Advanced Robust Optimization With Interval Uncertainty Using a Single-Looped Structure and Sequential Quadratic Programming

Jianhua Zhou

Mian Li[1]

e-mail: mianli@sjtu.edu.cn

University of Michigan-Shanghai Jiao Tong University Joint Institute,
Shanghai Jiao Tong University,
Shanghai 200240, China

Uncertainty is inevitable and has to be taken into consideration in engineering optimization; otherwise, the obtained optimal solution may become infeasible or its performance can degrade significantly. Robust optimization (RO) approaches have been proposed to deal with this issue. Most existing RO algorithms use double-looped structures in which a large amount of computational efforts have been spent in the inner loop optimization to determine the robustness of candidate solutions. In this paper, an advanced approach is presented where no optimization run is required for robustness evaluation in the inner loop. Instead, a concept of Utopian point is proposed and the corresponding maximum variable/parameter variation will be obtained just by performing matrix operations. The obtained robust optimal solution from the new approach may be conservative, but the deviation from the true robust optimal solution is small enough and acceptable given the significant improvement in the computational efficiency. Six numerical and engineering examples are tested to show the applicability and efficiency of the proposed approach, whose solutions and computational efforts are compared to those from a previously proposed double-looped approach, sequential quadratic program-robust optimization (SQP-RO). [DOI: 10.1115/1.4025963]

Keywords: robust optimization, single-looped structure, interval uncertainty, SQP

1 Introduction

Uncertainty is inevitable for real-world applications. Since Taguchi proposed the concept of RO [1], it has been an active research topic for a long time [2–5]. There are two types of uncertainties: aleatory and epistemic uncertainties. Aleatory uncertainty is the natural randomness in a system and can be modeled mathematically using probability theory, while epistemic uncertainty occurs due to incomplete knowledge or information [6] and research papers have been published on how to model and handle epistemic uncertainty [7–10]. Two categories of modeling uncertainties are considered in this area: one is probability-based [11–13] and another type is interval-based [14–16]. In the interval-based approaches, usually the nominal values of the uncertain parameters with their lower and upper bounds are used to calculate the largest possible deviations in the worst-case scenario. In our work, optimization problems with interval uncertainty are considered.

There have been methods proposed for robust linear problems [17–19] as well as RO algorithms for convex problems [20,21]. In addition, nonconvex RO programs are solved recently using the gradient-based approach [22,23] and using genetic algorithms (GAs) [24]. Most existing RO methods have the outer–inner or so-called double-looped structures where optimization performed in the inner loop is used to determine the robustness of candidate solutions, while the outer loop optimization is performed to find the optimal solution that satisfies robust requirements and all other constraints. This double-looped structure usually can damage the computational efficiency of RO approaches significantly.

To overcome the issue of the computational efficiency of double-looped RO approaches, several efficient methods have been proposed in the literature. A single-looped RO approach which is able to search the optimum point and the most probable failure points concurrently is proposed in Ref. [25]. However, this approach only considers probability-based uncertainties which may not be appropriate for the situations where required probabilities may not be available. In Ref. [26], linear, convex quadratic and nonconvex quadratic RO problems can be solved with the same complexity and efficiency as their deterministic counterparts using a worst-case analysis. In Ref. [27], a Bender's decomposition based approach, which is scalable and contains no nested structures, is proposed and proved to be computationally tractable. However, there were still subproblems that need to be solved by optimization runs. In Ref. [16], a novel approximation-assisted robust optimization approach using constraint cuts and online approximation for multi-objective problems is presented. It is a genetic algorithm based approach and the computational efficiency can still be a serious concern given the nature of GAs. In this regard, an efficient single-looped approach for nonlinear RO problems with interval uncertainty is still desirable.

In Ref. [23], a double-looped approach using SQP, named SQP-RO, is proposed to solve general nonlinear RO problems with interval uncertainties. This method is efficient compared to its deterministic counterpart and other GA-based RO approaches, in terms of the number of function evaluations. However, a large amount of computational efforts still have been spent in the inner loops to calculate the objective and constraint robustness measurements for candidate solutions, i.e., robustness indices. It is noted that the feasible domain of the inner optimization problem is the so-called uncertainty box, i.e., the hyper-box within which the

[1]Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received April 10, 2013; final manuscript received October 11, 2013; published online December 11, 2013. Assoc. Editor: Michael Kokkolaras.

uncertain parameters can vary. The uncertainty box is usually small (the parameter variation should not be very large in general). Notice that although the uncertainty box is not necessarily to be small, a smaller uncertainty box can indeed generate a solution with a smaller approximation error. In the proposed approach, to solve those robustness indices, first Taylor's expansion is applied to approximate those inner problems as quadratic optimization problems (i.e., QPs with box constraints).

There are lots of literatures dealing with quadratic optimization problems. In Ref. [28], optimality conditions for QPs have been discussed. In addition, methods for concave problems, such as extreme point ranking, cutting plane methods, convex envelopes, and reduction to bilinear programming and separable forms are summarized, and methods for indefinite QPs such as decomposition techniques are presented too. In Ref. [29], a canonical duality theory is proposed to solve quadratic minimization problems with the box or integer constraints. In Ref. [30], an efficient unconstrained minimization algorithm and gradient projection techniques are combined to solve convex quadratic minimization problems with many variables and box constraints. In Ref. [31], conjugate gradient projection methods and truncated projection methods are proposed, which can solve large-size quadratic problems subject to box constraints efficiently. In Ref. [32], a branch-and-cut approach based on several branching strategies is presented to solve nonconvex QPs with box constraints. In Ref. [33], based on a nonfinite heuristic algorithm, a finite algorithm is proposed to solve large and sparse QPs with box constraints. In Ref. [34], QPs and conic-QPs are discussed. However, in essence, the approaches mentioned above still use optimization runs to solve problems; thus using them to solve those inner problems in SQP-RO may not be helpful to improve the computational efficiency significantly.

In this paper, an advanced SQP-RO (A-SQP-RO) approach with a single-looped structure is proposed for nonlinear continuous RO problems with interval uncertainties. The concepts of the Utopian box associated with uncertain parameters and Utopian solutions are first proposed (please see the details later in Sec. 2.3). The Utopian solution of uncertain variable/parameter values used for robustness assessment can be easily and rapidly obtained by typical matrix operations instead of an optimization run, thus no optimization procedure is required for the inner loop. This solution will replace the nominal values of those uncertain variable/parameters in the objective and constraint functions, which will make the robust feasible domain shrink to some extent, possibly leading to a conservative robust solution. However, for constraints with single uncertain variable/parameter or whose Hessian matrix is already diagonal, the Utopian solution is exactly the true robust solution. Moreover, since the uncertainty box is small, the deviation from the true robust solution to the Utopian solution is rather small, as can be observed from six numerical and engineering test examples used in this work. Actually it is reasonable to believe that this deviation can be so small for large sparse matrices that it can be ignored.

The rest of this paper is organized as follows. Section 2 describes the background and terminologies used in this work, with the definitions of the proposed Utopian box and Utopian solution. Section 3 provides the procedure for the proposed approach with the discussion on its computational efficiency. Section 4 demonstrates six numerical and engineering examples with comparisons between their SQP-RO solutions and A-SQP-RO solutions, followed by the conclusions in Sec. 5.

2 Background and Terminology

In this section, the background of SQP-RO and the matrix decomposition method are briefly introduced, and the definition of the Utopian solution is provided in detail.

2.1 SQP-RO Approach. A robust optimization problem with interval uncertainty can be formulated as in the following equation:

$$\begin{aligned}
\min_{\boldsymbol{x}} \quad & f(\boldsymbol{x},\boldsymbol{p}) \\
\text{s.t.} \quad & g_j(\boldsymbol{x},\boldsymbol{p}) \le 0, \quad j=1,\dots,J \\
& \mathbf{lb} \le \boldsymbol{x} \le \mathbf{ub} \\
& \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \boldsymbol{p} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p}
\end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is the vector of bounded design variables, $\boldsymbol{p}$ represents the uncertain parameters whose values can vary within a symmetric hyper box $\boldsymbol{p}_0 \pm \Delta\boldsymbol{p}$ (the asymmetric interval case can be addressed by adding additional upper or lower bounds), $\boldsymbol{p}_0$ is the vector of the nominal values of $\boldsymbol{p}$, and $\Delta\boldsymbol{p}$ represents the half interval. f and g_j, $j=1,\dots, J$ are the objective and constraint functions, respectively. The goal of Eq. (1) is to find an optimal robust solution such that when $\boldsymbol{p}$ varies, the variation in the objective value of this optimal solution is still within the predefined acceptable range and none of the constraints is violated.

In Ref. [23], a novel double-looped SQP-RO has been proposed to solve the problem in Eq. (1). In this approach, the objective and constraint robustness indices are defined as the maximum variation in the objective and constraint functions, respectively, as shown in Eq. (2), where Δf_0 is a presumed acceptable variation range for the objective function and is usually determined by the designer

$$\begin{aligned}
& \eta_f = \max_{\boldsymbol{p}} \left| \frac{f(\boldsymbol{x},\boldsymbol{p}) - f(\boldsymbol{x},\boldsymbol{p}_0)}{\Delta f_0} \right| \le 1 \\
& \eta_{g,j} = \max_{\boldsymbol{p}} g_j(\boldsymbol{x},\boldsymbol{p}) \le 0 \\
& s.t. \quad \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \boldsymbol{p} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p}
\end{aligned} \tag{2}$$

As shown in Eq. (2), the objective and constraint robustness indices are solved in the inner problem in order that the variables/parameters' uncertainty that leads to the max objective and constraint variations can be identified. For a robust solution, its objective robustness index should always be less than one and its constraint robustness indices should always be no larger than zero. Then those obtained variables/parameters' values replace their nominal values in the original constraints, as shown in Eq. (3). After that, all the constraints in Eq. (3) will be linearized so that SQP can be applied to solve for a step size until the stopping criteria are met. Detailed information about SQP-RO can be found in Ref. [23]

$$\begin{aligned}
& \min_{\boldsymbol{x}} \quad f(\boldsymbol{x},\boldsymbol{p}_0) \\
& \text{s.t.} \quad g_j(\boldsymbol{x},\boldsymbol{p}_{j\max}) \le 0, \; j=1,\dots,J \\
& \qquad \left| \frac{f(\boldsymbol{x},\boldsymbol{p}_{\max}) - f(\boldsymbol{x},\boldsymbol{p}_0)}{\Delta f_0} \right| - 1 \le 0 \\
& \boldsymbol{p}_{j\max} = \arg\left\{ \max_{\boldsymbol{p}} g_j(\boldsymbol{x},\boldsymbol{p}) | \forall \boldsymbol{x}, \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \boldsymbol{p} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p} \right\}; \\
& \boldsymbol{p}_{\max} = \arg\left\{ \max_{\boldsymbol{p}} \left[\frac{f(\boldsymbol{x},\boldsymbol{p}) - f(\boldsymbol{x},\boldsymbol{p}_0)}{\Delta f_0} \right]^2 | \forall \boldsymbol{x}, \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \boldsymbol{p} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p} \right\} \\
& \qquad \boldsymbol{x}_{lb} \le \boldsymbol{x} \le \boldsymbol{x}_{ub}
\end{aligned} \tag{3}$$

Notice that each inner problem in Eq. (2) is defined on a small interval or box. Since this uncertainty box $[-\Delta\boldsymbol{p}, \Delta\boldsymbol{p}]$ is relatively small, Taylor's expansion can be utilized and Eq. (2) turns into a quadratic one, as shown in Eq. (4). The tail error induced by Taylor's expansion is measured by $O(d\boldsymbol{p}^3)$ when the function is expanded to the second order or $O(d\boldsymbol{p}^2)$ when expanded to the first order. As $d\boldsymbol{p}$ usually takes or converges to a very small positive value during the optimization procedure (e.g., 10^{-1} or 10^{-5} in this paper), this tail error could be much smaller. In this sense, Taylor's expansion can be considered accurate enough.

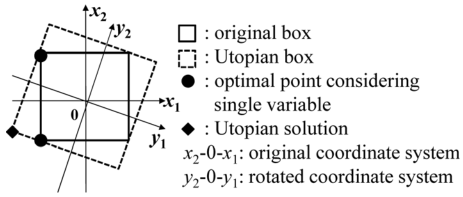

Fig. 1 Illustration of Utopian box and Utopian solution

$$\begin{aligned}\max|\Delta f(d\boldsymbol{p};\boldsymbol{x},\boldsymbol{p}_0)| &= |f(\boldsymbol{x},\boldsymbol{p}_0+d\boldsymbol{p})-f(\boldsymbol{x},\boldsymbol{p}_0)|\\ &\cong \left|f(\boldsymbol{x},\boldsymbol{p}_0)+\mathbf{c}_p^{\mathrm{T}}d\boldsymbol{p}+\frac{1}{2}d\boldsymbol{p}^{\mathrm{T}}\mathbf{H}_p d\boldsymbol{p}-f(\boldsymbol{x},\boldsymbol{p}_0)\right|\\ &= \left|\frac{1}{2}d\boldsymbol{p}^{\mathrm{T}}\mathbf{H}_p d\boldsymbol{p}+\mathbf{c}_p^{\mathrm{T}}d\boldsymbol{p}\right|\\ s.t. \quad & -\Delta\boldsymbol{p}\le d\boldsymbol{p}\le\Delta\boldsymbol{p}\end{aligned} \tag{4a}$$

$$\begin{aligned}\max \quad & g_j(d\boldsymbol{p};\boldsymbol{x},\boldsymbol{p}_0)\cong g_j(\boldsymbol{x},\boldsymbol{p}_0)+\mathbf{c}_{jp}d\boldsymbol{p}+\tfrac{1}{2}d\boldsymbol{p}^{\mathrm{T}}\mathbf{H}_{jp}d\boldsymbol{p}\\ s.t. \quad & -\Delta\boldsymbol{p}\le d\boldsymbol{p}\le\Delta\boldsymbol{p}\end{aligned} \tag{4b}$$

where $\mathbf{c}$ and $\mathbf{H}$ are gradient and Hessian matrix of f or g_j with respect to $\boldsymbol{p}$, respectively. Note that a design variable with interval uncertainty $[-\Delta\boldsymbol{x}, \Delta\boldsymbol{x}]$ can be treated as a parameter with nominal value $\boldsymbol{p}_0=\boldsymbol{x}^{(k)}$ in each iteration of SQP.

In this double-looped SQP-RO, one of the major problems is how to solve the inner QP problems with box constraints in Eq. (4) efficiently, or even remove them if possible.

2.2 Matrix Decomposition Method for QPs Subject to Box Constraints. In Ref. [28], a decomposition method is introduced to solve QPs as shown in the following equation:

$$\begin{aligned}\max/\min \quad & f(\boldsymbol{x})=\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\mathbf{H}\boldsymbol{x}+\mathbf{c}^{\mathrm{T}}\boldsymbol{x}\\ s.t. \quad & \mathbf{A}\boldsymbol{x}\le\mathbf{b}\end{aligned} \tag{5}$$

In Eq. (5), "max/min" means that this decomposition approach can be applied no matter it is to maximize or minimize the objective function. The matrix $\mathbf{H}$ can be decomposed as $\mathbf{H}=\mathbf{U}^{\mathrm{T}}\mathbf{D}\mathbf{U}$, with $\mathbf{U}^{\mathrm{T}}\mathbf{U}=\mathbf{I}$ and $\mathbf{D}$ a diagonal matrix. This can be easily realized for the symmetric Hessian matrix $\mathbf{H}$: calculate the eigenvalues $\lambda_1, \lambda_2,\ldots, \lambda_k$ (k is the dimension of $\boldsymbol{x}$) and the corresponding normalized eigenvectors $\boldsymbol{u}_1, \boldsymbol{u}_2,\ldots, \boldsymbol{u}_k$ of $\mathbf{H}$. Let $\mathbf{D}=\mathrm{diag}(\lambda_1, \lambda_2,\ldots, \lambda_k)$ and $\mathbf{U}=[\boldsymbol{u}_1, \boldsymbol{u}_2,\ldots, \boldsymbol{u}_k]$, then $\mathbf{H}=\mathbf{U}^{\mathrm{T}}\mathbf{D}\mathbf{U}$ with $\mathbf{U}^{\mathrm{T}}\mathbf{U}=\mathbf{I}$. As a result we have the following derivation:

$$\begin{aligned}\max/\min f(\boldsymbol{x}) &= \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\mathbf{H}\boldsymbol{x}+\mathbf{c}^{\mathrm{T}}\boldsymbol{x}\\ &= \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{D}\mathbf{U}\boldsymbol{x}+\mathbf{c}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{U}\boldsymbol{x}\\ &= \frac{1}{2}(\mathbf{U}\boldsymbol{x})^{\mathrm{T}}\mathbf{D}(\mathbf{U}\boldsymbol{x})+(\mathbf{U}\mathbf{c})^{\mathrm{T}}(\mathbf{U}\boldsymbol{x})\\ s.t. \quad & \mathbf{A}(\mathbf{U}^{\mathrm{T}}\mathbf{U})\boldsymbol{x}\le\mathbf{b}\end{aligned} \tag{6}$$

Let $\boldsymbol{y}=\mathbf{U}\boldsymbol{x}$ and $\boldsymbol{v}=\mathbf{U}\mathbf{c}$, Eq. (6) can be rewritten as

$$\begin{aligned}\max/\min \quad & f(\boldsymbol{y})=\frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\mathbf{D}\boldsymbol{y}+\boldsymbol{v}^{\mathrm{T}}\boldsymbol{y}\\ s.t. \quad & \mathbf{A}\mathbf{U}^{\mathrm{T}}\boldsymbol{y}\le\mathbf{b}\end{aligned} \tag{7}$$

In this way, the variables of the objective function in Eq. (5) can be decoupled. In fact, $\mathbf{U}$ plays the role of a rotation matrix. In section 3, it can be seen that by introducing the concept of the Utopian box, the optimization problem in Eq. (7) can be turned into a nonoptimization problem that can be solved by performing matrix operations.

2.3 Utopian Box and Utopian Solution to QPs Subject to Box Constraints. First, we define the Utopian box and Utopian solution.

(1) Utopian box: the box that is perpendicular to the axis of the rotated coordinate system and covers all the vertices of the original uncertainty box. Obviously, the Utopian box can be larger than the original uncertainty box. Figure 1 illustrates a Utopian box in the 2D case.

(2) Utopian solution: the optimal uncertain parameter values within or on the boundary of the Utopian box which gives the maximum function variation. The optimal point considering a single variable y_1 or y_2 refers to the optimal y_1 or y_2 value in the rotational coordinate system that individually makes the corresponding decomposed term in Eq. (7) take the maximum value. Note that this point is not always located within or on the boundary of the original box, as can be shown later in Sec. 3.

In order to determine the Utopian box, the optimal point in each dimension in the rotated coordinate system has to be found. The extreme vertices of the Utopian box and the Utopian solution can be determined as follows.

Since this part provides a general discussion on the QPs subject to box constraints, we use $\boldsymbol{x}$ instead of d$\boldsymbol{p}$ to present variables of the QPs in this subsection (d$\boldsymbol{p}$ will be used later specifically in our proposed approach discussed in Secs. 3 and 4). Usually it can be assumed that lower and upper bounds of $\boldsymbol{x}$ are of the same absolute value $\boldsymbol{x}_{\mathrm{b}}$ (i.e., $\boldsymbol{x}_{\mathrm{b}}$ is a vector with all elements positive), that is, the uncertainty box in RO is symmetric, and it is easy to convert an asymmetric box to a symmetric one. Then we may rewrite $\boldsymbol{x}$ as a Hadamard product: $\boldsymbol{x}=\boldsymbol{\alpha}\boldsymbol{o}\boldsymbol{x}_{\mathrm{b}}=[\alpha_1x_{\mathrm{b}1}, \alpha_2x_{\mathrm{b}2},\ldots,\alpha_kx_{\mathrm{b}k}]$, where $\alpha_j\in[-1, 1]$. We have $\boldsymbol{y}=\mathbf{U}\boldsymbol{x}=\mathbf{U}(\boldsymbol{\alpha}\boldsymbol{o}\boldsymbol{x}_{\mathrm{b}})$ and its elements $y_i=\sum_{j=1}^{k}u_{ij}x_j=\sum_{j=1}^{k}u_{ij}\alpha_jx_{bj}=y_i(\alpha_1,\alpha_2,...,\alpha_k)$, which indicates that y_i is a function of variables $\alpha_1, \alpha_2,\ldots, \alpha_k$. By substituting this Hadamard product into Eq. (7), Eq. (7) can be further decomposed as shown in the following equation:

$$\begin{aligned}\max/\min f(\boldsymbol{y}) &\cong \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\mathbf{D}\boldsymbol{y}+\mathbf{v}^{\mathrm{T}}\boldsymbol{y}\\ &= \frac{1}{2}\sum_{i=1}^{k}(\lambda_iy_i^2+2v_iy_i)\\ &= \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(y_i+\frac{v_i}{\lambda_i}\right)^2-\sum_{i=1}^{k}\frac{v_i}{2\lambda_i}^2\\ &= \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{j=1}^{k}u_{ij}\alpha_jx_{bj}+\frac{v_i}{\lambda_i}\right)^2-\frac{1}{2}\sum_{i=1}^{k}\frac{v_i^2}{\lambda_i}\\ \text{s.t.} \quad & \alpha_j\in[-1,1]\end{aligned} \tag{8}$$

Given the definiteness of the Hessian matrix $\mathbf{H}$ in Eq. (4), actually Eqs. (4a) and (4b) can possibly take different types of QP forms represented in Eq. (8), in terms of maximization or minimization, as discussed following.

2.3.1 Convex Maximization (or Concave Minimization) Problem. When the Hessian matrix $\mathbf{H}$ is positive definite (pd) or positive semi-definite (psd), all λ_i's are nonnegative. In this case, Eqs. (4a) and (4b) will become convex maximization problems. When $\mathbf{H}$ is negative definite (nd) or negative semi-definite (nsd), Eq. (4a) will become a concave minimization problem which can be easily transformed into a convex maximization one.

When $\lambda_i>0$, to ensure the maximum value of $f(\boldsymbol{y})$, each squared term $(y_i+(v_i/\lambda_i))^2=\left(\sum_{j=1}^{k}u_{ij}\alpha_jx_{bj}+(v_i/\lambda_i)\right)^2$ in Eq. (8) should be maximized. Given the known values of v_i, λ_i, x_{bj}, and u_{ij}, i,

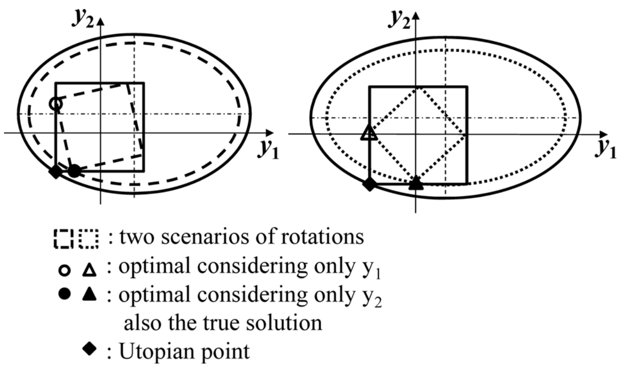

Fig. 2 Illustration of true solution and Utopian solution

$j = 1,\ldots,k$, only the $\boldsymbol{\alpha}$ vector in each squared term $\left(\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i)\right)^2$ needs to be determined based on the sign of v_i/λ_i (there are all k squared terms):

(1) if $v_i/\lambda_i > 0$, i.e. $v_i > 0$, let $\alpha_j = \mathrm{sign}(u_{ij}) = \mathrm{sign}(v_i)\mathrm{sign}(u_{ij})$, for $j = 1,\ldots, k$;
(2) if $v_i/\lambda_i < 0$, i.e. $v_i < 0$, let $\alpha_j = -\mathrm{sign}(u_{ij}) = \mathrm{sign}(v_i)\mathrm{sign}(u_{ij})$, for $j = 1,\ldots, k$;
(3) if $v_i/\lambda_i = 0$, i.e. $v_i = 0$, still we let $\alpha_j = \mathrm{sign}(u_{ij})$, for $j = 1,\ldots, k$.

When $\lambda_i = 0$ for some i, let $\alpha_j = \mathrm{sign}(u_{ij})$ for $v_i > 0$ and $\alpha_j = -sign(u_{ij})$ for $v_i < 0$, that is, $\alpha_j = \mathrm{sign}(v_i)\mathrm{sign}(u_{ij})$, for $j = 1,\ldots, k$.

As a summary, define $sign(x) = \begin{cases} +1, x \geq 0 \\ -1, x < 0 \end{cases}$, then each term $(\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i))^2$ can achieve its maximum value if we let $\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} = sign(v_i)\sum_{j=1}^{k} u_{ij} sign(u_{ij}) x_{bj} = sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj}$ which gives the maximum y_i, i.e.,$y_{i\max} = sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj}$. The Utopian point is located at $(y_{1\max}, y_{2\max},\ldots, y_{k\max})$, as illustrated in Fig. 1. To obtain the location of this Utopian point $\boldsymbol{x}^*$ in the original coordination system, we just need to solve

$$\mathbf{U}\boldsymbol{x} = \boldsymbol{y}_{\max} \tag{9}$$

where $\boldsymbol{y}_{\max} = [y_{1\max}, y_{2\max},\ldots, y_{k\max}]$. Since $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$, by multiplying $\mathbf{U}^{\mathrm{T}}$ on both sides of Eq.(9), it is easily obtained that

$$\boldsymbol{x}^* = \mathbf{U}^{\mathrm{T}}\boldsymbol{y}_{\max} \tag{10}$$

When $\mathbf{H}$ is nd or nsd, for the minimization problem, we just need to set $y_{i\max} = -sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj}$.

Still we may write $\boldsymbol{x}^*$ in Hadamard product as $\boldsymbol{x}^* = \boldsymbol{\alpha}^* o \boldsymbol{x}_{\mathrm{b}} = [\alpha_1^* x_{\mathrm{b1}}, \alpha_2^* x_{\mathrm{b2}},\ldots, \alpha_k^* x_{\mathrm{b}k}]^{\mathrm{T}}$ with $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*,\ldots, \alpha_k^*]^{\mathrm{T}}$ and $\alpha_j^* = x_j^*/x_{\mathrm{b}j}$. We would like to monitor $\boldsymbol{\alpha}^*$ value here because it gives us intuitive information about the location of the Utopian point (whether it is located within the original feasible domain or not, and if not, how far away it can be), as can be shown in the test examples. Besides, since this proposed method could be conservative, in which case $\boldsymbol{\alpha}^*$ value might be used in the future for error analysis.

2.3.2 Concave Maximization (or Convex Minimization) Problem. When $\mathbf{H}$ in Eq. (4*b*) is nd or nsd, it will become a concave maximization problem, which is also equivalent to a convex minimization problem.

When $\lambda_i < 0$, to ensure the maximum value of $f(\boldsymbol{y})$, each squared term $\left(\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i)\right)^2$ in Eq. (8) has to be minimized. In this case, obviously zero is the smallest value, so first we should determine if zero can be reached. Similar to the case of convex maximization, if $v_i/\lambda_i < 0, i.e.\, v_i > 0$, we set $\alpha_j = -sign(u_{ij}) = sign(v_i)sign(u_{ij})$, and if $v_i/\lambda_i \leq 0, i.e.\, v_i \geq 0$, we set $\alpha_j = sign(u_{ij}) = sign(v_i)sign(u_{ij})$. Then we compare the signs of $sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj} + v_i/\lambda_i$ with that of v_i/λ_i:

(1) if $sign(sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj} + v_i/\lambda_i) = sign(v_i/\lambda_i)$, it implies that $(\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i))^2$ cannot reach zero; then we set $\sum_{j=1}^{k} u_{ij} x_{bj}\alpha_j = sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj}$;
(2) if $sign(sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj} + (v_i/\lambda_i)) \neq sign(v_i/\lambda_i)$ which means zero can be reached, then we set $y_i + v_i/\lambda_i = \sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i) = 0$, i.e., $y_i = \sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} = -(v_i/\lambda_i)$.

If $\lambda_i = 0$, we set $\sum_{j=1}^{k} u_{ij} x_{bj}\alpha_j = sign(v_i)\sum_{j=1}^{k} |u_{ij}| x_{bj}$.

As a summary, each term $(\sum_{j=1}^{k} u_{ij}\alpha_j x_{bj} + (v_i/\lambda_i))^2$ can achieve its maximum value if we let

$$y_{i\max} = \begin{cases} -\dfrac{v_i}{\lambda_i}, \text{if } \lambda_i \neq 0 \text{ and } sign\left(sign(v_i)\displaystyle\sum_{j=1}^{k} |u_{ij}| x_{bj} + \frac{v_i}{\lambda_i}\right) \neq sign\left(\dfrac{v_i}{\lambda_i}\right) \\ sign(v_i)\displaystyle\sum_{j=1}^{k} |u_{ij}| x_{bj}, \text{ other cases} \end{cases} \tag{11}$$

By solving $\mathbf{U}\boldsymbol{x} = \boldsymbol{y}_{\max}$, the Utopian solution is obtained as $\boldsymbol{x}^* = \mathbf{U}^{\mathrm{T}}\boldsymbol{y}_{\max}$.

When $\mathbf{H}$ is pd or psd, for the minimization problem just set $y_{i\max}$ in Eq. (11) as

$$y_{i\max} = \begin{cases} -\dfrac{v_i}{\lambda_i}, \text{if } \lambda_i \neq 0 \; sign\left(-sign(v_i)\displaystyle\sum_{j=1}^{k} |u_{ij}| x_{bj} + \frac{v_i}{\lambda_i}\right) \neq sign\left(\dfrac{v_i}{\lambda_i}\right) \\ -sign(v_i)\displaystyle\sum_{j=1}^{k} |u_{ij}| x_{bj}, \text{other cases} \end{cases} \tag{12}$$

2.3.3 Indefinite Problem. When $\mathbf{H}$ is indefinite, the function f can be divided into two parts: those with nonnegative λ_i's $(i = 1,\ldots, m)$ and those with negative λ_i's $(i = m+1,\ldots, k)$, as shown in the following equation:

$$\begin{aligned} f(\boldsymbol{y}) &= \frac{1}{2}\boldsymbol{y}^{\mathbf{T}}\mathbf{D}\boldsymbol{y} + \mathbf{v}^{\mathbf{T}}\boldsymbol{y} \\ &= \frac{1}{2}\sum_{i=1}^{k}(\lambda_i y_i^2 + 2v_i y_i) \\ &= \left(\frac{1}{2}\sum_{i=1}^{m}\left(\lambda_i y_i^2 + 2\frac{v_i}{\lambda_i}\right)\right) + \left(\frac{1}{2}\left(\sum_{i=m+1}^{k}\lambda_i y_i^2 + 2\frac{v_i}{\lambda_i}\right)\right) \\ &:= Q_+(\boldsymbol{y}) + Q_-(\boldsymbol{y}) \\ y_i &= y_i(\alpha_1, \alpha_2, \ldots, \alpha_k), \alpha_j \in [-1, 1] \end{aligned} \tag{13}$$

If f is to be maximized, for those with nonnegative λ_i's $(i = 1,\ldots, m)$, the same procedure as that for convex maximization is applied; while for those with negative λ_i's $(i = m+1,\ldots, k)$, the same procedure as that for concave maximization can be applied.

On the contrary, if f is to be minimized, the same procedure as convex minimization is applied for those with nonnegative λ_i's $(i = 1,\ldots, m)$ while the same procedure as concave minimization is applied for those with negative λ_i's $(i = m+1,\ldots, k)$.

Note that for these three categories of problems, when the orders of magnitudes of uncertain variables/parameters are not of the same level, $\boldsymbol{x}_{\mathrm{b}}$'s should be normalized first.

Figure 2 illustrates a 2D case of the Utopian box and Utopian solution under two rotation scenarios. The ellipse in the solid line

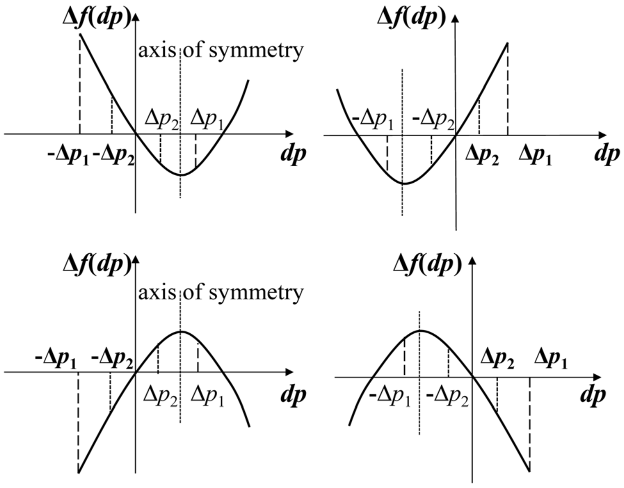

Fig. 3 Shapes of a parabolic function

represents the contour of the robustness index for the Utopian solution, while the ellipse in the dashed line is for that of the true optimal solution. The ratio (square of this ratio value) of the length of these two (ellipses in solid and dashed lines) semi-major axes can be used to measure the deviation of the Utopian solution to the true solution. It can be easily inferred that the smaller the rotation angle and the farther the centroid of the ellipse is away from the origin, the closer this ratio is to 1, and thus the closer this Utopian solution is to the true solution.

3 Advanced Sequential Quadratic Programming Approach for Robust Optimization (A-SQP-RO)

In this section, the proposed A-SQP-RO algorithm will be described in detail. The inner QPs in the proposed single-looped A-SQP-RO are solved by performing matrix operations. We first discuss the approach for objective robustness, followed by the discussion on feasibility robustness. After that the complete single-looped A-SQP-RO approach is described step by step.

3.1 Approach to Solve the Objective Robustness Index. First we discuss the case with a single uncertain variable/parameter; and then the method is extended for the case of multiple uncertain variables/parameters.

3.1.1 Solve the Objective Robustness Index With Single Uncertain Variable/Parameter. In this case, we have to obtain the maximum $|\Delta f\,(dp;\,x,\,p_0)|$ value as shown in Eq. (4*a*) with $\boldsymbol{p}$ being a scalar. To make it clear, it is rewritten as in the following equation:

$$\begin{aligned}\max\ |\Delta f(dp;x,p_0)| &= \left|\frac{1}{2}f_p''(x,p_0)dp^2 + f_p'(x,p_0)dp\right| \\ s.t.\quad & -\Delta p \le dp \le \Delta p\end{aligned} \tag{14}$$

In Eq. (14), Δf is a parabolic function with respect to dp that passes through the point (0, 0). The parabola may be convex or concave depending on the sign of the twice differentiation, and the symmetric axis of Δf may be to the left or right of the vertical axis, as shown in Fig. 3.

The solution to Eq. (14), called the extreme point, needs to be determined. It can be easily proved that the maximum value of the objective function in Eq. (14) within the bound of $[-\Delta p,\ \Delta p]$ occurs at $-\Delta p$ (or Δp) when $-f_p'(x,p_0)/f_p''(x,p_0) \ge 0$ (or ≤ 0). So searching for the objective robustness index becomes a maximization problem when the quadratic problem is convex or a minimization one if it is concave, which can be extended to the case where there are multiple uncertain variables/parameters. A concave minimization problem is actually equivalent to a convex maximization one. Thus we can have

$$p_{\max} = \begin{cases} p_0 - \Delta p, & \text{if } \dfrac{f_p'(x,p_0)}{f_p''(x,p_0)} < 0 \\[2ex] p_0 + \Delta p, & \text{if } \dfrac{f_p'(x,p_0)}{f_p''(x,p_0)} \ge 0 \end{cases} \tag{15}$$

When $f_p''(x,p_0)$ in Eq. (14) is zero, it becomes a linear equation and the solution is $-\Delta p$ or Δp depending on the sign of $f_p'(x,p_0)$. In fact, $dp_{\max} = sign(f_p'(x,p_0))\cdot\Delta p$.

3.1.2 Solve the Objective Robustness Index With Multiple Uncertain Variables/Parameters As discussed in Sec. 2.2, we have to find the maximum $|\Delta f\ (d\boldsymbol{p};\ \boldsymbol{x},\ \boldsymbol{p}_0)|$ value as shown in Eq. (4*a*) which can be decomposed to

$$\begin{aligned}\max|\Delta f(d\boldsymbol{p})| &= \left|\frac{1}{2}d\boldsymbol{p}^{\mathbf{T}}\mathbf{H}_p d\boldsymbol{p} + \mathbf{c}_p^{\mathbf{T}}d\boldsymbol{p}\right| \\ &= \left|\frac{1}{2}d\boldsymbol{p}^{\mathbf{T}}\mathbf{U}^{\mathbf{T}}\mathbf{D}\mathbf{U}d\boldsymbol{p} + \mathbf{c}^{\mathbf{T}}\mathbf{U}^{\mathbf{T}}\mathbf{U}d\boldsymbol{p}\right| \\ &= \left|\frac{1}{2}(\mathbf{U}d\boldsymbol{p})^{\mathbf{T}}\mathbf{D}(\mathbf{U}d\boldsymbol{p}) + (\mathbf{U}\mathbf{c})^{\mathbf{T}}(\mathbf{U}d\boldsymbol{p})\right| \\ s.t.\quad & \mathbf{lb} \le \mathbf{U}^{\mathbf{T}}(\mathbf{U}d\boldsymbol{p}) \le \mathbf{ub}\end{aligned} \tag{16}$$

We assume $|\mathbf{lb}_i| = |\mathbf{ub}_i| = \Delta \boldsymbol{p}_i$, $i = 1,\ldots,\ k$, where $\boldsymbol{x}_i$ indicates the ith component of vector $\boldsymbol{x}$, and let $\mathrm{d}\boldsymbol{p} = \boldsymbol{\alpha}o\Delta\boldsymbol{p} = [\alpha_1\Delta p_1,\ \alpha_2\Delta p_2,\ldots,\ \alpha_k\Delta p_k]$ where $\alpha_i\in[-1,\ 1]$, $\boldsymbol{y} = \mathbf{U}\mathrm{d}\boldsymbol{p} = \mathbf{U}(\boldsymbol{\alpha}o\Delta\boldsymbol{p})$, and $\boldsymbol{v} = \mathbf{U}\mathbf{c}_p$, then Eq. (16) can be rewritten as

$$\begin{aligned}\max|\Delta f(\boldsymbol{y})| &= \left|\frac{1}{2}\boldsymbol{y}^{\mathbf{T}}\mathbf{D}\boldsymbol{y} + \boldsymbol{v}^{\mathbf{T}}\boldsymbol{y}\right| \\ &= \left|\frac{1}{2}\sum_{i=1}^{k}\lambda_i y_i^2 + \sum_{i=1}^{k}v_i y_i\right| \\ &= \left|\frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(y_i + \frac{v_i}{\lambda_i}\right)^2 - \sum_{i=1}^{k}\frac{{v_i}^2}{2\lambda_i}\right| \\ &= \left|\frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{j=1}^{k}u_{ij}\alpha_j\Delta p_j + \frac{v_i}{\lambda_i}\right)^2 - \frac{1}{2}\sum_{i=1}^{k}\frac{{v_i}^2}{\lambda_i}\right| \\ s.t.\quad & \alpha_j \in [-1,1]\end{aligned} \tag{17}$$

As discussed previously, if $\mathbf{H}$ is pd or psd, then we have to solve

$$\begin{aligned}\max\quad & \Delta f(\boldsymbol{y}) = \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{j=1}^{k}u_{ij}\alpha_j\Delta p_j + \frac{v_i}{\lambda_i}\right)^2 - \frac{1}{2}\sum_{i=1}^{k}\frac{{v_i}^2}{\lambda_i} \\ s.t.\quad & \alpha_j \in [-1,1]\end{aligned} \tag{18}$$

which is a convex maximization problem. If $\mathbf{H}$ is nd or nsd, we have to solve the following concave minimization problem in Eq. (19).

$$\begin{aligned}\min\quad & \Delta f(\boldsymbol{y}) = \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{j=1}^{k}u_{ij}\alpha_j\Delta p_j + \frac{v_i}{\lambda_i}\right)^2 - \frac{1}{2}\sum_{i=1}^{k}\frac{{v_i}^2}{\lambda_i} \\ s.t.\quad & \alpha_j \in [-1,1]\end{aligned} \tag{19}$$

If $\mathbf{H}$ is indefinite, the objective function Δf has to be separated into two terms as shown in Eq. (13). Let $y_i = sign(v_i)\sum_{j=1}^{k}|u_{ij}|\Delta p_j$ for terms in $Q_+(\boldsymbol{y})$ and $y_i = -sign(v_i)\sum_{j=1}^{k}|u_{ij}|\Delta p_j$ for those in $Q_-(\boldsymbol{y})$. Then we could calculate $\Delta f(\boldsymbol{y})$. If $\Delta f(\boldsymbol{y}) \ge 0$

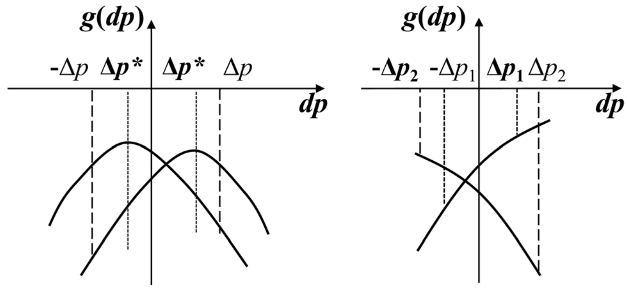

Fig. 4 Concave parabola for constraints

which implies that $Q_+(\boldsymbol{y})$ dominates $Q_-(\boldsymbol{y})$, then the maximization case for the indefinite problem discussed in Sec. 2.3 should be applied; otherwise, the minimization case for the indefinite problem should be applied. By solving those problems correspondingly, the optimal solution $\Delta\boldsymbol{p}_{\max}$ will be obtained, and $\boldsymbol{p}_{\max} = \boldsymbol{p}_0 + \Delta\boldsymbol{p}_{\max}$.

In addition, the variation in the objective function can be linearized as

$$f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_0) \cong f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}) + \mathbf{c}_{\max}\boldsymbol{d} - (f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0) + \mathbf{c}_0\boldsymbol{d}) \quad (20)$$

Thus, the objective robust requirement, i.e., the variation of the objective function due to uncertainty should always be less than the acceptable objective variation Δf_0, can lead to the conclusion stated in Eq. (21*b*) step by step

$$\begin{aligned} &\left|f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_0)\right| \le \Delta f_0 \\ \Leftrightarrow &-\Delta f_0 \le f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}_0) \le \Delta f_0 \\ \Leftrightarrow &-\Delta f_0 \le f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0) + \mathbf{c}_{\max}\boldsymbol{d} - \mathbf{c}_0\boldsymbol{d} \le \Delta f_0 \end{aligned} \quad (21a)$$

$$\Leftrightarrow$$

$$\begin{aligned} &(\mathbf{c}_{\max} - \mathbf{c}_0)\boldsymbol{d} \le \Delta f_0 - (f_m - f_0) \\ &\text{and} \quad -(\mathbf{c}_{\max} - \mathbf{c}_0)\boldsymbol{d} \le \Delta f_0 + (f_m - f_0) \\ &\text{where} \quad \mathbf{c}_{\max} = f'(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}), \mathbf{c}_0 = f'(\boldsymbol{x}^{(k)},\boldsymbol{p}_0) \\ &f_m = f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}), f_0 = f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0) \end{aligned} \quad (21b)$$

3.2 Approach to Solve the Constraint Robustness Index. In this section, the proposed approach to solve the constraint robustness indices is presented. Again, the case for the single uncertain variable/parameter is discussed first, followed by the case of multiple uncertain variables/parameters.

3.2.1 Solve the Constraint Robustness Index With Single Uncertain Variable/Parameter. Here we need to find the maximum $g(dp; x, p_0)$ value as shown in Eq. (4*b*) with $\boldsymbol{p}$ being a scalar. Eq. (4*b*) is rewritten as the following equation:

$$\begin{aligned} \max \quad & g_j(dp;x,p_0) = \frac{1}{2}g''_{jp}(x,p_0)dp^2 + g'_{jp}(x,p_0)dp + g_j(x,p_0) \\ s.t. \quad & -\Delta p \le dp \le \Delta p \end{aligned} \quad (22)$$

In Eq. (22) g_j is also a parabolic function with respect to dp; however, this function does not pass through the point (0, 0) and the corresponding discussion is more complex than that for the objective robustness index.

When g_j is convex, Eq. (22) becomes a convex maximization problem. Referring to convex cases shown in Fig. 3, we know that if the axis of symmetry is to the right (or left) of the vertical axis, the maximum value of g_j occurs at $dp = -\Delta p$ (or $dp = \Delta p$) in spite of the axis of symmetry being within the bound $[-\Delta p, \Delta p]$ or not.

When the parabola is concave as shown in Fig. 4, it becomes a concave maximization problem. If $\left|-g'_{jp}(x,p_0)/g''_{jp}(x,p_0)\right| \le \Delta p$, then the maximum g_j occurs at $dp = -g'_{jp}(x,p_0)/g''_{jp}(x,p_0)$. If $-g'_{jp}(x,p_0)/g''_{jp}(x,p_0) \ge \Delta p$ (or $\le -\Delta p$), the maximum value of g_j occurs at $dp = \Delta p$ or $(dp = -\Delta p)$. To simplify the procedure, the above situations can be classified into two cases:

(1) If $g''_{jp}(x,p_0) < 0 \left|-g'_{jp}(x,p_0)/g''_{jp}(x,p_0)\right| \le \Delta p$, the maximum g_j occurs at $dp = -g'_{jp}(x,p_0)/g''_{jp}(x,p_0)$;
(2) For other cases, a simple comparison between $g(x, p_0 - \Delta p)$ and $g(x, p_0 + \Delta p)$ can be made and the larger one should be chosen.

As a summary

$$p_{\max} = \begin{cases} p_0 - \dfrac{g'_{jp}(x,p_0)}{g''_{jp}(x,p_0)}, \text{if} g''_{jp}(x,p_0) < 0 \text{ and} \left|-\dfrac{g'_{jp}(x,p_0)}{g''_{jp}(x,p_0)}\right| \le \Delta p \\ p_0 - \Delta p, \text{other cases when } g_j(x,p_0-\Delta p) \text{ is larger} \\ p_0 + \Delta p, \text{ other cases when } g_j(x,p_0+\Delta p) \text{ is larger} \end{cases} \quad (23)$$

When $g''_{jp}(x,p_0)$ in Eq. (22) is zero, it becomes a linear equation and the solution is $\mathrm{d}p_{\max} = \mathrm{sign}(g_{jp}{}'(x,p_0))\cdot\Delta p$.

3.2.2 Solve the Constraint Robustness Index With Multiple Uncertain Variables/Parameters. Here we have to find the maximum g_j ($\boldsymbol{dp}$; $\boldsymbol{x}$, $\boldsymbol{p}_0$) value as shown in Eq. (4*b*) which can be decomposed to:

$$\begin{aligned} \max \quad & g_j(\boldsymbol{dp};\boldsymbol{x},\boldsymbol{p}_0) \cong g_j(\boldsymbol{x},\boldsymbol{p}_0) + \mathbf{c}_{jp}\boldsymbol{dp} + \frac{1}{2}\boldsymbol{dp}^{\mathbf{T}}\mathbf{H}_{jp}\boldsymbol{dp} \\ & = \frac{1}{2}\boldsymbol{dp}^{\mathbf{T}}\mathbf{U}_j^{\mathrm{T}}\mathbf{D}_j\mathbf{U}_j\boldsymbol{dp} + \mathbf{c}_j^{\mathbf{T}}\mathbf{U}_j^{\mathrm{T}}\mathbf{U}_j\boldsymbol{dp} \\ & = \frac{1}{2}(\mathbf{U}_j\boldsymbol{dp})^{\mathbf{T}}\mathbf{D}_j(\mathbf{U}_j\boldsymbol{dp}) + (\mathbf{U}_j\mathbf{c}_j)^{\mathbf{T}}(\mathbf{U}_j\boldsymbol{dp}) \\ s.t. \quad & \mathbf{lb} \le \mathbf{U}_j^{\mathbf{T}}(\mathbf{U}_j\boldsymbol{dp}) \le \mathbf{ub} \end{aligned} \quad (24)$$

With the assumption that $|\mathbf{lb}_i| = |\mathbf{ub}_i| = \Delta\boldsymbol{p}_i$, $i = 1,\dots, k$, we let $\mathrm{d}\boldsymbol{p} = \boldsymbol{\alpha}_j o\Delta\boldsymbol{p} = [\alpha_{j1}\Delta p_1, \ \alpha_{j2}\Delta p_2,\dots, \ \alpha_{jk}\Delta p_k]$, where $\alpha_{ji}\in[-1, \ 1]$, $\boldsymbol{y} = \mathbf{U}_j\mathrm{d}\boldsymbol{p} = \mathbf{U}_j\ (\boldsymbol{\alpha}_j o\Delta\boldsymbol{p})$, and $\boldsymbol{v}_j = \mathbf{U}_j\mathbf{c}_{jp}$, Eq. (25) can be rewritten as

$$\begin{aligned} \max \quad & g_j(\boldsymbol{y}) = \frac{1}{2}(\mathbf{U}_j\boldsymbol{dp})^{\mathbf{T}}\mathbf{D}_j(\mathbf{U}_j\boldsymbol{dp}) + (\mathbf{U}_j\mathbf{c}_j)^{\mathbf{T}}(\mathbf{U}_j\boldsymbol{dp}) \\ & = \frac{1}{2}\sum_{i=1}^{k}\lambda_i y_i^2 + \sum_{i=1}^{k} v_i y_i \\ & = \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(y_i + \frac{v_i}{\lambda_i}\right)^2 - \sum_{i=1}^{k}\frac{v_i{}^2}{2\lambda_i} \\ & = \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{r=1}^{k}u_{ir}\alpha_r\Delta p_r + \frac{v_i}{\lambda_i}\right)^2 - \frac{1}{2}\sum_{i=1}^{k}\frac{v_i{}^2}{\lambda_i} \\ s.t. \quad & \alpha_r \in [-1, 1] \end{aligned} \quad (25)$$

As discussed previously, we have to solve

$$\begin{aligned} \max \quad & g_j(\boldsymbol{y}) = \frac{1}{2}\sum_{i=1}^{k}\lambda_i\left(\sum_{r=1}^{k}u_{ir}\alpha_r\Delta p_r + \frac{v_i}{\lambda_i}\right)^2 - \frac{1}{2}\sum_{i=1}^{k}\frac{v_i{}^2}{\lambda_i} \\ s.t. \quad & \alpha_r \in [-1, 1] \end{aligned} \quad (26)$$

If $\mathbf{H}$ in Eq. (24) is pd or psd, then Eq. (26) is a convex maximization problem; if $\mathbf{H}$ is nd or nsd, Eq. (26) becomes a concave maximization problem; if $\mathbf{H}$ is indefinite, then the maximization case for the indefinite problem as discussed in Sec. 2.3 will be applied.

The optimal solution $\Delta \boldsymbol{p}_{j\max}$ can be obtained and $\boldsymbol{p}_{j\max} = \boldsymbol{p}_0 + \Delta \boldsymbol{p}_{j\max}$, and then the constraints can be linearized as follows:

$$g_j(\boldsymbol{x}^{(k+1)},\boldsymbol{p}) \cong g_j(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max}) + \mathbf{c}_{j\max}\boldsymbol{d} \tag{27}$$

The feasibility requirements (i.e., the constraints will always be less than zero even for the worst case of the uncertain parameter values) can lead to the following conclusion in Eq. (28*b*):

$$g_j(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max}) + \mathbf{c}_{j\max}\boldsymbol{d} \le 0 \tag{28a}$$

$$\Leftrightarrow \quad \mathbf{c}_{j\max}\boldsymbol{d} \le -g_j(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max}) \quad \text{where} \quad \mathbf{c}_{j\max} = g'(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max}) \tag{28b}$$

Up to now, both the objective and constraint robustness indices can be linearized in the form of $\mathbf{a}_r^T\boldsymbol{d} \le \boldsymbol{b}_r$, as shown in Eqs. (21*b*) and (28*b*), and all constraints in Eq. (3) can be written in the matrix form of $\mathbf{A}_r\boldsymbol{d} \le \mathbf{b}$. Except a set of linear equations with typical matrix operations, no optimization routine is necessary for those inner problems.

As pointed out in Ref. [23], for constraints in the form of lower and upper bounds of $\boldsymbol{x}$, $\mathbf{A}_{\rm lb} = diag(-1)_{n\times n}$ with $\mathbf{b}_{\rm lb} = (\boldsymbol{x}^{(k)} - \boldsymbol{x}_{lb} - \Delta\boldsymbol{x})$ and $\mathbf{A}_{\rm ub} = diag(1)_{n\times n}$ with $\mathbf{b}_{\rm ub} = (-\boldsymbol{x}^{(k)} + \boldsymbol{x}_{ub} - \Delta\boldsymbol{x})$ can be used to form the linear constraints. The feasible region can be defined as $\mathbf{A}\boldsymbol{d} \le \mathbf{b}$ where $\mathbf{A} = [\mathbf{A}_{\rm r};\ \mathbf{A}_{\rm lb};\ \mathbf{A}_{\rm ub}]$ and $\mathbf{b} = [\mathbf{b}_{\rm r},\ \mathbf{b}_{\rm lb},\ \mathbf{b}_{\rm ub}]$.

3.3 Advanced SQP for Robust Optimization (A-SQP-RO). After the discussion above, the QP formulation in this new A-SQP-RO can be shown as

$$\begin{aligned}
&\min_{\boldsymbol{d}} \quad \bar{f}(\boldsymbol{d}) \cong \mathbf{c}^T\boldsymbol{d} + \tfrac{1}{2}\boldsymbol{d}^T\mathbf{H}\boldsymbol{d}\\
&\text{s.t.} \quad \mathbf{A}\boldsymbol{d} \le \mathbf{b}\\
&\text{where} \quad \mathbf{c} = \frac{\partial f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)}{\partial \boldsymbol{x}}, \mathbf{H} = \frac{\partial^2 f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)}{\partial \boldsymbol{x}_l \partial \boldsymbol{x}_m}, l,m = 1,\ldots,n;\\
&\quad \mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_j, \ldots, \mathbf{a}_J, \mathbf{a}_{J+1}, \mathbf{a}_{J+2}, \mathbf{A}_{lb}, \mathbf{A}_{ub}];\\
&\quad \mathbf{b} = [b_1, \ldots, b_j, \ldots, b_J, b_{J+1}, b_{J+2}, \mathbf{b}_{lb}, \mathbf{b}_{ub}];\\
&\quad \mathbf{a}_j = \frac{\partial g_j(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max})}{\partial \boldsymbol{x}}, b_j = -g_j(\boldsymbol{x}^{(k)},\boldsymbol{p}_{j\max}) j = 1,\ldots,J,\\
&\quad \mathbf{a}_{J+1} = \frac{\partial f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max})}{\partial \boldsymbol{x}} - \frac{\partial f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)}{\partial \boldsymbol{x}},\\
&\quad b_{J+1} = \Delta f_0 - \left(f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)\right),\\
&\quad \mathbf{a}_{J+2} = -\left(\frac{\partial f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max})}{\partial \boldsymbol{x}} - \frac{\partial f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)}{\partial \boldsymbol{x}}\right),\\
&\quad b_{J+2} = \Delta f_0 + \left(f(\boldsymbol{x}^{(k)},\boldsymbol{p}_{\max}) - f(\boldsymbol{x}^{(k)},\boldsymbol{p}_0)\right),\\
&\mathbf{A}_{lb} = diag(-1)_{n\times n}, \mathbf{b}_{lb} = (\boldsymbol{x}^{(k)} - \mathbf{lb} - \Delta\boldsymbol{x}),\\
&\quad \mathbf{A}_{ub} = diag(1)_{n\times n}, \mathbf{b}_{ub} = (-\boldsymbol{x}^{(k)} + \mathbf{ub} - \Delta\boldsymbol{x})
\end{aligned} \tag{29}$$

and $\boldsymbol{p}_{\max}$ as well as $\boldsymbol{p}_{j\max}$ can be obtained as discussed in Secs. 3.1 and 3.2.

The basic steps of the proposed A-SQP-RO algorithm are summarized as follows:

Step 1. Initialization

Set a small convergence tolerance $\varepsilon > 0$ and the iteration counter $k = 0$; the starting point $\mathbf{x}^{(0)}$ is also initialized. The lower and upper bounds of $\mathbf{d}$ are given if necessary;

Step 2. Calculation of the objective robustness indices

By applying strategies discussed in Sec. 3.1, the variables/parameters values that give the maximum objective variation can be obtained. The objective robustness index in Eq. (3) is replaced by the new constraint as defined in Eq. (21b);

Step 3. Calculation of the constraint robustness indices

By applying strategies discussed in Sec. 3.2, the variables/parameters value that gives the corresponding maximum constraint variation is obtained for each constraint. The constraints in Eq. (3) are replaced by new ones as defined in Eq. (28*b*);

Step 4. A- SQP-RO definition

The QP defined in Eq. (29) is formulated and solved to find the optimal $\mathbf{d}^{(k)}$;

Step 5. Checking for convergence and updating

In this A-SQP-RO approach, the general convergence condition used in SQP is also applied: if $\left\|\boldsymbol{d}^{(k)}\right\| \le \varepsilon$, then $\mathbf{x}^* = \mathbf{x}^{(k)}$ and stop; otherwise, Set $k = k+1$, $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{d}^{(k)}$ and go to step 2. Note that when elements in $\mathbf{d}$ are of different magnitudes, normalization of those elements should be conducted first.

The proposed robust optimization formulation in Eq. (29) just includes two more additional constraints for objective robustness compared to Eq. (1). More importantly, there is no inner optimization procedure performed in A-SQP-RO anymore. Thus in nature this new single-looped RO approach is more computationally efficient than double-looped SQP-RO proposed previously.

3.4 Discussion of A-SQP-RO. The computational efficiency of A-SQP-RO and SQP-RO are compared as shown in Table 1. The differences lie in that: for each robustness constraint with at most k variables, SQP-RO has to solve an optimization problem, while A-SQP-RO just needs to solve for its derivatives and Hessian matrix, and the eigenvalues and corresponding eigenvectors for each matrix. Numbers of function evaluations are listed in Table 1 too. It is noted that for calculations of derivatives and Hessian matrix as well as matrix operations, the number of times that these operations are performed is utilized. It is obvious that the computational burden of the proposed A-SQP-RO approach is much less than that of SQP-RO.

The obtained Utopian solution can be conservative as can be observed from Fig. 2. However, this deviation is quite small as can be seen later in the test examples. Moreover, (1) if the constraints contain only one uncertain variable/parameter, or (2) if the Hessian matrix $\mathbf{H}$ is already diagonal, or (3) when the extreme point of the function is located within the uncertainty box, the Utopian solution is just the real solution of the approximated QPs in A-SQP-RO. For real engineering applications, cases 1 and 2 may be rare. However, it is reasonable to believe that for a large

Table 1 Analysis of function evaluations for A-SQP-RO and SQP-RO in one iteration

Operations	A-SQP-RO	SQP-RO
Number of times for calculation of derivatives and Hessian matrix for the objective at $\boldsymbol{x}^{(k)}$	1	1
For at most $(J+2)$ robustness constraints, number of times for calculation of derivatives and Hessian matrix, eigenvectors and eigenvalues for each Hessian matrix	$2(J+1)$*	0
Solve at most $(J+1)$ optimization problems, each with at most k variables, for worst-case uncertain values	0	$K(J+1)$
Number of times for calculation of function value and derivatives for at most $(J+2)$ constraints with worst-case values	$(J+1)$*	$(J+1)$*
Calculation of step size $\boldsymbol{d}$	L	L

Notes: *: Calculations of derivative and Hessian are the same for the additional two objective robustness constraints, J: Number of constraints, K: Number of function evaluations of fmincon which is between 8 and 16 for test examples in this paper (fmincon has been used in our approach to find extreme parameter values), L: Number of function evaluations of quadprog which is between 1 and 7 for test examples (quadprog has been used to find the step size $\boldsymbol{d}$).

and sparse Hessian matrix which is not a rare case, the obtained solution should be nearly the same as the true solution. If one is still not sure whether to adopt the obtained solution from A-SQP-RO, then this solution, which is close to the true solution, can be chosen as the starting point for other RO algorithms. In this regard, the computational efficiency of the entire procedure can still be largely improved.

For engineering problems with no explicit mathematical formulations, a surrogate model could be built and utilized to get approximated Hessian matrix and derivatives. Although this type of approximation may bring additional uncertainty (or approximation error), this error is usually relatively small and negligible as the feasible region shrinks during the optimization process and more and more sampling points are being used within this shrinking feasible region. This approximation error could also be considered as a type of "model uncertainty" which will be addressed in our future research.

In Sec. 4, six numerical and engineering examples will be used to demonstrate the applicability and computational efficiency of the proposed A-SQP-RO.

4 Test Examples and Comparison of Results

In this section, we test four nonlinear numerical and two engineering design optimization examples under interval uncertainties, with the results and computational time from A-SQP-RO compared to those from SQP-RO. The first numerical example is demonstrated step by step to show how the proposed approach works, and one engineering example is presented with results in each iteration too. The others are given with results from both A-SQP-RO and SQP-RO. In this paper, the computation platform is a Dell Optiplex 980 (3.2 GHz Intel(R) Core(TM) i5 CPU with 4GB of RAM). Table 2 summarizes the uncertainty occurrences in each example.

4.1 Nonlinear Numerical Example 1. Our first example is a nonlinear problem with uncertainty in design variables. The formulation is given in the following equation:

$$\begin{aligned}\min_{\boldsymbol{x}} \quad & x_1^3\sin(x_1+4)+10x_1^2+22x_1 \\ & +5x_1x_2+2x_2^2+3x_2+12 \\ \text{s.t.} \quad & x_1^2+3x_1-x_1\sin x_1+x_2-2.75\le 0 \\ & -\log(0.1x_1+0.41)+x_2e^{-x_1+3x_2-4}+x_2-3\le 0 \\ & \Delta x_1=\Delta x_2=0.4, \Delta f_0=2.5\end{aligned} \tag{30}$$

In this example, the design variables x_1 and x_2 have uncertainty ± 0.4 around their nominal. The acceptable objective function variation $\Delta f_0=2.5$. It will be demonstrated step by step in the first iteration to show how A-SQP-RO works.

Iteration 1:

Step 1: Initialization. Set $\varepsilon=1\times10^{-5}$, $k=0$, and $x^{(0)}=[-1,1]$.

Step 2: To solve the objective robustness index,

$$\begin{aligned}\max|\Delta f(d\boldsymbol{p};\boldsymbol{x},\boldsymbol{p}_0)| &\cong \left|\frac{1}{2}d\boldsymbol{p}^{\mathrm{T}}\mathbf{H}_p d\boldsymbol{p}+\mathbf{c}_p^{\mathrm{T}}d\boldsymbol{p}\right| \\ &= \left|\frac{1}{2}\begin{bmatrix}dp_1\\dp_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}13.355.00\\5.004.00\end{bmatrix}\begin{bmatrix}dp_1\\dp_2\end{bmatrix}+\begin{bmatrix}8.41\\2.00\end{bmatrix}^T\begin{bmatrix}dp_1\\dp_2\end{bmatrix}\right|\end{aligned} \tag{31}$$

By decomposing $\mathbf{H}_p$ into $\mathbf{U}^{\mathrm{T}}\mathbf{D}\mathbf{U}$, we get $\mathbf{D}=\begin{bmatrix}1.83\,0\\0\,15.52\end{bmatrix}$, $\mathbf{U}=\begin{bmatrix}0.40-0.92\\-0.92-0.40\end{bmatrix}$, and $\boldsymbol{v}=\mathbf{U}\mathbf{c}_p=\begin{bmatrix}1.51\\-8.51\end{bmatrix}$. Since $\mathbf{H}$ (or $\mathbf{D}$) is pd, $y_{i\max}=sign(v_i)\sum_{j=1}^{k}|u_{ij}|x_{bj}$ and thus $\boldsymbol{y}_{\max}=\begin{bmatrix}0.528\\-0.528\end{bmatrix}$ and $d\boldsymbol{p}_{\max}=\mathbf{U}^{\mathrm{T}}\boldsymbol{y}_{\max}=\begin{bmatrix}0.69\\-0.27\end{bmatrix}$. It is also obtained that $\boldsymbol{\alpha}^*=\begin{bmatrix}\alpha_1^*\\\alpha_2^*\end{bmatrix}=\begin{bmatrix}1.73\\-0.68\end{bmatrix}$.

Step 3: To solve the constraint robustness index

$$\begin{aligned}\max \quad g_1(d\boldsymbol{p};\boldsymbol{x}^{(0)},\boldsymbol{p}_0) &\cong g_1(\boldsymbol{x},\boldsymbol{p}_0)+\mathbf{c}_{1p}^{\mathrm{T}}d\boldsymbol{p}+\frac{1}{2}d\boldsymbol{p}^{\mathrm{T}}\mathbf{H}_{1p}d\boldsymbol{p} \\ &= -1.95+\begin{bmatrix}2.38\\1.00\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}dp_1\\dp_2\end{bmatrix}+\frac{1}{2}\begin{bmatrix}dp_1\\dp_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}1.760\\0\,0\end{bmatrix}\begin{bmatrix}dp_1\\dp_2\end{bmatrix}\end{aligned} \tag{32}$$

$\mathbf{H}_{1p}$ is diagonal and pd so $\mathbf{U}_1=\mathbf{I}=\begin{bmatrix}10\\01\end{bmatrix}$, $\boldsymbol{v}_1=\mathbf{U}_1\mathbf{c}_{1p}=\begin{bmatrix}2.38\\1.00\end{bmatrix}$, $y_{i\max}=sign(v_i)\sum_{j=1}^{k}|u_{ij}|x_{bj}$ and $\boldsymbol{y}_{\max}=\begin{bmatrix}0.4\\0.4\end{bmatrix}$, and thus $d\boldsymbol{p}_{1\max}=\mathbf{U}_1^{\mathrm{T}}\boldsymbol{y}_{\max}=\begin{bmatrix}0.4\\0.4\end{bmatrix}$. It is also obtained that $\boldsymbol{\alpha}_1^*=\begin{bmatrix}\alpha_1^*\\\alpha_2^*\end{bmatrix}=\begin{bmatrix}1\\1\end{bmatrix}$.

Similarly, we get $d\boldsymbol{p}_{2\max}=\begin{bmatrix}0.347\\0.600\end{bmatrix}$ and $\boldsymbol{\alpha}_2^*=\begin{bmatrix}\alpha_1^*\\\alpha_2^*\end{bmatrix}=\begin{bmatrix}0.87\\1.50\end{bmatrix}$.

Step 4: The A-SQP-RO formulation of this problem is given as in the following equation:

$$\begin{aligned}&\min_{d^{(0)}} \quad \mathbf{c}_0^T\boldsymbol{d}^{(0)}+\frac{1}{2}\boldsymbol{d}^{(0)T}\mathbf{H}_0\boldsymbol{d}^{(0)} \\ &\text{s.t.} \quad \mathbf{A}^T\boldsymbol{d}^{(0)}\le\mathbf{b} \\ &\text{where} \quad \mathbf{c}_0=\frac{\partial f(\boldsymbol{x}^{(0)})}{\partial\boldsymbol{x}},\mathbf{H}_0=\frac{\partial^2 f(\boldsymbol{x}^{(0)})}{\partial\boldsymbol{x}^2}, \\ &\mathbf{A}=\left[\frac{\partial g(\boldsymbol{x}^{(0)},\boldsymbol{p}_{\max})}{\partial\boldsymbol{x}},\mathbf{A_{lb}},\mathbf{A_{ub}}\right]^T, \\ &\mathbf{b}=[-g(\boldsymbol{x}^{(0)},\boldsymbol{p}_{\max}),\mathbf{b_{lb}},\mathbf{b_{ub}}], \\ &\mathbf{A_{lb}}=diag(-1)_{2\times2},\mathbf{A_{ub}}=diag(1)_{2\times2}, \\ &\mathbf{b_{lb}}=\boldsymbol{x}^{(0)}-\mathbf{lb}-\Delta\boldsymbol{x},\mathbf{b_{ub}}=\mathbf{ub}-\boldsymbol{x}^{(0)}-\Delta\boldsymbol{x}\end{aligned} \tag{33}$$

By solving this problem, we obtain $\boldsymbol{d}^{(0)}=[-0.45, -0.38]$.

Step 5: Since $\|d^{(0)}\|\ge\varepsilon$, set iteration counter $k=k+1=1$, $\boldsymbol{x}^{(1)}=\boldsymbol{x}^{(0)}+\boldsymbol{d}^{(0)}=[-1.45, 0.62]$ and go to iteration 2 that follows the same steps as in iteration 1. The results are summarized in Table 3.

The robust solution obtained from SQP-RO is $\boldsymbol{x}=[-1.44, 0.34]$ and $f=-1.7728$, while from the proposed approach $f_{\mathrm{new}}=-1.7287$ which is similar to the one obtained from SQP-RO.

Figure 5(a) shows the constraints contours with $p=p_0$ and $p=p_{\max}$ from both SQP-RO and A-SQP-RO. For g_1, two $p_{\max}$ solutions from SQP-RO and A-SQP-RO are the same, so the contours are overlapped while the two g_2 contours with $p=p_{\max}$ are

Table 2 Uncertainty occurrences in each example

	Numerical examples				Engineering examples	
Uncertainty occurrences	1	2	3	4	Two-bar Truss	Speed Reducer
Design variables $\boldsymbol{x}$	√	√	√		√	√
Parameters $\boldsymbol{p}$			√	√		

Table 3 Results of numerical example 1

k	α^*	α_1^*	α_2^*	$d^{(k-1)}$	$x^{(k)}$
1	[1.73,−0.68]	[1,1]	[0.87,1.50]	[−0.45, −0.38]	[−1.45,0.62]
2	[1.97, −0.23]	[1,1]	[−1.47,0.88]	[−0.06, −0.20]	[−1.51,0.42]
3	[−0.17, −1.99]	[1,1]	[−1.45,0.89]	[0.11, −0.06]	[−1.40,0.36]
…	……			……	……
7	[−0.33, −1.95]	[1,1]	[−1.45,0.89]	$[0.32,0.10]10_{-4}$	[−1.37,0.35]
8	[−0.33, −1.95]	[1,1]	[−1.45,0.89]	$[0.32,0.10]10_{-5}$	[−1.37,0.35]

almost overlapped. Figure 5(*b*) shows the solution comparison of SQP-RO and A-SQP-RO: the two solutions are very close to each other, and the objective contours are almost overlapped. When the obtained solution varies within the uncertain box, both are robust with respect to objective robustness ($\Delta f_0 = 2.5$) and feasibility robustness ($g_j(x,p) \le 0$) (i.e., the uncertainty box in Fig. 5(*b*) is within the feasible domain and also fall in the region between the contours of $f_+ -2.5$ and $f_+ +2.5$).

4.2 Additional Numerical Examples. In this section, another three numerical examples modified from the literature [23,27,35] are tested. The formulations and the real robust solutions as well as the Utopian solutions are listed in Table 4.

The numbers of iterations, # of function calls, and computational time (in seconds) for those four examples are listed in Table 5. From these two tables, it can be seen that this A-SQP-RO

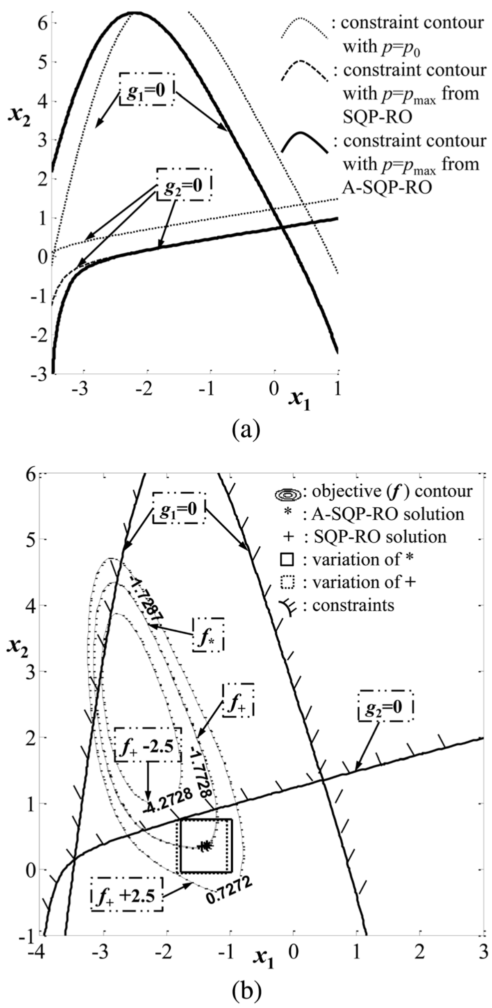

Fig. 5 Solution comparison of SQP-RO and A-SQP-RO: (*a*) Constraint contour comparison and (*b*) final solution comparison

method can improve the computational efficiency significantly (with respect to number of function calls and computational time) without worsening the solutions seriously or can even obtain the same solutions with that of SQP-RO.

4.3 Two-Bar Truss. The two-bar truss problem from Ref. [35] is tested in this paper. The uncertainty in design variables is set as $[\Delta x_1, \Delta x_2, \Delta x_3] = [0.000125, 0.000125, 0.075]$, and the acceptable objective variation is set as $\Delta f_0 = 1$. The formulation is as shown in the following equation:

$$\begin{aligned}
&\min_x \quad f(x) = \frac{20(16+x_3^2)^{\frac{1}{2}}}{10^3 x_1 x_3} \\
&\text{s.t.} \quad g_1 = f - 100 \le 0, g_2 = f_2 - 100 \le 0, g_3 = f_3 - 100 \le 0 \\
&\text{where} \quad f_2 = 10^3\left[x_1(16+x_3^2)^{\frac{1}{2}} + x_2(1+x_3^2)^{\frac{1}{2}}\right], f_3 = \frac{80(1+x_3^2)^{\frac{1}{2}}}{10^3 x_1 x_3} \\
&0.0001 \le x_1, x_2 \le 0.25, \quad 1.0 \le x_3 \le 3.0
\end{aligned} \tag{34}$$

The robust solution from SQP-RO is $\boldsymbol{x} = [0.0197, 0.0002, 2.9250]$ with $f = 1.7240$ [23] and the Utopian solution is $\boldsymbol{x} = [0.01956, 0.000225, 2.925]$ with $f_{new} = 1.7322$. Table 6 lists the detailed results from every iteration.

It can be seen from Table 6 that not all Utopian solutions from A-SQP-RO lie outside of the original uncertainty box in which case, the obtained solution is not over conservative, and the deviation of the final result is actually very small.

4.4 Speed Reducer. Speed reducer, as a typical mechanical design problem proposed in Ref. [35], is modified in Ref. [23] as follows:

$$\begin{aligned}
&\min_x \quad f = 0.7854 x_1 x_2^2\left(\frac{10x_3^2}{3} + 14.933x_3 - 43.0934\right) \\
&- 1.508x_1(x_6^2 + x_7^2) + 7.477(x_6^3 + x_7^3) + 0.7854(x_4 x_6^2 + x_5 x_7^2) \\
&\text{s.t.} \quad g_1 = \frac{1}{x_1 x_2^2 x_3} - \frac{1}{27} \le 0, \quad g_2 = \frac{1}{x_1 x_2^2 x_3^2} - \frac{1}{397.5} \le 0, \\
&g_3 = \frac{x_4^3}{x_2 x_3 x_6^4} - \frac{1}{1.93} \le 0, \quad g_4 = \frac{x_5^3}{x_2 x_3 x_7^4} - \frac{1}{1.93} \le 0, \\
&g_5 = x_2 x_3 - 40 \le 0, \quad g_6 = \frac{x_1}{x_2} - 12 \le 0, \\
&g_7 = 5 - \frac{x_1}{x_2} \le 0, \quad g_8 = 1.9 - x_4 + 1.5x_6 \le 0, \\
&g_9 = 1.9 - x_5 + 1.1x_7 \le 0, \quad g_{10} = f_2 - 1800 \le 0, \\
&g_{11} = f_3 - 1100 \le 0, \\
&\text{where} \quad f_2 = \frac{\sqrt{\left(\frac{745x_4}{x_2 x_3}\right)^2 + 1.69\times 10^7}}{0.1x_6^3}, \\
&f_3 = \frac{\sqrt{\left(\frac{745x_5}{x_2 x_3}\right)^2 + 1.575\times 10^8}}{0.1x_7^3} \\
&2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8, 17 \le x_3 \le 28, 7.3 \le x_4 \le 8.3 \\
&7.3 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9, 5.0 \le x_7 \le 5.5
\end{aligned} \tag{35}$$

Still we define that design variables x_2 and x_6 have variations $[\Delta x_2, \Delta x_6] = [0.01, 0.1]$ and the acceptable objective variation $\Delta f_0 = 100$. The robust solution obtained from SQP-RO is $\boldsymbol{x} = [3.6000, 0.7100, 17.0000, 7.3000, 7.7153, 3.4502, 5.2867]$ with the objective value $f = 3106.65$ [23]. From our new approach,

Table 4 Numerical examples: formulations and solutions

Numerical Examples	Formulation	Solutions		
		$\boldsymbol{x}$ & f	SQP-RO	A-SQP-RO
2	$\min_x \quad f(x) = -x \cdot \sin(3\pi x)$ s.t. $\quad 0 \le x \le 1$ with $\quad \Delta x = 0.05, \Delta f_0 = 0.06$	x f	2.153 −1.9308	2.153 −1.9308
3	$\min_x \quad (x_1 - 0.6)^2 + (x_2 - 0.6)^2 - x_3 x_4 + 10$ s.t. $\quad p_1 + x_1 + x_2 \le 0$ $p_2 + x_3 + x_4 \le 0$ $x_1, x_2, x_3, x_4 \ge 0$ where $\quad p_1 = p_2 = -1, \Delta x_3 = \Delta p_1 = \Delta p_2 = 0.1$	x_1 x_2 x_3 x_4 f	0.45 0.45 0.4 0.4 9.885	0.45 0.45 0.4 0.4 9.885
4	$\min_x \quad (x_1 - 0.6)^2 + (x_2 - 0.6)^2 - x_3^2 x_4 + 10$ s.t. $\quad (p_1 + x_1)^3 (p_2 + x_3)^2 + x_2 x_4 \le 0$ $x_1, x_2, x_3, x_4 \ge 0$ where $\quad p_1 = p_2 = -1, \Delta p_1 = \Delta p_2 = 0.1$	x_1 x_2 x_3 x_4 f	0.5901 0.5983 0.4500 0.0101 9.9981	0.5919 0.5988 0.4627 0.0067 9.9986

Table 5 Computational efficiency of numerical and engineering examples

	# of iterations		# of function calls		Computational time (s)	
Examples	SQP-RO	A-SQP-RO	SQP-RO	A-SQP-RO	SQP-RO	A-SQP-RO
1	6	8	159	96	2.1781	1.1428
2	7	6	49	30	2.2600	0.9416
3	2	2	56	26	3.2146	1.1669
4	14	14	196	135	3.1430	1.2338
Speed reducer	3	3	264	123	3.3781	1.4063
Two-bar truss	6	4	190	63	3.6906	1.1251

Table 6 Detailed record of two-bar truss

		g_1		g_2		g_3		Max $\lvert\Delta f\rvert$			
Iteration k		α_1^*	g_1	α_2^*	g_2	α_3^*	g_3	α^*	f	$d^{(k-1)}$	$x^{(k)}$
1	1	−1	−95.53	2.566	−32.92	−1	−91.06	1		3.89×10^{-3}	1.39×10^{-2}
	2	1		0.600		1		1.199	4.47	2.48×10^{-2}	1.25×10^{-2}
	3	−1		0.564		−1		0.996		5.56×10^{-1}	2.56
2	1	−1	−97.33	2.613	−0.17	−1	−93.81	−1		5.07×10^{-3}	1.90×10^{-2}
	2	1		0.499		1		1	2.67	-1.23×10^{-2}	2.25×10^{-4}
	3	−1		0.611		−1		−1		3.69×10^{-1}	2.925
3	1	−1	−98.21	2.635	−5.37	−1	−95.54	−1		6.04×10^{-4}	1.96×10^{-2}
	2	1		0.440		1		1	1.787	3.47×10^{-18}	2.25×10^{-4}
	3	−1		0.634		−1		−1		1.80×10^{-16}	2.925
4	1	−1	−98.27	2.635	−2.38	−1	−95.68	−1		7.15×10^{-9}	1.956×10^{-2}
	2	1		0.440		1		1	1.732	0	2.25×10^{-4}
	3	−1		0.634		−1		−1		1.80×10^{-16}	2.925
Final			−98.27		−2.38		−95.68		1.732		

$\boldsymbol{x}_{\text{new}} = [3.6000, 0.7100, 17.0000, 7.3000, 7.7153, 3.4502, 5.2867]$ which is the same as that of SQP-RO.

Table 5 lists the number of iterations and computational time (s) of these two engineering examples.

From Table 5, it can be observed that in most cases iteration numbers of both SQP-RO and A-SQP-RO are comparable. However, the computational burden in terms of the number of function calls and computational time of the latter approach is much less than the former one while the objective function values are slightly degraded or even not degraded.

5 Conclusion

In this paper, a new approach, A-SQP-RO, has been proposed to solve nonlinear RO problems with interval uncertainties. The concept of the Utopian box and Utopian solution for the objective and constraint robustness indices are proposed. In this new approach, the inner optimization problems in the original double-looped SQP-RO have been replaced by a few typical matrix operations. The computational burden that has to be paid for the robustness assessment in the double-looped RO approaches can be saved correspondingly. Although the approach proposed here in this paper is based on SQP, the idea of the Utopian boxes and Utopian solutions can be easily extended to other inner optimization problems in RO approaches with interval uncertainties.

Six examples have been used to demonstrate the applicability of the proposed approach. The applicability of A-SQP-RO is the same as that of SQP-RO. Although the obtained solutions can be conservative in some cases compared to that of SQP-RO, it is shown that the deviation from the true robust solutions is rather small for both the numerical and engineering examples while the computational efficiency has been improved significantly.

Acknowledgment

The work presented here is supported in part by National Natural Science Foundation of China through Grant No. 51375302. Such support does not constitute an endorsement by the funding agency of the opinions expressed in the paper.

References

[1] Taguchi, G., 1978, "Performance Analysis Design," Int. J. Prod. Res., **16**, pp. 521–530.
[2] Ben-Tal, A., and Nemirovski, A., 2002, "Robust Optimization-Methodology and Applications," Math. Program., Ser. B, **92**(3), pp. 453–480.
[3] Park, G.-J., Lee, T.-H., Lee, K. H., and Hwang, K.-H., 2006, "Robust Design: An Overview," AIAA J., **44**(1), pp. 181–191.
[4] Beyer, H.-G., and Sendhoff, B., 2007, "Robust Optimization—A Comprehensive Survey," Comput. Methods Appl. Mech. Eng., **196**(33–34), pp. 3190–3218.
[5] Bertsimas, D., Brown, D. B., Caramanis, C., 2010, "Theory and Applications of Robust Optimization," Math. Program., Ser. B, **107**, pp. 464–501.
[6] Hora, S. C., 1996, "Aleatory and Epistemic Uncertainty in Probability Elicitation With an Example From Hazardous Waste Management," Reliab. Eng. Syst. Saf., **54**, pp. 217–223.
[7] Oberkampf, W. L., Diegert, K. V., Alvin, K. F., Rutherford, B. M., 1998, "Variability, Uncertainty, and Error in Computational Simulation," ASME Proceedings of the 7th AIAA/ASME Joint Thermophysics and Heat Transfer Conference, Albuquerque, New Mexico, **357**(2), pp. 259–272.
[8] Hofer, E., Kloos, M., Krzykacz-Hausmann, B., Peschke, J., and Woltereck, M., 2002, "An Approximate Epistemic Uncertainty Analysis Approach in the Presence of Epistemic and Aleatory Uncertainties," Reliab. Eng. Syst. Saf., **77**, pp. 229–238.
[9] Agarwal, H., Renaud, J. E., Preston, E. L., and Padmanabhan, D., 2004, "Uncertainty Quantification Using Evidence Theory in Multidisciplinary Design Optimization," Reliab. Eng. Syst. Saf., **85**, pp.281–294.
[10] Helton, J. C., Johnson, J. D., Oberkampf, W. L., and Storlie, C. B., 2007, "A Sampling-Based Computational Strategy For The Representation of Epistemic Uncertainty in Model Predictions With Evidence Theory," Comput. Methods Appl. Mech. Eng., **196**, pp.3980–3998.
[11] Haldar, A., and Mahadevan S., 2000, Probability, Reliability and Statistical Methods in Engineering Design, Wiley, New York.
[12] Stacy, L. J., Lin, X., Floudas, C. A., 2007, "A New Robust Optimization Approach for Scheduling Under Uncertainty: II. Uncertainty With Known Probability Distribution," Comput. Chem. Eng., **31**(3), pp. 171–195.
[13] Shapiro, A., 2008, "Stochastic Programming Approach to Optimization Under Uncertainty," Math. Program., Ser. B, **112**, pp. 183–220.
[14] Lin, X., Stacy, L. J., and Floudas, C. A., 2003, "A New Robust Optimization Approach for Scheduling Under Uncertainty: I. Bounded Uncertainty," Comput. Chem. Eng., **28**(6–7), pp. 1069–1085.
[15] Li, M., and Azarm, S., 2008, "Multiobjective Collaborative Robust Optimization With Interval Uncertainty and Interdisciplinary Uncertainty Propagation," ASME J. Mech. Des., **130**(8), p. 081402.
[16] Hu, W., Li, M., Azarm, S., and Almansoori, A., 2011, "Multi-Objective Robust Optimization Under Interval Uncertainty Using Online Approximation and Constraint Cuts," ASME J. Mech. Des., **133**(6), p. 061002.
[17] Ben-Tal, A., and Nemirovski, A., 1999, "Robust Solutions of Uncertain Linear Programs," Oper. Res. Lett., **25**, pp. 1–13.
[18] Ben-Tal, A., and Nemirovski, A., 2000, "Robust Solutions of Linear Programming Problems Contaminated With Uncertain Data," Math. Program., Ser. A, **88**(3), pp. 411–424.
[19] Bertsimas, D., Pachamanova, D., and Sim, M., 2004, "Robust Linear Optimization Under General Norms," Oper. Res. Lett., **32**(6), pp. 510–516.
[20] Ben-Tal, A., and Nemirovski, A., 1998, "Robust Convex Optimization," Math. Oper. Res., **23**, pp. 769–805.
[21] Lobo, M., 2000, "Robust and Convex Optimization With Applications in Finance," Ph.D. thesis, Department of Electrical Engineering, Stanford University, Stanford, CA.
[22] Teo, K. M., 2007, "Nonconvex Robust Optimization," Ph.D. thesis, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA.
[23] Zhou, J. H., Cheng, S., and Li, M., 2012, "Sequential Quadratic Programming for Robust Optimization With Interval Uncertainty," ASME J. Mech. Des., **134**(10), p. 100913.
[24] Li, M., 2007, "Robust Optimization and Sensitivity Analysis With Multi-Objective Genetic Algorithms: Single- and Multi-Disciplinary Applications," Ph.D. thesis, Department of Mechanical Engineering, UMD, College Park, MD.
[25] Liang, J., Mourelatos, Z. P., and Nikolaidis, E., 2007. "A Single-Loop Approach for System Reliability-Based Design Optimization," ASME J. Mech. Des., **129**(12), pp. 1215–1224.
[26] Li, M., Gabriel, S. A., Shim, Y., and Azarm, S., 2011, "Interval Uncertainty-Based Robust Optimization for Convex and Non-Convex Quadratic Programs With Applications in Network Infrastructure Planning," Netw. Spatial Econ., **11**(1), pp. 159–191.
[27] Siddiqui, S., Azarm, S., and Gabriel, S., 2011, "A Modified Benders Decomposition Method for Efficient Robust Optimization Under Interval Uncertainty," Struct. Multidiscip. Optim., **44**(2), pp. 259–275.
[28] Floudas, C. A., and Visweswaran, V., 1995, "Quadratic optimization," *Handbook of Global Optimization*, R. Horst and P. M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 217–269.
[29] Gao, D. Y., and Ruan, N., 2010, "On the Solutions to Quadratic Minimization Problems With Box and Integer Constraints," J. Global Optim., **47**(3), pp. 463–484.
[30] Friedlander, A., Martinez, J. M., and Raydan, M., 1995, "A New Method for Large-Scale Box Constrained Convex Quadratic Minimization Problems," Optim. Methods Software, **5**, pp. 57–74.
[31] Dembo, R. S., and Tulowitzki, U., 1984, "On the Minimization of Quadratic Functions Subject to Box Constraints," Working Paper No.71, Series B, School of Organization and Management, Yale University (New Haven, CT, 1983).
[32] Vandenbussche, D., and Nemhauser, G. L., 2005, "A Branch-and-Cut Algorithm for Nonconvex Quadratic Programs With Box Constraints," Math. Program, Ser. A, **102**, pp. 559–575.
[33] Yang E. K., and Tolle, J. W., 1991, "A Class of Methods for Solving Large, Convex Quadratic Programs Subject to Box Constraints," Math. Programing, **51**, pp. 223–228.
[34] Ben-Tal, A., Nemirovski, A., and Roos, C., 2002, "Robust Solutions of Uncertain Quadratic and Conic-Quadratic Problems," SIAM, J. Optim., **13**(2), pp. 535–360.
[35] Gunawan, S., and Azarm, S., 2004, "Non-Gradient Based Parameter Sensitivity Estimation for Single Objective Robust Design Optimization," ASME J. Mech. Des., **126**(3), pp. 395–402.