}.$$

The assumption guarantees the existence of the first two moments of $h(\mathbf{Z};\mathbf{r})$, facilitating a solution of the RDO problem in (6).

3.1. Knot vector

For the coordinate direction $k = 1, \ldots, N$, denote by a positive integer $n_k \in \mathbb{N}$ and a non-negative integer $p_k \in \mathbb{N}_0$ the number of basis functions and degree, respectively. Then, a knot sequence or vector $\boldsymbol{\xi}_k$ for the interval $[a_k, b_k] \subset \mathbb{R}$, given $n_k > p_k \ge 0$, is a

non-decreasing sequence of real numbers

$$\begin{aligned}\boldsymbol{\xi}_k &:= \{\xi_{k,i_k}\}_{i_k=1}^{n_k+p_k+1} \\ &= \{a_k = \xi_{k,1}, \xi_{k,2}, \ldots, \xi_{k,n_k+p_k+1} = b_k\}, \\ &\quad \xi_{k,1} \le \xi_{k,2} \le \cdots \le \xi_{k,n_k+p_k+1},\end{aligned} \tag{8}$$

where ξ_{k,i_k} is the i_kth knot with $i_k = 1, 2, \ldots, n_k + p_k + 1$ representing the knot index for the coordinate direction k. The $n_k + p_k + 1$ knots in (8) may be equally spaced or unequally spaced, resulting in a uniform or non-uniform distribution. Furthermore, the knots, whether they are exterior or interior, may be repeated; for instance, the i_kth knot of $\boldsymbol{\xi}_k$ may appear $1 \le m_{k,i_k} \le p_k + 1$ times, where m_{k,i_k} is referred to as its multiplicity. The multiplicity has important consequences on the regularity properties of B-spline functions. Interested readers are strongly recommended to read the papers [16, 25]. To define knots without repetitions, consider r_k distinct knots $\zeta_{k,1}, \ldots, \zeta_{k,r_k}$ in $\boldsymbol{\xi}_k$ with respective multiplicities $m_{k,1}, \ldots, m_{k,r_k}$. Then the knot vector in (8) is re-expressed as

$$\begin{aligned}\boldsymbol{\xi}_k &:= \{a_k = \overbrace{\zeta_{k,1}, \ldots, \zeta_{k,1}}^{m_{k,1} \text{ times}}, \overbrace{\zeta_{k,2}, \ldots, \zeta_{k,2}}^{m_{k,2} \text{ times}}, \\ &\quad \ldots, \overbrace{\zeta_{k,r_k-1} \ldots, \zeta_{k,r_k-1}}^{m_{k,r_k-1} \text{ times}}, \overbrace{\xi_{k,r_k}, \ldots, \xi_{k,r_k}}^{m_{k,r_k} \text{ times}} = b_k\}, \\ &\quad a_k = \zeta_{k,1} < \zeta_{k,2} < \cdots < \zeta_{k,r_k-1} < \zeta_{r_k} = b_k,\end{aligned}$$

which consists of

$$\sum_{i_k=1}^{r_k} m_{k,i_k} = n_k + p_k + 1$$

knots.

3.1.1. Non-uniformly spaced knot sequence

A uniformly spaced knot vector is generally recommended when there is no prior knowledge of the output function and the input random variable Z_k follows a uniform distribution. However, when an input random variable Z_k follows a non-uniform distribution, such as a Gaussian or lognormal distribution, one may have a free choice of knot vectors for a better result. Since it is not well-known how to set a suitable knot vector, it is often determined arbitrarily at hand. In this section, a new method for generating a non-uniformly spaced knot vector is introduced for the first time, which transforms a uniformly spaced knot vector to following the non-uniform distribution of the input random variable Z_k or its modified version having different distribution parameters.

For each coordinate direction k, let the knots of $\boldsymbol{\xi}_k$ in (8) be equally spaced, resulting in the uniform distribution on $\bar{\mathbb{A}} = [a_k, b_k]$. Defining the kth new random variable U_k following the same uniform distribution of the knot vector $\boldsymbol{\xi}_k$, denote by $F_{U_k} : \bar{\mathbb{A}} \to [0, 1]$ the CDF of U_k. Also, define another new random variable $\bar{Z}_k$ that follows the same distribution type of the input random variable Z_k but may have different values of distribution parameters, such as its mean or standard deviation. This is because the output function may have an uncontrolled fluctuation at a place far from the mean point (say). In such a case, placing the knot near the place can be critical for better prediction. Thus, using the knot vector following a modified version of input distribution allows flexibility to deal with such output functions, thus becoming more beneficial than the original one. To explain this in more detail, let the input random variable Z_k follow a Guassian or lognormal distribution. As mentioned before, consider a new random variable $\bar{Z}_k$ following a Gaussian distribution with $\mathbb{E}[\bar{Z}_k] = \mathbb{E}[Z_k]$ and $\sqrt{\text{var}[\bar{Z}_k]} = t\sqrt{\text{var}[Z_k]}$ or a lognormal distribution with $\mathbb{E}[\ln(\bar{Z}_k)] = \mathbb{E}[\ln(Z_k)]$ and $\sqrt{\text{var}[\ln(\bar{Z}_k)]} = t\sqrt{\text{var}[\ln(Z_k)]}$, respectively, where $t \in \mathbb{R}^+$ is a controlling factor. Then, regarding the uniformly spaced knots ξ_{k,i_k}, $i_k = 1, 2, \ldots, n_k + p_k + 1$, as arbitrary points of a random variable U_k, transform the knot sequence $\boldsymbol{\xi}_k$ to one following the distribution of $\bar{Z}_k$ by

$$\bar{\boldsymbol{\xi}}_k = F_{\bar{Z}_k}^{-1}(F_{U_k}(\boldsymbol{\xi}_k)),$$

where $F_{\bar{Z}_k}^{-1}$ is the inverse CDF of $\bar{Z}_k$. The resulting knot vector $\bar{\boldsymbol{\xi}}_k$ can be expressed as

$$\begin{aligned}\bar{\boldsymbol{\xi}}_k &:= \{\overbrace{\bar{\zeta}_{k,1}, \ldots, \bar{\zeta}_{k,1}}^{m_{k,1} \text{ times}}, \overbrace{\bar{\zeta}_{k,2}, \ldots, \bar{\zeta}_{k,2}}^{m_{k,2} \text{ times}}, \\ &\quad \ldots, \overbrace{\bar{\zeta}_{k,r_k-1} \ldots, \bar{\zeta}_{k,r_k-1}}^{m_{k,r_k-1} \text{ times}}, \overbrace{\bar{\zeta}_{k,r_k}, \ldots, \bar{\zeta}_{k,r_k}}^{m_{k,r_k} \text{ times}}\}, \\ &\quad \bar{\zeta}_{k,1} < \bar{\zeta}_{k,2} < \cdots < \bar{\zeta}_{k,r_k-1} < \bar{\zeta}_{k,r_k}.\end{aligned}$$

Since the left and right exterior knots must be placed at a_k and b_k, respectively, the non-uniformly spaced knot sequence $\boldsymbol{\xi}_k^*$, used here, is

$$\begin{aligned}\boldsymbol{\xi}_k^* &:= \{a_k = \overbrace{\zeta_{k,1}, \ldots, \zeta_{k,1}}^{m_{k,1} \text{ times}}, \overbrace{\bar{\zeta}_{k,2}, \ldots, \bar{\zeta}_{k,2}}^{m_{k,2} \text{ times}}, \\ &\quad \ldots, \overbrace{\bar{\zeta}_{k,r_k-1} \ldots, \bar{\zeta}_{k,r_k-1}}^{m_{k,r_k-1} \text{ times}}, \overbrace{\zeta_{k,r_k}, \ldots, \zeta_{k,r_k}}^{m_{k,r_k} \text{ times}} = b_k\}, \\ &\quad a_k = \zeta_{k,1} < \bar{\zeta}_{k,2} < \cdots < \bar{\zeta}_{r_k-1} < \zeta_{r_k} = b_k.\end{aligned}$$

For the remainder of the sections, $\boldsymbol{\xi}_k$ is used to represent a general knot vector including the uniformly or non-uniformly spaced knot vector $\boldsymbol{\xi}_k^*$.

3.2. Measure-consistent orthonormalized B-splines

For the coordinate direction k, denote by $B_{i_k,p_k,\xi_k}^k(z_k)$ the i_kth univariate B-spline with degree $p_k \in \mathbb{N}_0$. Given either a uniformly or non-uniformly spaced knot vector $\boldsymbol{\xi}_k$ and *zero*-degree basis functions, all higher-order B-spline functions on $[a_k, b_k]$ are defined recursively, where $1 \le k \le N$, $1 \le i_k \le n_k$, and $1 \le p_k < \infty$. See the recursive formula in Appendix A.1 for deriving $B_{i_k,p_k,\xi_k}^k(z_k)$.

The B-splines are bestowed with several desirable properties, which can generally deliver tremendous approximating power to numerical methods. For more specific details, see Appendix A.2.

3.2.1. Univariate orthonormalized B-splines

The aforementioned B-splines, although they form a basis of the spline space of degree p_k and knot vector $\boldsymbol{\xi_k}$, are not orthogonal with respect to the probability measure $f_{Z_k}(z_k;\mathbf{g})dz_k$ of Z_k. A linear transformation, originally proposed in the prequel [16, 25], is summarized in Appendix B in three steps to generate an n_k-dimensional vector $\boldsymbol{\psi}_k(z_k;\mathbf{g})$, consisting of orthonormalized B-splines

$$\psi^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k),\ i_k = 1,\dots,n_k,\ k = 1,\dots,N.$$

The resulting orthonormalized B-splines $\psi^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k)$, $i_k = 2,\dots,n_k$, have *zero* means. Furthermore, the orthonormalized B-splines are neither non-negative nor locally supported. However, an orthonormal basis is an essential ingredient in constructing an SDD expansion.

3.2.2. Multivariate orthonormalized B-splines

Due to the product-type probability measure of input random variables, measure-consistent multivariate orthonormalized B-splines in N variables are easily constructed from the N-dimensional tensor product of measure-consistent univariate splines. However, building such a tensor product in a high-dimensional setting is not recommended. Instead, the authors advocate constructing a series of tensor products in a dimension-wise manner.

For a subset $\emptyset \neq u = \{k_1,\dots,k_{|u|}\} \subseteq \{1,\dots,N\}$, let $\mathbf{Z}_u := (Z_{k_1},\dots,Z_{k_{|u|}})^\intercal$ be a subvector of $\mathbf{Z}$ defined on the abstract probability space $(\Omega^u_{\mathbf{g}},\mathcal{F}^u_{\mathbf{g}},\mathbb{P}^u_{\mathbf{g}})$, where $\Omega^u_{\mathbf{g}}$ is the sample space of $\mathbf{Z}_u$, $\mathcal{F}^u_{\mathbf{g}}$ is a σ-algebra on $\Omega^u_{\mathbf{g}}$, and $\mathbb{P}^u_{\mathbf{g}}$ is a probability measure. As $\mathbf{Z}$ comprises independent random variables, the PDF of $\mathbf{Z}_u$ is

$$f_{\mathbf{Z}_u}(\mathbf{z}_u) = \prod_{k\in u} f_{Z_k}(z_k;\mathbf{g}) = \prod_{l=1}^{|u|} f_{Z_{k_l}}(z_{k_l}),$$

where $\mathbf{z}_u := (z_{k_1},\dots,z_{k_{|u|}})^\intercal$ and, for $M \in \mathbb{N}$, $\mathbf{g} = (g_1,\dots,g_M)^\intercal$ obtained from the scaling or shifting transformation of an M-dimensional design vector with non-empty closed set $\mathcal{D} \subset \mathbb{R}^M$, as presented in (3) or (5), respectively. Thus, the probability triple $(\Omega^u_{\mathbf{g}},\mathcal{F}^u_{\mathbf{g}},\mathbb{P}^u_{\mathbf{g}})$ of $\mathbf{Z}_u$ depends on $\mathbf{g}$. Define three multi-indices $\mathbf{i}_u := (i_{k_1},\dots,i_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$, $\mathbf{n}_u := (n_{k_1},\dots,n_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$, and $\mathbf{p}_u := (p_{k_1},\dots,p_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$, representing the knot indices, numbers of basis functions, and degrees of splines, respectively, in all $|u|$ coordinate directions. Associated with $\mathbf{i}_u$, define an index set

$$\mathcal{I}_{u,\mathbf{n}_u} := \Big\{\mathbf{i}_u = (i_{k_1},\dots,i_{k_{|u|}}) : 1 \le i_{k_l} \le n_{k_l},\ l = 1,\dots,|u|\Big\} \subset \mathbb{N}^{|u|}$$

with cardinality

$$|\mathcal{I}_{u,\mathbf{n}_u}| = \prod_{k\in u} n_k.$$

For the coordinate direction k_l, denote by

$$I_{k_l} = r_{k_l} - 1,$$

the number of subintervals corresponding to the knot vector $\boldsymbol{\xi}_{k_l}$ with r_{k_l} distinct knots. Then, the partition, defined by the knot sequences $\boldsymbol{\xi}_{k_1},\dots,\boldsymbol{\xi}_{k_{|u|}}$, decomposes the $|u|$-dimensional rectangle $\bar{\mathbb{A}}^u := \times_{k\in u}[a_k,b_k]$ into smaller rectangles

$$\bar{\mathbb{A}}^u_{\mathbf{i}_u} = \Big\{\mathbf{z}_u = (z_{k_1},\dots,z_{k_{|u|}}) : \zeta_{k_l,i_{k_l}} \le z_{k_l} \le \zeta_{k_l,i_{k_l}+1},\ l = 1,\dots,|u|\Big\},$$

$$\mathbf{i}_u \in \Big\{\mathbf{i}_u = (i_{k_1},\dots,i_{k_{|u|}}) : 1 \le i_{k_l} \le I_{k_l},\ l = 1,\dots,|u|\Big\} \subseteq \mathcal{I}_{u,\mathbf{n}_u},$$

where $\zeta_{k_l,i_{k_l}}$ is the i_{k_l}th distinct knot in the coordinate direction k_l. A mesh is defined by the partition of $\bar{\mathbb{A}}^u$ into rectangular elements $\bar{\mathbb{A}}^u_{\mathbf{i}_u}$, $\mathbf{i}_u \in \mathcal{I}_{u,\mathbf{n}_u}$.

For $\emptyset \neq u = \{k_1,\dots,k_{|u|}\} \subseteq \{1,\dots,N\}$, with $\mathbf{p}_u = (p_{k_1},\dots,p_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$ and $\boldsymbol{\Xi}_u = \{\boldsymbol{\xi}_{k_1},\dots,\boldsymbol{\xi}_{k_{|u|}}\}$, the multivariate splines in $\mathbf{z}_u = (z_{k_1},\dots,z_{k_{|u|}})$ consistent with the probability measure $f_{\mathbf{Z}_u}(\mathbf{z}_u;\mathbf{g})d\mathbf{z}_u$ are

$$\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}_u;\mathbf{g}) = \prod_{k\in u}\psi^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k;\mathbf{g}) = \prod_{l=1}^{|u|}\psi^k_{i_{k_l},p_{k_l},\boldsymbol{\xi}_{k_l}}(z_{k_l};\mathbf{g}),\quad \mathbf{i}_u = (i_{k_1},\dots,i_{k_{|u|}}) \in \mathcal{I}_{u,\mathbf{n}_u}. \tag{9}$$

When the input random variables $Z_1,\dots,Z_N$, instead of real variables $z_1,\dots,z_N$, are inserted in (9), the multivariate splines $\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}_u;\mathbf{g})$, $\emptyset \neq u \subseteq \{1,\dots,N\}$, $\mathbf{i}_u \in \mathcal{I}_{u,\mathbf{n}_u}$, become functions of input random variables. To describe their second-moment properties succinctly, limit the range of the index i_{k_l}, $l = 1,\dots,|u|$, associated with the k_lth variable z_{k_l}, to $2,\dots,n_{k_l}$. The exclusion of $i_{k_l} = 1$ is essentially the first constant element of $\Psi_k(Z_k)$. Hence, define a reduced index set

$$\bar{\mathcal{I}}_{u,\mathbf{n}_u} := \Big\{\mathbf{i}_u = (i_{k_1},\dots,i_{k_{|u|}}) : 2 \le i_{k_l} \le n_{k_l},\ l = 1,\dots,|u|\Big\} \subset (\mathbb{N}\backslash\{1\})^{|u|},$$

which has cardinality

$$|\bar{\mathcal{I}}_{u,\mathbf{n}_u}| := \prod_{k\in u}(n_k - 1).$$

The omission of $i_{k_l} = 1$ in $\bar{\mathcal{I}}_{u,n_\mathbf{u}}$ prevents reduction of the degree of interaction of the corresponding multivariate spline basis below $|u|$. Then, for $\emptyset \neq u,v \subseteq \{1,\dots,N\}$, $\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}$, and $\mathbf{j}_v \in \bar{\mathcal{I}}_{v,\mathbf{n}_v}$, the first-and second-order moments of multivariate

orthonormalized B-splines are [16]

$$\mathbb{E}_{\mathbf{g}}[\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{Z}_u;\mathbf{g})] = 0 \tag{10}$$

and

$$\begin{aligned}&\mathbb{E}_{\mathbf{g}}[\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{Z}_u;\mathbf{g})\Psi^{v}_{\mathbf{j}_v,\mathbf{p}_v,\Xi_v}(\mathbf{Z}_v;\mathbf{g})]\\ &= \begin{cases}1, & u = v \text{ and } \mathbf{i}_u = \mathbf{j}_v,\\ 0, & \text{otherwise,}\end{cases}\end{aligned} \tag{11}$$

respectively.

Note that the three steps mentioned earlier to create orthonormalized B-splines are described in terms of $\mathbf{z}$, not $\mathbf{x}$. This is intended to employ invariant $\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{z};\mathbf{g})$ throughout all design iterations by assigning unchanged values to $\mathbf{g}$ while the design vector $\mathbf{d}$ is updated at each iteration. To explain this further, consider the qth RDO sub-problem in 7, where the shifting or scaling transformation for the kth initial design variable yield

$$\mathbb{E}_{\mathbf{d}^{\{q\}}}[Z_{i_k}] = g_k^{\{q\}} = \begin{cases}d_k^{\{q\}} + r_{i_k}^{\{q\}}, & \text{shifting,}\\ d_k^{\{q\}} r_{i_k}^{\{q\}}, & \text{scaling.}\end{cases} \tag{12}$$

Here, when setting $d_k^{\{q\}}$ to $d_{k,0}^{\{q\}}$ at the qth iteration, one is free to choose the value of $g_k^{\{q\}}$ with respect to $d_{k,0}^{\{q\}}$ in (12). However, for the sake of convenience, choose the value of $g_k^{\{q\}}$ as *zero* or *one* in the shifting or scaling transformation, respectively. Then, $r_{i_k}^{\{q\}}$ is determined to be $-d_{k,0}^{\{q\}}$ and $1/d_{k,0}^{\{q\}}$, respectively, from (12). Solving the qth RDO sub-problem with the initial design $\mathbf{d}_0^{\{q\}}$ yields $\mathbf{d}_*^{\{q\}}$; thereby $g_k^{\{q\}}$ becomes $d_{k,*}^{\{q\}} - d_{k,0}^{\{q\}}$ and $d_{k,*}^{\{q\}}/d_{k,0}^{\{q\}}$ in the shifting or scaling transformation, respectively. In the repetitively updating process from qth to $(q+1)$th design iterations, choosing the same values of $g_k^{\{q\}}$ for $q = 1, 2, \ldots, Q$ contributes to only one sequence of calculation of the measure-consistent orthonormal polynomials $\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{z};\mathbf{g})$ throughout all design iterations.

3.3. SDD approximation

Suppose the degree and family of knot sequences in all coordinate directions have been specified as $\mathbf{p} = (p_1, \ldots, p_N) \in \mathbb{N}_0^{|u|}$ and $\Xi = \{\xi_1, \ldots, \xi_N\}$, respectively. For $\emptyset \neq u \subseteq \{1, \ldots, N\}$ and $\mathbf{Z}_u := (Z_{k_1}, \ldots, Z_{k_u})^{\intercal}$ over $(\Omega^u_{\mathbf{g}}, \mathcal{F}^u_{\mathbf{g}}, \mathbb{P}^u_{\mathbf{g}})$, with $\mathbf{p}_u = (p_{k_1}, \ldots, p_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$ and $\Xi_u = \{\xi_{k_1}, \ldots, \xi_{k_{|u|}}\}$, denote by $\{\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{Z}_u) : \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\}$ a set comprising multivariate orthonormalized B-splines that is consistent with the probability measure $f_{\mathbf{Z}_u}(\mathbf{z}_u)d\mathbf{z}_u$. Then, for any random variable $h(\mathbf{Z}) \in L^2(\Omega_{\mathbf{g}}, \mathcal{F}_{\mathbf{g}}, \mathbb{P}_{\mathbf{g}})$, there exists a hierarchically expanded Fourier-lie series in multivariate orthonormal splines in $\mathbf{Z}_u$, referred to as the SDD [16],

$$\begin{aligned}h_{\mathbf{p},\Xi}(\mathbf{Z};\mathbf{r}) \;:=\; & h_{\emptyset}(\mathbf{r}) + \sum_{\emptyset\neq u\subset\{1,\ldots,N\}}\sum_{\mathbf{i}_u\in\bar{\mathcal{I}}_{u,\mathbf{n}_u}} C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{r})\\ & \times\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{Z}_u;\mathbf{g})\end{aligned} \tag{13}$$

with its expansion coefficients

$$h_{\emptyset}(\mathbf{r}) := \int_{\bar{\mathbb{A}}^N} h(\mathbf{z};\mathbf{r}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}) d\mathbf{z}. \tag{14}$$

$$\begin{aligned}C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{r}) \;:=\; & \int_{\bar{\mathbb{A}}^N} h(\mathbf{z};\mathbf{r})\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{z}_u;\mathbf{g}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}) d\mathbf{z},\\ & \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}.\end{aligned} \tag{15}$$

In a practical setting, the output function $h(\mathbf{Z};\mathbf{r})$ is likely to have an effective dimension much lower than N, indicating that the right side of (13) can be effectively approximated by a sum of lower dimensional component functions of $h_{\mathbf{p},\Xi}(\mathbf{Z};\mathbf{r})$ but still preserve all random variables of N-dimension. Furthermore, the dimensional hierarchical structure of the basis functions permits ignoring safely a negligible amount to the function value. A straightforward approach is to keep all orthonormalized B-splines in at most $1 \leq S \leq N$ variables, resulting in an S-variate SDD approximation

$$\begin{aligned}h_{S,\mathbf{p},\Xi}(\mathbf{Z};\mathbf{r}) \;:=\; & h_{\emptyset}(\mathbf{r}) + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\leq|u|\leq S}}\sum_{\mathbf{i}_u\in\bar{\mathcal{I}}_{u,\mathbf{n}_u}} C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{r})\\ & \times\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\Xi_u}(\mathbf{Z}_u;\mathbf{g})\end{aligned} \tag{16}$$

of $h(\mathbf{Z};\mathbf{r})$, comprising

$$L_{S,\mathbf{p},\Xi} = 1 + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\leq|u|\leq S}}\prod_{k\in u}(n_k - 1) \leq \prod_{k=1}^{N} n_k = L_{\mathbf{p},\Xi}$$

expansion coefficients, including $h_{\emptyset}$. Due to its additive structure, the approximation in (16) includes degrees of interaction among at most S input variables $Z_{k_1}, \ldots, Z_{k_S}$, $1 \leq k_1 \leq \cdots \leq k_S \leq N$. For example, by selecting $S = 1$ and 2, the respective univariate and bivariate SDD approximations are

$$h_{1,\mathbf{p},\Xi}(\mathbf{Z};\mathbf{r}) = h_{\emptyset}(\mathbf{r}) + \sum_{k=1}^{N}\sum_{i_k=2}^{n_k} C_{ki_k}(\mathbf{r})\Psi^{k}_{i_k,p_k,\xi_k}(Z_k;\mathbf{r})$$

and

$$\begin{aligned}h_{2,\mathbf{p},\Xi}(\mathbf{Z};\mathbf{r}) \;=\; & h_{\emptyset}(\mathbf{r}) + \sum_{k=1}^{N}\sum_{i_k=2}^{n_k} C_{ki_k}(\mathbf{r})\Psi^{k}_{i_k,p_k,\xi_k}(Z_k;\mathbf{r})\\ & + \sum_{k_1=1}^{N-1}\sum_{k_2=k_1+1}^{N}\sum_{i_{k_1}=2}^{n_{k_1}}\sum_{i_{k_2}=2}^{n_{k_2}} C_{k_1k_2i_{k_1}i_{k_2}}(\mathbf{r})\\ & \times\Psi^{k1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{r})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{r}).\end{aligned}$$

Note that the univariate and bivariate SDD approximations should not be viewed as first- and second-order approximations, nor as limiting the nonlinearity of h. When $S \to N$ and $n_k \to \infty$, $h_{S,\mathbf{p},\Xi}$ converges to h in the mean-square sense, permitting (16) to generate a hierarchical convergent sequence of approximation of h.

3.4. Standard least-squares for expansion coefficient estimation

The SDD expansion coefficients are naturally computed by applying their respective definitions in (14) and (15). However, they are determined by various numerical integrations, which can be costly, in particular for high-dimensional problems. As an alternative, the standard least-squares (SLS) regression is an efficient way to estimate unknown coefficients in linear systems. In this work, the SLS method is used to estimate the SDD expansion coefficients, thereby sidestepping the expensive numerical integration. This is performed in the context of a single-index description of SDD as follows.

Consider the set of measure-consistent multivariate orthonormalized B-splines

$$\left\{\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{Z}_u;\mathbf{g}) : 1 \le |u| \le S,\ \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\right\}, \tag{17}$$

as previously defined in Section 3.2.2, which consists of $L_{S,\mathbf{p},\boldsymbol{\Xi}}$ basis functions. Without loss of generality and consistency of the orthonormalization procedure discussed in Section 3.2.1, with an arbitrary order of choice, arrange the elements of the set in (17) as

$$\begin{aligned}&\left\{\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{Z}_u;\mathbf{g}) : 1 \le |u| \le S,\ \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\right\}\\ &= \left\{\Psi_2(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi}),\ldots,\Psi_{L_{S,\mathbf{p},\boldsymbol{\Xi}}}(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi})\right\},\ \ \Psi_1(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi}) = 1,\end{aligned}$$

such that $\Psi_i(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi})$, $i = 1,\ldots,L_{S,\mathbf{p},\boldsymbol{\Xi}}$ represents the ith basis function in the truncated SDD approximation. For each basis function, there exists the corresponding expansion coefficient $C_i(\mathbf{r},\boldsymbol{\Xi}) \in \mathbb{R},\ i = 1,\ldots,L_{S,\mathbf{p},\boldsymbol{\Xi}}$. Hence, the SDD approximation in (16) is rewritten as

$$h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r}) := \sum_{i=1}^{L_{S,\mathbf{p},\boldsymbol{\Xi}}} C_i(\mathbf{r},\boldsymbol{\Xi})\Psi_i(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi}). \tag{18}$$

Then, employing the SLS method, the approximate SDD expansion coefficients of $h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})$ in (18) are computed through minimizing

$$\mathbb{E}_{\mathbf{g}}\left[h(\mathbf{Z};\mathbf{r}) - \sum_{i=1}^{L_{S,\mathbf{p},\boldsymbol{\Xi}}} C_i(\mathbf{r},\boldsymbol{\Xi})\Psi_i(\mathbf{Z};\mathbf{g},\boldsymbol{\Xi})\right]^2.$$

From random input $\mathbf{Z}$ with known distributions and an output function $h : \bar{\mathbb{A}}^N \to \mathbb{R}$, consider an input-output data set $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)};\mathbf{r})\}_{t=1}^L$ of a sample size $L \in \mathbb{N}$, where $\mathbf{r}$ is determined in (3) and (4). The data set, often referred to as the experimental design, is generated by evaluating the function h at each input sample $\mathbf{z}^{(t)}$. There exist various sampling methods, such as standard MCS, quasi MCS (QMCS), and Latin hypercube sampling (LHS), for constructing the input-output data set or the experimental design. In this work, an optimal LHS method, presented in Appendix B, is employed for all numerical examples. From the constructed data set $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)};\mathbf{r})\}_{t=1}^L$, the SLS method minimizes the mean-square error

$$e_{S,\mathbf{p},\boldsymbol{\Xi}} := \frac{1}{L}\sum_{t=1}^{L}\left[h(\mathbf{z}^{(t)};\mathbf{r}) - \sum_{i=1}^{L_{S,\mathbf{p},\boldsymbol{\Xi}}} C_i(\mathbf{r},\boldsymbol{\Xi})\Psi_i(\mathbf{z}^{(t)};\mathbf{g},\boldsymbol{\Xi})\right]^2,$$

which is done by estimating the expansion coefficients through

$$\begin{aligned}\hat{\mathbf{c}} &:= \left(\hat{C}_1(\mathbf{r},\boldsymbol{\Xi}),\ldots,\hat{C}_{L_{S,\mathbf{p},\boldsymbol{\Xi}}}(\mathbf{r},\boldsymbol{\Xi})\right)^{\mathsf{T}}\\ &= \left(\mathbf{A}^{\mathsf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{b}\end{aligned}$$

as the best estimation of $\mathbf{c} := (C_1(\mathbf{r},\boldsymbol{\Xi}),\ldots,C_{L_{S,\mathbf{p},\boldsymbol{\Xi}}}(\mathbf{r},\boldsymbol{\Xi}))^{\mathsf{T}}$, where

$$\begin{aligned}\mathbf{A} &:= \begin{bmatrix}\Psi_1(\mathbf{z}^{(1)};\mathbf{g},\boldsymbol{\Xi}) & \cdots & \Psi_{L_{S,\mathbf{p},\boldsymbol{\Xi}}}(\mathbf{z}^{(1)};\mathbf{g},\boldsymbol{\Xi})\\ \vdots & \ddots & \vdots\\ \Psi_1(\mathbf{z}^{(L)};\mathbf{g},\boldsymbol{\Xi}) & \cdots & \Psi_{L_{S,\mathbf{p},\boldsymbol{\Xi}}}(\mathbf{z}^{(L)};\mathbf{g},\boldsymbol{\Xi})\end{bmatrix},\\ \mathbf{b} &:= \left(h(\mathbf{z}^{(1)};\mathbf{r}),\ldots,h(\mathbf{z}^{(L)};\mathbf{r})\right)^{\mathsf{T}}.\end{aligned}$$

Here, the $L \times L_{S,\mathbf{p},\boldsymbol{\Xi}}$-matrix $\mathbf{A}$ carries the information about the basis functions evaluated at $\{\mathbf{z}^{(t)}\}_{t=1}^L$, and the L-dimensional column vector $\mathbf{b}$ has the responses by a deterministic solver, such as the finite element method.

Furthermore, the $L_{S,\mathbf{p},\boldsymbol{\Xi}} \times L_{S,\mathbf{p},\boldsymbol{\Xi}}$-square matrix $\mathbf{A}^{\mathsf{T}}\mathbf{A}$, which is referred to as the information matrix, plays a pivotal role in the SLS method. The accuracy of the regression method hinges on the condition number of the information matrix, which itself depends on how many samples are used in the data set and how they are chosen for analysis. In other words, the matrix $\mathbf{A}^{\mathsf{T}}\mathbf{A}$ should be nonsingular and well-conditioned. While there are rules of thumb stating that two or three times the number of expansion coefficients $L_{S,\mathbf{p},\boldsymbol{\Xi}}$ is ordinarily acceptable for the sample size L, the selection of L heavily depends on applications at hand and may yield either inaccurate, if attainable, results or unnecessarily expensive computations if L is too small or too large, respectively. Having said this, a necessary condition for the SLS method reads $L > L_{S,\mathbf{p},\boldsymbol{\Xi}}$.

3.5. Statistical moment analysis

The S-variate SDD approximation $h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})$ can be viewed as an inexpensive surrogate of an expensive-to-calculate function $h(\mathbf{Z};\mathbf{r})$. Therefore, pertinent statistical properties of $h(\mathbf{Z})$, such as its first two moments, can be estimated from those of $h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z})$.

Applying the expectation operator on $h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z})$, its mean

$$\mathbb{E}_{\mathbf{g}}[h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})] = h_{\emptyset}(\mathbf{r}) = \mathbb{E}_{\mathbf{g}}[h(\mathbf{Z};\mathbf{r})]$$

is independent of S, $\mathbf{p}$, and $\boldsymbol{\Xi}$. More importantly, the SDD approximation always yields the exact mean, provided that the expansion coefficient $h_{\emptyset}(\mathbf{r})$ is determined exactly.

Applying the expansion operator again, such that $\mathbb{E}_{\mathbf{g}}[h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z}) - h_{\emptyset}(\mathbf{r})]^2$, then the result of the variance of

$h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z})$ is

$$\begin{aligned}\mathbb{E}_{\mathbf{g}}[h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z}) - h_{\emptyset}(\mathbf{r})]^2 &= \mathrm{var}_{\mathbf{g}}[h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})] \\ &= \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\le|u|\le S}}\sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^{u2}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}) \\ &\le \mathrm{var}[h(\mathbf{Z};\mathbf{r})]\end{aligned}$$

of $h(\mathbf{Z};\mathbf{r})$.

In this paper, only a brief exposition of SDD is given. Readers interested in theoretical details of SDD [16], including applications to isogeometric analysis [17] are referred to the authors' past works.

4. Proposed method for design sensitivity analysis

Solving an RDO problem by any gradient-based optimization method, such as linear or sequential quadratic programming, demands at least the computations of the first-order derivatives of the first- and second-order moments $h(\mathbf{Z};\mathbf{r})$ with respect to each design variable d_l, $l = 1,\ldots,M$. Here, a novel analytical sensitivity formulation is derived by coupling SDD expansion coefficients with score functions for input random variables. The assessment of the following regularity conditions should precede the sensitivity analysis:

1. The probability density function $f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})$ of $\mathbf{Z}$ is continuous. Also, the partial derivative $\partial f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})/\partial g_l$, $l = 1,\ldots,M$, exists for all possible values of $\mathbf{z}$ and g_l. Furthermore, the statistical moments of $h(\mathbf{Z};\mathbf{r})$ are differentiable with respect to g_l.

2. There exists a Lebesgue integrable dominating function $b(\mathbf{z})$ such that

$$\left|h^r(\mathbf{z};\mathbf{r})\frac{\partial f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})}{\partial g_l}\right| \le b(\mathbf{z}),\ r = 1,2;\ l = 1,\ldots,M.$$

4.1. Score functions

Taking the partial derivative of the first two moments with respect to d_l and then interchanging the order of the differential and integral operators, by the Lebesgue dominated convergence theorem, yields the sensitivities

$$\begin{aligned}\frac{\partial\mathbb{E}_{\mathbf{g}}[h^r(\mathbf{Z};\mathbf{r})]}{\partial d_l} &= \frac{\partial g_l}{\partial d_l}\frac{\partial}{\partial g_l}\int_{\bar{\mathbb{A}}^N} h^r(\mathbf{z};\mathbf{r}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})d\mathbf{z} \\ &= \frac{\partial g_l}{\partial d_l}\int_{\bar{\mathbb{A}}^N} h^r(\mathbf{z};\mathbf{r})\frac{\partial \ln f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})}{\partial g_l} f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})d\mathbf{z}, \\ &\quad r = 1,2;\ l = 1,\ldots,M.\end{aligned} \tag{19}$$

In (19), $\partial \ln f_{\mathbf{Z}}(\mathbf{Z};\mathbf{g})/\partial g_l$ is known as the first-order score function for g_l or, simply, the lth score function. As input $\mathbf{Z}$ are assumed as independent random variables, the log of multivariate density function

$$\ln f_{\mathbf{Z}}(\mathbf{Z};\mathbf{g}) = \sum_{k=1}^{N}\ln f_{Z_k}(Z_k;\mathbf{g}),$$

which is a sum of univariate functions. If g_l is any distribution parameter of Z_{k_l}, then the score function for the g_l can be simplified to $\partial \ln f_{\mathbf{Z}}(\mathbf{Z};\mathbf{g})/\partial g_l = \partial \ln f_{Z_{k_l}}(Z_{k_l};\mathbf{g})/\partial g_l$. Denoting by $s_l(Z_{k_l};\mathbf{g}) := \partial \ln f_{Z_{k_l}}(Z_{k_l};\mathbf{g})/\partial g_l$ the lth score function, the sensitivity is obtained from

$$\begin{aligned}\frac{\partial\mathbb{E}_{\mathbf{g}}[h^r(\mathbf{Z};\mathbf{r})]}{\partial d_l} &= \frac{\partial g_l}{\partial d_l}\int_{\bar{\mathbb{A}}^N} h^r(\mathbf{z};\mathbf{r}) s_l(z_{k_l};\mathbf{g}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g})d\mathbf{z}, \\ &=: \frac{\partial g_l}{\partial d_l}\mathbb{E}_{\mathbf{g}}[h^r(\mathbf{Z};\mathbf{r})s_l(Z_{k_l};\mathbf{g})], \\ &\quad r = 1,2;\ l = 1,\ldots,M.\end{aligned} \tag{20}$$

4.2. Design sensitivities

For independent coordinates of $\mathbf{Z}$, consider the Fourier spline expansion of the lth score function $s_l(Z_{k_l};\mathbf{g})$

$$\begin{aligned}s_{l,p_{k_l},\xi_{k_l}}(Z_{k_l};\mathbf{g}) &= s_{l,\emptyset}(\mathbf{g}) + \sum_{i_{k_l}=2}^{n_{k_l}} D_{k_l,i_{k_l}}(\mathbf{g})\psi^{k_l}_{i_{k_l},p_{k_l},\xi_{k_l}}(Z_{k_l};\mathbf{g}), \\ &= \sum_{i_{k_l}=2}^{n_{k_l}} D_{k_l,i_{k_l}}(\mathbf{g})\psi^{k_l}_{i_{k_l},p_{k_l},\xi_{k_l}}(Z_{k_l};\mathbf{g}),\end{aligned} \tag{21}$$

where

$$D_{k_l,i_{k_l}}(\mathbf{g}) := \int_{\bar{\mathbb{A}}^{\{k_l\}}} s_l(z_{k_l};\mathbf{g})\psi^{k_l}_{i_{k_l},p_{k_l},\xi_{k_l}}(z_{k_l};\mathbf{g}) f_{Z_{k_l}}(z_{k_l};\mathbf{g})dz_{k_l}$$

and

$$\begin{aligned}s_{l,\emptyset}(\mathbf{g}) &= \int_{\bar{\mathbb{A}}^{\{k_l\}}} s_l(z_{k_l};\mathbf{g}) f_{Z_{k_l}}(z_{k_l};\mathbf{g})dz_{k_l} \\ &=: \mathbb{E}_{\mathbf{g}}[s_{l,\emptyset}(Z_{k_l};\mathbf{g})] = 0.\end{aligned}$$

Then, applying the SDD approximation of an output h in (16) and the Fourier spline expansion of the lth score function in (21) to the integrand $h^r(\mathbf{Z};\mathbf{r})s_l(Z_{k_l};\mathbf{g})$ of (20) yields

$$\begin{aligned}&h^r_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})s_{l,p_{k_l},\xi_{k_l}}(Z_{k_l};\mathbf{g}) \\ &= \left(h_{\emptyset}(\mathbf{r}) + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\le|u|\le S}}\sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r})\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{Z}_u,\mathbf{g})\right)^r \\ &\times\left(\sum_{i_{k_l}=2}^{n_{k_l}} D_{k_l,i_{k_l}}(\mathbf{g})\psi^{k_l}_{i_{k_l},p_{k_l},\xi_{k_l}}(Z_{k_l};\mathbf{g})\right).\end{aligned} \tag{22}$$

Thereafter, applying the expectation operator on (22) and using the second-moment properties in (10) and (11), the sensitivities of the first-order ($r = 1$) and second-order ($r = 2$) moments with respect to d_l are derived as

$$\frac{\partial\mathbb{E}_{\mathbf{g}}[h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})]}{\partial d_l} = \frac{\partial g_l}{\partial d_l}\sum_{i_{k_l}=2}^{n_{k_l}} C_{k_l i_{k_l}}(\mathbf{r})D_{k_l,i_{k_l}}(\mathbf{g}), \tag{23}$$

and

$$\frac{\partial \mathbb{E}_{\mathbf{g}}[h^2_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})]}{\partial d_l} = \frac{\partial g_l}{\partial d_l}\Bigg(2h_{\emptyset}(\mathbf{r})\sum_{i_{k_l}=2}^{n_{k_l}} C_{k_l i_{k_l}}(\mathbf{r})D_{k_l,i_{k_l}}(\mathbf{g}) + \sum_{j=1}^{J} T_{l,j}\Bigg), J \in \mathbb{N}, \tag{24}$$

respectively. The closed-form expressions of the moment sensitivities in (23) and (24) mainly consist of the SDD and Fourier spline expansion coefficients of the response and score functions, respectively. For (24), the last terms $T_{l,j}$, $j = 1, 2, 3$, if $J = 3$, are

$$\begin{aligned} T_{l,1} = & \sum_{k_1=1}^{N}\sum_{k_2=1}^{N}\sum_{i_{k_1}=2}^{n_{k_1}}\sum_{i_{k_2}=2}^{n_{k_2}}\sum_{i_{k_3}=2}^{n_{k_3}} C_{k_1 i_{k_1}}(\mathbf{r})C_{k_2 i_{k_2}}(\mathbf{r}) \\ & \times D_{k_l,i_{k_3}}(\mathbf{g})\mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g}) \\ & \times \Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_l}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Big], \end{aligned}$$

where

$$\begin{aligned} &\mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Big] \\ &= \begin{cases} \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g}) \\ \times\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Big], & \text{if } k_1 = k_2 = k_3, \\ 0, \quad \text{otherwise}, \end{cases} \end{aligned} \tag{25}$$

$$\begin{aligned} T_{l,2} = & 2\sum_{k_1=1}^{N}\sum_{k_2=1}^{N-1}\sum_{k_3=k_2+1}^{N}\sum_{i_{k_1}=2}^{n_{k_1}}\sum_{i_{k_2}=2}^{n_{k_2}}\sum_{i_{k_3}=2}^{n_{k_3}}\sum_{i_{k_4}=2}^{n_{k_4}} C_{k_1 i_{k_1}}(\mathbf{r}) \\ & \times C_{k_2 k_3 i_{k_2} i_{k_3}}(\mathbf{r})D_{k_l,i_{k_4}}(\mathbf{g})\mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g}) \\ & \times \Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g}) \\ & \times \Psi^{k_l}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_l};\mathbf{g})\Big], \end{aligned}$$

where

$$\begin{aligned} &\mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g}) \\ &\times\Psi^{k_l}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_l};\mathbf{g})\Big] \\ &= \begin{cases} 1, & \text{if } k_1 = k_2;\ k_3 = k_l;\ p_{k_1} = p_{k_2};\ p_{k_3} = p_{k_l} \\ & ;\ \boldsymbol{\xi}_{k_1} = \boldsymbol{\xi}_{k_2};\ \boldsymbol{\xi}_{k_3} = \boldsymbol{\xi}_{k_l}, \\ & \text{if } k_1 = k_3;\ k_2 = k_l;\ p_{k_1} = p_{k_3};\ p_{k_2} = p_{k_l} \\ & ;\ \boldsymbol{\xi}_{k_1} = \boldsymbol{\xi}_{k_3};\ \boldsymbol{\xi}_{k_2} = \boldsymbol{\xi}_{k_l}, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{26}$$

and

$$\begin{aligned} T_{l,3} = & \sum_{k_1=1}^{N-1}\sum_{k_2=k_1+1}^{N}\sum_{k_3=1}^{N-1}\sum_{k_4=k_3+1}^{N}\sum_{i_{k_1}=2}^{n_{k_1}}\sum_{i_{k_2}=2}^{n_{k_2}}\sum_{i_{k_3}=2}^{n_{k_3}}\sum_{i_{k_4}=2}^{n_{k_4}}\sum_{i_{k_5}=2}^{n_{k_5}} \\ & \times C_{k_1 k_2 i_{k_1} i_{k_2}}(\mathbf{r})C_{k_3 k_4 i_{k_3} i_{k_4}}(\mathbf{r})D_{k_l,i_{k_5}}(\mathbf{g}) \\ & \times \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g}) \\ & \times \Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Psi^{k_4}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_4};\mathbf{g}) \\ & \times \Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big], \end{aligned}$$

where

$$\begin{aligned} &\mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g}) \\ &\times\Psi^{k_4}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_4};\mathbf{g})\Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big] \\ &= \begin{cases} \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_4}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_4};\mathbf{g})\Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big], \\ \text{if } k_1 = k_3;\ k_2 = k_4 = k_l;\ p_{k_1} = p_{k_3};\ \boldsymbol{\xi}_{k_1} = \boldsymbol{\xi}_{k_3}, \\ \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_2}_{i_{k_2},p_{k_2},\xi_{k_2}}(Z_{k_2};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big], \\ \text{if } k_1 = k_4;\ k_2 = k_3 = k_l;\ p_{k_1} = p_{k_4};\ \boldsymbol{\xi}_{k_1} = \boldsymbol{\xi}_{k_4}, \\ \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_3}_{i_{k_3},p_{k_3},\xi_{k_3}}(Z_{k_3};\mathbf{g})\Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big], \\ \text{if } k_2 = k_4;\ k_1 = k_3 = k_l;\ p_{k_2} = p_{k_4};\ \boldsymbol{\xi}_{k_2} = \boldsymbol{\xi}_{k_4} \\ \mathbb{E}_{\mathbf{g}}\Big[\Psi^{k_1}_{i_{k_1},p_{k_1},\xi_{k_1}}(Z_{k_1};\mathbf{g})\Psi^{k_4}_{i_{k_4},p_{k_4},\xi_{k_4}}(Z_{k_4};\mathbf{g})\Psi^{k_l}_{i_{k_5},p_{k_5},\xi_{k_5}}(Z_{k_l};\mathbf{g})\Big], \\ \text{if } k_2 = k_3;\ k_1 = k_4 = k_l;\ p_{k_2} = p_{k_3};\ \boldsymbol{\xi}_{k_2} = \boldsymbol{\xi}_{k_3} \\ 0, \quad \text{otherwise}. \end{cases} \end{aligned} \tag{27}$$

The last J additive terms $T_{l,j}$ in (24) are determined by at most degree (S) of interaction among input variables. For example, selecting $S = 1$ and 2 produces the univariate second-moment sensitivity approximation

$$\frac{\partial \mathbb{E}_{\mathbf{g}}[h^2_{1,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})]}{\partial d_l} = \frac{\partial g_l}{\partial d_l}\Bigg(2h_{\emptyset}(\mathbf{r})\sum_{i_{k_l}=2}^{n_{k_l}} C_{k_l i_{k_l}}(\mathbf{r})D_{k_l,i_{k_l}}(\mathbf{g}) + T_{l,1}\Bigg)$$

and the bivariate second-moment sensitivity approximation

$$\frac{\partial \mathbb{E}_{\mathbf{g}}[h^2_{2,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})]}{\partial d_l} = \frac{\partial g_l}{\partial d_l}\Bigg(2h_{\emptyset}(\mathbf{r})\sum_{i_{k_l}=2}^{n_{k_l}} C_{k_l i_{k_l}}(\mathbf{r})D_{k_l,i_{k_l}}(\mathbf{g}) + T_{l,1} + T_{l,2} + T_{l,3}\Bigg),$$

respectively.

Note that the sensitivities in (23) and (24) are exact as $S \to N$ and $n_{k_l} \to \infty$, provided that expansion coefficients and expectations of products of orthonormalized B-splines are exact. How-

ever, for (24), given the last J additive terms $T_{l,j}$ for $j = 1$ ($J = 1$) or $j = 1, 2, 3$ ($J = 3$), the second-moment sensitivity is exact when the response function h is univariate or at most bivariate, respectively. One may further exploit the formulations of $T_{l,j}$ for $j \geq 4$ ($J \geq 4$) if high degrees of interaction among input variables are critical to the sensitivity solution. However, in many practical problems, such a high-variate interaction effect is often negligible for sensitivity analysis to find an optimal solution.

In (25),(26), and (27), many solutions of their expectations are either *one* or *zero*, depending on the spline indices. Otherwise, the expectations of various products of three random orthonormalized B-splines can be determined numerically, using Gauss quadrature or sampling methods, such as MC, QMC, or LHS, without any generic output function evaluations, say, FEA. Therefore, the calculations are not computationally expensive. More importantly, in the design process, the expectations of products of these splines need not be recalculated since the orthonormalized B-splines of the transformed input $\mathbf{Z}$ are preserved throughout all design iterations.

5. Proposed method for design optimization

The SDD approximations described in the preceding sections are easily employed to evaluate the objective and constraint functions, along with their sensitivity analysis. Such SDD-based stochastic and design sensitivity analyses are now integrated into any gradient-based algorithm to solve RDO problems, in which there are multiple methods. This section describes two distinct methods for the integration, resulting in the direct SDD and multi-point single-step (MPSS) SDD methods.

5.1. Direct SDD

The direct SDD method is the most straightforward way to combine the SDD-based stochastic and design sensitivity analyses with a chosen gradient-based optimization algorithm. During a design optimization process, given a current design vector and the corresponding values of the objective and/or constraint functions and their sensitivities, the following design is calculated from the optimization algorithm. In such a design update, recalculations of the SDD expansion coefficients, used for new statistical moment and sensitivity analyses, are only performed through new original output function evaluations that are usually expensive to run and required at every design iteration.

Consider a change of design variables, say, from an old design $\mathbf{d}$ to a new design $\mathbf{d}'$ during the design iteration process. Thereby, the probability measure of $\mathbf{X}$ varies from $f_{\mathbf{X}}(\mathbf{x}; \mathbf{d})d\mathbf{x}$ to $f_{\mathbf{X}}(\mathbf{x}; \mathbf{d}')d\mathbf{x}$, corresponding to the old and new designs, respectively. However, from either shifting or scaling transformations, the probability measure $f_{\mathbf{Z}}(\mathbf{z}; \mathbf{g})d\mathbf{z}$ of $\mathbf{Z}$ remains unchanged as $\mathbf{g}$ is assigned as the same value *zero* or *one*, respectively, throughout all design iterations. Instead, the respective old vector $\mathbf{r}$ of an output function evolves to its new vector $\mathbf{r}'$, as described in Section 2. As a result, a new set of output data is called for whenever the design changes.

Let the input-output data sets generated independently for the old and new designs be $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)}; \mathbf{r})\}_{t=1}^{L}$ and $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)}; \mathbf{r}')\}_{t=1}^{L}$, respectively. In these two sets, note that the input data are the same, but the output data are different due to two distinct vectors $\mathbf{r}$ and $\mathbf{r}'$. Denote by $C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r})$ and $C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}')$ the expansion coefficients for the old and new designs, respectively. Then the expansion coefficients for both designs are obtained by minimizing the associated mean square error

$$\hat{e}_{S,\mathbf{p},\boldsymbol{\Xi}} := \frac{1}{L}\sum_{t=1}^{L}\left[\left[h(\mathbf{z}^{(t)};\mathbf{r}) - \left[h_{\emptyset}(\mathbf{r}) + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\leq|u|\leq S}}\sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r})\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}^{(t)};\mathbf{g})\right]\right]^2, \tag{28}$$

and

$$\hat{e}'_{S,\mathbf{p},\boldsymbol{\Xi}} := \frac{1}{L}\sum_{t=1}^{L}\left[\left[h(\mathbf{z}^{(t)};\mathbf{r}') - \left[h_{\emptyset}(\mathbf{r}') + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\leq|u|\leq S}}\sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}')\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}^{(t)};\mathbf{g})\right]\right]^2, \tag{29}$$

respectively, using the SLS method explained in Section 3. According to (28) and (29), there is no demand for any regeneration of the input data or any recalculation of the multivariate orthonormalized B-splines, but, still, new output data sets are mandated throughout all design iterations. Consequently, the direct SDD method can be expensive, depending on the cost of evaluating the objective and constraint functions and the requisite number of design iterations to attain convergence.

In addition, the direct SDD method is not limited to using the SLS method. That is, the SDD expansion coefficients can be determined using an analytical or numerical integration method, although such methods are generally more expensive than the SLS method. In this work, due to its simplicity in the optimization algorithm, the direct SDD method was mainly employed to evaluate the accuracy power of the SDD-based stochastic and sensitivity methods to find an optimal solution in solving an elementary RDO problem.

5.2. Multi-point single-step SDD

The multi-point single-step SDD method, fusing statistical moment and sensitivity analyses, and a suitable gradient-based optimization algorithm, is intended to solve RDO problems for large design space from a smaller number of stochastic analyses through single-step, SDD approximations on a series of local subregions of the design space. Therefore, this method is built on (1) the multi-point approximation and (2) the single-step procedure.

5.2.1. Multi-point approximation

For the rectangular design space

$$\mathcal{D} = \bigtimes_{l=1}^{l=M}[d_{l,L}, d_{l,U}] \subseteq \mathbb{R}^M$$

of the RDO problem described in (6), denote by $q' = 1, 2, \ldots, Q'$, $Q' \in \mathbb{N}$, an index representing the q'th subregion of $\mathcal{D}$ with the initial design vector $\mathbf{d}_0^{(q')} = (d_{1,0}^{(q')}, \ldots, d_{M,0}^{(q')})^\intercal$. Given a sizing factor $0 < \beta_l^{(q')} \leq 1$, the domain of the q'th subregion is expressed by

$$\begin{aligned} \mathcal{D}^{(q')} &= \bigtimes_{l=1}^{l=M} \left[d_{l,L}^{(q')},\ d_{l,U}^{(q')} \right] \\ &= \bigtimes_{l=1}^{l=M} \left[d_{l,0}^{(q')} - \beta_l^{(q')} \frac{(d_{l,U} - d_{l,L})}{2} \right. \\ &\quad \left. , d_{l,0}^{(q')} + \beta_l^{(q')} \frac{(d_{l,U} - d_{l,L})}{2} \right] \subseteq \mathcal{D} \subseteq \mathbb{R}^M, \\ &\quad q' = 1, \ldots, Q', \end{aligned} \tag{30}$$

where $d_{l,U}^{(q')} = d_{l,0}^{(q')} + \beta_l^{(q')}(d_{l,U} - d_{l,L})/2$ and $d_{l,L}^{(q')} = d_{l,0}^{(q')} - \beta_l^{(q')}(d_{l,U} - d_{l,L})/2$.

According to the multi-point approximation [2, 9, 26], the RDO problem in (6) is converted to a succession of local RDO problems defined for Q' subregions. For the q'th subregion, the local RDO problem requires one to

$$\begin{aligned} \min_{\mathbf{d} \in \mathcal{D}^{(q')} \subseteq \mathbb{R}^M} \quad & \tilde{c}_{0,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{d}) \quad := w_1 \frac{\mathbb{E}_{\mathbf{g}}[\tilde{h}_{0,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})]}{\mu_0^*} \\ & \qquad + w_2 \frac{\sqrt{\mathrm{var}_{\mathbf{g}}[\tilde{h}_{0,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})]}}{\sigma_0^*}, \\ \text{subject to} \quad & \tilde{c}_{a,m}^{(q')}(\mathbf{d}) \quad := \alpha_a \sqrt{\mathrm{var}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})]} \\ & \qquad - \mathbb{E}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})] \leq 0, \\ & \qquad [d_{l,0}^{(q')} - \beta_l^{(q')}(d_{l,U} - d_{l,L}) \,/\, 2, d_{l,0}^{(q')} \\ & \qquad + \beta_l^{(q')}(d_{l,U} - d_{l,L}) \,/\, 2], \\ & \qquad a = 1, \ldots, K,\ l = 1, \ldots, M, \end{aligned} \tag{31}$$

where

$$\mathbb{E}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})] := \int_{\tilde{\mathbb{A}}^N} \tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{z};\mathbf{r}) f_{\mathbf{Z}}(\mathbf{z};\mathbf{g}(\mathbf{d})) d\mathbf{z},$$

$$\begin{aligned} \mathrm{var}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})] \quad := \quad & \mathbb{E}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r}) \\ & - \mathbb{E}_{\mathbf{g}}[\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})]]^2, \end{aligned}$$

and $\tilde{c}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{d})$, $\tilde{y}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{X})$, and $\tilde{h}_{a,S,\mathbf{p},\boldsymbol{\Xi}}^{(q')}(\mathbf{Z};\mathbf{r})$, $a = 0, 1, \ldots, K$, are local S-variate SDD approximations of $c_a(\mathbf{d})$, $y_a(\mathbf{X})$, and $h_a(\mathbf{Z};\mathbf{r})$, respectively, for the q'th subregion; $w_1 \in \mathbb{R}^*$ and $w_2 = 1 - w_1$ are two non-negative, real-valued weights; and $\mu_0^* \in \mathbb{R}\backslash\{0\}$ and $\sigma_0^* \in \mathbb{R}_0^+\backslash\{0\}$ are two non-zero, real-value scaling factors. Furthermore, $d_{l,0}^{(q')} - \beta_l^{(q')}(d_{l,U} - d_{l,L})/2$ and $d_{l,0}^{(q')} + \beta_l^{(q')}(d_{l,U} - d_{l,L})/2$, also known as the move limits, are the lower and upper bounds, respectively, of the subregion $\mathcal{D}^{(q')}$. Here, the iterations with respect to q' are associated with solving q'th local RDO problems and should not be confused with q describing the iteration count for design iterations in (6).

5.2.2. Single-step procedure

The single-step method is intended to solve each local RDO problem in (31) from a single stochastic analysis by sidestepping the demand for the recalculation of new input-output data sets to determine new SDD expansion coefficients at every design iteration. However, it is predicated on two important assumptions: (1) an S-variate SDD approximation $h_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{Z};\mathbf{r})$ of $h(\mathbf{Z};\mathbf{r})$ at the initial design is adequate for evaluating objective and/or constraint functions and their sensitivities on the entire design space; and (2) the SDD expansion coefficients for a new design, derived by reusing those generated for an old design, are acceptably accurate.

Under the above two assumptions, consider the vectors $\mathbf{r}$ and $\mathbf{r}'$, determined by $\mathbf{d}$ and $\mathbf{d}'$, respectively. Assume that the SDD expansion coefficients $C_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}^u(\mathbf{r})$, $\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}$, for the old design have been calculated from the input-output data $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)};\mathbf{r})\}_{t=1}^L$ already. Then the SDD expansion coefficients $C_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}^u(\mathbf{r}')$, $\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}$, for the new design are estimated by changing the old input data (say) $\{\mathbf{z}^{(t)}\}_{t=1}^L$ to the new input data (say) $\{\mathbf{z}'^{(t)}\}_{t=1}^L$, depending on the scaling or shifting transformation, as follows.

$$\mathbf{z}'^{(t)} = \begin{cases} \mathbf{z}^{(t)} - \mathbf{r}' + \mathbf{r} & \text{shifting}, \\ \mathrm{diag}\left(\dfrac{r_1}{r'_1}, \ldots, \dfrac{r_N}{r'_N}\right)\mathbf{z}^{(t)}, & \text{scaling}. \end{cases}$$

To further explain these modifications, first, consider the shifting transformation. In this case, the tth sample of the output function is

$$\begin{aligned} h(\mathbf{z}^{(t)};\mathbf{r}') \quad &:= \quad y(\mathbf{z}^{(t)} - \mathbf{r}') = y(\mathbf{z}^{(t)} - \mathbf{r}' + \mathbf{r} - \mathbf{r}) \\ &= \quad y(\mathbf{z}'^{(t)} - \mathbf{r}) =: h(\mathbf{z}'^{(t)};\mathbf{r}), \end{aligned}$$

where $\mathbf{z}'^{(t)} := \mathbf{z}^{(t)} - \mathbf{r}' + \mathbf{r}$ is the modified tth input sample. Secondly, for the scaling transformation, the tth sample of the output function is

$$\begin{aligned} h(\mathbf{z}^{(t)};\mathbf{r}') \quad &:= \quad y\left(\mathrm{diag}\left[\frac{1}{r'_1}, \ldots, \frac{1}{r'_N}\right]\mathbf{z}^{(t)}\right) \\ &= \quad y\left(\mathrm{diag}\left[\frac{1}{r_1}, \ldots, \frac{1}{r_N}\right]\right. \\ &\qquad \left. \times \mathrm{diag}\left[\frac{r_1}{r'_1}, \ldots, \frac{r_N}{r'_N}\right]\mathbf{z}^{(t)}\right) \\ &= \quad y\left(\mathrm{diag}\left[\frac{1}{r_1}, \ldots, \frac{1}{r_N}\right]\mathbf{z}'^{(t)}\right) =: h(\mathbf{z}'^{(t)};\mathbf{r}), \end{aligned}$$

where $\mathbf{z}'^{(t)} := \mathrm{diag}[r_1/r'_1, \ldots, r_N/r'_N]\mathbf{z}^{(t)}$ is the modified tth input sample. These modifications are intended to evaluate the output function at the new design to be approximated by the output

function at the old design, that is,

$$\begin{aligned} h(\mathbf{z}^{(t)};\mathbf{r}') &= h(\mathbf{z}'^{(t)};\mathbf{r}) \\ &\approx h_{\emptyset}(\mathbf{r}) + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\le|u|\le S}} \sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}) \\ &\quad\times\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}'^{(t)};\mathbf{g}), \end{aligned} \tag{32}$$

where the second and third lines reflect the S-variate SDD approximation. Applying (32) to (29) produces the following mean square error:

$$\begin{aligned} \hat{e}''_{S,\mathbf{p},\boldsymbol{\Xi}} := \frac{1}{L}\sum_{t=1}^{L}\Bigg[& h_{\emptyset}(\mathbf{r}) + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\le|u|\le S}} \sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} C^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}) \\ & \times\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}'^{(t)};\mathbf{g}) - \Bigg(h_{\emptyset}(\mathbf{r}') + \sum_{\substack{\emptyset\neq u\subseteq\{1,\ldots,N\}\\ 1\le|u|\le S}} \sum_{\mathbf{i}_u\in\bar{I}_{u,\mathbf{n}_u}} \\ & \times C^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{r}')\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z}^{(t)};\mathbf{g})\Bigg)\Bigg]^2, \end{aligned} \tag{33}$$

the minimization of which by the SLS method yields SDD expansion coefficients for the new design. Compared with the minimization of $\hat{e}'_m$ in (29), new output data obtained from the generic output function, that is, $h(\mathbf{z}^{(l)};\mathbf{r}')$, are not demanded. Instead, the output data, involved in (33), are generated by reusing the old expansion coefficients, initially obtained, and employing their SDD approximation for estimating new output data. Subsequently, new statistical moment and design sensitivity analyses, all employing an S-variate SDD approximation at the initial design, are conducted with little extra cost throughout all design iterations. However, notwithstanding that the single-step procedure holds the potential to substantially curtail such computational effort, employing it alone easily produces inaccurate or computationally inefficient RDO solutions for large design spaces because of the two assumptions mentioned above.

5.2.3. Computational flow

When combining the multi-point approximation with the single-step procedure, the result is an accurate and efficient design process to solve the RDO problem [2, 9]. The multi-point single-step SDD method is schematically illustrated in Figure 1a. Here, $\mathbf{d}^{(q')}_*$ is the optimal design solution obtained by the single-step procedure for the q'th local RDO problem in (31). The local optimal design solution is assigned to the initial design at the next local RDO problem, which is successively repeated along the design process to obtain the final convergent solution $\mathbf{d}^*$.

As mentioned in Section 5.2.2, in the single step procedures to solve each local subregion problem, SDD approximations at the respective initial design $\mathbf{d}^{(q')}_0$, including orthonormalized B-splines with the closed boundary domain $\times_{k=1}^{N}[a_k^{(q')}, b_k^{(q')}]$, are reused to represent new output data. Therefore, the corresponding new input data should exist within the supports of the splines. For the q'th subregion problem, consider L number of arbitrary points $\mathbf{x}^{(t)} = (x_1^{(t)},\ldots,x_N^{(t)})^\intercal$ of $\mathbf{X}$, $t = 1,\ldots,L$, and an M dimensional design vector $\mathbf{d}^{(q')} \subseteq \mathcal{D}^{(q')}$, where $\mathcal{D}^{(q')} = \times_{l=1}^{M}[d_{l,L}^{(q')}, d_{l,U}^{(q')}]$, $N \ge M$. Let $x_{k_l,\max}^{(q')} = \max\{x_{k_l}^{(t)} : t = 1,\ldots,L \in \mathbb{N}\}$ be the maximum value among the total arbitrary points of X_{k_l} following $f_{X_{k_l}}(x_{k_l}; d_{l,U}^{(q')})$, and let $x_{k_l,\min}^{(q')} = \min\{x_{k_l}^{(t)} : t = 1,\ldots,L \in \mathbb{N}\}$ be the minimum value among the total arbitrary points of X_{k_l} following $f_{X_{k_l}}(x_{k_l}; d_{l,L}^{(q')})$. Then, the conditions which must be satisfied to place the input data within the supports of the splines are

$$|b_{k_l}^{(q')} - d_{l,U}^{(q')}| > |x_{k_l,\max}^{(q')} - d_{l,U}^{(q')}| \tag{34}$$

and

$$|a_{k_l}^{(q')} - d_{l,L}^{(q')}| > |x_{k_l,\min}^{(q')} - d_{l,L}^{(q')}|. \tag{35}$$

Figure 1b illustrates the above conditions in a two-dimensional case ($N = 2$ and $M = 2$). Applying (34) and (35) to (30) produces the inequality conditions of sizing factors $\beta_l^{(q')}$, $l = 1,\ldots,M$, as

$$\begin{aligned} \beta_l^{(q')} < \min\Bigg[& \frac{2\left(b_{k_l}^{(q')} - d_{l,0}^{(q')} - |x_{k_l,\max}^{(q')} - d_{l,U}^{(q')}|\right)}{d_{l,U} - d_{l,L}}, \\ & \frac{2\left(d_{l,0}^{(q')} - a_{k_l}^{(q')} - |d_{l,L}^{(q')} - x_{k_l,\min}^{(q')}|\right)}{d_{l,U} - d_{l,L}}\Bigg]. \end{aligned} \tag{36}$$

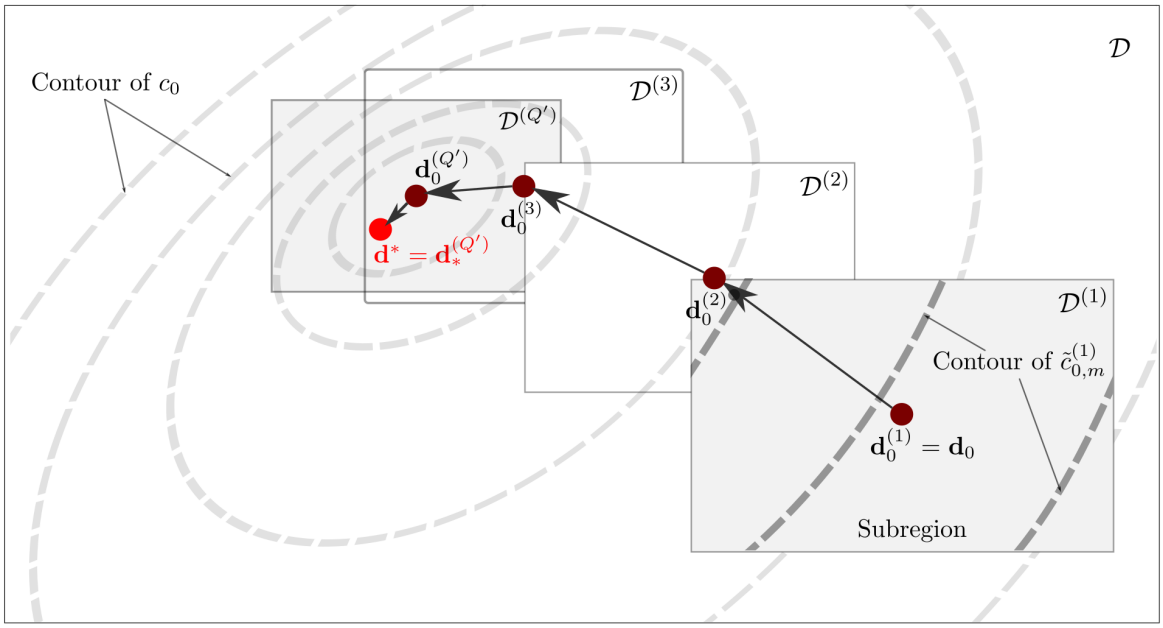
Finally, the computational flow of the multi-point single-step SDD method, including the step to satisfy (34) and (35) in sizing subregions, is presented in Figures 2 and 3 with supplementary descriptions of each step as follows.

1. Set termination criteria $0 < \epsilon_1, \epsilon_2 << 1$; set tolerances for sizing subregions $0 < \epsilon_3, \epsilon_4, \epsilon_5, \epsilon_6, \epsilon_7 < 1$; initialize size parameters $0 < \beta_k^{(q')} \le 1$, $k = 1,\ldots,M$, of $\mathcal{D}^{(q')}$; set an initial design vector $\mathbf{d}_0^{(q')} = (d_{1,0}^{(q')},\ldots,d_{M,0}^{(q')})$. The initial design can be in either feasible or infeasible domains with respect to the probabilistic constraints.

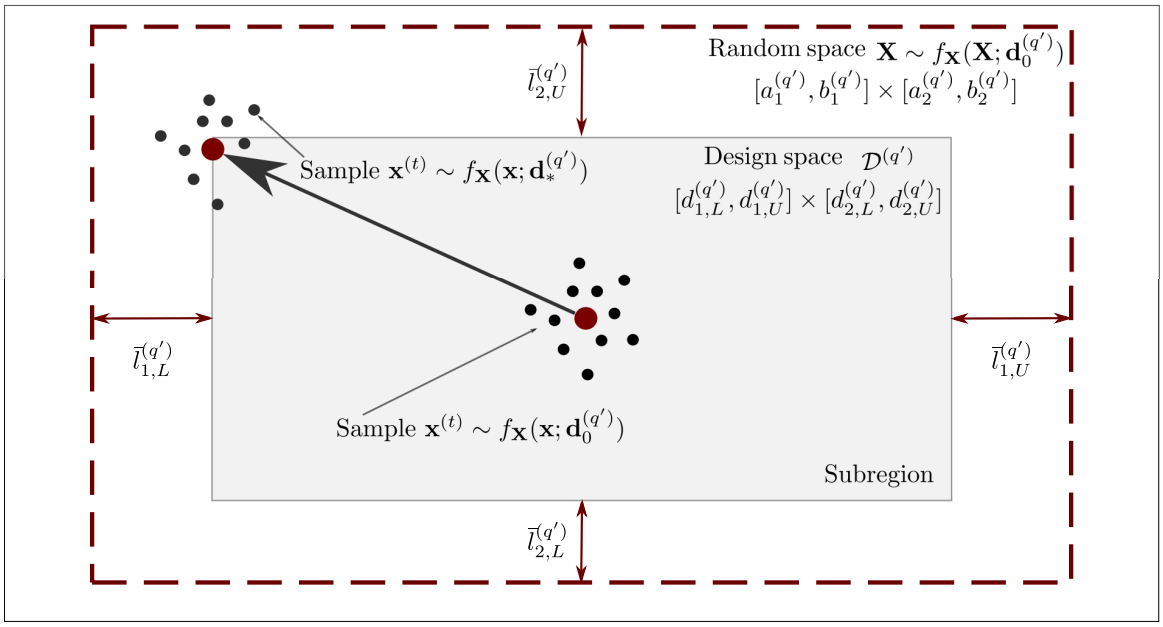
2. Transform the input random vector $\mathbf{X}$ to a new random vector $\mathbf{Z}$ such that $\mathbb{E}_{\mathbf{d}}[Z_{k_l}] = g_l = 0$ or 1, $l = 1,\ldots,M$, by the shifting or the scaling, respectively, described in Section 2.2.

3. Select the at most degree (S) of interaction among input variables for the SDD approximations of performance functions $h_a(\mathbf{z};\mathbf{r})$, $a = 1,\ldots,K$. For the coordinate direction k, $k = 1,\ldots,N$, choose p_k, and create a knot sequence $\boldsymbol{\xi}_k$, following in Section 3.1. Construct measure-consistent univariate orthonormalized B-splines $\boldsymbol{\psi}_k(z_k;\mathbf{g})$ through three steps in Section 3.2.1. Then, construct measure-consistent multivariate orthonormalized B-splines $\Psi^u_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z};\mathbf{g})$ in a dimensionwise manner in Section 3.2.2.

4. Update the current design vector $\mathbf{d}$, as follows. If $q' = 1$, create input samples $\mathbf{z}^{(t)}$, $t = 1,\ldots,L$, via the optimal LHS method, introduced in Appendix B. Then, construct an input-output data set $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)};\mathbf{r})\}_{t=1}^{L}$ of a sample size $L > L_{S,\mathbf{p},\boldsymbol{\Xi}}$,



(a)



(b)

Figure 1: A schematic description of the multi-point single-step SDD: (a) design process during Q' iterations to get the final optimum $\mathbf{d}^*$; (b) the necessity condition of the q'th subregion size relative to the intervals of spline supports at the initial design $\mathbf{d}_0^{(q')}$ for the single-step procedure; for $l = 1, 2$, $\bar{l}_{l,U}^{(q')} = |b_l^{(q')} - d_{l,U}^{(q')}| > |x_{l,\max}^{(q')} - d_{l,U}^{(q')}|$, $\bar{l}_{l,L}^{(q')} = |a_l^{(q')} - d_{l,L}^{(q')}| > |x_{l,\min}^{(q')} - d_{l,L}^{(q')}|$, where $x_{l,\max}^{(q')} = \max\{x_l^{(t)} : t = 1, \ldots, L\}$ for X_l following $f_{X_l}(x_l; d_{l,U}^{(q')})$ and $x_{l,\min}^{(q')} = \min\{x_l^{(t)} : t = 1, \ldots, L\}$ for X_l following $f_{X_l}(x_l; d_{l,L}^{(q')})$, $L \in \mathbb{N}$.

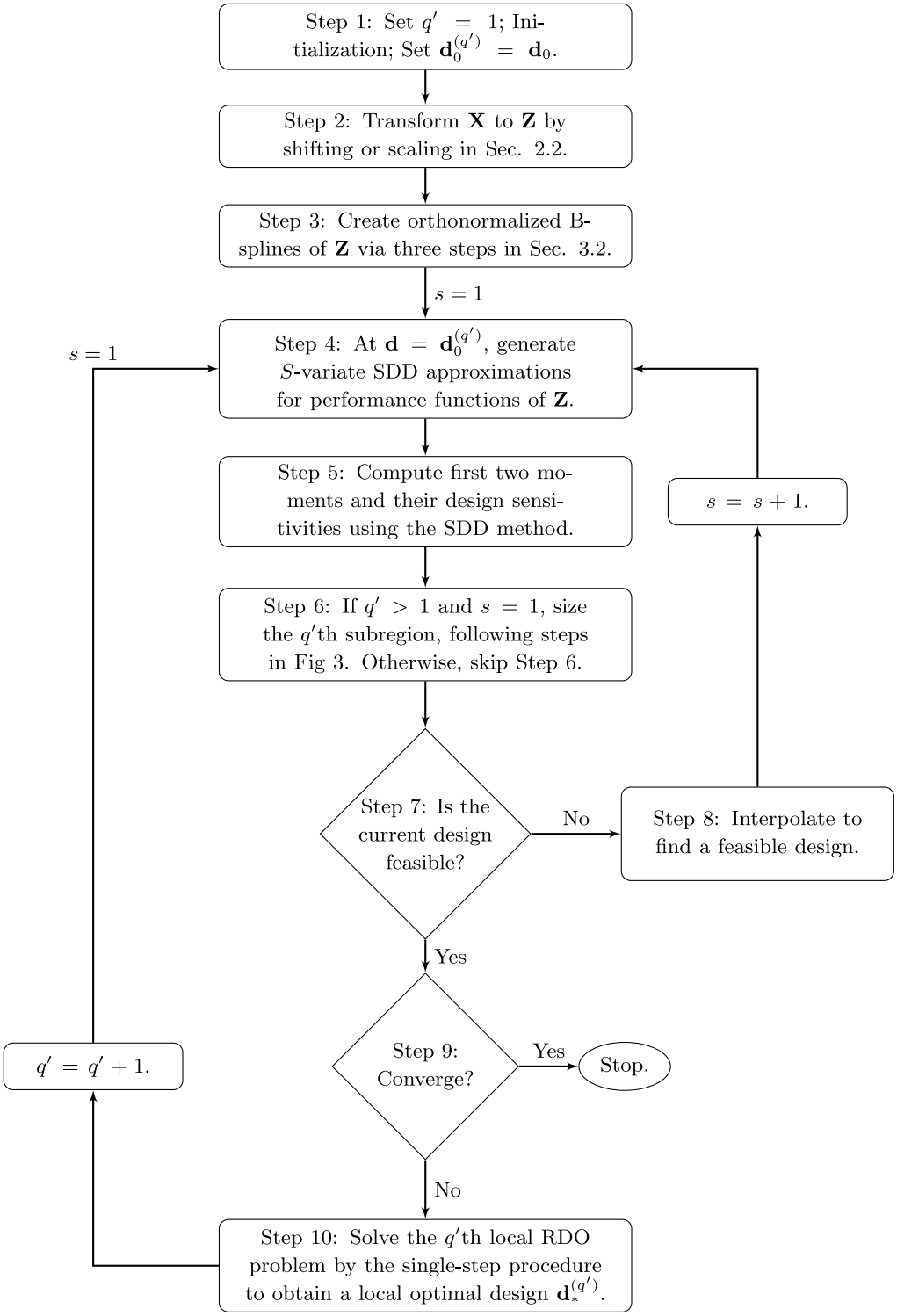


Figure 2: A flow chart of the multi-point single-step SDD method.

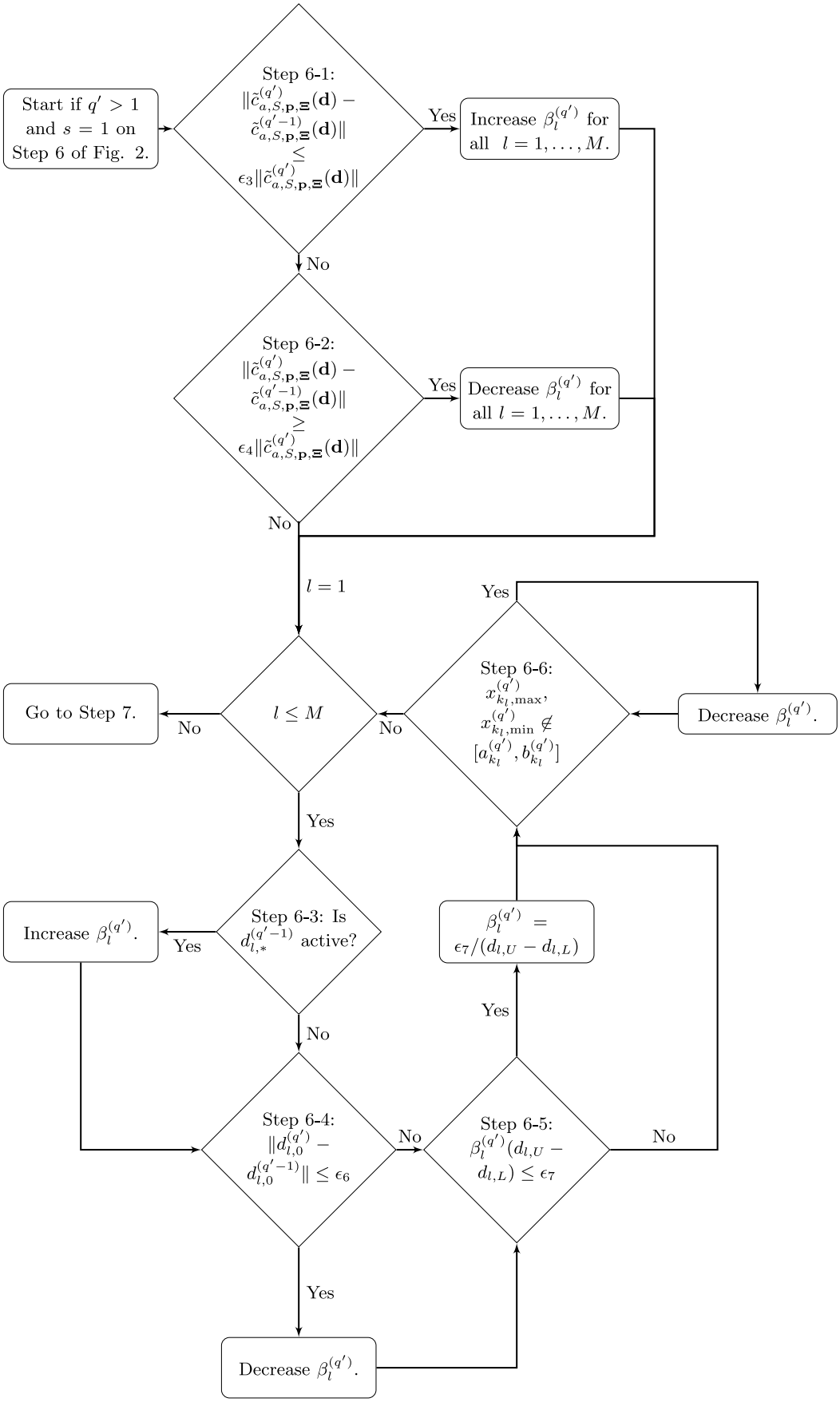


Figure 3: A flow chart of sizing the q'th sub-region in the multi-point single-step SDD method.

$L/L_{S,\mathbf{p},\boldsymbol{\Xi}} \geq 3$ (say). If $q' > 1$, reuse the input samples to generate new input-output data sets $\{\mathbf{z}^{(t)}, h(\mathbf{z}^{(t)}; \mathbf{r}')\}_{t=1}^{L}$. In every q' step, use SLS to estimate SDD expansion coefficients with respect to $\Psi^{u}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u}(\mathbf{z};\mathbf{g})$ for a performance function h.

5. In each iteration, conduct the SDD-based statistical moment and sensitivity analyses. For the design sensitivity analysis, if $q' = 1$, construct an input-output data set for score functions, $\{\mathbf{z}^{(t)}, s_l(\mathbf{z}^{(t)};\mathbf{g})\}_{t=1}^{L}$, $l = 1,\ldots,M$. Otherwise, reuse the input-output data set created in $q' = 1$. Then, obtain the objective and/or constraint function values and their gradient values at $\mathbf{d} = \mathbf{d}_0^{(q')}$.

6. If $q' = 1$ and $s = 1$, use the initial values of size parameters $0 < \beta_l^{(q')} \leq 1$, $l = 1,\ldots,M$, in Step 1. If $q' > 1$ and $s = 1$, modify the size parameters according to three criteria: (1) an estimated accuracy of SDD approximations, (2) an active/inactive condition of subregion boundaries, and (3) a converged condition of current designs. Otherwise, skip Step 6. Detail explanations of the above three conditions are presented in the following.

6-1. First condition: For any $a = 1,\ldots,K$, if $\|\tilde{c}^{(q')}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')}) - \tilde{c}^{(q'-1)}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')})\| \leq \epsilon_3 \|\tilde{c}^{(q')}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')})\|$, increase $\beta_l^{(q')}$ for all $l = 1,\ldots,M$. Otherwise, go to Step 6-2. One may need to control the enlargement rate, depending on the problems at hand. For instance, set $\beta_l^{(q')} = (2 - 1/\phi)\beta_l^{(q'-1)}$, where the golden ratio $\phi \approx 1.618$.

6-2. First condition: For any $a = 1,\ldots,K$, if $\|\tilde{c}^{(q')}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')}) - \tilde{c}^{(q'-1)}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')})\| \geq \epsilon_4 \|\tilde{c}^{(q')}_{a,S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{d}_0^{(q')})\|$, decrease $\beta_l^{(q')}$ for all $l = 1,\ldots,M$. As an instance of the decrement rate, set $\beta_l^{(q')} = \beta_l^{(q'-1)}/\phi$, where the golden ratio $\phi \approx 1.618$. Otherwise, go to Step 6-3.

6-3. Second condition: If $\|d_{l,0}^{(q')} - d_{l,L}^{(q'-1)}\| \leq \epsilon_5$ or $\|d_{l,0}^{(q')} - d_{l,U}^{(q'-1)}\| \leq \epsilon_5$, increase $\beta_l^{(q')}$, $l = 1,\ldots,M$. As an instance of the enlargement rate, set $\beta_l^{(q')} = (2 - 1/\phi)\beta_l^{(q'-1)}$, where the golden ratio $\phi \approx 1.618$. Otherwise, go to Step 6-4.

6-4. Third condition: If $\|d_{l,0}^{(q')} - d_{l,0}^{(q'-1)}\| \leq \epsilon_6$, decrease $\beta_l^{(q')}$, $l = 1,\ldots,M$. As an instance of the decrement rate, set $\beta_l^{(q')} = \beta_l^{(q'-1)}/\phi$, where the golden ratio $\phi \approx 1.618$. Otherwise, go to Step 6-5.

6-5. Move limit: If $\beta_l^{(q')}(d_{l,U} - d_{l,L}) < \epsilon_7$, set $\beta_l^{(q')} = \epsilon_7/(d_{l,U} - d_{l,L})$. Otherwise, go to Step 6-6.

6-6. If $x_{k_l,max}^{(q')}$, $x_{k_l,min}^{(q')} \notin [a_{k_l}^{(q')}, b_{k_l}^{(q')}]$ through checking the conditions (34) and (35), decrease $\beta_l^{(q')}$, $l = 1,\ldots,M$. One may use the inequalities of sizing parameters in (36). Otherwise, $l = l + 1$, and repeat the process until the loop condition $l \leq M$ is satisfied.

7. If the current design $\mathbf{d}$ is not feasible, that is, at least one constraint is violated, go to Step 8. Otherwise, set $\mathbf{d}$ to the current feasible design $\mathbf{d}_f^{(q')}$, then go to Step 9.

8. Interpolate between the current design $\mathbf{d}$ and the previous feasible design $\mathbf{d}_f^{(q'-1)}$. For instance, set $\mathbf{d} = \mathbf{d}_f^{(q'-1)}/\phi + (1 - 1/\phi)\mathbf{d}$, where the golden ratio $\phi \approx 1.618$. If an initial design at $q' = 1$ is infeasible, interpolate it with upper or lower bounds of the design space, or another initial guess, depending on the problems at hand.

9. If any of the two termination conditions, such that (1) $\|\mathbf{d}_f^{(q')} - \mathbf{d}_f^{(q'-1)}\| \leq \epsilon_1$ and/or (2) $\|\tilde{c}^{(q')}_{0,m}(\mathbf{d}_f^{(q')}) - \tilde{c}^{(q')}_{0,m}(\mathbf{d}_f^{(q'-1)})\| \leq \epsilon_2$, are met, terminate the optimization process and set the final optimal design as $\mathbf{d}^* = \mathbf{d}_f^{(q')}$. Otherwise, go to Step 10.

10. Solve the q'th local RDO problem with the single-step process using a gradient-based algorithm, such as sequential quadratic programming, to obtain a local optimal solution $\mathbf{d}_*^{(q')}$. Then, increase the subregion count as $q' = q' + 1$. Set $\mathbf{d}_0^{(q')} = \mathbf{d}_*^{(q'-1)}$ and go to Step 4.

6. Numerical examples

Four numerical examples are presented to demonstrate the efficacy of the proposed methods, as follows: the direct SDD method in Examples 1 and 2 and the multi-point single-step SDD method in Examples 3 and 4. All examples employed a single objective function or the weighted sum approach. The objective and constraint functions are associated with elementary mathematical functions or engineering applications from a simple truss to an industrial-scale robotic gripper jaw design problem. In such practical engineering problems, both sizing and shape design optimizations in the context of RDO were studied. In all examples, the design variables are the statistical means of the two through at most the twenty-six dimensional input random variables, which are mutually statistically independent.

For all examples, the coordinate degrees of the SDD approximation are identical, say, $p_1 = \ldots, p_N = p$. The selected degree of interaction S, coordinate spline degree p, or subinterval number I of the SDD methods are tabulated in Table 1. Also, all knot sequences of input $\mathbf{Z}$ or $\mathbf{X}$, used in Examples 1–4, are $(p + 1)$-open and were transformed from uniform spacing to ones following the probability distributions of input random variables. The controlling factors, chosen to generate knot sequences in Examples 1–4, are listed in Table 1. In all examples, orthonormalized B-splines were determined by the Cholesky factorization of the spline moment matrix obtained analytically.

In Examples 1–3, the proposed SDD-based solutions are compared with those obtained by the corresponding PCE-based methods whenever possible. For instance, in Examples 1 and 2, the PCE approximation was employed in the direct method, resulting in the direct PCE method for RDO solution. In Example 3, the PCE approximation was applied to the multi-point single-step method as well, then named the MPSS PCE method, for comparison with the proposed MPSS SDD method. However, while in Example 1, the PCE approximation consists of Hermite orthonormal polynomials as their basis functions, in Examples 2 and 3, the PCE bases were determined by the Cholesky factorization of the monomial moment matrix, introduced in the

literature [27]. For both the SDD and PCE methods, the respective expansion coefficients, with respect to their basis functions, were calculated analytically in Example 1 and numerically using SLS in Examples 2–4.

To show the computational efficiency of the proposed RDO methods or others, in Example 1 the requisite numbers of basis functions are provided, invoked by the fact that the computational cost increases in the proportion to the number of basis functions employed. In contrast, in Examples 2–4, the numbers of function evaluations or FEA required to obtain input-output data set for SLS are provided to ascertain the computational cost. The input samples were created by the optimal LHS, explained in Appendix C. The sample sizes L, three to five times the number of basis functions or expansion coefficients, say, $3 \times L_{S,\mathbf{p},\Xi}$ or $5 \times L_{S,\mathbf{p},\Xi}$, are listed in Table 1 for Examples 2–4. The expectations of various products of three random orthonormalized B-splines, in (25), (26), and (27), are determined analytically in Example 1 and estimated numerically employing a Gauss quadrature rule in Examples 2–4. The selected numbers of Gauss points ($\bar{L}$) in each coordinate direction, used in Examples 2–4, are reported in Table 1.

As a gradient-based optimization, the sequential quadratic programming was used to solve RDO problems in all examples. For the multi-point, single-step design SDD or PCE methods used in Examples 3 and 4, their used tolerances or parameters are $\epsilon_1 = 1 \times 10^{-3}$, $\epsilon_2 = 1 \times 10^{-3}$, $\epsilon_3 = 0.01$, $\epsilon_4 = 0.07$, $\epsilon_5 = 0.01$, $\epsilon_6 = 0.5$, $\epsilon_7 = 0.05$, and $\epsilon_8 = 1 \times 10^{-4}$. The sizing parameters $\beta_l^{(1)}$, $l = 1, \ldots, M$, are 0.3 and 0.5 in Examples 3 and 4, respectively.

All numerical results were generated using MATLAB [28], CREO parametric [29], and ABAQUS [30] on an Intel Core i7-7700K 4.20 GHz processor with 64 GB of RAM.

6.1. Example 1: Optimization of mathematical functions

Consider a mathematical problem involving a bivariate, independent Gaussian random vector $\mathbf{X} = (X_1, X_2)^\intercal$ which has means $\mathbb{E}_{\mathbf{d}}[X_1] = d_1$ and $\mathbb{E}_{\mathbf{d}}[X_2] = d_2$. Given the design vector $\mathbf{d} = (d_1, d_2)^\intercal$, the problem requires one to

$$\begin{aligned} \min_{\mathbf{d} \in \mathcal{D}} \quad & c_0(\mathbf{d}) \quad := \frac{\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}}{\sqrt{\mathrm{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}, \\ \text{subject to} \quad & c_1(\mathbf{d}) \quad := 3\sqrt{\mathrm{var}_{\mathbf{d}}[y_1(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})] \le 0. \end{aligned} \tag{37}$$

In (37), the initial design vector $\mathbf{d}_0 = (5, 5)^\intercal$, and y_0 and y_1 are two random functions of $\mathbf{X}$. The approximate optimal solution is denoted by $\tilde{\mathbf{d}}^* = (\tilde{d}_1^*, \tilde{d}_2^*)^\intercal$. The performance functions y_0 and y_1, used in $c_0(\mathbf{d})$ and $c_1(\mathbf{d})$, respectively, are considered in two distinct cases: smooth functions (Case 1) and non-smooth functions (Case 2), as follows:

Case 1: The smooth functions y_0 and y_1 are defined as

$$y_0(\mathbf{X}) = (X_1 - 4)^3 + (X_1 - 3)^4 + (X_2 - 5)^2 + 10$$

and

$$y_1(\mathbf{X}) = X_1 + X_2 - 6.45.$$

The standard deviations σ_1 and σ_2 of X_1 and X_2 are the same as 0.4. The bounds of the two design variables are given as $1 \le d_1, d_2 \le 10$.

Case 2: The non-smooth functions y_0 and y_1 are defined as

$$y_0(\mathbf{X}) = g_1(X_1) + g_1(X_2) + \frac{1}{50} g_1(X_1) g_1(X_2),$$

where, for $k = 1, 2$,

$$g_1(x_k) = \begin{cases} 10\exp(3x_k - 18), & x_k < 6, \\ 10\exp(-3x_k + 18), & x_k \ge 6, \end{cases}$$

and

$$y_1(\mathbf{X}) = 8g_2(X_1) + 10g_2(X_2) + \frac{1}{10} g_2(X_1) g_2(X_2) - 165,$$

where, for $k = 1, 2$,

$$g_2(x_k) = \begin{cases} 3x_k, & x_k < 6, \\ -3x_k + 36, & x_k \ge 6. \end{cases}$$

The standard deviations σ_1 and σ_2 of X_1 and X_2 are the same as 0.8. The bounds of the two design variables are given as $1 \le d_1, d_2 \le 5$.

Since SDD basis functions are defined on a finite domain, it is assumed that the PDF of X_l is truncated Gaussian with an adequately large support. Subsequently, the shifting transformation leads to Z_l, which is also truncated Gaussian. Consequently, the PDFs of X_l and Z_l used in this example are as follows:

$$f_{X_l}(x_l; d_l) = \frac{\phi\left(\dfrac{x_l - d_l}{\sigma_l}\right)}{\Phi\left(\dfrac{D_l}{\sigma_l}\right) - \Phi\left(-\dfrac{D_l}{\sigma_l}\right)}, \; l = 1, 2,$$

when $d_l - D_l \le x_l \le d_l + D_l$, and *zero* otherwise, and

$$f_{Z_l}(z_l; g_l) = \frac{\phi\left(\dfrac{z_l - g_l}{\bar{\sigma}_l}\right)}{\Phi\left(\dfrac{\bar{D}_l}{\bar{\sigma}_l}\right) - \Phi\left(-\dfrac{\bar{D}_l}{\bar{\sigma}_l}\right)}, \; l = 1, 2,$$

when $g_l - \bar{D}_l \le z_l \le g_l + \bar{D}_l$, and *zero* otherwise. Here, $\Phi(\cdot)$ and $\phi(\cdot)$ are the cumulative distribution and probability density functions, respectively, of a standard Gaussian random variable; for $l = 1, 2$, $1 \le d_l \le 10$ or $1 \le d_l \le 5$ in Case 1 or 2, respectively, $g_l = 0$, $\sigma_l = \bar{\sigma}_l = 0.4$ or 0.8 in Case 1 or 2, respectively; and for $l = 1, 2$, $D_l = \bar{D}_l = 6\sigma_l$.

Figures 4a–b illustrate that for Case 1, the functions y_0 and y_1 have smooth, continuous derivatives on the region $[0, 10]^2$. In contrast, Figures 4c–d indicate that for Case 2, the functions y_0 and y_1 have a peak point at $(6, 6)$, and then they fall off along all directions exponentially and linearly, respectively, both forming sharp corners along $x_1 = 6$ and $x_2 = 6$. As a result, the functions y_0, y_1 in Case 2 have discontinuous partial derivatives while

Table 1: The list of parameters (Examples 1–4): SDD truncation parameters (S, p, I), sample sizes (L), and quadrature point sizes ($\bar{L}$).

Methods	S[a]		p[b]		I[c]		t[d]	L[e]		$\bar{L}$[f]
	y_0	y_1	y_0	y_1	y_0	y_1	–	y_0,y_1		–
					Example 1 (Case 1)					
Direct SDD	1	1	1	1	4	4	2	–		–
	1	1	1	1	6	6	2	–		–
	1	1	2	2	2	2	2	–		–
	1	1	2	2	4	4	2	–		–
					Example 1 (Case 2)					
Direct SDD	1	1	1	1	2[g]	2[g]	1	–		–
	1	1	1	1	4[g]	4[g]	1	–		–
	1	1	2	2	2[g]	2[g]	1	–		–
	1	1	2	2	4[g]	4[g]	1	–		–
	2	2	2	2	2[g]	2[g]	1	–		–
	y_0	y_1–y_3	y_0	y_1–y_3	y_0	y_1–y_3	–	y_0	y_1–y_3	–
					Example 2					
Direct SDD	1	1	1	1	4	4	1.7[h]/1[i]	65	105	32
	1	1	2	2	3	3	1.7[h]/1[i]	65	105	34
	1	1	2	2	4	4	1.7[h]/1[i]	80	130	34
	2	2	2	2	2	2	1.7[h]/1[i]	185	530	34
	y_0	y_1–y_{11}	y_0	y_1–y_{11}	y_0	y_1–y_{11}	–	y_0–y_{11}		–
					Example 3 (Case 1)					
Multi-point single-step SDD	1	1	1	1	2	2	1.5	105		32
	1	1	1	1	4	4	1.5	205		32
	1	1	2	2	2	2	1.5	155		34
	1	1	2	2	4	4	1.5	255		34
	y_0	y_1–y_3	y_0	y_1–y_3	y_0	y_1–y_3	–	y_0–y_3		–
					Example 3 (Case2)					
Multi-point single-step SDD	1	1	1	1	4	4	1	205		32
	1	1	1	1	6	6	1	305		32
	1	1	2	2	2	2	1	155		34
	1	1	2	2	4	4	1	255		34
	y_0	y_1	y_0	y_1	y_0	y_1	–	y_0,y_1		–
					Example 4					
Multi-point single-step SDD	1	1	1	1	2	2	1	159		32

a. The degree of interaction of the SDD approximation for an output function or a score function
b. The spline degree or order of the SDD approximation for an output function or a score function
c. The number of subintervals of the SDD approximation for an output function or a score function
d. A controlling factor, used for the transformation of a knot vector, introduced in Section 3.1.
e. The sample size of the input-output data set used for estimating expansion coefficients of an output function or a score function
f. The number of quadrature points used for calculating the expectation of multiple products of three orthonormal polynomials on a subinterval ($[\xi_{k,i_k}, \xi_{k,i_k+1}]$) of a coordinate ($k$)
g. Repeated knots (multiplicity of two) at $(6,6)^\intercal$ were employed during the design iterations. At every iteration, if any knots are not initially placed at $(6,6)^\intercal$, additional repeated knots (multiplicity of two) are inserted at $(6,6)^\intercal$
h. A controlling factor, used for the transformation of a knot vector, for Z_1–Z_3
i. A controlling factor, used for the transformation of a knot vector, for X_5

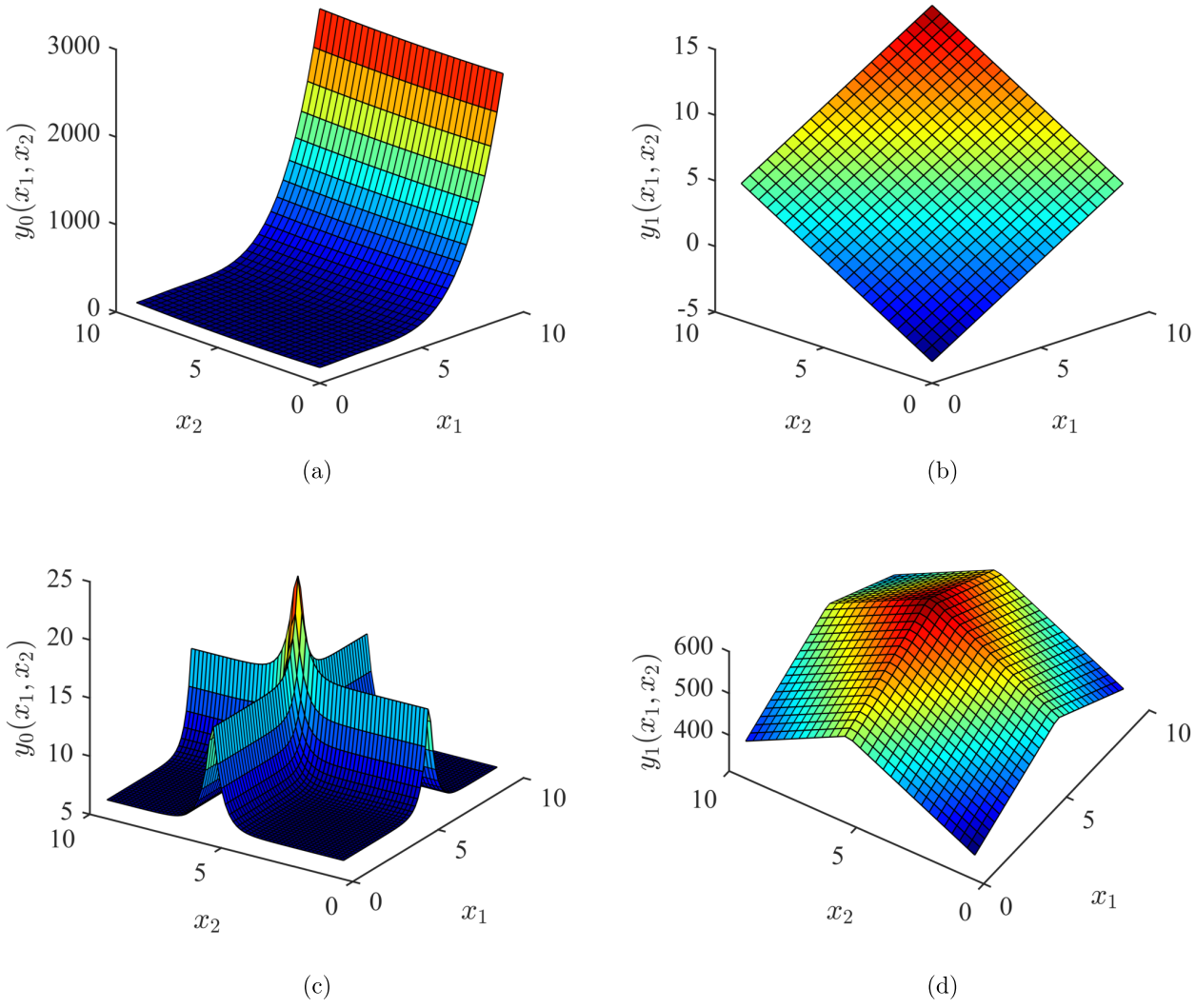


Figure 4: Graphs of functions (Example 1): (a) smooth function y_0 (Case 1); (b) smooth function y_1 (Case 1); (c) nonsmooth function y_0 (Case 2); (d) nonsmooth function y_1 (Case 2).

they are all continuous. Such functions are generally difficult to approximate by polynomials.

Table 2 presents the means and variances of $y_0(\mathbf{X})$ and $y_1(\mathbf{X})$ in Cases 1 and 2, including their first-order design sensitivities, obtained by the SDD and PCE methods, at the initial design $\mathbf{d}_0 = (5, 5)^\intercal$. For comparison between the two methods, the respective exact solutions, existing for smooth or nonsmooth functions y_0 and y_1 in the two cases, are reported in the eighth (Case 1) or ninth (Case 2) column from the left in Table 2. In Case 1, for the fourth-order function y_0, the univariate, linear ($S = 1, p = 1$) and univariate, quadratic ($S = 1, p = 2$) SDD approximations, presented in the second through fifth columns, yield practically identical estimates of response moments and their design sensitivities with exact ones, while the second-order ($m = 2$) PCE approximation has relatively less accuracy than the SDD methods or the fourth-order ($m = 4$) PCE approximation. However, the estimates of the means and the first-order sensitivities by all aforementioned methods are accurate, as expected for smooth functions. Also, in Case 1 of Table 2, for the linear function y_1, both the SDD and PCE methods with all cases of parameters (p, m, I) reproduce response moments and their sensitivities faithfully, since such linear function can be exactly represented by at least first-order basis functions.

In contrast, Case 2 of Table 2 shows that the SDD estimates converge more rapidly to the exact ones than the PCE estimates when their parameter values, p, I, or m, increase. Between the SDD methods, the bivariate ($S = 2$) SDD approximation yields the most close-to-exact solutions for both nonsmooth functions y_0 and y_1. However, even univariate, linear ($S = 1, p = 1$) SDD approximations, presented in the second and third columns, achieve more accurate results of the second moment or its sensitivities than the seventh-order ($m = 7$) PCE approximation. In particular, when comparing the first-order SDD method and the seventh-order PCE method, presented in the second and eighth columns, respectively, their estimates are very close to each other, while the SDD's number of basis functions (7) is much less than PCE's (36). Thus, the results clearly demonstrate the more outstanding performance of the SDD method in terms of both accuracy and computational efficiency when dealing with such nonsmooth functions.

Table 3 summarizes the approximate optimal solutions for Cases 1 and 2, including the requisite numbers of design iterations, by the direct SDD method. For comparison, the approximate solutions by the direct PCE method are included in sixth to seventh (Case 1) or seventh to eighth (Case 2) columns from the left in Table 3. Also, the exact solution, obtained from the exact expressions of objective and constraint functions and their design sensitivities, are tabulated in its eighth (Case 1) or ninth (Case 2) column. For Case 1 in Table 3, all SDD- or PCE-based RDO solutions and the respective exact one are practically identical, all indicating that the constraint is inactive ($c_1 < 0$), although the SDD solutions, presented in the second, third, and fifth columns of Table 3, are closer to the exact one than those by the second-order ($m = 2$) direct PCE method. For each order $p = 1$ and 2 of the direct SDD method, the respective optimal solution converges to the exact one as its subinterval number increases from $I = 4$ to $I = 6$ or from $I = 2$ to $I = 4$, respectively. However, since the fourth-order ($m = 4$) direct PCE method achieves the exact RDO solution, the merit of the direct SDD method over the direct PCE method is not fully realized in Case 1. In contrast, in Case 2, even the first- or second-order ($p = 1, 2$) direct SDD methods yield more accurate optimal solutions than the fifth-order ($m = 5$) direct PCE method. Furthermore, the proposed RDO solutions by the bivariate, quadratic ($S = 1,\ p = 2$) SDD approximation with two subintervals ($I = 2$) or the univariate, quadratic ($S = 1,\ p = 2$) SDD approximation with four subintervals ($I = 4$) are very close to the exact one. The direct PCE method, on the other hand, demands much higher order, say, at least the seventh-order ($m = 7$) approximation, to get the similar level of the accurate RDO solution. Indeed, the number (36) of basis functions in the seventh-order ($m = 7$) PCE is more than those (15 or 26) in the univariate or bivariate, quadratic ($S = 1,\ p = 2,\ I = 4$) or ($S = 2,\ p = 2,\ I = 2$) SDD, respectively, thus demanding more computational cost than the SDD method when considering the 8–9 iterations to obtain the final optimal solution. Thus, the optimization results in Case 2 also show the superiority of the proposed direct SDD method over the PCE counterpart for such nonsmooth functions.

6.2. Example 2: Bi-objective optimization of an eccentric loaded column

The second example was created by recasting an eccentrically loaded column problem, previously investigated by [31] and [32]. This example intends to prove the technical merit of the proposed RDO method when confronted with highly nonlinear performance functions in solving a bi-objective optimization problem. As shown in Fig. 5, there are five input random variables $\mathbf{X} = (X_1, X_2, X_3, X_4, X_5)^\intercal$, where X_1 and X_2 are means of the average radius (average of outer and inner radii) and the thickness, respectively, of a hollow circular tube of length X_3. The tubular column is subject to a random load X_5 with eccentricity $0.01X_4X_1$ about its center axis. It is made of cast iron, which has Young's modulus $E = 210$ GPa. The design variables are $d_1 = \mathbb{E}_{\mathbf{d}}[X_1]$, $d_2 = \mathbb{E}_{\mathbf{d}}[X_2]$, and $d_3 = \mathbb{E}_{\mathbf{d}}[X_3]$. The probability distributions of all input random variables are listed in Table 4.

The objective of this RDO problem is to minimize the second-moment properties of the surface area of the hollow circular tube, subject to three constraints described in (39), (40), and (41), limiting the normal stress, buckling load, and deflection at or below the respective thresholds of the allowable stress $\sigma_a = 2.5 \times 10^8$ Pa and lateral deflection $\Delta = 0.25$ m. The deterministic constraint c_4 in (38) restricts the ratio of d_1/d_2, not to exceed the value of 50 for practical considerations. Mathematically, such

Table 2: The results of the first two moments and design sensitivities of y_0 and y_1 at $\mathbf{d}_0 = (5,5)^\intercal$ in Cases 1 and 2 (Example 1)

	SDD approximations				PCE approximations		Exact[a]
	Case 1 (Smooth functions y_0 and y_1)						
	$S=1,\ p=1$		$S=1,\ p=2$				
	$I=4$	$I=6$	$I=2$	$I=4$	$m=2$[b]	$m=4$[b]	
	(1) $y_0(X_1,X_2) = (X_1-4)^3 + (X_1-3)^4 + (X_2-5)^2 + 10$						
$\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]$	31.5568	31.5568	31.5568	31.5568	31.5568	31.5568	31.5568
$\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]$[c]	288.6518	289.2490	289.3101	289.4472	287.4474	289.4538	289.4538
$\partial\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]/\partial d_1$	39.3200	39.3200	39.3200	39.3200	39.3200	39.3200	39.3200
$\partial\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]/\partial d_2$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\partial\mathbb{E}_{\mathbf{d}}[y_0^2(\mathbf{X})]/\partial d_1$	3244.1178	3258.8149	3257.5744	3263.9702	3185.2346	3264.3078	3264.3078
$\partial\mathbb{E}_{\mathbf{d}}[y_0^2(\mathbf{X})]/\partial d_2$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
No. of basis functions	9[d]	13[d]	7[d]	11[d]	6	15	–
	(2) $y_1(X_1,X_2) = X_1 + X_2 - 6.45$						
$\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]$	3.5500	3.5500	3.5500	3.5500	3.5500	3.5500	3.5500
$\mathrm{var}_{\mathbf{d}}[y_1(\mathbf{X})]$[c]	0.3200	0.3200	0.3200	0.3200	0.3200	0.3200	0.3200
$\partial\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]/\partial d_1$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$\partial\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]/\partial d_2$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$\partial\mathbb{E}_{\mathbf{d}}[y_1^2(\mathbf{X})]/\partial d_1$	7.1000	7.1000	7.1000	7.1000	7.1000	7.1000	7.1000
$\partial\mathbb{E}_{\mathbf{d}}[y_1^2(\mathbf{X})]/\partial d_2$	7.1000	7.1000	7.1000	7.1000	7.1000	7.1000	7.1000
No. of basis functions	9[d]	13[d]	7[d]	11[d]	6	15	–
	Case 2 (Nonsmooth functions y_0 and y_1)						
	$S=1,\ p=1$		$S=1,\ p=2$		$S=2,\ p=2$		
	$I=2$	$I=4$	$I=2$	$I=4$	$I=2$	$m=5$[b]	$m=7$[b]	
	(1) $y_0(X_1,X_2) = g_1(X_1)^{(e)} + g_1(X_2)^{(e)} + \frac{1}{50} g_1(X_1)^{(e)} g_1(X_2)^{(e)}$							
$\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]$	3.2067	3.2067	3.2067	3.2067	3.2067	3.2067	3.2067	3.2067
$\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]$[c]	10.0067	10.6451	11.0350	11.0384	11.0458	7.7798	8.6212	11.2044
$\partial\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]/\partial d_1$	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810
$\partial\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]/\partial d_2$	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810	1.9810
$\partial\mathbb{E}_{\mathbf{d}}[y_0^2(\mathbf{X})]/\partial d_1$	16.7739	18.2764	18.0123	17.9950	18.4335	15.0469	16.4851	18.5946
$\partial\mathbb{E}_{\mathbf{d}}[y_0^2(\mathbf{X})]/\partial d_2$	16.7739	18.2764	18.0123	17.9950	18.4335	15.0469	16.4851	18.5946
No. of basis functions	7[f]	11[f]	11[g]	15[g]	26[g]	21	36	–
	(2) $y_1(X_1,X_2) = 8g_2(X_1)^{(h)} + 10g_2(X_2)^{(h)} + \frac{1}{10} g_2(X_1)^{(h)} g_2(X_2)^{(h)} - 165$							
$\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]$	122.4067	122.4067	122.4067	122.4067	122.4067	122.4067	122.4067	122.4067
$\mathrm{var}_{\mathbf{d}}[y_1(\mathbf{X})]$[c]	939.9974	939.9974	939.9974	939.9974	940.1775	930.0175	933.8534	940.1776
$\partial\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]/\partial d_1$	22.4205	22.4205	22.4205	22.4205	22.4205	22.4205	22.4205	22.4205
$\partial\mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})]/\partial d_2$	27.1527	27.1527	27.1527	27.1527	27.1527	27.1527	27.1527	27.1527
$\partial\mathbb{E}_{\mathbf{d}}[y_1^2(\mathbf{X})]/\partial d_1$	5250.5125	5250.5125	5250.5126	5250.5126	5273.4479	5259.6837	5267.9116	5273.4479
$\partial\mathbb{E}_{\mathbf{d}}[y_1^2(\mathbf{X})]/\partial d_2$	6297.7954	6297.7954	6297.7954	6297.7954	6316.7139	6296.7600	6308.6966	6316.7139
No. of basis functions	7[f]	11[f]	11[g]	15[g]	26[g]	21	36	–

a. The exact closed-form solutions of sensitivities are used
b. The truncation order (m) is total degree
c. $\mathrm{var}_{\mathbf{d}}[y_a(\mathbf{X})] := \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X}) - \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})]]^2,\ a = 0, 1$
d. Simple knots were used to determine the basis functions
e. $g_1(x_k) = \begin{cases} 10\exp(3x_k - 18), & x_k < 6, \\ 10\exp(-3x_k + 18), & x_k \geq 6, \end{cases} k = 1, 2$
f. Simple knot (multiplicity of one) at $(6,6)^\intercal$ was employed
g. Repeated knots (multiplicity of two) at $(6,6)^\intercal$ were employed
h. $g_2(x_k) = \begin{cases} 3x_k, & x_k < 6, \\ -3x_k + 36, & x_k \geq 6, \end{cases} k = 1, 2$

Table 3: Optimization results of mathematical formulations in Cases 1 and 2 (Example 1)

	Direct SDD				Direct PCE		Exact [a]
	Case 1 (Smooth functions y_0 and y_1)						
	$S=1, p=1$		$S=1, p=2$				
	$I=4$	$I=6$	$I=2$	$I=4$	$m^{(b)}=2$	$m^{(b)}=4$	
$\tilde{d}_1^*$	3.3742	3.3637	3.3844	3.3595	3.3844	3.3577	3.3577
$\tilde{d}_2^*$	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000
$c_0(\tilde{\mathbf{d}}^*)$	0.0661	0.0665	0.0662	0.0666	0.0624	0.0666	0.0666
$c_1(\tilde{\mathbf{d}}^*)$	−0.2271	−0.2166	−0.2374	−0.2125	−0.2374	−0.2107	−0.2107
$\sqrt{\text{var}_{\mathbf{x}}[y_0(\mathbf{X})]}$	1.1234	1.1306	1.1252	1.1332	1.0574	1.1338	1.1338
No. of iterations	7	7	7	7	7	7	7
No. of basis functions	9	13	7	11	6	15	–
	Case 2 (Nonsmooth functions y_0 and y_1)						
	$S=1,\ p=1$		$S=1,\ p=2$		$S=2,\ p=2$			
	$I=2$	$I=4$	$I=2$	$I=4$	$I=2$	$m^{(b)}=5$	$m^{(b)}=7$	
$\tilde{d}_1^*$	4.4811	4.4516	4.4143	4.3678	4.3883	4.0086	4.3440	4.3022
$\tilde{d}_2^*$	4.6667	4.6889	4.7168	4.7512	4.7364	5.0000	4.7658	4.7993
$c_0(\tilde{\mathbf{d}}^*)$	0.6913	0.7323	0.7270	0.7375	0.7264	0.7528	0.7591	0.7369
$c_1(\tilde{\mathbf{d}}^*)$	2×10^{-7}	-5×10^{-6}	-1×10^{-4}	3×10^{-6}	-1×10^{-4}	-3×10^{-10}	-2×10^{-7}	-4×10^{-8}
$\sqrt{\text{var}_{\mathbf{x}}[y_0(\mathbf{X})]}$	2.1869	2.3892	2.4150	2.4569	2.4141	2.0996	2.2288	2.4666
No. of iterations	8	8	8	8	7	9	9	15
No. of basis functions	7	11	11	15	26	21	36	–

a. The closed-form expressions of objective, constraint, and their gradient functions were used
b. The truncation order (m) is total degree

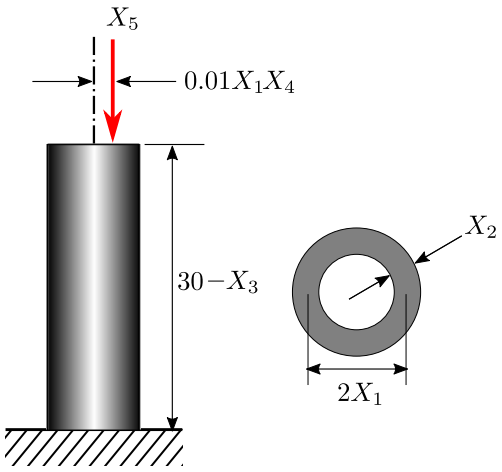


Figure 5: Configuration of the vertical column with an eccentric load (Example 2).

Table 4: Statistical properties of random variables of the column with an eccentric load (Example 2)

Random variables	Mean	Standard deviation	Probability distribution
X_1, m	d_1	$0.05d_1$	Lognormal
X_2, m	d_2	$0.05d_2$	Lognormal
X_3, m	d_3	$0.15d_3$	Lognormal
X_4	2	0.5	Weibull
X_5, kN	50	5	Normal

an RDO problem is devised to

$$\begin{aligned}
\min_{\mathbf{d}\in\mathcal{D}\subseteq\mathbb{R}^M} \quad & c_0(\mathbf{d}) \quad := \{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})],\ \sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}\},\\
\text{subject to} \quad & c_1(\mathbf{d}) \quad := 3\sqrt{\mathrm{var}_{\mathbf{d}}[y_1(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})] \le 0,\\
& c_2(\mathbf{d}) \quad := 3\sqrt{\mathrm{var}_{\mathbf{d}}[y_2(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_2(\mathbf{X})] \le 0,\\
& c_3(\mathbf{d}) \quad := 3\sqrt{\mathrm{var}_{\mathbf{d}}[y_3(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_3(\mathbf{X})] \le 0,\\
& c_4(\mathbf{d}) \quad := -1 + \frac{d_1}{50d_2} \le 0,\\
& \qquad\quad 0.1\ \mathrm{m} \le d_1 \le 2.0\ \mathrm{m},\\
& \qquad\quad 0.005\ \mathrm{m} \le d_2 \le 0.2\ \mathrm{m},\\
& \qquad\quad 4\ \mathrm{m} \le d_3 \le 20\ \mathrm{m},
\end{aligned} \tag{38}$$

where

$$y_0(\mathbf{X}) = 4\pi X_1(30 - X_3 + X_2),$$

is the random surface area of the column, and

$$\begin{aligned}
y_1(\mathbf{X}) \quad = \quad & 1 - \frac{X_5}{2\pi X_1 X_2 \sigma_a}\left[1 + \frac{2\times 0.01 X_4(X_1 + 0.5X_2)}{X_1}\right.\\
& \left.\times \sec\left(\frac{\sqrt{2}(30 - X_3)}{X_1}\sqrt{\frac{X_5}{E(2\pi X_1 X_2)}}\right)\right],
\end{aligned} \tag{39}$$

$$y_2(\mathbf{X}) = 1 - \frac{4(30 - X_3)^2 X_5}{\pi^3 E X_1^3 X_2}, \tag{40}$$

$$y_3(\mathbf{X}) = 1 - \frac{0.01 X_4 X_1}{\Delta}\left[\sec\left((30 - X_3)\sqrt{\frac{X_5}{E\pi X_1^3 X_2}}\right) - 1\right] \tag{41}$$

are three random performance functions of the normal stress, buckling load, and deflection, respectively. The initial design is $\mathbf{d}_0 = (1, 0.2, 5)^\intercal$ m. The approximate optimal solution is denoted by $\tilde{\mathbf{d}}^* = (\tilde{d}_1^*, \tilde{d}_2^*, \tilde{d}_3^*)^\intercal$.

Since SDD basis functions are defined on a finite domain, it is assumed that, for l = 1–3, the PDF of X_l is truncated lognormal with an adequately large support. Subsequently, the scaling transformation leads to Z_l, which is also truncated lognormal. Consequently, the PDFs of X_l and Z_l used in this example are as follows:

$$f_{X_l}(x_l; d_l) = \frac{\exp\left[-\frac{1}{2\tilde{\sigma}_l^2}(\ln x_l - \tilde{\mu}_l)^2\right]}{\sqrt{2\pi}\tilde{\sigma}_l\left[\Phi\left(\frac{\ln 2d_l - \tilde{\mu}_l}{\tilde{\sigma}_l}\right)\right]x_l},\ l = 1, \ldots, 3,$$

if $0 \le x_l \le 2d_l$, and *zero* otherwise, and

$$f_{Z_l}(z_l; g_l) = \frac{\exp\left[-\frac{1}{2\bar{\tilde{\sigma}}_l^2}(\ln z_l - \bar{\tilde{\mu}}_l)^2\right]}{\sqrt{2\pi}\bar{\tilde{\sigma}}_l\left[\Phi\left(\frac{\ln 2 - \bar{\tilde{\mu}}_l}{\bar{\tilde{\sigma}}_l}\right)\right]z_l},\ l = 1, \ldots, 3,$$

if $0 \le z_l \le 2$, and *zero* otherwise. Here, $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian random variable; for $l = 1, 2$, $\tilde{\mu}_l = \ln\left[d_l^2/\sqrt{d_l^2 + (0.05d_l)^2}\right]$, $\bar{\tilde{\mu}}_l = \ln\left[g_l^2/\sqrt{g_l^2 + (0.05g_l)^2}\right]$, and $\tilde{\sigma} = \bar{\tilde{\sigma}} = 0.05$; for $l = 3$, $\tilde{\mu}_l = \ln\left[d_l^2/\sqrt{d_l^2 + (0.15d_l)^2}\right]$, $\bar{\tilde{\mu}}_l = \ln\left[g_l^2/\sqrt{g_l^2 + (0.15g_l)^2}\right]$, and $\tilde{\sigma} = \bar{\tilde{\sigma}} = 0.15$; and for l = 1–3, $d_{l,L} \le d_l \le d_{l,U}$, defined in (38), and $g_l = 1$. Also, the PDFs of X_4 and X_5 are assumed as being represented by the truncated version on the domain $[0, 7.6]$ of the Weibull defined in Table 4 and the truncated version on $[20, 80]$ of the Gaussian defined in Table 4, respectively.

The bi-objective functions in (38) are linearly aggregated by weights $w_1 \in \mathbb{R}_0^+$ and $w_2 \in \mathbb{R}_0^+$, such that $w_1 + w_2 = 1$. By selecting the weighted sum approach, the scalarized objective function is defined as

$$c_0(\mathbf{d}) := w_1\frac{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]}{\mathbb{E}_{\mathbf{d}_0}[y_0(\mathbf{X})]} + w_2\frac{\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}}{\sqrt{\mathrm{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}.$$

Table 5 summarizes the approximate optimal solutions for the case of $w_1 = 0.5$ and $w_2 = 0.5$, including the requisite numbers of design iterations and function evaluations, by the direct SDD and direct PCE methods. To confirm the accuracy of their results, the reference solution, obtained by employing the QMCS method with the sample size of 5×10^6 for stochastic analysis and central finite-difference approximation for design sensitivities, are tabulated in the eighth column from the left in Table 5. Four approximate optimal designs, obtained by univariate ($S = 1$) direct SDD method—one with linear SDD approximation ($p = 1$, $I = 4$) and the other two with quadratic SDD approximations ($p = 2$, $I = 3, 4$)—and bivariate ($S = 2$) direct SDD method with quadratic SDD approximations ($p = 2$, $I = 2$), are tabulated from the second to fifth columns, respectively, from the left in Table 5. For the univariate ($S = 1$) cases, the respective optimal designs converge with the increase to four subintervals regardless of either linear or quadratic SDD approximations, while for the bivariate ($S = 2$) case, the respective optimal design converges with only two subintervals. On the other hand, the direct PCE method needs a higher-order ($m = 3$ or 4) of polynomial basis to obtain the converged optimal solutions compared with the order ($p = 1$ or 2) of spline basis in the proposed method. Furthermore, the converged optimal designs by the proposed method with the linear ($p = 1$) or quadratic ($p = 2$) SDD approximation are closer to the reference solution by QMCS than one by the PCE counterpart with the higher order ($m = 4$) of polynomial basis. Having said so, the requisite numbers of function evaluations (4,745 in c_0 or 7,665 in c_a, a = 1–3) by the proposed method ($S = 1$, $p = 1$, $I = 4$) are one-half or one-fifth of ones (11,200 in c_0 or 40,320 in

c_a, a = 1–3) by the PCE counterpart ($m = 4$). The results show that the proposed method is not only more accurate but also more computationally efficient than the direct PCE method in obtaining an optimal solution for the RDO problem with highly nonlinear functions.

The direct SDD with the univariate SDD approximation ($S = 1$, $p = 1$, $I = 4$) was selected to obtain Pareto optimal solutions for 17 cases of evenly distributed combinations of weights w_1 and w_2. Table 6 summarizes the results of the weighted sum approach for aforementioned 17 cases of w_1 and w_2. For the middle values (0.3–0.7) of w_1 and w_2, the Pareto optimal solutions $\tilde{d}_1^*$, $\tilde{d}_2^*$, and $\tilde{d}_3^*$ are noticeably or slightly different in each case of the combinations of weights. Indeed, in Figure 6, the Pareto frontier, generated from the Pareto solutions, signifies the trade-off between two individual objectives—the mean $\mathbb{E}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and standard deviation $\sqrt{\text{var}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]}$. From the Pareto frontier, the designer can select the most suitable design. The result demonstrates how the direct SDD can be employed in obtaining the Pareto optimal solutions for the bi-objective RDO problem.

6.3. Example 3: Optimal sizing design of a ten-bar truss

In Example 3, a linear-elastic ten-bar truss, studied in earlier work [2], was used to evaluate the multi-point single-step SDD method. As shown in Fig. 7, the truss is simply supported at nodes 1 and 4 and is subjected to two vertically downward concentrated forces of F_1 and F_2 at nodes 2 and 3, respectively, and a horizontal concentrated force of F_3 at node 3. The material is aluminum alloy, which has Young's modulus of 10^7 psi and a mass density of 0.1 lb/in^3. There are ten ($N = 10$) random variables $\mathbf{X} = (X_1, \ldots, X_{10})^\intercal$, representing random cross-sectional areas of all ten bars. The ten random variables, X_k, $k = 1 \ldots, 10$, are mutually independent, each of which follows a lognormal distribution with the mean $\mathbb{E}_{\mathbf{d}}[X_k]$ and standard deviation $0.05\mathbb{E}_{\mathbf{d}}[X_k]$. There are ten ($M = 10$) design variables, such that $d_l = \mathbb{E}_{\mathbf{d}}[X_l]$, $l = 1, \ldots, 10$. The objective is to minimize the second-moment properties of the mass of the entire truss structure, subject to two cases of constraints: (1) Case 1 involving axial stresses in all ten bars and the vertical displacement (v_3) at node 3; (2) Case 2 entailing maximum of axial stresses at nine bars, the axial stress of the tenth bar, and the vertical displacement (v_3) at node 3. The scale factors of 3 in the constraints are intended to satisfy the constraint conditions with at least 99.865% probability if the performance functions $y_a(\mathbf{X})$, $a = 1, \ldots, 11$ (Case 1) and $a = 1, \ldots, 3$ (Case 2), follow the standard Gaussian distribution. More specifically, the RDO problem is defined to

$$\begin{aligned}
\min_{\mathbf{d}\in\mathcal{D}\subseteq\mathbb{R}^M} \quad & c_0(\mathbf{d}) := 0.5\frac{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]}{\mathbb{E}_{\mathbf{d}_0}[y_0(\mathbf{X})]} + 0.5\frac{\sqrt{\text{var}_{\mathbf{d}}[y_0(\mathbf{X})]}}{\sqrt{\text{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}, \\
\text{subject to} \quad & c_a(\mathbf{d}) := 3\sqrt{\text{var}_{\mathbf{d}}[y_a(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_a(\mathbf{X})] \le 0, \\
& a = 1, \ldots, K,\ K = 11 \text{ (Case 1) or 3 (Case 2)}, \\
& 1 \text{ in}^2 \le d_l \le 35 \text{ in}^2,\ l = 1, \ldots, 10,
\end{aligned}$$

where

$$y_0(\mathbf{X}) = 0.1\sum_{k=1}^{10} l_k X_k$$

is the random mass of the truss with l_k, k=1,…,10, representing bar lengths. Two distinct cases of performance functions y_a are as follows:

Case 1:

$$y_a(\mathbf{X}) = \begin{cases} 25,000 - |\sigma_a(\mathbf{X})| \text{ (psi)}, & \text{if } a = 1, \ldots, 9, \\ 75,000 - |\sigma_a(\mathbf{X})| \text{ (psi)}, & \text{if } a = 10, \\ 5 - |v_3(\mathbf{X})| \text{ (in)}, & \text{if } a = 11, \end{cases}$$

are eleven ($K = 11$) stochastic performance functions.

Case 2:

$$y_a(\mathbf{X}) = \begin{cases} 25,000 - \max\left(|\sigma_1(\mathbf{X})|, \ldots, |\sigma_9(\mathbf{X})|\right) \text{ (psi)}, & \text{if } a = 1, \\ 75,000 - |\sigma_{10}(\mathbf{X})| \text{ (psi)} & \text{if } a = 2, \\ 5 - |v_3(\mathbf{X})| \text{ (in)}, & \text{if } a = 3, \end{cases}$$

are three ($K = 3$) stochastic performance functions.

Cases 1 and 2 are devised with the same intention of limiting the upper values of axial stresses and displacement. However, only Case 2 involves the maximum function of axial stresses between X_1 and X_9, which generally introduces discontinuities to the response. Case 2 is intended to evaluate the ability of the proposed method in handling discontinuities in the response h_2 to find an RDO solution. The initial design is $\mathbf{d}_0 = (30, \ldots, 30)^\intercal$ in^2. The approximate optimal solution is denoted by $\tilde{\mathbf{d}}^* = (\tilde{d}_1^*, \ldots, \tilde{d}_{10}^*)^\intercal$.

Since SDD basis functions are defined on a finite domain, it is assumed that the PDF of X_l is truncated lognormal with an adequately large support. Subsequently, the scaling transformation leads to Z_l, which is also truncated lognormal. Consequently, the PDFs of X_l and Z_l used in this example are as follows:

$$f_{X_l}(x_l; d_l) = \frac{\exp\left[-\frac{1}{2\tilde{\sigma}_l^2}\left(\ln x_l - \tilde{\mu}_l\right)^2\right]}{\sqrt{2\pi}\tilde{\sigma}_l\left[\Phi\left(\frac{\ln 2d_l - \tilde{\mu}_l}{\tilde{\sigma}_l}\right)\right]x_l}, \ l = 1, \ldots, 10,$$

if $0 \le x_l \le 2d_l$, and *zero* otherwise, and

$$f_{Z_l}(z_l; g_l) = \frac{\exp\left[-\frac{1}{2\bar{\tilde{\sigma}}_l^2}\left(\ln z_l - \bar{\tilde{\mu}}_l\right)^2\right]}{\sqrt{2\pi}\bar{\tilde{\sigma}}_l\left[\Phi\left(\frac{\ln 2 - \bar{\tilde{\mu}}_l}{\bar{\tilde{\sigma}}_l}\right)\right]z_l}, \ l = 1, \ldots, 10,$$

if $0 \le z_l \le 2$, and *zero* otherwise. Here, $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian random variable; and for $l = 1, \ldots, 10$, $\tilde{\mu}_l = \ln\left[d_l^2/\sqrt{d_l^2 + (0.05d_l)^2}\right]$, $\bar{\tilde{\mu}}_l = \ln\left[g_l^2/\sqrt{g_l^2 + (0.05g_l)^2}\right]$, $\tilde{\sigma}_l^2 = \bar{\tilde{\sigma}}_l = 0.0025$, $1 \le d_l \le 35$, and $g_l = 1$.

Table 5: Optimization results of an eccentric loaded column (Example 2)

	Direct SDD			Direct PCE			
	$S = 1, p = 1$ $I = 4$	$S = 1, p = 2$ $I = 3$	$I = 4$	$S = 2, p = 2$ $I = 2$	$m = 3$ $m' = 3$	$m = 4$ $m' = 3$	QMCS[a),(b)]
$\tilde{d_1}^*$,m	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1014
$\tilde{d_2}^*$,m	0.0259	0.0388	0.0266	0.0299	0.0308	0.0329	0.0266
$\tilde{d_3}^*$,m	7.4953	7.6163	7.2634	7.5961	6.8263	7.2169	7.5939
$c_0(\tilde{\mathbf{d}}^*)$	0.1879	0.0989	0.0987	0.0990	0.0988	0.0989	0.1005
$c_1(\tilde{\mathbf{d}}^*)$	−0.9765	−0.9873	−0.9776	−0.9820	−0.9819	−0.9842	−0.7812
$c_2(\tilde{\mathbf{d}}^*)$	−0.0014	−0.3361	−0.0055	−0.1372	−0.1116	−0.1918	−0.0661
$c_3(\tilde{\mathbf{d}}^*)$	−0.9237	−0.9838	−0.9330	−0.9668	−0.9515	−0.9693	0.2943
$c_4(\tilde{\mathbf{d}}^*)$	−0.0239	−0.0368	−0.0246	−0.0279	−0.0288	−0.0309	−0.0246
$\mathbb{E}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]$,m^2	28.3108	28.1742	28.6047	28.1918	29.1595	28.6714	28.5730
$\sqrt{\text{var}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]}$,m^2	1.9900	2.0015	1.9665	2.0105	1.9456	1.9775	2.0374
No. of iterations	12	11	13	10	11	289	4
No. of y_0 eval.	4, 745	4, 355	7, 040	15, 355	7, 100	11, 200	2, 660, 000, 000
No. of y_a eval. $a = 1 - 3$	7, 665	7, 035	11, 440	43, 990	19, 880	40, 320	2, 660, 000, 000
No. of y_4 eval.	73	67	88	83	71	64	532

a.QMCS with the sample size of 5×10^6 for statistical moment analysis and design sensitivity analysis based on the central finite difference method was employed.
b.$\mathbf{d}_0$ was signed as $\tilde{\mathbf{d}}^*$ in its fifth column.

Table 6: Bi-optimization results of an eccentric loaded column (Example 2)

Weights		Pareto solutions[(a)]			Mean	Standard deviation
w_1	w_2	$\tilde{d}_1^*$	$\tilde{d}_2^*$	$\tilde{d}_3^*$	$\mathbb{E}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]$	$\sqrt{\text{var}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]}$
0.1	0.9	0.1000	0.1203	5.1447	31.3838	1.8399
0.15	0.85	0.1000	0.0342	5.2542	31.1379	1.8401
0.2	0.8	0.1000	0.0339	5.2304	31.1379	1.8401
0.25	0.75	0.1000	0.0314	5.2164	31.1820	1.8382
0.3	0.7	0.1000	0.0324	5.2115	31.1894	1.8381
0.35	0.65	0.1000	0.0328	5.2138	31.1870	1.8382
0.4	0.6	0.1000	0.0306	5.5103	30.8116	1.8524
0.45	0.55	0.1000	0.0481	6.3602	29.7653	1.9037
0.5	0.5	0.1000	0.0259	7.4953	28.3108	1.9900
0.55	0.45	0.1000	0.0242	8.5750	26.9514	2.0920
0.6	0.4	0.1000	0.0250	9.9705	25.1984	2.2469
0.65	0.35	0.1000	0.0169	13.2153	21.1098	2.6819
0.7	0.3	0.1000	0.0090	20.0000	12.5721	3.7834
0.75	0.25	0.1000	0.0090	20.0000	12.5721	3.7834
0.8	0.2	0.1000	0.0090	20.0000	12.5721	3.7834
0.85	0.15	0.1000	0.0090	20.0000	12.5721	3.7834
0.9	0.1	0.1000	0.0090	20.0000	12.5721	3.7834

a.Optimal solutions were obtained by the direct SDD of $S = 1,\ p = 1,\ I = 4$.

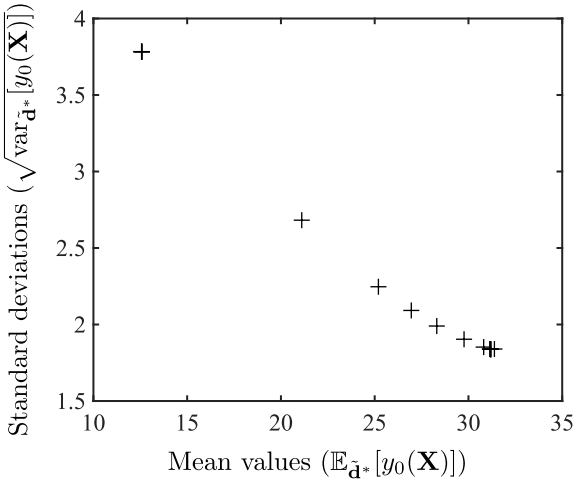


Figure 6: Pareto optimal set of mean and standard deviation of y_0 for RDO problem (Example 2).

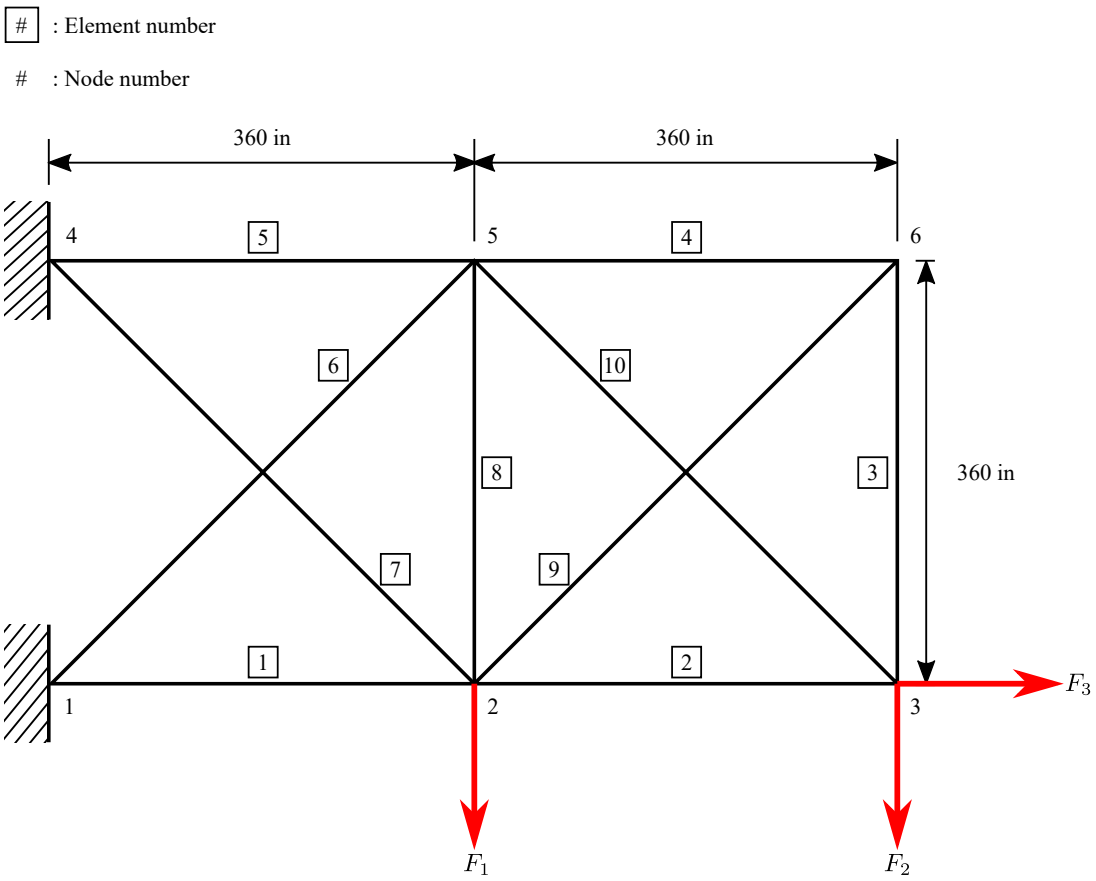


Figure 7: A ten-bar truss structure (Example 3).

Table 7: Optimization results of a ten-bar truss in Cases 1 and 2 (Example 3)

	MPSS SDD[a]				MPSS PCE[b]	QMCS[c]	
	Case 1 (Performance functions y_a, $a = 1, \ldots, 11$)						
	$S = 1$, $p = 1$		$S = 1$, $p = 2$				
	$I = 2$	$I = 4$	$I = 2$	$I = 4$	$m = 2$, $m' = 3$	I[d]	II[e]
$\tilde{d}_1^*$,in^2	4.9433	5.0679	4.9421	5.1525	5.1564	5.1760	5.1759
$\tilde{d}_2^*$,in^2	14.7865	14.7350	14.7802	14.6224	14.7813	14.7920	14.7920
$\tilde{d}_3^*$,in^2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$\tilde{d}_4^*$,in^2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$\tilde{d}_5^*$,in^2	4.9433	5.0652	4.9411	5.1513	5.1478	5.1767	5.1767
$\tilde{d}_6^*$,in^2	1.0000	1.3102	1.0000	1.5968	1.5320	1.6680	1.6679
$\tilde{d}_7^*$,in^2	12.4335	12.2041	12.4368	12.0881	12.0903	12.0540	12.0541
$\tilde{d}_8^*$,in^2	3.2491	3.1941	3.2520	3.1866	3.0266	3.0016	3.0016
$\tilde{d}_9^*$,in^2	1.3733	1.2926	1.3740	1.1484	1.3812	1.3828	1.3828
$\tilde{d}_{10}^*$,in^2	2.6274	2.7412	2.6289	2.9917	2.6185	2.6166	2.6166
$c_0(\tilde{\mathbf{d}}^*)$	0.1879	0.1876	0.1879	0.1879	0.1875	0.1878	0.1878
$c_1(\tilde{\mathbf{d}}^*)$	-1.29×10^{-1}	-1.45×10^{-1}	-1.29×10^{-1}	-1.82×10^{-1}	-1.26×10^{-1}	-1.28×10^{-1}	-1.28×10^{-1}
$c_2(\tilde{\mathbf{d}}^*)$	-1.25×10^{-4}	-2.73×10^{-3}	2.36×10^{-6}	-8.31×10^{-5}	-3.83×10^{-3}	-7.29×10^{-11}	1.93×10^{-7}
$c_3(\tilde{\mathbf{d}}^*)$	-3.13×10^{-4}	-6.61×10^{-4}	4.15×10^{-5}	-9.39×10^{-6}	3.02×10^{-4}	-7.02×10^{-11}	-2.57×10^{-11}
$c_4(\tilde{\mathbf{d}}^*)$	-6.30×10^{-3}	-6.07×10^{-2}	-5.74×10^{-3}	-1.65×10^{-1}	-4.15×10^{-4}	-1.50×10^{-4}	-1.51×10^{-4}
$c_5(\tilde{\mathbf{d}}^*)$	-5.20×10^{-3}	-6.11×10^{-2}	-5.66×10^{-3}	-1.65×10^{-1}	-2.41×10^{-4}	-1.52×10^{-10}	-7.64×10^{-9}
$c_6(\tilde{\mathbf{d}}^*)$	1.58×10^{-4}	-1.97×10^{-3}	-7.76×10^{-7}	-5.52×10^{-5}	-2.29×10^{-3}	4.01×10^{-11}	1.93×10^{-7}
$c_7(\tilde{\mathbf{d}}^*)$	-4.27×10^{-1}	-4.30×10^{-1}	-4.28×10^{-1}	-4.40×10^{-1}	-4.33×10^{-1}	-4.42×10^{-1}	-4.42×10^{-1}
$c_8(\tilde{\mathbf{d}}^*)$	4.53×10^{-4}	3.75×10^{-3}	-1.26×10^{-6}	3.25×10^{-4}	2.23×10^{-3}	1.71×10^{-11}	-1.26×10^{-7}
$c_9(\tilde{\mathbf{d}}^*)$	6.40×10^{-4}	-2.28×10^{-3}	-1.38×10^{-4}	5.87×10^{-4}	5.07×10^{-3}	-1.70×10^{-10}	-2.36×10^{-7}
$c_{10}(\tilde{\mathbf{d}}^*)$	3.28×10^{-4}	2.27×10^{-3}	4.64×10^{-5}	-1.09×10^{-3}	7.78×10^{-4}	$-2.12E-10$	-9.01×10^{-9}
$c_{11}(\tilde{\mathbf{d}}^*)$	-3.63×10^{-1}	-3.79×10^{-1}	-3.63×10^{-1}	-4.13×10^{-1}	-3.61×10^{-1}	$-3.61E-01$	-3.61×10^{-1}
$\mathbb{E}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]$,lb	1964.8006	1975.6463	1964.8390	1991.5560	1981.2082	1987.4125	1987.4095
$\sqrt{\text{var}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]}$,lb	44.4141	44.1036	44.4135	43.9781	43.9940	43.9904	43.9905
No. of iterations	291	334	266	249	289	3	13
No. of FEA	1,785	4,100	2,945	4,335	5,280	14,700,000	65,100,000
	Case 2 (Performance functions y_a, $a = 1, \ldots, 3$)						
	$S = 1$, $p = 1$		$S = 1$, $p = 2$				
	$I = 4$	$I = 6$	$I = 2$	$I = 4$	$m = 3$, $m' = 3$	III[f]	IV[g]
$\tilde{d}_1^*$,in^2	5.1082	5.0150	5.2165	5.2076	5.8179	5.3729	5.3728
$\tilde{d}_2^*$,in^2	14.7759	14.9630	15.5024	14.9090	16.0398	14.7451	14.7451
$\tilde{d}_3^*$,in^2	2.8592	1.5288	2.2347	1.5141	4.8875	1.0000	1.0000
$\tilde{d}_4^*$,in^2	1.3469	1.8268	2.3502	1.4698	3.6214	1.0000	1.0000
$\tilde{d}_5^*$,in^2	5.0852	5.0289	5.1988	5.2002	6.0391	5.3722	5.3723
$\tilde{d}_6^*$,in^2	1.9325	1.0000	1.4037	1.2510	4.0446	1.6439	1.6438
$\tilde{d}_7^*$,in^2	11.7774	12.2523	11.6594	12.0193	10.9542	12.0165	12.0166
$\tilde{d}_8^*$,in^2	3.6045	2.8308	2.0302	2.8016	3.1001	3.1555	3.1555
$\tilde{d}_9^*$,in^2	1.9147	1.9578	2.8631	1.9752	1.9222	1.3319	1.3319
$\tilde{d}_{10}^*$,in^2	2.4432	2.2946	1.7094	2.4105	3.6412	2.7890	2.7890
$c_0(\tilde{\mathbf{d}}^*)$	0.1923	0.1899	0.1919	0.1893	0.2143	0.1889	0.1889
$c_1(\tilde{\mathbf{d}}^*)$	2.40×10^{-2}	2.72×10^{-2}	3.77×10^{-2}	1.62×10^{-2}	1.14×10^{-2}	1.48×10^{-8}	1.90×10^{-8}
$c_2(\tilde{\mathbf{d}}^*)$	-3.94×10^{-1}	-3.74×10^{-1}	-3.52×10^{-1}	-3.85×10^{-1}	-5.75×10^{-1}	-3.88×10^{-1}	-3.88×10^{-1}
$c_3(\tilde{\mathbf{d}}^*)$	-1.97×10^{-1}	-1.43×10^{-1}	-1.14×10^{-1}	-1.50×10^{-1}	-3.88×10^{-1}	-1.57×10^{-1}	-1.57×10^{-1}
$\mathbb{E}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]$,lb	2099.9415	2014.1591	2069.0385	2018.5866	2469.0707	2008.4488	2008.4456
$\sqrt{\text{var}_{\tilde{\mathbf{d}}^*}[y_0(\mathbf{X})]}$,lb	43.9958	44.4004	44.3570	44.0991	46.9739	44.1154	44.1155
No. of iterations	369	450	342	367	304	64	81
No. of FEA	5,330	9,760	3,720	7,140	32,890	294,000,000	392,700,000

a. The multi-point single-step SDD method was employed
b. The multi-point single-step PCE method was employed
c. QMCS with the sample size of 1×10^5 for statistical moment analysis and design sensitivity analysis based on the central finite difference method was employed
d. I: $\mathbf{d}_0$ was signed as $\tilde{\mathbf{d}}^*$ in its fifth column in Case 1
e. II: $\mathbf{d}_0$ was signed as $\tilde{\mathbf{d}}^*$ in its sixth column in Case 1
f. III: $\mathbf{d}_0$ was signed as $\tilde{\mathbf{d}}^*$ in its fifth column in Case 2
g. IV: $\mathbf{d}_0$ was signed as $\tilde{\mathbf{d}}^*$ in its sixth column in Case 2

Table 7 summarizes the assorted RDO results by multi-point single-step SDD methods for the two cases of performance functions. For Case 1, four approximate optimal designs, presented in the second through fifth columns from the left in Table 7, were obtained by univariate ($S = 1$) multi-point single-step SDD methods–two with linear SDD approximations ($p = 1,\ I = 2, 4$) and the other two with quadratic SDD approximations ($p = 2,\ I = 2, 4$), respectively. For either linear or quadratic SDD approximations of the proposed method, the respective optimal designs converge with the increase in number of subintervals in each coordinate direction from $I = 2$ to $I = 4$. For comparison, the optimization results from the multi-point single-step PCE method using the second-order ($m = 2$) and the third-order ($m' = 3$) PCE approximations for response and score functions, respectively, are tabulated in the sixth column of Table 7. All of these optimal designs, obtained from the proposed method and its PCE counterpart, are very close to each other, rendering active or inactive constraints. To seek further credibility for the accuracy of the proposed method's RDO solutions, QMCS entailing 1×10^5 samples for stochastic moment analysis and design sensitivity analysis based on the central finite-difference method was employed. However, due to its extensive computational cost, the RDO problem was resolved for two different initial designs assigned as two optimal designs by the proposed method and its PCE counterpart, presented in the fifth and sixth columns of Table 7, respectively. The resulting two reference solutions, denoted by QMCS I and II for the former and the latter cases of initial design, respectively, are listed in the seventh and eighth columns of Table 7. As expected, the two reference solutions are almost the same as their initial designs, indicating that the optimal designs, obtained by the proposed method and its PCE counterpart, are all accurate and reliable. Although the requisite numbers of FEA (1,785–4,335) by four MPSS SDD methods are slightly less than one (5,280) by the PCE counterpart, the advantage of the proposed method over its PCE counterpart is not clearly visible in Case 1.

For Case 2, four approximate optimal designs, obtained by univariate ($S = 1$) multi-point single-step SDD methods (two with linear SDD approximations ($p = 1,\ I = 4, 6$) and the rest two with quadratic SDD approximations ($p = 2,\ I = 2, 4$)), are presented in second and fifth columns from the left in Table 7, respectively. Whether the proposed method's RDO solutions are obtained from the linear ($p = 1$) or quadratic ($p = 2$) SDD approximations, their optimal designs converge as the number of subintervals increases from $I = 4$ to $I = 6$ or $I = 2$ to $I = 4$, respectively, yielding all active constraints ($c_1 \simeq 0, c_2 \simeq 0, c_3 \simeq 0$). For comparison, the approximate optimal design, obtained by the third-order ($m = 3,\ m' = 3$) multi-point single-step PCE method, is presented in the sixth column of Table 7. Notwithstanding that the multi-point single-step PCE method employs higher-order ($m = 3,\ m' = 3$) approximations than first- or second-order ($p = 1$ or 2) SDD approximations of the univariate ($S = 1$) proposed method, the resultant PCE solution shows moderate to significant differences with the converged one by the proposed method. Indeed, compared with two reference solutions denoted by QMCS III and IV obtained from QMCS for stochastic analysis and design sensitivity analysis based on the central finite difference method for two different initial designs assigned as optimal designs in the fifth and sixth columns of Table 7, respectively, the proposed method's optimal design is closer than that of the PCE counterpart. Furthermore, the proposed multi-point single-step SDD method requires only 3,720–7,140 FEA to obtain the better optimal design, while the PCE counterpart demands 32,890 FEA. Therefore, the proposed RDO method is not only accurate but also more computationally efficient than its PCE counterpart, particularly in solving RDO problems involving discontinuous performance functions.

6.4. Example 4: Shape optimization of a robotic gripper jaw

This last example demonstrates the high performance of the proposed RDO method in solving an industrial-scale shape design problem, encompassing an industrial robotic arm, as shown in Figure 8. The robotic arm includes a gripper jaw for holding and transmitting workpieces. The gripper jaw is pivoted on a hinge in the interior of the gripper head, which has a rotation axis, and its rotation motion is caused by a rack gear. By rotating the gripper jaw, a friction force between the two surfaces of the gripper jaw and a workpiece is created, resulting in a gripping force for holding the workpiece while it is moved between work positions.

In general, such a gripper jaw needs to be designed to maintain fatigue durability under repetitive loading cycles during operations, sustaining its performances during the expected service life. Otherwise, it may cause manufacturing and maintenance costs to rise. However, uncertainties in manufacturing variables or material properties exist inherently, resulting in the randomness of fatigue life. As a conservative design approach, a large safety factor is applied to ensure its satisfactory long-term performance, but it may cause an unnecessary gripper jaw weight increment, thereby resulting in operating efficiency loss. Thus, uncertainty in fatigue life should be accounted for while designing a lightweight gripper jaw. Such optimal robust design can be achieved by the RDO method developed.

Twenty-six input random variables were employed for modeling the uncertainty in manufacturing tolerances of a robotic gripper jaw geometry. Figure 8b presents a computer-aided design (CAD) model of the gripper jaw with the twenty-six random manufacturing variables X_1 through X_{26}, which are marked in the side and top views. The first two random variables X_1 and X_2 represent the outer shape of the gripper jaw head, and X_3 and X_4 depict the thicknesses of the outer/inner frames on the upper and lower sides of the gripper jaw body, respectively. Also, X_5 depicts the thickness of the total gripper jaw body, and X_6 through X_{26} describe the six holes of the jaw, devised to reduce the weight of the gripper jaw as much as possible.

The random variables are statistically independent and follow lognormal distributions with the means $\mathbb{E}_{\mathbf{d}}[X_l]$, $l = 1, \ldots, 26$, and standard deviations $0.02\mathbb{E}_{\mathbf{d}}[X_l]$, $l = 1, \ldots, 5$, and $0.01\mathbb{E}_{\mathbf{d}}[X_l]$, $l = 6, \ldots, 26$. There are twenty-six design variables, that is, $d_l = \mathbb{E}_{\mathbf{d}}[X_l]$, $l = 1, \ldots, 26$. The gripper jaw is made of Titanium Alloy Ti-6A1-4V with the following deterministic material properties: elastic modulus $E = 115$ GPa and Poisson's ratio $\nu = 0.33$. Also, the deterministic fatigue parameters are

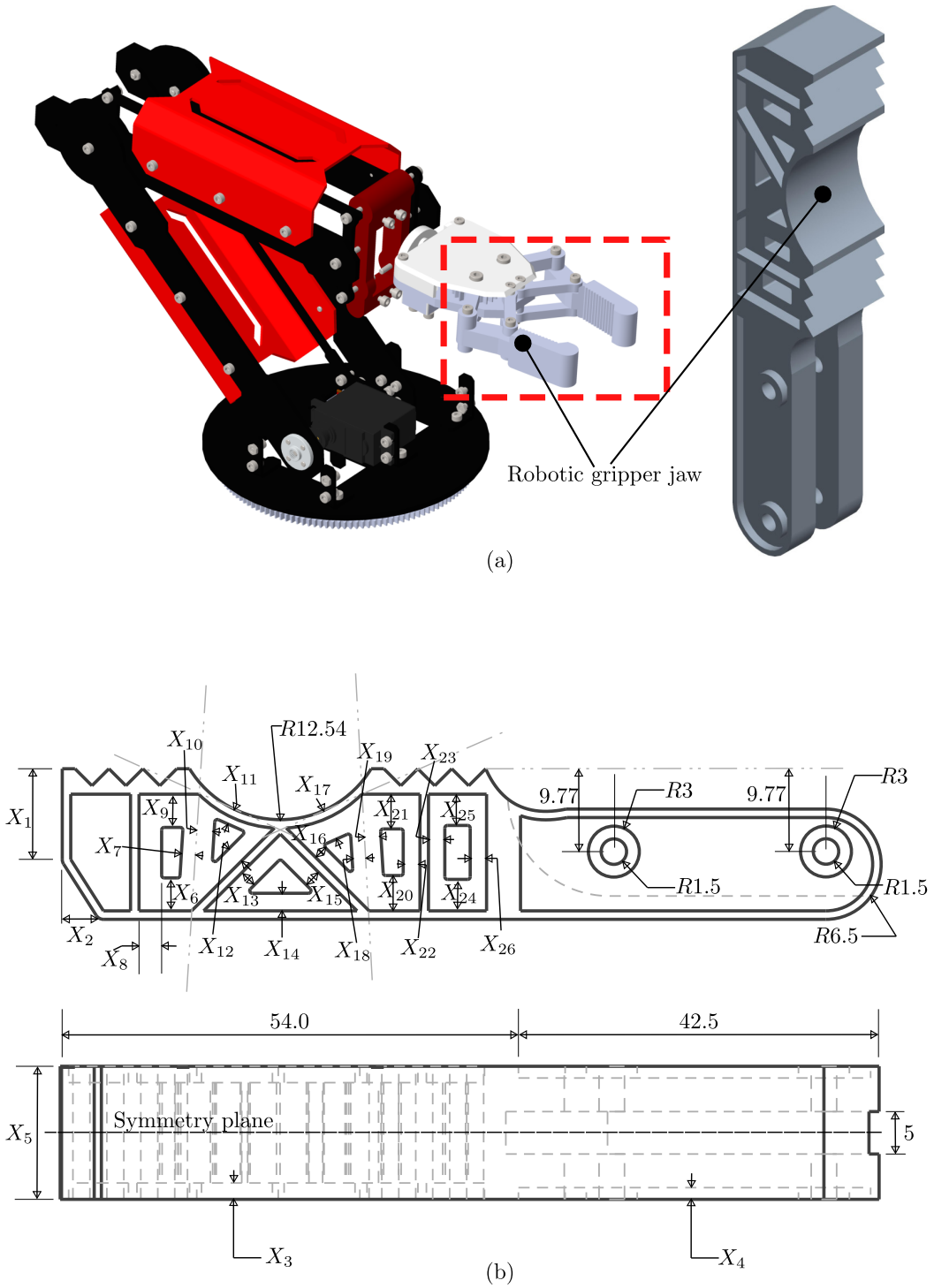


Figure 8: A robotic gripper jaw (Example 4): (a) photos of an industrial robotic arm (left) and the gripper jaw (right); (b) a CAD model of the gripper jaw in side view (upper) and top view (lower) (unit: mm).

as follows: fatigue strength coefficient $\sigma'_f = 2,030$ MPa, fatigue ductility coefficient $\epsilon'_f = 0.841$, fatigue strength exponent $b = -0.104$, and fatigue ductility exponent $c = -0.69$.

The stochastic performance of the gripper jaw was determined by fatigue durability analysis under pressure loading conditions P on the circular surface on the top side of the gripper jaw, as shown in Figure 9a. The loading condition is created when the gripper holds and then releases a rod-like specimen by closing and opening the jaw. As a result, the gripper jaw experiences constant-amplitude cyclic loads with the maximum and minimum values of the load, as follows: $0 \le P \le 57.34/X_5$ MPa (normal direction to the contact surface of the gripper jaw). The essential boundary condition involves fixing the center of the two holes of the gripper jaw in all three directions. The fatigue durability analysis involved (1) calculating maximum principal strain and mean stress at a critical point; and (2) calculating the fatigue crack-initiation life at the critical point from the well-known Coffin-Manson-Morrow equation [33]. The critical point is the location where the von Mises stress is the largest, identified from an FEA where all input random variables are assigned as their mean values.

The objective is to minimize both the mean and the standard deviation of the mass of the gripper jaw by changing the shape of the geometry. At the same time, the fatigue crack-initiation life $N_1(\mathbf{X})$, under the loading condition $P(\mathbf{X})$ at the critical point, exceeds a design threshold of one hundred thousand loading cycles with 98.21% probability. Mathematically, the RDO for this problem is defined to

$$\begin{aligned} \min_{\mathbf{d}\in\mathcal{D}} \quad & c_0(\mathbf{d}) \quad := 0.5\frac{\mathbb{E}_{\mathbf{d}}[y_0(\mathbf{X})]}{\mathbb{E}_{\mathbf{d}_0}[y_0(\mathbf{X})]} + 0.5\frac{\sqrt{\mathrm{var}_{\mathbf{d}}[y_0(\mathbf{X})]}}{\sqrt{\mathrm{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}, \\ \text{subject to} \quad & c_1(\mathbf{d}) \quad := 2.1\sqrt{\mathrm{var}_{\mathbf{d}}[y_1(\mathbf{X})]} - \mathbb{E}_{\mathbf{d}}[y_1(\mathbf{X})] \le 0, \\ & d_{l,L} \le d_l \le d_{l,U},\ l = 1,\ldots,26, \end{aligned}$$

where

$$y_0(\mathbf{X}) = \rho\int_{\mathcal{D}'} d\mathcal{D}'$$

is the random mass of the gripper jaw, and

$$y_1(\mathbf{X}) = \frac{N_1(\mathbf{X})}{1\times 10^5} - 1,$$

is a stochastic performance function based on normalized fatigue crack-initiation life. The initial design $\mathbf{d}_0 = (d_{1,0},\ldots,d_{26,0})^\intercal$ mm; the upper and lower bounds of the design vector $\mathbf{d} = (d_1,\ldots,d_{26})^\intercal$ mm $\in \mathcal{D} \subset \mathbb{R}^{26}$ are tabulated in Table 8. Figure 9b presents the FEA mesh for the initial gripper jaw design, which comprises 114,599 linear tetrahedral elements. The approximate optimal solution is denoted by $\tilde{\mathbf{d}}^* = (\tilde{d}_1^*,\ldots,\tilde{d}_{26}^*)^\intercal$.

Since SDD basis functions are defined on a finite domain, it is assumed that the PDF of X_l is truncated lognormal with an adequately large support. Subsequently, the scaling transformation leads to Z_l, which is also truncated lognormal. Consequently, the PDFs of X_l and Z_l used in this example are as follows:

$$f_{X_l}(x_l;d_l) = \frac{\exp\left[-\frac{1}{2\tilde{\sigma}_l^2}(\ln x_l - \tilde{\mu}_l)^2\right]}{\sqrt{2\pi}\tilde{\sigma}_l\left[\Phi\left(\frac{\ln 2d_l - \tilde{\mu}_l}{\tilde{\sigma}_l}\right)\right]x_l},\ l = 1,\ldots,26,$$

if $0 \le x_l \le 2d_l$, and *zero* otherwise, and

$$f_{Z_l}(z_l;g_l) = \frac{\exp\left[-\frac{1}{2\bar{\tilde{\sigma}}_l^2}(\ln z_l - \bar{\tilde{\mu}}_l)^2\right]}{\sqrt{2\pi}\bar{\tilde{\sigma}}_l\left[\Phi\left(\frac{\ln 2 - \bar{\tilde{\mu}}_l}{\bar{\tilde{\sigma}}_l}\right)\right]z_l},\ l = 1,\ldots,26,$$

if $0 \le z_l \le 2$, and *zero* otherwise. Here, $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian random variable; for $l = 1,\ldots,5$, $\tilde{\mu}_l = \ln\left[d_l^2/\sqrt{d_l^2 + (0.02d_l)^2}\right]$, $\bar{\tilde{\mu}}_l = \ln\left[g_l^2/\sqrt{g_l^2 + (0.02g_l)^2}\right]$, and $\tilde{\sigma} = \bar{\tilde{\sigma}} = 3.9992 \times 10^{-4}$; for $l = 6,\ldots,26$, $\tilde{\mu}_l = \ln\left[d_l^2/\sqrt{d_l^2 + (0.01d_l)^2}\right]$, $\bar{\tilde{\mu}}_l = \ln\left[g_l^2/\sqrt{g_l^2 + (0.01g_l)^2}\right]$, and $\tilde{\sigma} = \bar{\tilde{\sigma}} = 9.9995 \times 10^{-5}$; and for $l = 1,\ldots,26$, $d_{l,L} \le d_l \le d_{l,U}$, defined in Table 8, and $g_l = 1$.

The multi-point single-step method, employing univariate quadratic ($S = 1, p = 2$) SDD approximations with two subintervals ($I = 2$), was employed to solve this robotic gripper jaw design problem. As a result, the optimal design is presented in the third and eighth columns from the left in Table 8. At the optima, the value of the fifth design variable d_5^* reached the lower limit, and the values of the remaining design variables are between their lower and upper limits, satisfying inactive constraints $c_1 \simeq -27.99$. The mean and the standard deviation of the optimal gripper jaw mass is 4.27×10^{-2} kg and 1.34×10^{-3} kg, which represent about 52 % and 37 % reductions, respectively, from the initial mass of 8.91×10^{-2} kg and the initial standard deviation of 2.13×10^{-3} kg. To complete the design process, the requisite number of FEA is 3, 180.

Figures 10a–d present the contour plots of the logarithm of the fatigue crack-initiation life at the mean shapes of the robotic gripper jaw for several design iterations, including the initial and optimal designs. By using the proposed RDO method with tolerances and subregion size parameters appropriately selected, a total of 511 iterations (q) led to the final optimal design. Indeed, there is a significant reduction of the overall volume of the gripper jaw, satisfying the constraint condition exceeding the logarithm of target fatigue life $11.51 = \ln 10^5$.

Figure 11 presents the iteration histories of the objective function c_0 and the normalized twenty-six design variables $d_l/d_{l,0}$, $l = 1,\ldots,26$, during the RDO process attained by the multi-point single-step SDD method employing univariate, linear ($S = 1, p = 1, I = 2$) SDD approximations. In Figure 11a, the value of the objective function converges monotonically from 1.00 at the initial design to nearly 0.56 at the optimal design, about a 44% decrease. According to Figures 11b–f, the values of all twenty-six design variables have changed significantly from their initial values, thus resulting in substantial modifications

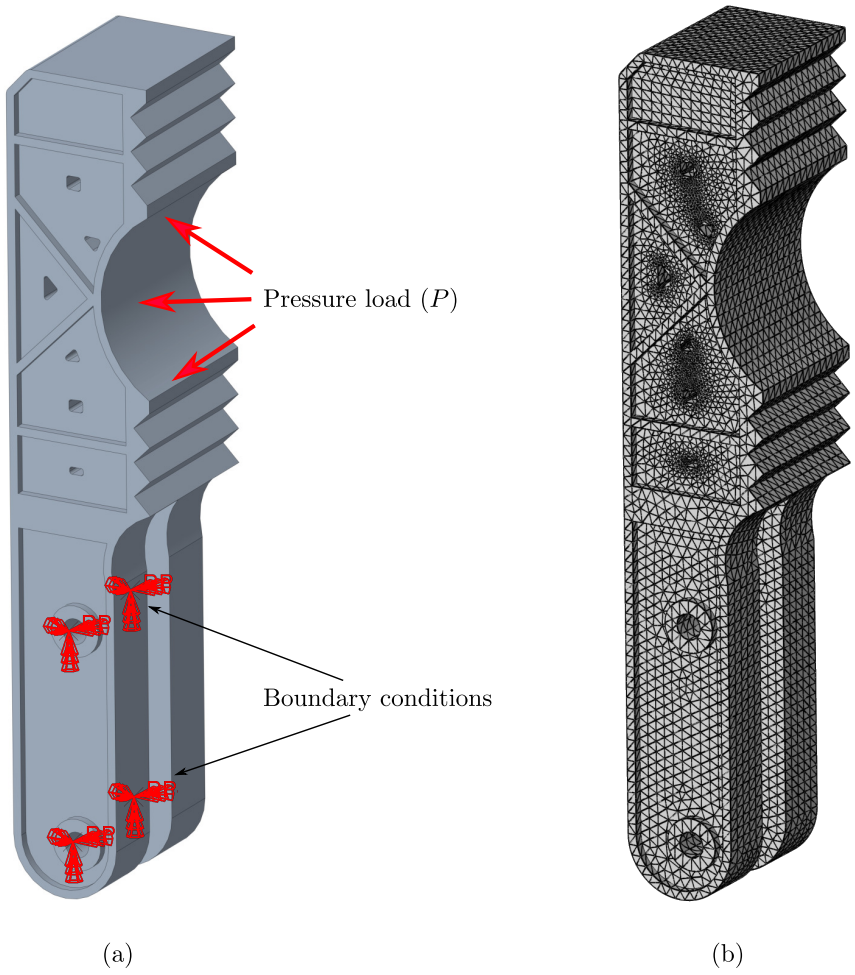
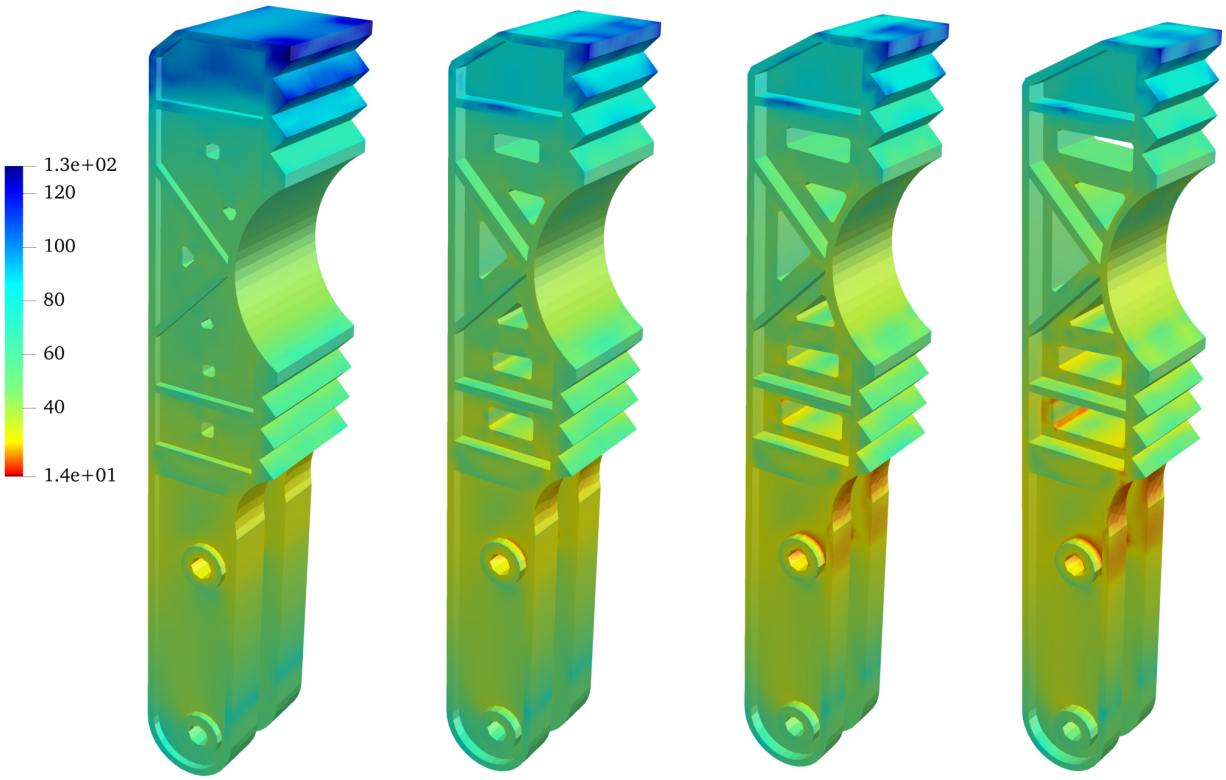


Figure 9: An FEA of the robotic gripper jaw (Example 4): (a) pressure load and boundary conditions; (b) a tetrahedral mesh comprising 114,599 elements

Table 8: Initial and optimal values, and bounds of design variables for the robotic gripper jaw problem (Example 4)

l	$d_{l,0}$ mm	$\tilde{d}_l^*$ mm	$d_{l,L}$ mm	$d_{l,U}$ mm	l	$d_{l,0}$ mm	$\tilde{d}_l^*$ mm	$d_{l,L}$ mm	$d_{l,U}$ mm
1	14	7.2288	5	14	14	3	1.3379	1	3
2	3	5.7282	3	6	15	3	1.4062	1	3
3	1	2.7416	1	3	16	2	1.3371	1	2
4	1	1.7170	1	2	17	5	2.3001	2	5
5	18	13.0000	13	18	18	2	1.3153	1	2
6	6	1.9828	1	6	19	2.5	1.4805	1	2.5
7	2	1.3444	1	2	20	6	2.4292	1	6
8	2	1.2839	1	2	21	6	2.3824	1	6
9	6	1.7168	1	6	22	2.5	1.6261	1	2.5
10	3	1.3698	1	3	23	2.5	1.4940	1	2.5
11	2	1.6774	1.5	2	24	6	1.7633	1	6
12	3	1.5084	1	3	25	6	1.5763	1	6
13	3	1.3881	1	3	26	2.5	1.4298	1	2.5



(a) Initial design (b) Iteration 100 (c) Iteration 200 (d) Iteration 511

Figure 10: Contours of logarithmic fatigue crack-initiation life of the robotic gripper jaw (Example 4): (a) initial design; (b) iteration 100; (c) iteration 200; (d) iteration 511 (optimum), obtained from the multi-point single-step SDD method employing univariate linear ($S = 1, p = 1$) SDD approximations with two subintervals ($I = 2$)

Table 9: Reductions in the mean and standard deviation of y_0 from initial to optimal designs

Examples	$\frac{\mathbb{E}_{\mathbf{d}^*}[y_0(\mathbf{X})] - \mathbb{E}_{\mathbf{d}_0}[y_0(\mathbf{X})]}{\mathbb{E}_{\mathbf{d}_0}[y_0(\mathbf{X})]}$	$\frac{\sqrt{\text{var}_{\mathbf{d}^*}[y_0(\mathbf{X})]} - \sqrt{\text{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}{\sqrt{\text{var}_{\mathbf{d}_0}[y_0(\mathbf{X})]}}$
1-1[a]	68.83 %	93.39 %
1-2[b]	46.27 %	27.95 %
2[c]	91.06 %	89.16 %
3-1[d]	84.32 %	78.11 %
3-2[e]	83.71 %	78.12 %
4	52 %	37 %

a. The value of $\mathbb{E}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and $\sqrt{\text{var}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]}$ is the average of all corresponding SDD results in Table 3-(Case 1)
b. The value of $\mathbb{E}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and $\sqrt{\text{var}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]}$ is the average of all corresponding SDD results in Table 3-(Case 2)
c. The value of $\mathbb{E}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and $\sqrt{\text{var}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]}$ is the average of all corresponding SDD results in Table 5
d. The value of $\mathbb{E}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and $\sqrt{\text{var}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]}$ is the average of all corresponding SDD results in Table 7-(Case 1)
e. The value of $\mathbb{E}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]$ and $\sqrt{\text{var}_{\bar{\mathbf{d}}^*}[y_0(\mathbf{X})]}$ is the average of all corresponding SDD results in Table 7-(Case 2)

of the gripper jaw geometry and its hollow shapes and sizes. This concluding example shows that the multi-point single step SDD method developed is capable of solving industrial-scale engineering design problems using only a few thousand FEA.

The accomplishments of the proposed RDO method in all three examples are summarized in Table 9, which presents the percentage changes in the mean and standard deviation of y_0 from initial to optimal designs. The second-moment statistics at optimal designs are averages of all SDD-based method solutions. While the largest reduction of the mean is 84.32%, the standard deviation decreases by at most 93.39%. Apparently, the proposed RDO methods have played a crucial role in lessening the statistical moments of objective functions, thus achieving insensitive designs.

7. Discussion

The computational efficiency of the multi-point single-step SDD method largely relies on the total iteration count (Q'), where each q'th iteration calls for a single stochastic analysis to solve the local RDO problem for the q'th subregion. Since the q'th subregion size is always less than the domain size of input random variables at the q'th initial design as depicted in Figure 1b, the smaller intervals of the input will increase the total iteration count (Q'), causing a subsequent increase in the computational costs of the proposed RDO method. In such cases, a new optimization algorithm may need to be developed to tackle the aforementioned computational issue. In addition, for unbounded input, which occur in many engineering systems, the input domains are assumed as large, closed, bounded ones, say, greater than or equal to six times their standard deviations, as presented in Examples 1–4. Therefore, for many real-life applications, this is not a limitation of the multi-point single-step SDD method to solve the RDO problems at a reasonable computational price.

The score functions and RDO problems studied in this work are applicable when the design variables are solely the distributional parameters of $\mathbf{X}$. Many engineering design tasks fall under this class of problems. However, there are exceptions that require delving into design problems that depend on structural parameters too, comprising deterministic design variables [34]. Under similar regularity conditions, the corresponding score functions can be be derived, but will now involve the partial derivatives of the performance function with respect to both distributional and structural design parameters, which must exist and be finite. However, RDO problems addressing both types of design parameters are outside the scope of this study. For the same reason, considerations of dynamic problems or reliability analysis, not pursued in this work, are left for future works.

8. Conclusion

Two novel spline-based computational methods, comprising the direct SDD and multi-point single-step SDD methods, were invented for robust design optimization of complex engineering systems. The methods feature SDD of a high-dimensional, discontinuous, or nonsmooth stochastic response for statistical moment analysis, a novel fusion of SDD and score functions for calculating the second-moment sensitivities with respect to the design variables, and standard gradient-based optimization algorithms, constructing direct and multi-point single-step design processes. In these SDD-based methods, the orthonormal basis functions are derived from compactly supported B-splines, resulting in an excellent approximation power for locally pronounced, highly nonlinear, discontinuous, or nonsmooth stochastic output functions. Hence, a low-variate and/or low-degree SDD approximation with an adequate mesh size can produce a remarkably accurate and convergent estimates of response moments. When integrated with score functions, SDD leads to explicit formulae, expressed in terms of the expansion coefficients, for approximating the design sensitivities of moments that are also accurate and theoretically convergent. More importantly, the statistical moments and design sensitivities are determined simultaneously and hence inexpensively from a single stochastic analysis or simulation.

Between the two methods developed, the direct SDD method is most straightforward, but it demands re-calculations of the SDD expansion coefficients at each design iteration. Therefore, it easily becomes computationally expensive. However, due to simplicity in the optimization algorithm, it was employed to evaluate the performance of SDD approximations in solving the elementary mathematical RDO problems. In contrast, the multi-point single-step SDD method adjusts local enforcement of SDD approximation, where the original RDO problem is converted into a series of local RDO problems defined on subregions of the entire design space. As a result, the method permits employing a low-degree SDD approximation to obtain a reliable RDO solution in the case of a large design space and locally highly nonlinear stochastic responses. Also, the latter method avoids the necessity of recomputing the expansion coefficients by reprocessing the old expansion coefficients, thus dramatically reducing the computational cost. Therefore, the multi-point single-step SDD method is capable of solving practical engi-

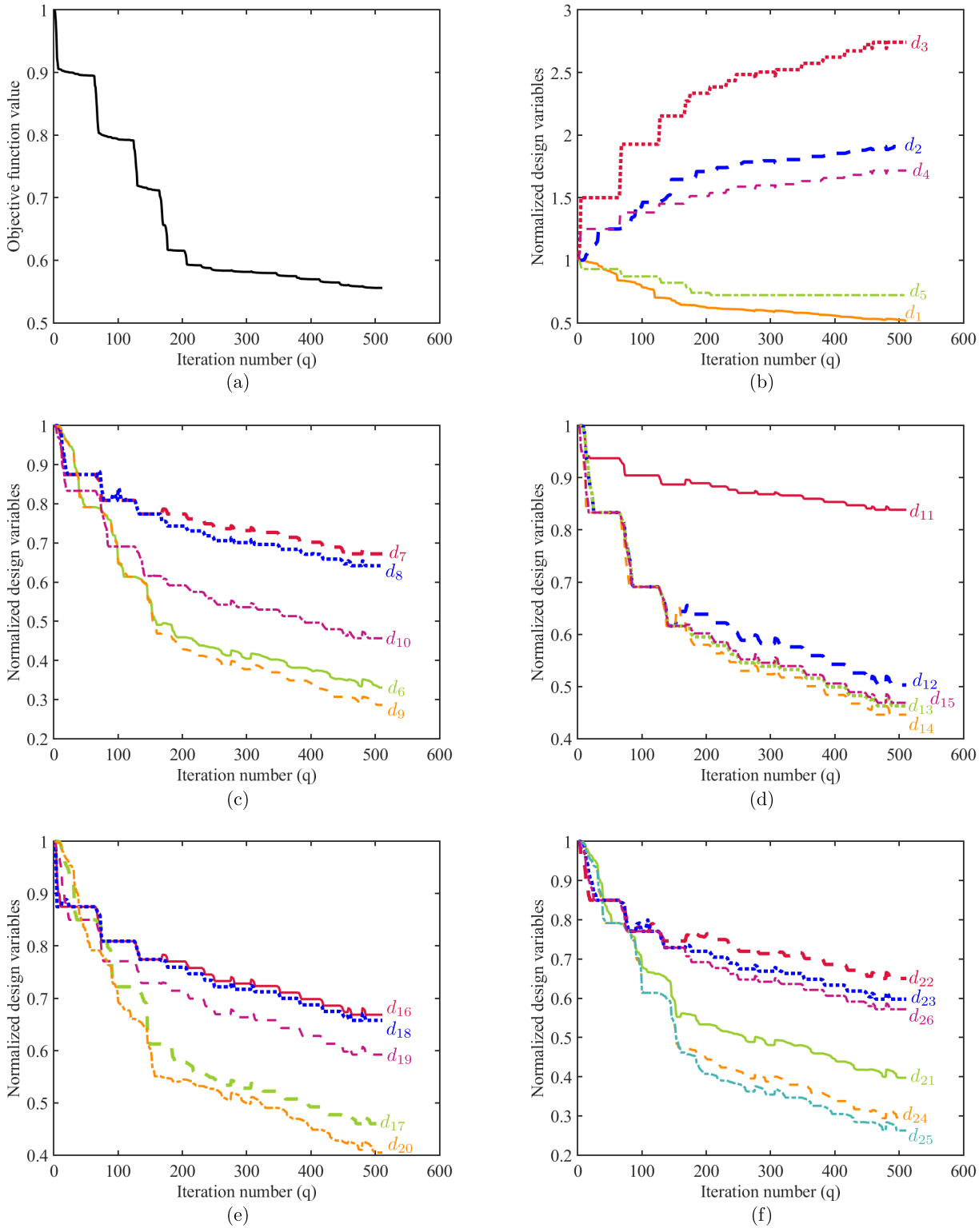


Figure 11: RDO iteration histories for the robotic gripper jaw (Example 4): (a) objective function value; (b) normalized design variables d_1–d_5; (c) normalized design variables d_6–d_{10}; (d) normalized design variables d_{11}–d_{15}; (e) normalized design variables d_{16}–d_{20}; (f) normalized design variables d_{21}–d_{26}

neering problems, as successfully demonstrated by shape design optimization of an industrial-scale robotic gripper jaw.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work received financial support from the U.S. National Science Foundation under Grant No. CMMI-1933114.

Appendix A. B-splines

A.1. Recursive formula

Let $\boldsymbol{\xi}_k$ be a general knot vector of length at least $p_k + 2$ for the interval $[a_k, b_k]$. Denote by $B^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k)$ the i_kth univariate B-spline function with degree $p_k \in \mathbb{N}_0$ for the coordinate direction k. Given the *zero*-degree basis functions such that, for $k = 1, \ldots, N$,

$$B^k_{i_k,0,\boldsymbol{\xi}_k}(z_k) = \begin{cases} 1, & \xi_{k,i_k} \le x_k < \xi_{k,i_k+1}, \\ 0, & \text{otherwise}, \end{cases}$$

all higher-order B-spline functions on $\mathbb{R}$ are determined recursively as

$$\begin{aligned} B^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k) \quad := \quad & \frac{z_k - \xi_{k,i_k}}{\xi_{k,i_k+p_k} - \xi_{k,i_k}} B^k_{i_k,p_k-1,\boldsymbol{\xi}_k}(z_k) + \frac{\xi_{k,i_k+p_k+1} - z_k}{\xi_{k,i_k+p_k+1} - \xi_{k,i_k+1}} \\ & \times B^k_{i_k+1,p_k-1,\boldsymbol{\xi}_k}(z_k), \end{aligned}$$

where $1 \le k \le N$, $1 \le i_k \le n_k$, $1 \le p_k < \infty$, and $0/0$ is considered as *zero*.

A.2. Properties

The B-splines are: (1) non-negative; (2) locally supported on the interval $[\xi_{k,i_k}, \xi_{k,i_k+p_k+1})$ for all i_k; (3) linearly independent; (4) committed to a partition of unity; and (5) pointwise C^∞-continuous everywhere except at the knots ζ_{k,j_k} of multiplicity m_{k,j_k} for all $j_k = 1, \ldots, r_k$, where they are $C^{p_k - m_{k,j_k}}$-continuous, provided that $1 \le m_{k,j_k} < p_k + 1$.

Appendix B. Three-steps for Orthonormalized B-splines

1. Given a set of B-splines of degree p_k, create an auxiliary set by replacing any element, arbitrarily chosen to be the first, with *one*. Arrange the elements of the set into an n_k-dimensional vector

   $$\mathbf{P}_k(z_k) := (1, B^k_{2,p_k,\boldsymbol{\xi}_k}(z_k), \ldots, B^k_{n_k,p_k,\boldsymbol{\xi}_k}(z_k))^\intercal$$

   comprising the auxiliary B-splines. The linear independence of the auxiliary B-splines is preserved [25].

2. Construct an $n_k \times n_k$ spline moment matrix of $\mathbf{P}_k(Z_k)$ as follows:

   $$\mathbf{G}_k := \mathbb{E}_{\mathbf{g}}[\mathbf{P}_k(Z_k)\mathbf{P}_k^\intercal(Z_k)].$$

   The matrix $\mathbf{G}_k$ exists because Z_k has finite moments up to order $2p_k$. Also, $\mathbf{G}_k$ is positive-definite, thus, invertible. As a result, there is a non-singular whitening matrix $\mathbf{W}_k \in \mathbb{R}^{n_k \times n_k}$ such that the factorization of the form

   $$\mathbf{W}_k^\intercal \mathbf{W}_k = \mathbf{G}_k^{-1} \text{ or } \mathbf{W}_k^{-1}\mathbf{W}_k^{-\intercal} = \mathbf{G}_k$$

   holds.

3. Using a whitening transformation, create an n_k-dimensional vector of orthonormalized B-splines

   $$\boldsymbol{\psi}_k(z_k; \mathbf{g}) = \mathbf{W}_k \mathbf{P}_k(z_k),$$

   consisting of uncorrelated components

   $$\psi^k_{i_k,p_k,\boldsymbol{\xi}_k}(z_k),\ i_k = 1, \ldots, n_k,\ k = 1, \ldots, N.$$

   Note that the invertibility of $\mathbf{G}_k$ does not uniquely determine $\mathbf{W}_k$. Indeed, there are several ways to choose $\mathbf{W}_k$ such that the condition described in Step 2 is satisfied. One prominent choice of $\mathbf{W}_k$, employed in this paper, is to invoke the Cholesky factoriziation, such that $\mathbf{G}_k = \mathbf{Q}_k\mathbf{Q}_k^\intercal$, yielding

   $$\mathbf{W}_k = \mathbf{Q}_k^{-1}.$$

The effectiveness of the three-step process is determined by reliable construction of a well-conditioned spline moment matrix $\mathbf{G}_k$ in Step 2. The spline moment matrix can be created analytically, or by a numerical method, such as a Gauss-type quadrature rule. Since the numerical integration does not include any potentially expensive output function evaluations, it can be implemented with an arbitrary level of precision. While it is achievable to do this using a single step of measure-consistent quadrature points and weights of adequately high order corresponding to the entire domain $[a_k, b_k]$, it is often more accurate to generate measure-consistent quadrature points and weights for each subinterval defined between successive, unique knots. In this case, the integral is split into several integrals, each operating on a polynomial of order $2p_k$ multiplied by a part of the PDF.

Appendix C. Optimal Latin Hypercube Sampling

Let $\mathbf{X} = (X_1, \ldots, X_N)^\intercal$ be N random variables, each with a uniform distribution over $[0, 1]$. Also, let $\mathbf{x}^{(t)} = (x_1^{(t)}, \ldots, x_N^{(t)})^\intercal$ be the tth realization of $\mathbf{X}$. In Latin hypercube sampling (LHS), for $k = 1, \ldots, N$, a sample $x_k^{(t)}$ is generated where each random variable X_k is stratified into L equal strata. Thus, a formula for LHS is as follows [35]:

$$x_k^{(t)} = \frac{\alpha_k(t-1) + u_k^{(t)}}{L},\ 1 \le k \le N,\ 1 \le t \le L,$$

where $\alpha_1(\cdot), \ldots, \alpha_N(\cdot)$ are uniform random permutation functions of $\{0, 1, \ldots, L-1\}$, $u_k^{(t)}$ is a realization of uniform distribution on $[0, 1)$, and all the $u_k^{(t)}$ and α_k are independent. For all $k = 1, \ldots, N$ and $j = 0, \ldots, L-1$, LHS satisfies

$$\#\left\{1 \le t \le L : \ \frac{j}{L} \le x_k^{(t)} < \frac{j+1}{L}\right\} = 1,$$

where $j = \alpha_i(t-1)$, and for each $k = 1, \ldots, N$, there is precisely one such that $t - 1 = \alpha_k^{-1}(j)$.

However, a better choice is desired to more uniformly distribute points in the domain. As one of the preferred options, the maximin LHS is to maximize the minimal distance among all pairs of points, resulting in an optimal Latin hypercube design. For $t_1, t_2 = 1, \ldots, L$, the maximin LHS is to maximize

$$\min_{t_1 \ne t_2} d(\mathbf{x}^{(t_1)}, \mathbf{x}^{(t_2)}) = \min_{t_1 \ne t_2} \left[\sum_{k=1}^{N} |x_k^{(t_1)} - x_k^{(t_2)}|^r\right]^{1/r}, \ r = 1 \text{ or } 2,$$

where $d(\mathbf{x}^{(t_1)}, \mathbf{x}^{(t_2)})$ is the general inter-point distance between any points $\mathbf{x}^{(t_1)}$ and $\mathbf{x}^{(t_2)}$.

References

[1] G. Taguchi. *Taguchi on robust technology development: bringing quality engineering upstream*. ASME Press series on international advances in design productivity. ASME Press, 1993.

[2] D. Lee and S. Rahman. Robust design optimization under dependent random variables by a generalized polynomial chaos expansion. *Structural and Multidisciplinary Optimization*, 63(5):2425–2457, 2021.

[3] W. Chen, J. Allen, K. Tsui, and F. Mistree. Procedure for robust design: Minimizing variations caused by noise factors and control factors. *Journal of Mechanical Design, Transactions of the ASME*, 118(4):478–485, 1996.

[4] X. Du and W. Chen. Towards a better understanding of modeling feasibility robustness in engineering design. *Journal of Mechanical Design*, 122(4):385–394, 2000.

[5] Z. Mourelatos and J. Liang. A methodology for trading-off performance and robustness under uncertainty. *Journal of Mechanical Design*, 128(4):856–863, 2006.

[6] K. Zaman, M. McDonald, S. Mahadevan, and L. Green. Robustness-based design optimization under data uncertainty. *Structural and Multidisciplinary Optimization*, 44(2):183–197, 2011.

[7] G. Park, T. Lee, H. Kwon, and K. Hwang. Robust design: an overview. *AIAA journal*, 44(1):181–191, 2006.

[8] W. Yao, X. Chen, W. Luo, M. Van Tooren, and J. Guo. Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. *Progress in Aerospace Sciences*, 47(6):450–479, 2011.

[9] X. Ren and S. Rahman. Robust design optimization by polynomial dimensional decomposition. *Structural and Multidisciplinary Optimization*, 48(1):127–148, 2013.

[10] T. Chatterjee, S. Chakraborty, and R. Chowdhury. A critical review of surrogate assisted robust design optimization. *Archives of Computational Methods in Engineering*, 26(1):245–274, 2019.

[11] N. Qiu, Z. Jin, J. Liu, L. Fu, Z. Chen, and N. H. Kim. Hybrid multiobjective robust design optimization of a truck cab considering fatigue life. *Thin-Walled Structures*, 162:107545, 2021.

[12] S. Sundaresan, K. Ishii, and D. R. Houser. A robust optimization procedure with variations on design variables and constraints. *Engineering Optimization+ A35*, 24(2):101–117, 1995.

[13] B. Huang and X. Du. Analytical robustness assessment for robust design. *Structural and Multidisciplinary Optimization*, 34(2):123–137, 2007.

[14] S. Lee, W. Chen, and B. Kwak. Robust design with arbitrary distributions using gauss-type quadrature formula. *Structural and Multidisciplinary Optimization*, 39(3):227–243, 2009.

[15] R. Jin, X. Du, and W. Chen. The use of metamodeling techniques for optimization under uncertainty. *Structural and Multidisciplinary Optimization*, 25(2):99–116, 2003.

[16] S. Rahman and R. Jahanbin. A spline dimensional decomposition for uncertainty quantification. *SIAM/ASA Journal on Uncertainty Quantification (accepted)*, 2022.

[17] R. Jahanbin and S. Rahman. Stochastic isogeometric analysis in linear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 364:112928, 2020.

[18] R. T. Marler and J. S. Arora. The weighted sum method for multi-objective optimization: new insights. *Structural and multidisciplinary optimization*, 41(6):853–862, 2010.

[19] M. Bashiri, A. Moslemi, and S. T. Akhavan Niaki. Robust multi-response surface optimization: a posterior preference approach. *International Transactions in Operational Research*, 27(3):1751–1770, 2020.

[20] W. Chen, M. M. Wiecek, and J. Zhang. Quality utility—a compromise programming approach to robust design. *Journal of Mechanical Design*, 121(2):179–187, 1999.

[21] S. Shin and B. R. Cho. Development of a sequential optimization procedure for robust design and tolerance design within a bi-objective paradigm. *Engineering Optimization*, 40(11):989–1009, 2008.

[22] V. T. Nha, S. Shin, and S. H. Jeong. Lexicographical dynamic goal programming approach to a robust design optimization within the pharmaceutical environment. *European Journal of Operational Research*, 229(2):505–517, 2013.

[23] M. Bhushan, S. Narasimhan, and R. Rengaswamy. Robust sensor network design for fault diagnosis. *Computers & Chemical Engineering*, 32(4-5):1067–1084, 2008.

[24] K. Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 2012.

[25] S. Rahman. A spline chaos expansion. *SIAM/ASA Journal on Uncertainty Quantification*, 8(1):27–57, 2020.

[26] V. Toropov, A. Filatov, and A. Polynkin. Multiparameter structural optimization using FEM and multipoint explicit approximations. *Structural and Multidisciplinary Optimization*, 6(1):7–14, 1993.

[27] D. Lee and S. Rahman. Practical uncertainty quantification analysis involving statistically dependent random variables. *Applied Mathematical Modelling*, 84:324–356, 2020.

[28] MATLAB. *version 9.7.0 (R2019b)*. The MathWorks Inc., Natick, Massachusetts, 2019.

[29] CREO. *version 4.0*. PTC, 2016.

[30] ABAQUS. *version 2019*. Dassault Systèmes Simulia Corp., 2019.

[31] J. S. Arora. *Introduction to optimum design–4th ed.* New York: McGraw-Hill, 2004.

[32] D. Lee and S. Rahman. Reliability-based design optimization under dependent random variables by a generalized polynomial chaos expansion. *Structural and Multidisciplinary Optimization*, 65(21), 2022.

[33] R. Stephens, A. Fatemi, R. R. Stephens, and H. Fuchs. *Metal fatigue in engineering*. Wiley-Interscience, 2000.

[34] X. Ren and S. Rahman. Stochastic design optimization accounting for structural and distributional design variables. *Engineering Computations*, 35(8):2654–2695, 2018.

[35] A. B. Owen. *Monte Carlo theory, methods and examples*. 2013.