Modal dynamic residual-based model updating through regularized semidefinite programming with facial reduction

Dan Li [1], Yang Wang [1, 2, *]

[1] School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA

[2] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA

*yang.wang@ce.gatech.edu

Abstract: Structural model updating techniques optimize model parameter values for improving the predication accuracy of a numerical model. Formulated as optimization problems, the objective of these techniques is to minimize the difference between model prediction and the experimental data. Most of these optimization problems are nonconvex, and consequently, the global optimum is very hard to solve. The sum of squares (SOS) approach and its sparse variant have been reported to relax a nonconvex polynomial optimization problem into a convex semidefinite programming (SDP) problem, which is more readily solvable. However, the corresponding convex SDP problem may fail the Slater condition qualification. The failure increases the difficulty for numerical algorithms, e.g. the interior point method, to solve the SDP problem. This paper proposes to utilize the facial reduction technique to regularize an SDP problem which fails the Slater condition qualification. In addition, the regularized SDP problem has smaller size, which helps to improve the computation efficiency. The performance of the proposed model updating approach is evaluated through a plane truss example.

Keywords: model updating, modal dynamic residual, sum of squares (SOS), semidefinite programming (SDP), facial reduction

1 Introduction

Finite element (FE) models are widely used to simulate the behaviors of civil structures. However, the structural responses simulated by an FE model are usually different from the measured results obtained in the field. The difference can be caused by many reasons, such as inaccurate material properties and idealized boundary conditions adopted in the FE model. Consequently, it is desirable to update the model parameter values so that the predicted behaviors better match the experimental results.

Considerable research efforts have been devoted to FE model updating using modal properties of structures, including natural frequencies and mode shapes. One common approach is to find a suitable set of model parameters through minimizing an objective function that measures the difference between the simulated and experimental results. These optimization-based model updating techniques have been utilized for parameter identification, condition assessment, and damage detection with limited success [1-5]. Some studies attempted to minimize the dynamic force/input residual of the equations of motion in frequency domain, which is the difference between the two sides of the equation and calculated using updated model matrices and measured dynamic forces and displacements [6, 7]. When evaluated at resonance frequencies and assuming normal modes, the dynamic force vanishes and the input residual simplifies to the modal dynamic residual of the generalized eigenvalue equation [8-10]. Typically, these formulated optimization problems for model updating are nonconvex, which means there can be unknown number of local minima over the feasible domain. Traditional optimization algorithms, e.g. the gradient based algorithm, can be easily trapped at a local minimum and fail to find the global solution. Attempts have been made to seek the global optimum of the model updating problems. Most of these global optimization methods exploit either the stochastic search [11], or gradient search from multiple starting points [12]. The nature of these optimization methods is heuristic and cannot guarantee global optimality.

In this paper, the sum-of-squares (SOS) approach is proposed to solve the FE model updating problem. It has been shown that if an optimization problem consists of only polynomials, the SOS approach and its variant can be applied to approximate the nonconvex problem with a convex semidefinite programming (SDP) problem. The re-formulated SDP problem can be solved efficiently by many existing algorithms, such as the interior-point method. As the SDP problem is convex, the solution is guaranteed to be globally optimal. The SOS approach has made significant impact on global optimization in many different fields, for example tensor decomposition [13, 14], computational geometry [15], and control theory [16].

In the field of FE model updating, the authors have shown the effectiveness of the SOS approach in identifying correct parameter values of structural models. In [17], the authors applied the standard SOS approach for FE model updating by minimizing modal dynamic residuals. Both numerical simulation and experimental study of a four-story shear frame structure have shown that the standard SOS approach can reliably solve the global solution to the FE model updating problem. However, despite the appeal of yielding global optimality, the computational complexity of the SOS approach grows rapidly when the polynomial optimization problem has a large number of variables and/or high degree [18, 19]. One method to alleviate the difficulty is to exploit the sparsity in the polynomial optimization problem and eliminate some redundant constraints in the formulated SDP problem. The so-called sparse SOS approach is shown to be able to efficiently solve a 2D truss updating example that the standard SOS approach cannot [20]. Nevertheless, it is discovered that the sparse SOS approach still requires excessively long time to solve some moderately-sized problems, or encounters significant difficulty when solving relatively large problems.

Building upon previous work, this paper proposes a new technique that can much more efficiently solve the SDP problems arising from both the standard and the sparse SOS approaches. Although effectiveness of these approaches on global optimality has been demonstrated, it turns out that the SDP problems formulated by both the standard and the sparse SOS approaches may fail to satisfy the Slater condition (strict feasibility), which plays an important role in the convergence of most interior-point solvers [21]. The failure to satisfy the Slater condition increases the difficulty of finding the optimal solution of the SDP problem using numerical algorithms. In this paper, we adopt facial reduction technique to overcome this challenge. The technique restricts such an SDP problem onto a feasible set with lower dimension and yields an equivalent SDP problem for which there are strictly feasible points. The smaller equivalent SDP can then be solved by a numerical solver in a more stable manner.

The rest of this paper is organized as follows. Section 2 introduces the formulation of the modal dynamic residual approach for model updating. This formulation entails an optimization problem whose objective function and inequality constraints are all polynomials. Section 3 covers the SOS approach for relaxing the nonconvex polynomial problems into convex SDP problems and investigates the sparsity in the SOS approach. In Section 4, we describe the facial reduction algorithms for regularizing SDP problems which fail the Slater condition qualification, emphasizing a practical algorithm which only inspects constraints of SDP problems. In Section 5, we validate the proposed approach through simulation of a plane truss structure. Finally, Section 6 provides a summary and discussion.

2 Modal dynamic residual approach for model updating

In this paper, we assume only the stiffness values of a model require updating. Consider a linear structure with N degrees of freedom (DOFs). The stiffness matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ of the structure can be parameterized by an updating vector $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$:

$$\mathbf{K}(\boldsymbol{\theta}) = \mathbf{K}_0 + \sum_{i=1}^{n_{\boldsymbol{\theta}}} \theta_i \mathbf{K}_i \tag{1}$$

where $\mathbf{K}_0 \in \mathbb{R}^{N \times N}$ is the initial structural stiffness matrix, and $\mathbf{K}_i \in \mathbb{R}^{N \times N}$ is the i^{th} influence matrix corresponding to the updating parameter θ_i. In theory, given a pair of resonance frequency ω and mode shape vector $\boldsymbol{\psi} \in \mathbb{R}^N$, only the actual/correct value of updating parameter $\boldsymbol{\theta}^*$ can provide the stiffness matrix $\mathbf{K}(\boldsymbol{\theta}^*)$ that satisfies the generalized eigenvalue equation:

$$[\mathbf{K}(\boldsymbol{\theta}^*) - \omega^2 \mathbf{M}]\boldsymbol{\psi} = \mathbf{0} \tag{2}$$

Based on this concept, the parameter $\boldsymbol{\theta}$ can be updated using the experimentally measured modal properties as a baseline. To acquire modal properties of the structure, sensors are instrumented at specific locations of the structure. In general, not all DOFs of the structure can be measured by sensors. The set of DOFs measured by sensors is denoted as $\mathcal{M}$, and the set of the remaining DOFs is denoted as $\mathcal{U}$. The number of DOFs in the set $\mathcal{M}$ is denoted as $n_{\mathcal{M}}$, and similarly the number of DOFs in the set $\mathcal{U}$ is denoted as $n_{\mathcal{U}}$. The measured modal properties of the structure usually include the first several resonance frequencies, $\boldsymbol{\omega} \in \mathbb{R}^{n_{\text{modes}}}$, and measured entries $\boldsymbol{\psi}_{\mathcal{M}} \coloneqq \left(\boldsymbol{\psi}_{\mathcal{M},1}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{M},2}^{\mathrm{T}}, \cdots, \boldsymbol{\psi}_{\mathcal{M},n_{\text{modes}}}^{\mathrm{T}}\right)^{\mathrm{T}} \in \mathbb{R}^{n_{\mathcal{M}} \cdot n_{\text{modes}} \times 1}$ in corresponding mode shapes.

The model updating problem can be stated as: given n_{modes} number of measured resonance frequencies, ω_i, $i = 1, 2, \cdots, n_{\text{modes}}$, and corresponding mode shapes with only entries measured by sensors $\boldsymbol{\psi}_{\mathcal{M},i}$, find the actual/correct value for the unknown stiffness parameter $\boldsymbol{\theta}$. Treating the stiffness parameter $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ and the unmeasured entries $\boldsymbol{\psi}_{\mathcal{U}} \coloneqq \left(\boldsymbol{\psi}_{\mathcal{U},1}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},2}^{\mathrm{T}}, \cdots, \boldsymbol{\psi}_{\mathcal{U},n_{\text{modes}}}^{\mathrm{T}}\right)^{\mathrm{T}} \in \mathbb{R}^{n_{\mathcal{U}} \cdot n_{\text{modes}} \times 1}$ in the mode shapes as optimization variables, the model updating problem can be formulated by minimizing following residual r:

$$
\begin{aligned}
\underset{\boldsymbol{\theta},\, \boldsymbol{\psi}_{\mathcal{U}}}{\text{minimize}} \quad & r = \sum_{i=1}^{n_{\text{modes}}} \left\| \left[\mathbf{K}(\boldsymbol{\theta}) - \omega_i^2 \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{\mathcal{M},i} \\ \boldsymbol{\psi}_{\mathcal{U},i} \end{Bmatrix} \right\|_2^2 \\
\text{subject to} \quad & \mathbf{L}_{\boldsymbol{\theta}} \leq \boldsymbol{\theta} \leq \mathbf{U}_{\boldsymbol{\theta}}
\end{aligned} \tag{3}
$$

Here $\|\cdot\|_2$ denotes the ℓ_2-norm; constant vectors $\mathbf{L}_{\boldsymbol{\theta}}$ and $\mathbf{U}_{\boldsymbol{\theta}}$ denote the lower and upper bounds for $\boldsymbol{\theta}$, respectively. The sign "≤" is overloaded to represent the entry-wise inequality. The matrices $\mathbf{K}(\boldsymbol{\theta})$ and $\mathbf{M}$ are permutated according the sets $\mathcal{M}$ and $\mathcal{U}$.

A special case leads to a convex optimization problem if all DOFs in the structure are measured, i.e. $\mathcal{M} = \{1, 2, \cdots, N\}$ and $\mathcal{U} = \emptyset$. In this case, $\boldsymbol{\theta}$ is the only optimization variable and the problem becomes a least-squares problem, which is convex. However, in general, $\mathcal{U}$ is non-empty and the problem in Eq. (3) is a nonconvex optimization problem [17]. When the problem is nonconvex, most off-the-shelf optimization algorithms can only find some local optima, which may differ greatly from the global optimum. To address this challenge, we utilize the fact that all functions in Eq. (3) are equivalently polynomials and propose to apply the SOS approach to relax the problem into a convex SDP problem. The SDP problem then can be reliably solved by existing optimization algorithms, such as the interior-point method [21]. It should be clarified that this research investigates modal dynamic residual formulation using noise-free data. More systematic studies are needed in the future regarding the influence of measurement noise on model updating results. One example approach is through regularization with the zero-based correction factors [6].

3 Sum-of-squares (SOS) approach

The SOS approach is applicable for finding the global solutions for polynomial optimization problems. We start by introducing notations used in this approach. The set of integers is denoted by $\mathbb{Z}$, and the set of nonnegative (positive) integers is denoted by $\mathbb{Z}_+$ ($\mathbb{Z}_{++}$). The set of real symmetric matrices of size $n \times n$ is denoted as $\mathbb{S}^n$, and the set of positive semidefinite (definite) matrices of size $n \times n$ is denoted as $\mathbb{S}_+^n$ ($\mathbb{S}_{++}^n$). A matrix $\mathbf{A} \succcurlyeq \mathbf{0}$ ($\mathbf{A} \succ \mathbf{0}$) means that $\mathbf{A} \in \mathbb{S}_+^n$ ($\mathbf{A} \in \mathbb{S}_{++}^n$) is positive semidefinite (definite). The inner product between matrices $\mathbf{A} \in \mathbb{S}^n$ and $\mathbf{B} \in \mathbb{S}^n$ is denoted as $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}(\mathbf{A}^{\mathrm{T}}\mathbf{B}) = \sum a_{ij} b_{ij}$. A monomial in $\mathbf{x} \in \mathbb{R}^n$ is denoted as $\mathbf{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_n) \in \mathbb{Z}_+^n$ representing the corresponding non-negative powers of variables $(x_1, x_2, \cdots, x_n)$. The degree of a monomial is the summation of all the powers $\sum_{i=1}^n \alpha_i$. A polynomial in $\mathbf{x}$ is the linear combination of monomials $f(\mathbf{x}) = \sum_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}}$ with $c_{\boldsymbol{\alpha}} \in \mathbb{R}$ as the coefficient for each monomial. The degree of a polynomial $f(\mathbf{x})$ is the largest degree among its monomials, $\deg(f) = \max_{\boldsymbol{\alpha}}(\sum_{i=1}^n \alpha_i)$.

3.1 Sum-of-squares certificate for nonnegative polynomials

The SOS approach is developed based on the relationship between nonnegative polynomials and the sum of squared polynomials. Nonnegative polynomials are of practical importance in numerous optimization applications. In general, checking whether a given polynomial is nonnegative or not is a hard problem, and there is no efficient algorithm to solve this problem. However, if a polynomial $f(\mathbf{x})$ can be written as a sum of squared polynomials, then $f(\mathbf{x})$ is clearly nonnegative over its domain. This explicit expression of $f(\mathbf{x})$ as a sum of squares (SOS) acts as a certificate of nonnegativity, which gives an immediate proof of the nonnegativity of $f(\mathbf{x})$.

A necessary condition for nonnegativity of a polynomial $f(\mathbf{x})$ is that the degree $\deg(f)$ is even. Considering a polynomial $f(\mathbf{x}) = \sum_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}}$ with even degree $\deg(f) = 2t,\ \ t \in \mathbb{Z}_{++}$, $f(\mathbf{x})$ has a SOS decomposition, i.e. we can use a sum of squared polynomials to represent $f(\mathbf{x})$, if and only if there is a positive semidefinite matrix $\mathbf{W} \succcurlyeq \mathbf{0}$ such that:

$$f(\mathbf{x}) = \mathbf{z}^{\mathrm{T}}(\mathbf{x})\mathbf{W}\mathbf{z}(\mathbf{x}) \tag{4}$$

where $\mathbf{z}(\mathbf{x}) \in \mathbb{R}^{n_{\mathbf{z}}}$ is a vector containing all the base monomials with degree up to t:

$$\mathbf{z}(\mathbf{x}) = (1, x_1, x_2, \cdots, x_n, x_1^2, x_1 x_2, \cdots, x_n^2, \cdots, x_n^t)^{\mathrm{T}}. \tag{5}$$

According to the theory of combinatorics [22], the number of monomials in n variables with degree up to t is $n_{\mathbf{z}} = \binom{n+t}{n}$. The equality in Eq. (4) implies that the polynomials on both sides should have the same coefficient for the same monomial $\mathbf{x}^{\boldsymbol{\alpha}}$. To explicitly describe this coefficient matching condition, we define a group of matrices $\{\mathbf{A}_{\boldsymbol{\alpha}}\}$, and each matrix $\mathbf{A}_{\boldsymbol{\alpha}} \in \mathbb{R}^{n_{\mathbf{z}} \times n_{\mathbf{z}}}$ is an indicator matrix for monomial $\mathbf{x}^{\boldsymbol{\alpha}}$ in the matrix $\mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x})$:

$$(\mathbf{A}_{\boldsymbol{\alpha}})_{\boldsymbol{\beta},\boldsymbol{\gamma}} = \begin{cases} 1 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} = \boldsymbol{\alpha} \\ 0 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} \neq \boldsymbol{\alpha} \end{cases} \tag{6}$$

Here the natural ordering of multi-indices $\boldsymbol{\beta} \in \mathbb{Z}_+^n$ and $\boldsymbol{\gamma} \in \mathbb{Z}_+^n$ are used to index the entries of $\mathbf{A}_{\boldsymbol{\alpha}}$. Note the equality $\mathbf{z}^{\mathrm{T}}(\mathbf{x})\mathbf{W}\mathbf{z}(\mathbf{x}) = \langle \mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x}), \mathbf{W} \rangle$. Fig. 1 shows the coefficient matching condition between $f(\mathbf{x})$ and $\langle \mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x}), \mathbf{W}\rangle$. As shown in the figure, on the left-hand side, the coefficient of $\mathbf{x}^{\boldsymbol{\alpha}}$ in polynomial $f(\mathbf{x})$ is $c_{\boldsymbol{\alpha}}$; on the right-hand side, the coefficient of $\mathbf{x}^{\boldsymbol{\alpha}}$ in polynomial $\langle \mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x}), \mathbf{W}\rangle$ is $\sum_{\boldsymbol{\beta}+\boldsymbol{\gamma}=\boldsymbol{\alpha}} \mathbf{W}_{\boldsymbol{\beta},\boldsymbol{\gamma}}$, which can be written as $\langle \mathbf{A}_{\boldsymbol{\alpha}}, \mathbf{W}\rangle$. The indicator matrix $\mathbf{A}_{\boldsymbol{\alpha}}$ indicates the position of monomial $\mathbf{x}^{\boldsymbol{\alpha}}$ in the matrix $\mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x})$. It is easy to verify that $\mathbf{A}_{\boldsymbol{\alpha}} = \mathbf{A}_{\boldsymbol{\alpha}}^{\mathrm{T}}$, and $\mathbf{A}_{\boldsymbol{\alpha}}$ is sparse which means that only a few entries are one and all the others are zero.

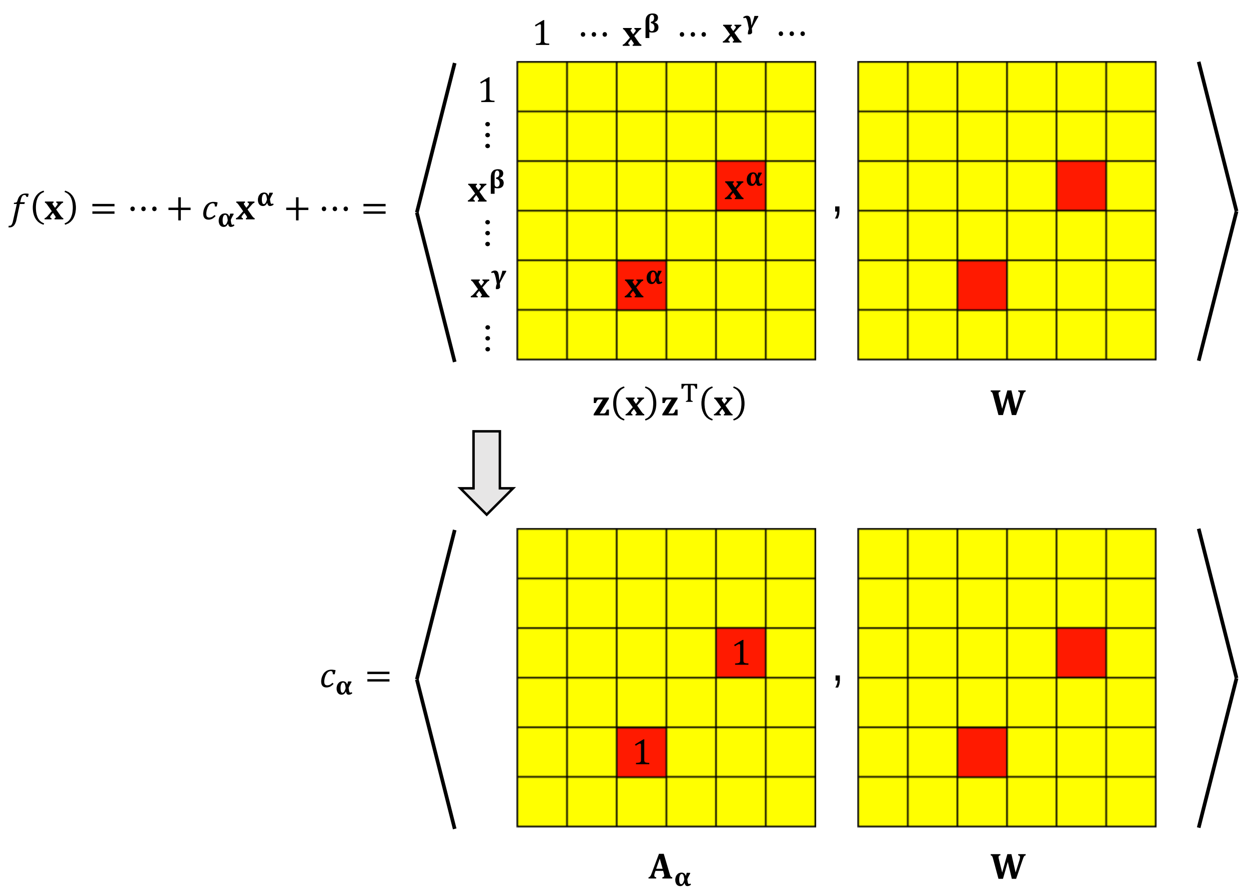


Fig. 1 Coefficient matching

Using the notation shown above, checking whether a given polynomial $f(\mathbf{x})$ has an SOS decomposition can be formulated as a feasibility SDP problem:

$$\begin{aligned} &\text{find} \quad \mathbf{W} \\ &\text{subject to} \quad \langle \mathbf{A}_{\boldsymbol{\alpha}}, \mathbf{W}\rangle = c_{\boldsymbol{\alpha}}, \quad \forall \boldsymbol{\alpha} \text{ in } f(\mathbf{x}) = \textstyle\sum_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} \end{aligned} \tag{7}$$

$$\mathbf{W} \succcurlyeq \mathbf{0}$$

Example: Consider a polynomial $f(\mathbf{x})$ in $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ with degree of $\deg(f) = 2$:

$$f(\mathbf{x}) = 1 + 4x_1 - 6x_2 + 5x_1^2 - 16x_1x_2 + 13x_2^2 \tag{8}$$

The polynomial has six monomials. The power index, monomials, and corresponding coefficients are shown below:

Table 1 Monomials in the polynomial Eq. (8)

$\boldsymbol{\alpha}$	$(0,0)$	$(1,0)$	$(0,1)$	$(2,0)$	$(1,1)$	$(0,2)$
$\mathbf{x}^{\boldsymbol{\alpha}}$	1	x_1	x_2	x_1^2	x_1x_2	x_2^2
$c_{\boldsymbol{\alpha}}$	1	4	-6	5	-16	13

The base monomial vector $\mathbf{z}(\mathbf{x})$ and the matrix $\mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x})$ are shown as:

$$\mathbf{z}(\mathbf{x}) = \begin{Bmatrix} 1 \\ x_1 \\ x_2 \end{Bmatrix} \qquad \mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x}) = \begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & x_1^2 & x_1x_2 \\ x_2 & x_1x_2 & x_2^2 \end{pmatrix}$$

According to Eq. (6), the indicator matrices $\{\mathbf{A}_{\boldsymbol{\alpha}}\}$ are shown as:

$$\mathbf{A}_{(0,0)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{A}_{(1,0)} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{A}_{(0,1)} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

$$\mathbf{A}_{(2,0)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{A}_{(1,1)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \qquad \mathbf{A}_{(0,2)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Solving the feasibility problem in Eq. (7), the solution is calculated as:

$$\mathbf{W} = \begin{pmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ -3 & -8 & 13 \end{pmatrix}$$

Here we use the monomial x_1x_2 to illustrate the coefficient matching. The monomial has power vector $\boldsymbol{\alpha} = (1,1)$, according to which Eq. (6) is used to find matrix $\mathbf{A}_{\boldsymbol{\alpha}}$. The two possible pairs of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ satisfying $\boldsymbol{\beta} + \boldsymbol{\gamma} = \boldsymbol{\alpha}$ are: (*i*) $\boldsymbol{\beta} = (1,0)$ and $\boldsymbol{\gamma} = (0,1)$; (*ii*) $\boldsymbol{\beta} = (0,1)$ and $\boldsymbol{\gamma} = (1,0)$. The two pairs of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are then used to find matrix $\mathbf{A}_{(1,1)}$ as above. Finally, to verify $\langle \mathbf{A}_{\boldsymbol{\alpha}}, \mathbf{W} \rangle = c_{\boldsymbol{\alpha}}$, the coefficient of x_1x_2 in polynomial $\langle \mathbf{z}(\mathbf{x})\mathbf{z}^{\mathrm{T}}(\mathbf{x}), \mathbf{W} \rangle$ is $\langle \mathbf{A}_{(1,1)}, \mathbf{W} \rangle = -16$, which equals the coefficient of x_1x_2 in $f(\mathbf{x})$.

This positive semidefinite matrix $\mathbf{W}$ can be decomposed as $\mathbf{L}^{\mathrm{T}}\mathbf{L}$ by many different decomposition algorithms, such as Cholesky decomposition or eigen decomposition. The Cholesky decomposition provides

$$\mathbf{L} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -2 \end{pmatrix}$$

Thus, the polynomial in Eq. (8) can be written as a sum of two squared polynomials

$$f(\mathbf{x}) = \mathbf{z}^{\mathrm{T}}(\mathbf{x})\mathbf{W}\mathbf{z}(\mathbf{x}) = \big(\mathbf{L}\mathbf{z}(\mathbf{x})\big)^{\mathbf{T}}\big(\mathbf{L}\mathbf{z}(\mathbf{x})\big) = (1 + 2x_1 + 3x_2)^2 + (x_1 - 2x_2)^2$$

3.2 Solving polynomial optimization problems through SDP

Consider a general polynomial optimization problem:

$$
\begin{aligned}
&\underset{\mathbf{x}}{\text{minimize}} \quad f_0(\mathbf{x}) = \sum\nolimits_{\boldsymbol{\alpha}_0} c_{\boldsymbol{\alpha}_0} \mathbf{x}^{\boldsymbol{\alpha}_0} \\
&\text{subject to} \quad f_i(\mathbf{x}) = \sum\nolimits_{\boldsymbol{\alpha}_i} c_{\boldsymbol{\alpha}_i} \mathbf{x}^{\boldsymbol{\alpha}_i} \geq 0 \qquad i = 1, 2, \cdots, k
\end{aligned}
\tag{9}
$$

where $f_0(\mathbf{x})\colon \mathbb{R}^n \to \mathbb{R}$ and $f_i(\mathbf{x})\colon \mathbb{R}^n \to \mathbb{R}$ are polynomials with degrees of $\deg(f_0)$ and $\deg(f_i)$, respectively. We denote the optimal objective function value of the problem in Eq. (9) as f^*. From a "dual" point of view, finding optimal objective value $f_0^* = f_0(\mathbf{x}^*)$ can be equivalently reformulated as solving for the maximum lower bound of the function $f_0(\mathbf{x})$ over the feasible set $\boldsymbol{\Omega} = \{\mathbf{x} \in \mathbb{R}^n | f_i(\mathbf{x}) \geq 0, i = 1, 2, \cdots, k\}$:

$$
\begin{aligned}
&\underset{\gamma}{\text{maximize}} \quad \gamma \\
&\text{subject to} \quad f_0(\mathbf{x}) - \gamma \geq 0 \qquad \forall \mathbf{x} \in \boldsymbol{\Omega}
\end{aligned}
\tag{10}
$$

The optimization problem in Eq. (10) is convex, as the objective function is affine, and the feasible set is defined as an intersection of an infinite number of halfspaces. On the other hand, the constraints in Eq. (10) are intractable because there are infinitely many halfspaces involved. To implement the constraints, the SOS decomposition is utilized [23]. A sufficient condition for $f_0(\mathbf{x}) - \gamma \geq 0$ over the feasible set $\boldsymbol{\Omega}$ is that there exist SOS polynomials $s_0(\mathbf{x})$ and $s_i(\mathbf{x})$, $i = 1, 2, \cdots, k$, satisfying the following SOS decomposition of $f_0(\mathbf{x}) - \gamma$:

$$
f_0(\mathbf{x}) - \gamma = s_0(\mathbf{x}) + \sum\nolimits_{i=1}^{k} s_i(\mathbf{x}) f_i(\mathbf{x}) \tag{11}
$$

Substituting $f_0(\mathbf{x})$ and $f_i(\mathbf{x})$ from Eq. (9), equivalently we obtain:

$$\sum_{\boldsymbol{\alpha}_0} c_{\boldsymbol{\alpha}_0}\mathbf{x}^{\boldsymbol{\alpha}_0} - \gamma = s_0(\mathbf{x}) + \sum_{i=1}^{k} s_i(\mathbf{x}) \sum_{\boldsymbol{\alpha}_i} c_{\boldsymbol{\alpha}_i}\mathbf{x}^{\boldsymbol{\alpha}_i} \tag{12}$$

The function $f_0(\mathbf{x}) - \gamma$ is then represented as a polynomial with degree of $2t$, where t is the smallest integer satisfying the inequality $2t \geq \max_{i=0,1,\cdots,k}\left(\deg(f_i)\right)$. Here the SOS polynomial $s_0(\mathbf{x})$ has degree of $\deg(s_0) = 2t$ and $s_i(\mathbf{x})$ has degree of $\deg(s_i) = 2e_i$, where e_i is the largest integer satisfying the condition $2e_i \leq 2t - \deg(f_i)$. Indeed, if we evaluate the above equation in any $\mathbf{x} \in \boldsymbol{\Omega}$, nonnegativity of the SOS polynomials implies that $f_0(\mathbf{x}) - \gamma \geq 0$. We introduce the indicator matrices $\{\mathbf{A}_{\boldsymbol{\alpha}_0}\}$ and $\{\mathbf{B}_{i,\boldsymbol{\alpha}_0}\}$, $i = 1, 2, \cdots, k$, for coefficient matching. Recall $\boldsymbol{\alpha}_0$ is the variable index in function $f_0(\mathbf{x})$ and $\boldsymbol{\alpha}_i$ is the variable index in function $f_i(\mathbf{x})$ in Eq. (9). Analogous to Eq. (6), each $\mathbf{A}_{\boldsymbol{\alpha}_0}$ and each $\mathbf{B}_{i,\boldsymbol{\alpha}_0}$ are defined as:

$$\begin{aligned} \left(\mathbf{A}_{\boldsymbol{\alpha}_0}\right)_{\boldsymbol{\beta},\boldsymbol{\gamma}} &= \begin{cases} 1 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} = \boldsymbol{\alpha}_0 \\ 0 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} \neq \boldsymbol{\alpha}_0 \end{cases} \\ \left(\mathbf{B}_{i,\boldsymbol{\alpha}_0}\right)_{\boldsymbol{\beta},\boldsymbol{\gamma}} &= \begin{cases} c_{\boldsymbol{\alpha}_i} & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} + \boldsymbol{\alpha}_i = \boldsymbol{\alpha}_0 \\ 0 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} + \boldsymbol{\alpha}_i \neq \boldsymbol{\alpha}_0 \end{cases} \end{aligned} \tag{13}$$

Here the natural ordering of multi-indices $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are used to index the entries of $\mathbf{A}_{\boldsymbol{\alpha}_0}$ and $\mathbf{B}_{i,\boldsymbol{\alpha}_0}$. Representing the inequality constraints in Eq. (10), $f_0(\mathbf{x}) - \gamma \geq 0$, $\forall \mathbf{x} \in \boldsymbol{\Omega}$, through coefficient matching, the optimization problem is then relaxed to an SDP problem with γ, $\mathbf{W}$, and $\mathbf{Q}_i$ as optimization variables:

$$\begin{aligned} &\underset{\gamma,\mathbf{W},\mathbf{Q}_i}{\text{maximize}} \quad \gamma \\ &\text{subject to} \quad \mathbf{A}_{\mathbf{0}} \cdot \mathbf{W} + \sum_{i=1}^{k} \mathbf{B}_{i,\mathbf{0}} \cdot \mathbf{Q}_i = c_{\mathbf{0}} - \gamma \end{aligned} \tag{14}$$

$$\mathbf{A}_{\boldsymbol{\alpha}_0} \cdot \mathbf{W} + \sum_{i=1}^{k} \mathbf{B}_{i,\boldsymbol{\alpha}_0} \cdot \mathbf{Q}_i = c_{\boldsymbol{\alpha}_0}, \qquad \forall \boldsymbol{\alpha}_0 \neq \mathbf{0}$$

$$\mathbf{W} \succcurlyeq \mathbf{0}, \qquad \mathbf{Q}_i \succcurlyeq \mathbf{0}, \qquad i = 1, 2, \cdots, k$$

Solving the SDP problem in Eq. (14) provides the maximum lower bound γ^* for the original optimization problem in Eq. (9). Although it is possible that $\gamma^* < f^*$, in practical applications, this lower bound achieved by SOS relaxation usually reaches the optimal value of the original optimization problem, i.e. $\gamma^* = f^*$ [24].

The minimizer of the original optimization problem in Eq. (9) can be extracted from the solution of the dual problem of the SDP problem in Eq. (14). Define the dual variables, including Lagrangian multiplier vector $\mathbf{y}$ and matrices $\mathbf{U} \succcurlyeq \mathbf{0}$ and $\mathbf{V}_i \succcurlyeq \mathbf{0}$, $i = 1, 2, \cdots, k$. Here we adopt the natural order for the dual variable $\mathbf{y}$ as:

$$\mathbf{y} = \{\mathbf{y}_{\boldsymbol{\alpha}_0}\} = \{y_{(0,0,\cdots,0)}, y_{(1,0,\cdots,0)}, y_{(0,1,\cdots,0)}, \cdots, y_{(0,0,\cdots,1)}, \cdots, y_{(0,0,\cdots,2t)}\}^{\mathrm{T}} \tag{15}$$

The Lagrangian for the problem in Eq. (14) can be written as:

$$\begin{aligned}
\mathcal{L}(\gamma, \mathbf{W}, \mathbf{Q}_i, \mathbf{y}, \mathbf{U}, \mathbf{V}_i) &= \gamma + y_{\mathbf{0}} \left(c_{\mathbf{0}} - \gamma - \langle \mathbf{A}_{\mathbf{0}}, \mathbf{W} \rangle - \sum_{i=1}^{k} \langle \mathbf{B}_{i,\mathbf{0}}, \mathbf{Q}_i \rangle \right) \\
&+ \sum_{\boldsymbol{\alpha}_0 \neq \mathbf{0}} y_{\boldsymbol{\alpha}_0} \left(c_{\boldsymbol{\alpha}_0} - \langle \mathbf{A}_{\boldsymbol{\alpha}_0}, \mathbf{W} \rangle - \sum_{i=0}^{k} \langle \mathbf{B}_{i,\boldsymbol{\alpha}_0}, \mathbf{Q}_i \rangle \right) + \langle \mathbf{U}, \mathbf{W} \rangle + \sum_{i=1}^{k} \langle \mathbf{V}_i, \mathbf{Q}_i \rangle \\
&= \sum_{\boldsymbol{\alpha}_0} c_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} + \gamma(1 - y_{\mathbf{0}}) + \left\langle \mathbf{U} - \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{A}_{\boldsymbol{\alpha}_0}, \mathbf{W} \right\rangle \\
&+ \sum_{i=1}^{k} \left\langle \mathbf{V}_i - \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{B}_{i,\boldsymbol{\alpha}_0}, \mathbf{Q}_i \right\rangle
\end{aligned} \tag{16}$$

The dual function is then formulated as the supremum of the Lagrangian with respect to primal variables γ, $\mathbf{W}$, and $\mathbf{Q}_i$. The dual function is found as:

$$\begin{aligned}\mathcal{D}(\mathbf{y}, \mathbf{U}, \mathbf{V}_i) &= \sup_{\gamma, \mathbf{W}, \mathbf{Q}_i} \mathcal{L}(\gamma, \mathbf{W}, \mathbf{Q}_i, \mathbf{y}, \mathbf{U}, \mathbf{V}_i) \\ &= \begin{cases} \sum_{\boldsymbol{\alpha}_0} c_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} & \text{if } y_{\mathbf{0}} = 1, \mathbf{U} = \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{A}_{\boldsymbol{\alpha}_0}, \mathbf{V}_i = \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{B}_{i,\boldsymbol{\alpha}_0} \\ +\infty & \text{otherwise} \end{cases}\end{aligned} \tag{17}$$

Thus, the dual problem of the SDP problem in Eq. (13) can be written as:

$$\begin{aligned} \underset{\mathbf{y}}{\text{minimize}} \quad & \sum_{\boldsymbol{\alpha}_0} c_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \\ \text{subject to} \quad & y_{\mathbf{0}} = 1 \\ & \mathbf{U} = \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{A}_{\boldsymbol{\alpha}_0} \succcurlyeq \mathbf{0} \\ & \mathbf{V}_i = \sum_{\boldsymbol{\alpha}_0} y_{\boldsymbol{\alpha}_0} \mathbf{B}_{i,\boldsymbol{\alpha}_0} \succcurlyeq \mathbf{0} \qquad i = 1, 2, \cdots, k \end{aligned} \tag{18}$$

It has been shown that if $\gamma^* = f^*$, the optimal solution of the dual problem in Eq. (18) can be calculated as [25]:

$$\mathbf{y}^* = (1, x_1^*, x_2^*, \cdots, x_n^*, \cdots, (x_n^*)^{2t})^{\mathrm{T}} \tag{19}$$

where the entries correspond to the monomials $\mathbf{x}^{\boldsymbol{\alpha}_0}$. Thus, the optimal solution $\mathbf{x}^*$ for the original problem in Eq. (9) can be easily extracted from $\mathbf{y}^*$, as the 2nd to the $(n+1)$th entries. Since most of the primal-dual interior methods simultaneously solve both the primal and dual problems, the optimal solution $\mathbf{x}^*$ can be computed efficiently.

In summary, the SOS approach provides the great theoretical advantage of converting a nonconvex polynomial optimization problem into a convex SDP problem. Thus, all the desirable properties of convex problems can be exploited to analyze and solve the problem. However, the size of the SDP problem remains a significant challenge. Recall that there are n optimization variables in the optimization problem in Eq. (9). In order to achieve the SOS representation in Eq. (11), the degree of SOS polynomial $s_0(\mathbf{x})$ should be $\deg(s_0) = d_0 = 2t$, where t is the smallest integer such that $2t \geq \max_i \deg(f_i)$, $i = 0, 1, 2, \cdots, k$. Similarly, the degree of SOS polynomial $s_i(\mathbf{x})$ should be $\deg(s_i) = d_i = d_0 - e_i$, where e_i is the smallest even integer such that $e_i \geq \deg(f_i)$, $i = 1, 2, \cdots, k$. The sizes of the matrices $\mathbf{W}$ and $\mathbf{Q}_i$ are $\binom{n+d_0}{n} \times \binom{n+d_0}{n}$ and $\binom{n+d_i}{n} \times \binom{n+d_i}{n}$, respectively, and the number of equality constraints in Eq. (14) is $\binom{n+2d_0}{n}$ [17]. The size of the SDP problem in Eq. (14) can be very large as n and/or d_0 grow, making the problem while theoretically convex, practically unsolvable. This difficulty necessitates exploring the problem structure to improve the solvability of the SDP problem.

3.3 Exploring sparsity in the SOS approach

As discussed in the previous section, the SDP problem formulated by the SOS approach can be very computationally expensive when n and/or d_0 are large. One approach taking advantage of the sparsity of the underlying polynomials can be applied to reduce the size of the SDP problem. Here, we examine a specific sparsity pattern that the polynomial objective function consists of several polynomials only involving a small number of variables. Take the modal dynamic residual formulation in Eq. (3) as an example. The total number of optimization variables in Eq. (3) is $n_{\boldsymbol{\theta}} + n_{u} \cdot n_{\text{modes}}$, including the stiffness parameter $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ and the unmeasured entries $\boldsymbol{\psi}_{\mathcal{U}} = \left(\boldsymbol{\psi}_{\mathcal{U},1}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},2}^{\mathrm{T}}, \cdots, \boldsymbol{\psi}_{\mathcal{U},n_{\text{modes}}}^{\mathrm{T}}\right)^{\mathrm{T}} \in \mathbb{R}^{n_{\mathcal{U}} \cdot n_{\text{modes}} \times 1}$ in the mode shapes. Nevertheless, the objective function consists of n_{modes} polynomials, each of which involves only $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ and $\boldsymbol{\psi}_{\mathcal{U},i} \in \mathbb{R}^{n_{\mathcal{U}}}$. This fact enlightens that we can represent each polynomial in SOS form, so that the cross terms between $\boldsymbol{\psi}_{\mathcal{U},i}$ and $\boldsymbol{\psi}_{\mathcal{U},j}$, $i \neq j$, need not to be considered.

Consider a constrained polynomial optimization problem in which the objective function consists of several polynomials:

$$
\begin{aligned}
&\underset{\mathbf{x}}{\text{minimize}} \quad f_0(\mathbf{x}) = \sum_{j=1}^{m} f_{0,j}(\mathbf{x}) = \sum_{j=1}^{m} \sum_{\boldsymbol{\alpha}_{0,j}} c_{\boldsymbol{\alpha}_{0,j}} \mathbf{x}^{\boldsymbol{\alpha}_{0,j}} \\
&\text{subject to} \quad f_i(\mathbf{x}) = \sum_{\boldsymbol{\alpha}_i} c_{\boldsymbol{\alpha}_i} \mathbf{x}^{\boldsymbol{\alpha}_i} \geq 0 \qquad i = 1, 2, \cdots, k
\end{aligned}
\tag{20}
$$

Instead of representing $f_0(\mathbf{x})$ as an SOS directly, each polynomial $f_{0,j}(\mathbf{x})$ is represented as an SOS. Here we only consider the sparsity in the objective function. The condition for $f_0(\mathbf{x}) - \gamma \geq 0$ over the feasible set $\boldsymbol{\Omega}$ is that there exist SOS polynomials $s_{0,j}(\mathbf{x})$, $j = 1, 2, \cdots, m$, and $s_i(\mathbf{x})$, $i = 1, 2, \cdots, k$, satisfying the following SOS decomposition of $f_0(\mathbf{x}) - \gamma$:

$$
f_0(\mathbf{x}) - \gamma = \sum_{j=1}^{m} s_{0,j}(\mathbf{x}) + \sum_{i=1}^{k} s_i(\mathbf{x}) f_i(\mathbf{x}) \tag{21}
$$

Substituting $f_0(\mathbf{x})$ and $f_i(\mathbf{x})$ from Eq. (9), equivalently we have:

$$
\sum_{j=1}^{m} \sum_{\boldsymbol{\alpha}_{0,j}} c_{\boldsymbol{\alpha}_{0,j}} \mathbf{x}^{\boldsymbol{\alpha}_{0,j}} - \gamma = \sum_{j=1}^{m} s_{0,j}(\mathbf{x}) + \sum_{i=1}^{k} s_i(\mathbf{x}) \sum_{\boldsymbol{\alpha}_i} c_{\boldsymbol{\alpha}_i} \mathbf{x}^{\boldsymbol{\alpha}_i} \tag{22}
$$

As the sparsity in $f_i(\mathbf{x})$ is not considered, the indicator matrices $\{\mathbf{B}_{i,\boldsymbol{\alpha}_0}\}$ corresponding to function $s_i(\mathbf{x})$ are the same as those in Eq. (13). The indicator matrices $\{\mathbf{A}_{j,\boldsymbol{\alpha}_0}\}$ corresponding to $s_{0,j}(\mathbf{x})$ are introduced below:

$$\left(\mathbf{A}_{j,\boldsymbol{\alpha}_0}\right)_{\boldsymbol{\beta},\boldsymbol{\gamma}} = \begin{cases} 1 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} = \boldsymbol{\alpha}_0 \\ 0 & \text{if } \boldsymbol{\beta} + \boldsymbol{\gamma} \neq \boldsymbol{\alpha}_0 \end{cases} \tag{23}$$

Note that although we represent each polynomial $f_{0,j}(\mathbf{x})$ as an SOS separately, the equality constraint on coefficient $c_{\boldsymbol{\alpha}_0} = \sum_{j=1}^{m} c_{\boldsymbol{\alpha}_{0,j}}$ should hold for every monomial $\mathbf{x}^{\boldsymbol{\alpha}_0}$ in $f_0(\mathbf{x}) - \gamma$. The SDP problem through the sparse SOS approach can be formulated as:

$$\begin{aligned} \underset{\gamma,\mathbf{W}_j,\mathbf{Q}_i}{\text{maximize}} \quad & \gamma \\ \text{subject to} \quad & \sum\nolimits_{j=1}^{m} \mathbf{A}_{j,\mathbf{0}} \cdot \mathbf{W}_j + \sum\nolimits_{i=1}^{k} \mathbf{B}_{i,\mathbf{0}} \cdot \mathbf{Q}_i = c_{\mathbf{0}} - \gamma \\ & \sum\nolimits_{j=1}^{m} \mathbf{A}_{j,\boldsymbol{\alpha}_0} \cdot \mathbf{W}_j + \sum\nolimits_{i=1}^{k} \mathbf{B}_{i,\boldsymbol{\alpha}_0} \cdot \mathbf{Q}_i = c_{\boldsymbol{\alpha}_0}, \quad \forall \boldsymbol{\alpha}_0 \neq \mathbf{0} \\ & \mathbf{W}_j \succcurlyeq \mathbf{0}, \quad j = 1, 2, \cdots, m \quad \mathbf{Q}_i \succcurlyeq \mathbf{0}, \quad i = 1, 2, \cdots, k \end{aligned} \tag{24}$$

Exploiting the sparsity in model updating problem can greatly improve the computation efficiency of the SOS approach. For instance, in [20] the authors have shown for the same model updating problem on a truss model, the standard SOS approach leads to an SDP problem with 123,256 variables, while the sparse SOS approach leads to an SDP problem with only 39,267 variables, about 1/3 of the original size.

4 Facial reduction for regularizing SDP problems

4.1 Facial reduction

Although the approach taking advantage of the sparsity in the polynomial has been proven useful in reducing the size of the SDP problems, there is still significant room for improvement in computational efficiency. In this section, we focus on the redundancy caused by the failure of the Slater condition (strict feasibility), i.e., when there is no feasible positive definite matrices $\mathbf{W}_j \succ \mathbf{0}$ and $\mathbf{Q}_i \succ \mathbf{0}$ for the SDP problems in Eq. (14) and Eq. (24). When the Slater condition fails for the SDP problem, SDP solvers, especially those based on the interior-point methods, often struggle to find the optimal point. For simplicity of discussion, we use the standard primal form of the SDP problem with $\mathbf{C} \in \mathbb{S}^n$ and $\mathbf{A}_i \in \mathbb{S}^n$:

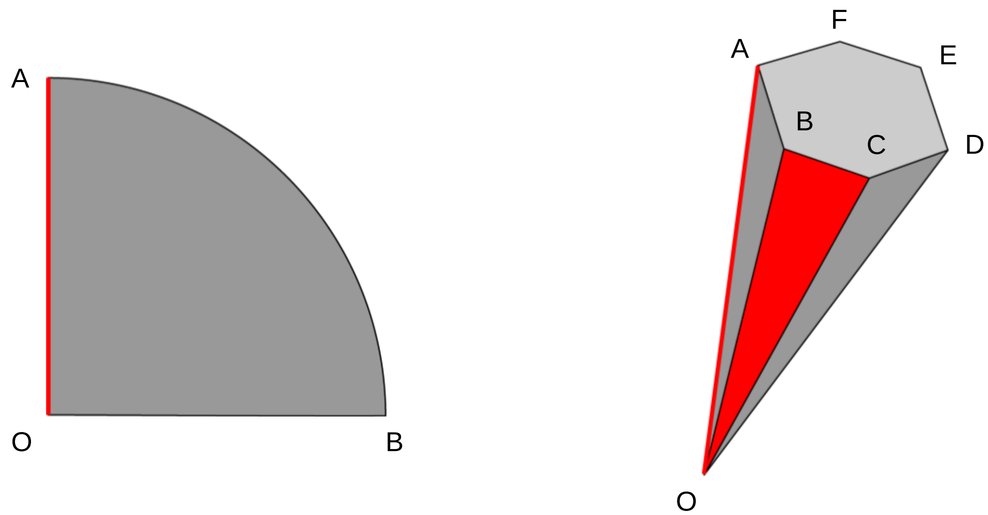
$$\begin{aligned} \underset{\mathbf{X}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{X} \rangle \\ \text{subject to} \quad & \langle \mathbf{A}_i, \mathbf{X} \rangle = b_i \qquad\qquad i = 1, 2, \cdots, k \\ & \mathbf{X} \succcurlyeq \mathbf{0} \end{aligned} \tag{25}$$

The initial step of an interior-point method is finding a strictly feasible point, a positive definite matrix for the SDP problem. If there is no strictly feasible point, i.e. there is no $\mathbf{X} \succ \mathbf{0}$ that satisfies $\langle \mathbf{A}_i, \mathbf{X} \rangle = b_i$, $i = 1, 2, \cdots, k$, the Slater condition qualification fails. As a result, slight perturbation can make the SDP problem infeasible, which increases the difficulty for the numerical algorithms solving the problem. In this case, several techniques can be applied to reformulate SDP problems which fail the Slater condition qualification, such as adding bound constraints [26] and using the homogeneous self-dual embedding method [27]. Here we apply the facial reduction technique to regularize the SDP problems. The idea of facial reduction is to reformulate the SDP problem onto a feasible domain with lower dimension. Thus, the equivalent SDP problem is not only more robust for numerical algorithms to solve but also smaller in size.

Let $\mathbb{C} \subseteq \mathbb{R}^n$ be a convex set. A subset $\mathbb{F} \subseteq \mathbb{C}$ is called a face of $\mathbb{C}$, if and only if

$$\text{For any } \mathbf{X}, \mathbf{Y} \in \mathbb{C} \text{ such that } \frac{\mathbf{X} + \mathbf{Y}}{2} \in \mathbb{F}, \qquad \mathbf{X}, \mathbf{Y} \in \mathbb{F} \text{ holds.}$$

A face $\mathbb{F}$ is a proper face if it is non-empty and not equal to $\mathbb{C}$. Fig. 2 shows examples of proper faces of two convex sets. In Fig. 2(a), the edge OA is a proper face of the convex set in $\mathbb{R}^2$. In Fig. 2(b), both the edge OA and facet OBC are proper faces of the convex set in $\mathbb{R}^3$.



(a) Convex set in $\mathbb{R}^2$ (b) Convex set in $\mathbb{R}^3$

Fig. 2 Proper faces of convex sets

The feasible set of the semidefinite problem in Eq. (25) can be described by the intersection of an affine subspace $\mathcal{A} = \{\mathbf{X} \in \mathbb{S}^n | \langle \mathbf{A}_i, \mathbf{X} \rangle = b_i, i = 1, 2, \cdots, k\}$ with the semidefinite cone $\mathbb{S}_+^n$. If this semidefinite optimization problem is feasible but not strictly feasible, it can be reformulated as an optimization problem over a lower dimensional face (proper face) of $\mathbb{S}_+^n$ [28, 29].

Facial reduction algorithms were first proposed for general conic programming (CP) problems and later found many applications in SDP problems. The goal of facial reduction algorithms is to reformulate an SDP problem as one over the proper face with lower dimension. In the cases of semidefinite problems, finding the proper face containing the feasible set can be achieved by solving a sequence of SDP subproblems, which may be as difficult as solving the original SDP problem. To address these issues, here we adopt an alternative approach to achieve facial reduction by simply inspecting the constraints of the SDP problem [30].

Example: Consider an SDP problem:

$$\begin{aligned} \underset{x_1,x_2,x_3}{\text{minimize}} \quad & \left\langle \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} x_1 & x_2 \\ x_2 & x_3 \end{pmatrix} \right\rangle \\ \text{subject to} \quad & \left\langle \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} x_1 & x_2 \\ x_2 & x_3 \end{pmatrix} \right\rangle = 0 \\ & \begin{pmatrix} x_1 & x_2 \\ x_2 & x_3 \end{pmatrix} \succcurlyeq \mathbf{0} \end{aligned} \tag{26}$$

The equality constraint requires that $x_1 = 0$. The positive semidefinite matrix $\mathbf{X} = \begin{pmatrix} x_1 & x_2 \\ x_2 & x_3 \end{pmatrix} \succcurlyeq \mathbf{0}$ requires that $x_1 x_3 \geq x_2^2$, and thus $x_2 = 0$. As $\mathbf{X} = \begin{pmatrix} 0 & 0 \\ 0 & x_3 \end{pmatrix}$ is not positive definite no matter what value x_3 takes, the Slater condition (strict feasibility) qualification fails. This SDP problem in Eq. (26) can then be reformulated as a regularized SDP problem with lower dimension. In this simple example, the SDP problem degenerates to a linear programming (LP) problem, a special case of SDP problems. Compared to the SDP problem in Eq. (26), the following equivalent problem is regularized and smaller in size.

$$\begin{aligned} \underset{x_3}{\text{minimize}} \quad & x_3 \\ \text{subject to} \quad & x_3 \geq 0 \end{aligned} \tag{27}$$

Geometrically, an SDP problem in Eq. (25) minimizes an affine function $\langle \mathbf{C}, \mathbf{X} \rangle$ over an intersection between an affine subspace (defined by linear equalities $\langle \mathbf{A}_i, \mathbf{X} \rangle = b_i$, $i = 1, 2, \cdots, k$) and a positive semidefinite cone $\mathbf{X} \succcurlyeq \mathbf{0}$. Fig. 3 illustrates the feasible set of the SDP example in Eq. (26). The intersection of the affine subspace $x_1 = 0$ and the positive semidefinite cone $\mathbf{X} \succcurlyeq \mathbf{0}$ is one proper face of the positive semidefinite cone. The proper face is described simply as $x_3 \geq 0$, i.e. a halfline in this $\mathbb{R}^3$ space.

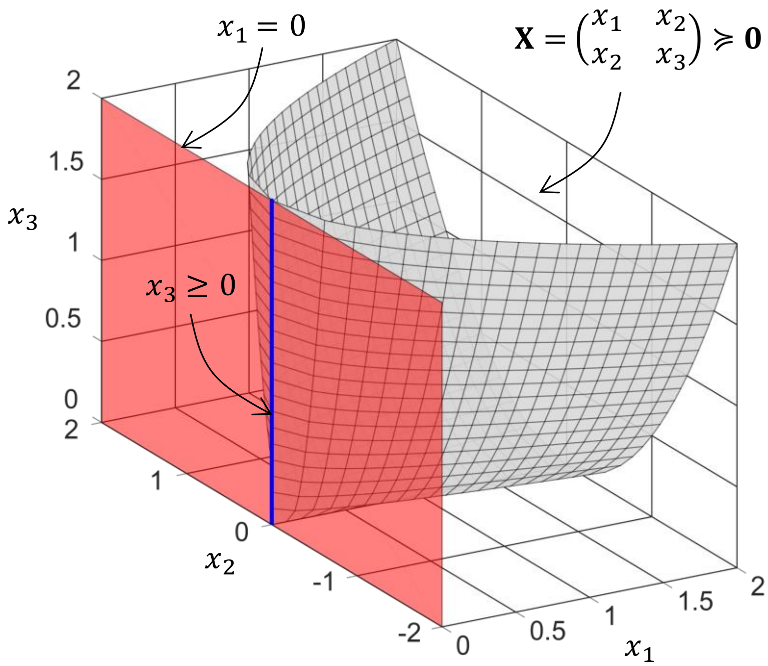


Fig. 3 Feasible set of the SDP example in Eq. (26)

This example motivates and illustrates the approach to achieve facial reduction by inspecting the constraints of the SDP problem. This facial reduction approach is based on the property of a positive semidefinite matrix $\mathbf{X}$ whose leading principal minors are all nonnegative. For the i-th linear equality constraint $\langle \mathbf{A}_i, \mathbf{X} \rangle = b_i$, we first check whether the following equivalent form can be obtained by permutating rows and columns of matrices $\mathbf{A}_i$ and $\mathbf{X}$:

$$\left\langle \begin{pmatrix} \mathbf{D}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \begin{pmatrix} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{pmatrix} \right\rangle = b_i', \quad \text{with } \mathbf{D}_i \succ \mathbf{0} \text{ and } b_i' = b_i \text{ or } b_i' = -b_i. \tag{28}$$

If $b_i' > 0$, no facial reduction can be performed according to this constraint. If $b_i' = 0$, $\mathbf{X}_{11}$ has to be a zero matrix. Therefore, we can eliminate this redundant constraint, and delete all the rows and columns of $\mathbf{X}$ corresponding to the nullified variable $\mathbf{X}_{11}$, and delete the corresponding rows and columns in other matrices $\mathbf{A}_j$, $j \neq i$. Note that if b_i' is found to be negative, then we can claim that the SDP problem is infeasible as there is no such positive semidefinite matrix $\mathbf{X}$ satisfying this constraint. The idea of inspecting constraints to achieve facial reduction is proposed by Zhu, *et al* [30]. This facial reduction technique is named as Sieve-SDP, since a sieve-like structure is finally obtained as the rows and columns are eliminated in the matrices $\mathbf{X}$ and $\mathbf{A}_i$. The implementation of the Sieve-SDP algorithm is available as public domain software `SieveSDP` [30], and we will apply this facial reduction technique to SDP problems arising from FE model updating.

4.2 SDP problems arising from FE model updating

Consider the original model updating problem in Eq. (3), the objective in general is a 4-th order polynomial, and the monomials with degree of 4 are the cross terms $\theta_i^2 \psi_{u,j}^2$, $u \in \mathcal{U}$. The standard SOS approach Eq. (14) generates an SDP problem with redundant constraints. For example, consider coefficient matching equality constraint for monomial $\psi_{u,j}^4$ that does not exist in the objective function. For simplicity in discussion, we denote the index of this monomial as $\boldsymbol{\alpha}_0 = (\mathbf{0}, 4, \mathbf{0})$, which means that only the power of $\psi_{u,j}$ is 4 and others are 0.

$$
\begin{array}{rl}
\mathbf{x} = (& \theta_1, \cdots \theta_{n_{\boldsymbol{\theta}}}, \psi_{\mathcal{U}_1,1}, \cdots \psi_{u,j}, \cdots \psi_{\mathcal{U}_{n_u}, n_{\text{mode}}})^{\mathrm{T}} \\
\boldsymbol{\alpha}_0 = (& 0, \quad \cdots \quad 0, \quad 0, \quad \cdots \quad 4, \quad \cdots \quad 0 \quad)^{\mathrm{T}}
\end{array}
$$

As there is no $\psi_{u,j}^4$ in the objective function $f_0(\mathbf{x})$, the corresponding coefficient $c_{(\mathbf{0},4,\mathbf{0})} = 0$. From coefficient matching in the standard SOS approach, the indicator matrix for monomial $\psi_{u,j}^4$ has only one non-zero entry:

$$
\mathbf{A}_{(\mathbf{0},4,\mathbf{0})} = \begin{pmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{pmatrix}.
$$

Meanwhile, the model updating problem in Eq. (3) only incorporates bounds on $\boldsymbol{\theta}$, and the bounds can be equivalently rewritten as polynomials $f_i(\mathbf{x}) = (\theta_i - L_i)(U_i - \theta_i) \geq 0$. Since $\sum_{i=1}^{n_{\boldsymbol{\theta}}} s_i(\mathbf{x}) f_i(\mathbf{x})$ as in the R.H.S. of Eq. (11) cannot produce monomial $\psi_{u,j}^4$, all the indicator matrices $\mathbf{B}_{i,(\mathbf{0},4,\mathbf{0})} = \mathbf{0}$. Thus, the coefficient matching equality constraint in Eq. (14) for monomial $\psi_{u,j}^4$ is $\langle \mathbf{A}_{(\mathbf{0},4,\mathbf{0})}, \mathbf{W} \rangle = 0$. By Eq. (28), the constraint is redundant and can be eliminated; the corresponding diagonal entry in the matrix variable $\mathbf{W}$ should be zeroed out.

The sparse SOS approach through Eq. (24) generates similar redundancy. When representing the objective function as n_{modes} SOS polynomials, only the *j*-th polynomial contains the monomial $\psi_{u,j}^4$. This fact implies that the indicator matrices in Eq. (24) as:

$$\mathbf{A}_{j,(\mathbf{0},4,\mathbf{0})} = \begin{pmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{pmatrix} \qquad \text{and} \qquad \mathbf{A}_{p,(\mathbf{0},4,\mathbf{0})} = \mathbf{0}, \qquad p \neq j.$$

The indicator matrix corresponding to $s_i(\mathbf{x})$ is $\mathbf{B}_{i,(\mathbf{0},4,\mathbf{0})} = \mathbf{0}$ for the same reason as the standard SOS approach. Thus, the coefficient matching equality constraint in Eq. (24) for monomial $\psi_{u,j}^4$ is $\langle \mathbf{A}_{j,(\mathbf{0},4,\mathbf{0})}, \mathbf{W}_j \rangle = 0$, which is redundant and can be eliminated. The corresponding entry in matrix variable $\mathbf{W}_j$ should be zerod out.

Example: Consider the four-story shear frame structure in [17]. A scalar stiffness updating variable θ represents the relative change from the nominal value of the stiffness parameter of the 4-th story. Another two variables are the unmeasured 4-th entries in the first and second mode shape vectors, $\boldsymbol{\psi}_{\mathcal{U}} = \left(\psi_{4,1}, \psi_{4,2}\right)^{\mathrm{T}}$. The optimization variable vector is $\mathbf{x} = \left(\theta, \psi_{4,1}, \psi_{4,2}\right)^{\mathrm{T}}$. Plugging in the numerical vales of the example structure, the optimization problem is found as follows according the formulation in Eq. (3):

$$\begin{aligned} \underset{\theta,\psi_{4,1},\psi_{4,2}}{\text{minimize}} \; f_0(\mathbf{x}) &= 104.21 + 136.71\theta + 231.65\psi_{4,1} + 106.34\psi_{4,2} + 66.21\theta^2 \\ &+461.18\theta\psi_{4,1} + 89.60\theta\psi_{4,2} + 177.45\psi_{4,1}^2 + 100.14\psi_{4,2}^2 \\ &+376.02\theta\psi_{4,1}^2 + 207.39\theta\psi_{4,2}^2 + 229.53\theta^2\psi_{4,1} - 16.74\theta^2\psi_{4,2} \\ &+200.00\theta^2\psi_{4,1}^2 + 200.00\theta^2\psi_{4,2}^2 \\ \text{subject to} \; f_1(\mathbf{x}) &= 1 - \theta^2 \geq 0 \end{aligned} \tag{29}$$

As shown in Eq. (29), the coefficients for $\psi_{4,1}^4$ and $\psi_{4,2}^4$ are 0, and the coefficient matching equality constraint for these monomials are redundant when using the standard and the sparse SOS approaches. Take the monomial $\psi_{4,2}^4$ as an example. Using the standard SOS approach Eq. (14), the formulated SDP problem has two matrix variables $\mathbf{W}$ and $\mathbf{Q}$ corresponding to SOS polynomials $s_0(\mathbf{x})$ and $s_1(\mathbf{x})$. The indicator matrix of $\psi_{4,2}^4$ for $s_0(\mathbf{x})$ is

$$\mathbf{A}_{(0,0,4)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

As the inequality constraint $f_1(\mathbf{x})$ incorporates only variable θ and the SOS polynomial $s_1(\mathbf{x})$ is a second order polynomial, the polynomial $s_1(\mathbf{x})f_1(\mathbf{x})$ cannot generate the monomial $\psi_{4,2}^4$, and thus the indicator matrix $\mathbf{B}_{(0,0,4)} = \mathbf{0} \in \mathbb{R}^{4\times 4}$. The coefficient matching equality constraint for monomial $\psi_{4,2}^4$ is then found as $\langle \mathbf{A}_{(0,0,4)}, \mathbf{W} \rangle = 0$, which is redundant and can be eliminated.

Using the sparse SOS approach Eq. (24), the formulated SDP problem has three matrix variables $\mathbf{W}_1$, $\mathbf{W}_2$, and $\mathbf{Q}$ corresponding to SOS polynomials $s_{0,1}(\mathbf{x})$, $s_{0,2}(\mathbf{x})$, and $s_1(\mathbf{x})$. As the SOS polynomial $s_{0,1}(\mathbf{x})$ is a function of variable θ and $\psi_{4,1}$, the indicator matrix of $\psi_{4,2}^4$ for $s_{0,1}(\mathbf{x})$ is $\mathbf{A}_{1,(0,0,4)} = \mathbf{0} \in \mathbb{R}^{6\times 6}$. On the other hand, the SOS polynomial $s_{0,2}(\mathbf{x})$ is a function of variables θ and $\psi_{4,2}$, and the indicator matrix of $\psi_{4,2}^4$ for $s_{0,2}(\mathbf{x})$ is

$$\mathbf{A}_{2,(0,0,4)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

The indicator matrix of $\psi_{4,2}^4$ for $s_1(\mathbf{x})$ is $\mathbf{B}_{(0,0,4)} = \mathbf{0} \in \mathbb{R}^{4\times4}$ for the same reason as the standard SOS approach. The coefficient matching equality constraint for monomial $\psi_{4,2}^4$ is then found as $\langle \mathbf{A}_{2,(0,0,4)}, \mathbf{W}_2 \rangle = 0$, which is redundant and can be eliminated.

This example demonstrates that there are many redundancies in the SDP problems formulated by the standard and the sparse SOS approaches. The facial reduction algorithm can be applied to eliminate these types of redundancy and result in more solvable SDP problems.

5 Validation examples

To validate the proposed approaches for model updating and the facial reduction algorithm for regularizing the SDP problems, a plane truss structure, which has been analyzed previously [20], is simulated. The truss model has 10 nodes, and each node has a vertical and a horizontal DOF. All member cross-sectional areas are set as $8\times10^{-5}\ \mathrm{m}^2$, and the material density is set as $7{,}849\ \mathrm{kg/m}^3$. Flexible support conditions are considered in this structure. Vertical and horizontal springs (k_1 and k_2) are allocated at the left support, while a vertical spring (k_3) is allocated at the right support. The Young's moduli of the truss bars are divided into three groups, including E_1 of the top-level truss bars, E_2 of the diagonal and vertical truss bars, and E_3 of the bottom-level truss bars. The mechanical properties of the structure are summarized in Table 2, including the initial/nominal values and the "as-built"/actual values. The same measurement layout, with eight DOFs measured by sensors, is used and illustrated in Fig. 4. Mode shapes extracted from the "experimental" data are only available at these eight measured DOFs. The number of unmeasured DOFs is therefore twelve.

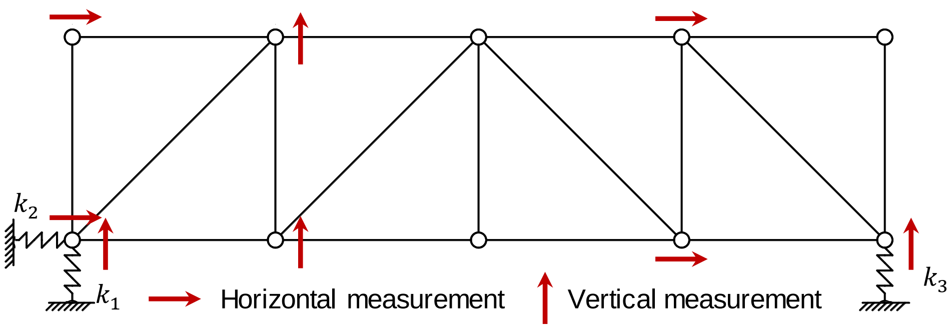


Fig. 4 Plane truss structure with 8 DOFs measured

Table 2. Model updating parameters

Property		Initial/Nominal	"As-built"/Actual	Ideal updating result for θ_i
Young's moduli ($\times 10^{11}$ N/m^2)	Top (E_1)	2	2.2	0.100
	Diagonal & Vertical (E_2)	2	1.8	−0.100
	Bottom (E_3)	2	1.9	−0.050
Springs ($\times 10^6$ N/m)	k_1	6	7	0.167
	k_2	6	3	−0.500
	k_3	6	5	−0.167

5.1 Using the data set involving the first two modes

For comparison with previous results, it is also assumed that only the first two modes (associated with the two lowest resonance frequencies) are available for model updating. Fig. 5 shows the first two resonance frequencies and mode shapes of the plane truss structure.

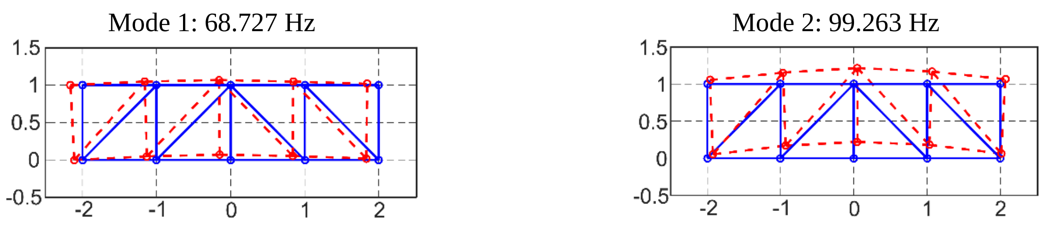


Fig. 5 Modal properties of the plane truss structure

Six stiffness parameters $\boldsymbol{\theta} \in \mathbb{R}^6$ are updated, including three Young's moduli in the structure (E_1, E_2, and E_3) and the spring stiffness values (k_1, k_2, and k_3). The ideal updating result for each θ_i is shown in the last column of Table 2. The lower bound for $\boldsymbol{\theta}$ is set as $\mathbf{L} = -\mathbf{1}_{6\times1}$ and the upper bound is set as $\mathbf{U} = \mathbf{1}_{6\times1}$. The bounds effectively restrict the variation range of the stiffness parameters as $\pm$100%. In total, $n_u = 12$ DOFs of the structure are unmeasured. As per Eq. (3), all unmeasured entries in the two available mode shapes, $\boldsymbol{\psi}_{\mathcal{U}} = \begin{Bmatrix} \boldsymbol{\psi}_{\mathcal{U},1} \\ \boldsymbol{\psi}_{\mathcal{U},2} \end{Bmatrix} \in \mathbb{R}^{24}$, are treated as optimization variables together with $\boldsymbol{\theta}$. The total number of optimization variables is $n = n_{\boldsymbol{\theta}} + n_{\text{modes}} \cdot n_{\mathcal{U}} = 6 + 2 \times 12 = 30$. To minimize the modal dynamic residual r, the model updating problem can be formulated as follows with optimization variables $\mathbf{x} = (\boldsymbol{\theta}, \boldsymbol{\psi}_{\mathcal{U}})$.

$$
\begin{aligned}
&\underset{\mathbf{x}=(\boldsymbol{\theta},\,\boldsymbol{\psi}_{\mathcal{U}})}{\text{minimize}} \quad f(\mathbf{x}) = r = \sum_{j=1}^{2} \left\| \left[\mathbf{K}(\boldsymbol{\theta}) - \omega_j^2 \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{\mathcal{M},j} \\ \boldsymbol{\psi}_{\mathcal{U},j} \end{Bmatrix} \right\|_2^2 \\
&\text{subject to} \quad 1 - \theta_i^2 \geq 0 \qquad\qquad i = 1, 2, \cdots, 6
\end{aligned}
\tag{30}
$$

Using the standard SOS approach, the nonconvex problem in Eq. (30) is recast into an equivalent convex SDP problem Eq. (14). In the SDP problem, optimization variables γ, $\mathbf{W}$, $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ are introduced. The variable γ is a scalar. With $d = 2t = 4$ and $n = 30$, the size of variable $\mathbf{W}$ is $\binom{n+t}{n} \times \binom{n+t}{n} = \binom{30+2}{30} \times \binom{30+2}{30}$=$496 \times 496$. Recall in Eq. (12), e_i is the largest integer satisfying the condition $2e_i \leq 2t - \deg(f_i)$. In this example, with $e_i = 1$, the size of variable $\mathbf{Q}_i$ is $\binom{n+t-e_i}{n} \times \binom{n+t-e_i}{n} = \binom{30+2-1}{30} \times \binom{30+2-1}{30} = 31 \times 31$. In addition, coefficient matching generates $\binom{n+d}{n} = \binom{30+4}{30} = 46{,}376$ linear equality constraints.

It took 457h-16min-49s to solve this SDP problem formulated by the regular SOS approach on a computing clusters using 16 CPUs and 84.56 GB RAM memory [20].

Now we apply the facial reduction technique on the SDP problem. The size of variable $\mathbf{W}$ is reduced to 196×196, about 40% of the original size. The size of variable $\mathbf{Q}_i$ is not changed, remaining as 31×31. As a result, the number of linear equality constraints is reduced to 10,626, about 22% of the original number of 46,376. It took only 1h-55min to solve this SDP problem regularized by the facial reduction technique on the same computing cluster.

Next, we consider applying the sparse SOS approach Eq. (24) to the problem. The objective function of problem in Eq. (30) consists of two polynomials, each of which contains updating parameters $\boldsymbol{\theta}$ and unmeasured entries in the mode shape $\boldsymbol{\psi}_{\mathrm{U},j}$, $j = 1, 2$. Each polynomial has $n_j = 18$ variables and degree of $d_j = 2t_j = 4$. Applying the sparse SOS approach, the nonconvex problem in Eq. (30) can be recast into an equivalent convex SDP problem, with optimization variables γ, $\mathbf{W}_j$ $(j = 1, 2)$, and $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$. The variables γ and $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ share the same size as those produced by the SOS approach. With $d_j = 2t_j = 4$ and $n_j = 18$, the size of variable $\mathbf{W}_j$ is $\binom{n_j + t_j}{n_j} \times \binom{n_j + t_j}{n_j} = \binom{18 + 2}{18} \times \binom{18 + 2}{18}$=$190 \times 190$. The coefficient matching also generates $\binom{n + d}{n} = \binom{30 + 4}{30} = 46{,}376$ linear equality constraints. Solving the SDP problem formulated by the sparse SOS approach, without facial reduction, took 3h-13min-14s on the same computing cluster [20].

Next, we apply the facial reduction technique on the SDP problem. The size of variable $\mathbf{W}_j$ is reduced to 112×112, about 60% as the original size. The size of variable $\mathbf{Q}_i$ is not changed, remaining as 31×31. The number of the linear equality constraints is reduced to 7,602, about 16%

of the original number of 46,376. It took only 10min-53s to solve this SDP problem regularized by the facial reduction algorithm on the same computing cluster.

Table 3 summarizes the updating results obtained from the standard SOS and the sparse SOS approaches with facial reduction. Both approaches can solve the model updating problem with acceptable accuracy. The updating results of the standard SOS and sparse SOS approaches slightly deviate from the ideal updating results $\boldsymbol{\theta}^*$. These numerical inaccuracies of the SDP solutions are inevitable as the problems are solved on double precision floating point SDP solvers [31, 32]. To further refine the updating results, we adopt the data processing method proposed in [33]. The identified parameters from the SDP solutions are used as the initial points and the function `lsqnonlin` in MATLAB Optimization Toolbox [34] is applied to solve the problem in Eq. (30). The updating results shows that the SDP solutions serve as good starting points, and the model updating results from the local optimization solver reach the global optimal solution.

Table 3. Updating results for the structure with the first 2 modes measured at 8 DOFs

Variables	Ideal updating results $\boldsymbol{\theta}^*$	The standard SOS approach $\boldsymbol{\theta}^*_{\text{SOS}}$	The sparse SOS approach $\boldsymbol{\theta}^*_{\text{SpSOS}}$	$\boldsymbol{\theta}^*_{\text{SOS}}$ as initial point	$\boldsymbol{\theta}^*_{\text{SpSOS}}$ as initial point
θ_1	0.100	0.096	0.097	0.100	0.100
θ_2	−0.100	−0.100	−0.102	−0.100	−0.100
θ_3	−0.050	−0.051	−0.052	−0.050	−0.050
θ_4	0.167	0.164	0.164	0.167	0.167
θ_5	−0.500	−0.500	−0.501	−0.500	−0.500
θ_6	−0.167	−0.166	−0.168	−0.167	−0.167

5.2 Using the data set involving the first five modes

In practice, incorporating more experimental modes in model updating usually provides better updating results. However, for SOS approaches, utilizing more modes introduces more optimization variables and makes the SDP problems more difficult to solve. With the help of facial reduction, such unsolvable problems can be simplified to be solvable again. To study this issue, we now assume that the first five modes are available for model updating. Modal properties of the first two modes are already shown in Fig. 5. Resonance frequencies and mode shapes of the third to the fifth modes are consequently shown in Fig. 6.

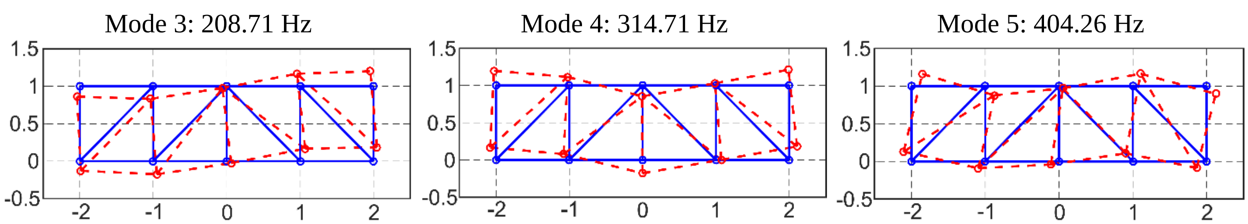


Fig. 6 Modal properties of the plane truss structure

The same stiffness updating variables, $\boldsymbol{\theta} \in \mathbb{R}^6$ corresponding to three Young's moduli in the structure (E_1, E_2, and E_3) and the spring stiffness values (k_1, k_2, and k_3), are updated through the optimization process. To formulate the optimization problem, all unmeasured entries in the five available mode shapes, $\boldsymbol{\psi}_{\mathcal{U}} = \{\boldsymbol{\psi}_{\mathcal{U},1}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},2}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},3}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},4}^{\mathrm{T}}, \boldsymbol{\psi}_{\mathcal{U},5}^{\mathrm{T}}\}^{\mathrm{T}} \in \mathbb{R}^{60}$, are treated as optimization variables together with $\boldsymbol{\theta}$. The total number of optimization variables is $n = n_{\boldsymbol{\theta}} + n_{\text{modes}} \cdot n_{\mathcal{U}} = 6 + 5 \times 12 = 66$, which is notably higher than the example shown in the previous section. The same lower bound and upper bound for $\boldsymbol{\theta}$ are adopted, and the optimization problem can be formulated in a similar way as shown in Eq. (30).

Discussion in Section 5.1 indicates that the sparse SOS approach with facial reduction is the most efficient method to solve the FE model updating problem. For this problem, there are five polynomials of $\left\| [\mathbf{K}(\boldsymbol{\theta}) - \omega_i^2 \mathbf{M}] \begin{Bmatrix} \boldsymbol{\psi}_{\mathcal{M},i} \\ \boldsymbol{\psi}_{\mathcal{U},i} \end{Bmatrix} \right\|_2^2$ in the objective function, each of which involves updating parameters $\boldsymbol{\theta}$ and unmeasured entries in the mode shape $\boldsymbol{\psi}_{\mathrm{U},j}$, $j = 1, \cdots, 5$. Each polynomial has $n_j = 18$ variables and a degree of $d_j = 2t_j = 4$. The sparse SOS approach introduces optimization variables γ, $\mathbf{W}_j$ $(j = 1, \cdots, 5)$, and $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$. The variable γ is a scalar. With $d_j = 2t_j = 4$ and $n_j = 18$, the size of variable $\mathbf{W}_j$ is $\binom{n_j + t_j}{n_j} \times \binom{n_j + t_j}{n_j} = \binom{18+2}{18} \times \binom{18+2}{18}$=190 × 190. With $n = 66$, $t = 2$ and $e_i = 1$, the size of variable $\mathbf{Q}_i$ is $\binom{n + t - e_i}{n} \times \binom{n + t - e_i}{n} = \binom{66+2-1}{66} \times \binom{66+2-1}{66} = 67 \times 67$. The coefficient matching generates $\binom{n+d}{n} = \binom{66+4}{66} = 916{,}895$ linear equality constraints. Without using facial reduction technique, the SDP solver was not able to solve the problem after continuously running for five days on the cluster; at the end, the process was manually terminated due to lack of progress.

On the other hand, when the facial reduction algorithm is applied on the SDP problem, the size of variable $\mathbf{W}_j$ is reduced to 112×112, about 60% of the original size. The size of variable $\mathbf{Q}_i$ is not changed, remaining as 67×67. The number of linear equality constraints is reduced to 26,250, about 3% of the original number. Solution of this regularized SDP problem took 1h-28min-6s on the same computing cluster to successfully complete. Table 4 summarizes the updating results obtained from the sparse SOS approach with facial reduction. Compared to the results with only 2 modes, the updating results with 5 modes available are more accurate. Therefore, it is not necessary to further optimize the parameters using the function `lsqnonlin` as in Section 5.1.

Table 4. Updating results for the structure with the first 5 modes measured at 8 DOFs

Variables	Ideal updating results $\boldsymbol{\theta}^*$	The sparse SOS approach $\boldsymbol{\theta}^*_{\mathrm{SSOS}}$
θ_1	0.100	0.100
θ_2	−0.100	−0.100
θ_3	−0.050	−0.050
θ_4	0.167	0.167
θ_5	−0.500	−0.500
θ_6	−0.167	−0.167

6 Summary and discussion

This paper proposes facial reduction technique for solving the SDP problems in FE model updating. Modal dynamic residual approach formulates a polynomial optimization problem to update the unknown structural parameters as well as unmeasured entries in the mode shape vectors. The SOS approaches convert a polynomial optimization problem into a convex SDP problem, of which the global optimality is guaranteed. Prior work has demonstrated that the standard SOS and the sparse SOS approaches are capable of reliably solving the global optimum for the FE model updating problem. Although the sparse SOS approach exploits the sparsity in the polynomial and greatly reduces the computation effort, the complexity of the SDP problems still remains as a major challenge.

In this paper, we take advantage of the failure of the Slater constraint qualification and apply the facial reduction algorithm to regularize the SDP problems into ones with smaller size. Model updating for a plane truss structure is conducted to validate the proposed approach. The simulation shows that the facial reduction algorithm can efficiently reduce the size of the SDP problems derived from the standard SOS and the sparse SOS approaches. As demonstrated in the first example with two modes available, the facial reduction technique reduces computation time from 457h-16min-49s to 1h-55min for the standard SOS approach, and from 3h-13min-14s to only 10min-53s for the sparse SOS approach. As demonstrated in the second example with five modes available, not mentioning the standard SOS approach, even the sparse SOS approach cannot solve the problem after running continuously for five days; whereas the facial reduction technique finds the solution in 1h-28min-6s. The optimal solutions calculated from the proposed approach are verified to be the global minimum.

While the proposed SOS optimization with facial reduction demonstrates promising performance, the authors acknowledge that current development is not feasible for larger-scale structural models with hundreds of thousands DOFs. Toward larger applications and under realistic constraints on computing resources, model order reduction techniques, such as dynamic condensation [35], can be investigated for incorporation with the proposed formulation using SOS optimization. Another future research direction is extending the proposed SOS approach to include uncertainties. The perturbed model parameters would result in an SDP problem with uncertain data. In this scenario, the robust optimization methodology [36] can be a promising way to find the reliable solution of such an SDP whose data are contaminated with perturbation.

References

[1] B. Moaveni, X. He, J. P. Conte, and J. I. Restrepo, "Damage identification study of a seven-story full-scale building slice tested on the UCSD-NEES shake table," *Structural Safety*, vol. 32, pp. 347-356, 2010.

[2] A. Teughels and G. De Roeck, "Damage detection and parameter identification by finite element model updating," *Revue Européenne de Génie Civil*, vol. 9, pp. 109-158, 2005.

[3] N. Grip, N. Sabourova, and Y. Tu, "Sensitivity-based model updating for structural damage identification using total variation regularization," *Mechanical Systems and Signal Processing*, vol. 84, pp. 365-383, 2017.

[4] B. Jaishi and W. X. Ren, "Damage detection by finite element model updating using modal flexibility residual," *Journal of Sound and Vibration*, vol. 290, pp. 369-387, Feb 2006.

[5] J. E. Mottershead, M. Link, and M. I. Friswell, "The sensitivity method in finite element model updating: a tutorial," *Mechanical Systems and Signal Processing*, vol. 25, pp. 2275-2296, 2011.

[6] A. Fregolent, W. D'Ambrogio, P. Salvini, and A. Sestieri, "Regularisation techniques for dynamic model updating using input residual," *Inverse Problems in Engineering*, vol. 2, pp. 171-200, 1996.

[7] H. Natke, "Updating computational models in the frequency domain based on measured data: a survey," *Probabilistic Engineering Mechanics*, vol. 3, pp. 28-35, 1988.

[8] C. Farhat and F. M. Hemez, "Updating finite element dynamic models using an element-by-element sensitivity methodology," *AIAA Journal,* vol. 31, pp. 1702-1711, 1993.

[9] S. Ibrahim, C. Stavrinidis, E. Fissette, and O. Brunner, "A direct two response approach for updating analytical dynamic models of structures with emphasis on uniqueness," *Journal of Vibration and Acoustics,* vol. 112, p. 5, 1990.

[10] D. Zhu, X. Dong, and Y. Wang, "Substructure stiffness and mass updating through minimization of modal dynamic residuals," *ASCE Journal of Engineering Mechanics,* vol. 142, p. 04016013, 2016.

[11] R. Levin and N. Lieven, "Dynamic finite element model updating using simulated annealing and genetic algorithms," *Mechanical Systems and Signal Processing,* vol. 12, pp. 91-120, 1998.

[12] P. G. Bakir, E. Reynders, and G. De Roeck, "An improved finite element model updating method by the global optimization technique 'Coupled Local Minimizers'," *Computers & Structures,* vol. 86, pp. 1339-1352, 2008.

[13] T. Ma, J. Shi, and D. Steurer, "Polynomial-time tensor decompositions with sum-of-squares," in *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS),* 2016, pp. 438-446.

[14] J. Nie, "Symmetric tensor nuclear norms," *SIAM Journal on Applied Algebra and Geometry,* vol. 1, pp. 599-625, 2017.

[15] A. A. Ahmadi, G. Hall, A. Makadia, and V. Sindhwani, "Geometry of 3D environments and sum of squares polynomials," in *Proceedings of Robotics: Science and Systems 2017*, Cambridge, MA, 2017.

[16] J. Anderson and A. Papachristodoulou, "Advances in computational Lyapunov analysis using sum-of-squares programming," *Discrete & Continuous Dynamical Systems-Series B,* vol. 20, 2015.

[17] D. Li, X. Dong, and Y. Wang, "Model updating using sum of squares (SOS) optimization to minimize modal dynamic residuals," *Structural Control and Health Monitoring,* vol. 25, p. e2263, 2018.

[18] J. B. Lasserre, "Convergent SDP-relaxations in polynomial optimization with sparsity," *SIAM Journal on Optimization,* vol. 17, pp. 822-843, 2006.

[19] H. Waki, S. Kim, M. Kojima, and M. Muramatsu, "Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity," *SIAM Journal on Optimization,* vol. 17, pp. 218-242, 2006.

[20] D. Li and Y. Wang, "Sparse sum-of-squares optimization for model updating through minimization of modal dynamic residuals," *Journal of Nondestructive Evaluation, Diagnostics and Prognostics of Engineering Systems,* vol. 2, p. 011005, 2019.

[21] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review,* vol. 38, pp. 49-95, 1996.

[22] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry* vol. 4: Springer Heidelberg, 2003.

[23] J. B. Lasserre, *An Introduction to Polynomial and Semi-Algebraic Optimization* vol. 52: Cambridge University Press, 2015.

[24] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming,* vol. 96, pp. 293-320, 2003.

[25] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization,* vol. 11, pp. 796-817, 2001.

[26] S. J. Benson, Y. Ye, and X. Zhang, "Solving large-scale sparse semidefinite programs for combinatorial optimization," *SIAM Journal on Optimization,* vol. 10, pp. 443-461, 2000.

[27] S. Zhang, "A new self-dual embedding method for convex programming," *Journal of Global Optimization*, vol. 29, pp. 479-496, 2004.

[28] F. Permenter and P. Parrilo, "Partial facial reduction: simplified, equivalent SDPs via approximations of the PSD cone," *Mathematical Programming,* pp. 1-54, 2014.

[29] Y.-L. Cheung, "Preprocessing and reduction for semidefinite programming via facial reduction: theory and practice," PhD Thesis, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada, 2013.

[30] Y. Zhu, G. Pataki, and Q. Tran-Dinh, "Sieve-SDP: a simple facial reduction algorithm to preprocess semidefinite programs," *Mathematical Programming Computation,* vol. 11, pp. 503-586, 2019.

[31] J. Nie and L. Wang, "Regularization methods for SDP relaxations in large-scale polynomial optimization," *SIAM Journal on Optimization*, vol. 22, pp. 408-428, 2012.

[32] J. B. Lasserre and V. Magron, "In SDP relaxations, inaccurate solvers do robust optimization," *SIAM Journal on Optimization*, vol. 29, pp. 2128-2145, 2019.

[33] P. Biswas, T.-C. Liang, K.-C. Toh, Y. Ye, and T.-C. Wang, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Transactions on Automation Science and Engineering*, vol. 3, pp. 360-371, 2006.

[34] MathWorks Inc., *Optimization Toolbox™ User's Guide*, Version 6. ed. Natick, MA: MathWorks Inc., 2016.

[35] M. Paz, "Dynamic condensation," *AIAA Journal,* vol. 22, pp. 724-727, 1984.

[36] L. El Ghaoui, F. Oustry, and H. Lebret, "Robust solutions to uncertain semidefinite programs," *SIAM Journal on Optimization,* vol. 9, pp. 33-52, 1998.