Sparse Sum-of-Squares Optimization for Model Updating through Minimization of Modal Dynamic Residuals

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

[1] School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, USA
[2] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA

*yang.wang@ce.gatech.edu

Abstract: This research investigates the application of sum-of-squares (SOS) optimization method on finite element model updating through minimization of modal dynamic residuals. The modal dynamic residual formulation usually leads to a nonconvex polynomial optimization problem, the global optimality of which cannot be guaranteed by most off-the-shelf optimization solvers. The sum-of-squares (SOS) optimization method can recast a nonconvex polynomial optimization problem into a convex semidefinite programming (SDP) problem. However, the size of the SDP problem can grow very large, sometimes with hundreds of thousands of variables. To improve the computation efficiency, this study exploits the sparsity in SOS optimization to significantly reduce the size of the SDP problem. A numerical example is provided to validate the proposed method.

Key words: Sum-of-squares (SOS) optimization, sparse SOS, modal dynamic residual approach, finite element model updating

1 Introduction

Finite element (FE) model updating refers to methods and techniques to improve and fine-tune a numerical structural model, based on experimental measurements from the as-built structure. By minimizing the discrepancies between the characteristics of an as-built structure and its FE model, model updating can achieve higher simulation accuracy. Various FE model updating algorithms have been investigated and applied in practice. Generally, these algorithms can be categorized into two groups: time-domain approaches and frequency-domain approaches. The time-domain approaches directly utilize the measured time history data for model updating. Among these approaches, the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) have shown good performance on structural parameter identification (Ebrahimian *et al.* 2015; Hoshiya and Saito 1984; Wu and Smyth 2007; Yang *et al.* 2006). Other approaches, such as Bayesian approach, have also been reported for FE model updating (Astroza *et al.* 2017). On the other hand, the frequency-domain approaches conduct model updating using frequency-domain structural properties extracted from measured structural responses, such as acceleration, velocity, and displacement. The extracted modal properties can be utilized to update the FE model so that the model generates similar modal properties.

This paper focuses on frequency-domain approaches, which usually formulate an optimization problem that minimizes the difference between experimental and simulated modal properties. Early researchers in FE model updating field attempted to obtain better agreement between simulated resonance frequencies and those extracted from the field measurement data. Although these approaches are straightforward and easy to implement, only using the resonance frequency data could not ensure successful model updating (Salawu 1997). Alternatively, other modal properties, such as mode shapes and modal flexibility, are included in the optimization objective function to utilize more information and thus provide better updating results (Jaishi and Ren 2006; Koh and Shankar 2003; Moaveni *et al.* 2013; Nozari *et al.* 2017; Sanayei *et al.* 2001; Zhang and Johnson 2013). To this end, the modal dynamic residual approach accomplishes FE model updating by forming an optimization problem that minimizes the residuals of the generalized eigenvalue equations in structural dynamics (Farhat and Hemez 1993; Kosmatka and Ricles 1999; Zhu *et al.* 2016). Despite previous efforts, these optimization problems in FE model updating are generally nonconvex. Most off-the-shelf optimization algorithms, including gradient search methods and trust-region methods, can only find some local optima, while providing little or no knowledge on the global optimality.

Although the optimization problems in model updating are generally nonconvex, the objective function, as well as equality and inequality constraints, can be formulated as polynomial functions. Each polynomial function is a summation of monomial functions with the same or a lower degree. For example, polynomial function $1 - 4x_1 + 6x_2 + 8x_1^2 - 8x_1x_2 + 10x_2^2$ has degree two, and has contribution from monomials 1, x_1, x_2, x_1^2, x_1x_2, and x_2^2, while each monomial has degree zero, one, or two. If an optimization problem has polynomial objective and constraint functions, it becomes possible to find the global optimum of the nonconvex problem by sum-of-squares (SOS) optimization method. The SOS method tackles the problem by decomposing the original objective function into SOS polynomials to find the best lower bound of the objective function, which makes the problem more solvable. In recent years, extensive research efforts have been dedicated to SOS method for calculating the global bounds of polynomial functions (Nie *et al.* 2006; Parrilo 2003). It has also been reported that the dual problem of the SOS formulation provides information about the optimal solution of the original polynomial optimization problem (Henrion and Lasserre 2005; Lasserre 2001; Laurent 2009). Utilizing primal and dual problems of SOS optimization, the authors found that the global optimum can be reliably solved for nonconvex model updating problems using the modal dynamic residual formulation (Li *et al.* 2018).

While our previous work shows the SOS optimization method is promising in solving nonconvex polynomial problems, the formulated optimization problem can be very expensive to solve. Therefore, only a 4-DOF lumped mass example was presented. For the model updating of larger structures, the number of variables or the degrees of the polynomial can become exponentially larger. To address this challenge, some researchers in mathematics community have recently investigated the so-called sparse SOS optimization method, which exploits the sparsity in the polynomial objective or constraint functions. Here sparsity refers to the property of a polynomial function (of certain degree) that contains a relatively small number of monomials with nonzero coefficient. For example, $-4x_1 + 10x_2^2$ can be considered as a sparse polynomial (function). The sparsity in objective or constraint polynomials is found to significantly reduce the computation load (Nie and Demmel 2008). Leveraging this recent progress, the paper exploits sparse SOS optimization method to reduce the number of optimization variables in the modal dynamic residual approach towards model updating. To this end, the paper demonstrates the model updating of a larger 2D truss structure with sparse SOS optimization.

The rest of this paper is organized as follows. Section 2 summarizes the formulation of the modal dynamic residual approach for model updating. Section 3 briefly reviews the SOS optimization method and its application on modal dynamic residual approach. Section 4 investigates the application of sparsity in SOS method to reduce the size of the corresponding optimization problem for model updating with modal dynamic residual approach. Section 5 shows numerical simulation of a 2D truss that demonstrates the advantage of the sparse SOS method. In the end, Section 6 provides a summary and discussion.

2 Modal dynamic residual approach for FE model updating

The purpose of FE model updating is to identify an accurate numerical model of an as-built preexisting structure using measurement data from the structure. For brevity, only stiffness values are considered as updating parameters in this paper (although the formulation can be easily extended for updating mass and damping). The stiffness parameters can be represented by updating variable $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$, where each entry θ_i denotes the relative change from the initial/nominal value of the *i-th* stiffness parameter being updated. For a linear elastic structure with N degrees of freedom (DOFs), the overall stiffness matrix can be written as an affine matrix function of the updating variable $\boldsymbol{\theta}$:

$$\mathbf{K}(\boldsymbol{\theta}) = \mathbf{K}_0 + \sum_{i=1}^{n_{\boldsymbol{\theta}}} \theta_i \mathbf{K}_{0,i} \tag{1}$$

where $\mathbf{K}_0 \in \mathbb{R}^{N \times N}$ denotes the initial stiffness matrix prior to model updating; $\mathbf{K}_{0,i} \in \mathbb{R}^{N \times N}$ denotes the *i*-th (constant) stiffness influence matrix corresponding to the contribution of the *i*-th stiffness parameter being updated. Finally, $\mathbf{K}(\boldsymbol{\theta}): \mathbb{R}^{n_{\boldsymbol{\theta}}} \to \mathbb{R}^{N \times N}$ represents that the structural stiffness matrix is written as an affine matrix function of vector variable $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$. When not all the stiffness parameters need updating, it is not required that $\mathbf{K}_0 = \sum_{i=1}^{n_{\boldsymbol{\theta}}} \mathbf{K}_{0,i}$.

In theory, given the resonance frequency ω_i and mode shape vector $\boldsymbol{\psi}_i$, no other value of the updating variable $\boldsymbol{\theta}$ except the actual/correct value can provide the exact stiffness matrix $\mathbf{K}$ that satisfies the generalized eigenvalue equation:

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

Based on this concept, the modal dynamic residual approach achieves model updating by minimizing the residual of the generalized eigenvalue equation of structural dynamics in Eq. (2). The residual can be calculated using the matrices generated by the FE model and modal properties obtained by experiment. The stiffness matrix $\mathbf{K}(\boldsymbol{\theta})$ is parameterized by updating variable $\boldsymbol{\theta}$. For brevity, the mass matrix $\mathbf{M}$ is considered as accurate and requires no updating. The modal properties usually include the first few resonance frequencies (ω_i, $i = 1, 2, \cdots, n_{\text{modes}}$) and corresponding mode shapes. Here, n_{modes} denotes the number of measured modes. For mode shapes, the experimental results can only include the measured DOFs, denoted as $\boldsymbol{\psi}_{i,\text{m}}$. As for entries in mode shapes corresponding to the unmeasured DOFs, $\boldsymbol{\psi}_{i,\text{u}}$ are unknown and needs to be treated as optimization variables. The optimization problem is formulated as follows to minimize modal dynamic residual r, where the optimization variables are $\boldsymbol{\theta}$ and $\boldsymbol{\psi}_{i,\text{u}}$:

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

Here $\|\cdot\|_2$ denotes the $\mathcal{L}_2$-norm; constant vectors $\mathbf{L}$ and $\mathbf{U}$ denote the lower bound and upper bound for vector $\boldsymbol{\theta}$, respectively. Note that the sign "$\le$" is overloaded to represent entry-wise inequality. The formulation implies that both $\mathbf{K}(\boldsymbol{\theta})$ and $\mathbf{M}$ are reordered by the instrumented and un-instrumented DOFs in $\boldsymbol{\psi}_{i,\text{m}}$ and $\boldsymbol{\psi}_{i,\text{u}}$.

3 Sum-of-squares (SOS) optimization method

The sum-of-squares (SOS) optimization method is applicable to polynomial optimization problems. The core idea of this method is to represent nonnegative polynomials in terms of a sum of squared polynomials. Using the SOS method, many nonconvex polynomial optimization problems can be recast as convex SDP problems, for which the global optimum can be reliably solved.

3.1 Nonnegative polynomials

A monomial $m(\mathbf{x})$: $\mathbb{R}^n \to \mathbb{R}$ is defined as the product form below:

$$m(\mathbf{x}) = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} \tag{4}$$

where $\alpha_i \in \mathbb{Z}_+$ (the nonnegative integer set) is the exponent of each variable. The degree of a monomial is calculated as $\sum_{i=1}^{n} \alpha_i$. With $c_k \in \mathbb{R}$ as the real-valued coefficient, a polynomial $p(\mathbf{x}): \mathbb{R}^n \to \mathbb{R}$ is defined as a linear combination of monomials:

$$p(\mathbf{x}) = \sum_{k=1}^{n_p} c_k x_1^{\alpha_{k,1}} x_2^{\alpha_{k,2}} \cdots x_n^{\alpha_{k,n}} \tag{5}$$

where $n_p \in \mathbb{Z}_{++}$ (the positive integer set) is the number of monomials; $\alpha_{k,i} \in \mathbb{Z}_+$ is the exponent of each variable. The degree $d \in \mathbb{Z}_+$ of the polynomial $p(\mathbf{x})$ refers to the highest degree of its constituting monomials, $d = \max_k \left(\sum_{i=1}^{n} \alpha_{k,i} \right)$.

A large variety of optimization problems involve positive semidefinite (PSD) polynomials. A polynomial $p(\mathbf{x})$ with even degree of $d = 2t$ is called PSD if $p(\mathbf{x}) \geq 0$ for any $\mathbf{x} \in \mathbb{R}^n$. However, except for limited cases, e.g. $n = 1$ or $d = 2$, it is very difficult to test whether a given polynomial $p(\mathbf{x})$ is PSD or not. Alternatively, a sufficient condition for a polynomial to be PSD is that $p(\mathbf{x})$ can be expressed as a sum-of-squares (SOS) form $p(\mathbf{x}) = \sum_i s_i^2(\mathbf{x})$ for a finite number of polynomials $s_i: \mathbb{R}^n \to \mathbb{R}$. Consider the vector including all the base monomials of degree $t \in \mathbb{Z}_{++}$ or lower:

$$\mathbf{z}(\mathbf{x}) = (1, x_1, x_2, \cdots, x_n, x_1^2, x_1 x_2, \cdots, x_{n-1} x_n^{t-1}, x_n^t)^{\mathrm{T}} \in \mathbb{R}^{n_{\mathbf{z}}} \tag{6}$$

According to combination theory, the number of base monomials in n variables of degree t or lower is $n_{\mathbf{z}} = \binom{n+t}{n}$ (Basu *et al.* 2003). Any polynomial, regardless being PSD or not, can be expressed in a quadratic form using the base monomial vector $\mathbf{z}(\mathbf{x})$ (Lall 2011):

$$p(\mathbf{x}) = \sum_{k=1}^{n_p} c_k x_1^{\alpha_{k,1}} x_2^{\alpha_{k,2}} \cdots x_n^{\alpha_{k,n}} = \mathbf{z}(\mathbf{x})^{\mathrm{T}} \mathbf{W} \mathbf{z}(\mathbf{x}) \tag{7}$$

where $\mathbf{W} \in \mathbb{S}^{n_{\mathbf{z}}}$ is a constant coefficient matrix determined by the coefficients c_k and $\mathbb{S}$ denotes the set of real symmetric matrices. The condition that $p(\mathbf{x})$ has a SOS decomposition turns out to be equivalent to that $\mathbf{W} \succcurlyeq 0$ is a positive semidefinite matrix (Nesterov 2000; Parrilo 2000):

$$p(\mathbf{x}) = \sum_i s_i^2(\mathbf{x}) \quad \Leftrightarrow \quad p(\mathbf{x}) = \mathbf{z}(\mathbf{x})^{\mathrm{T}}\mathbf{W}\mathbf{z}(\mathbf{x}), \mathbf{W} \succcurlyeq 0 \tag{8}$$

A polynomial $p(\mathbf{x})$ is called SOS if $p(\mathbf{x})$ has a SOS decomposition. Recall that t is the highest degree among monomials in $\mathbf{z}(\mathbf{x})$, d is the even-valued degree of $p(\mathbf{x})$. The equality $p(\mathbf{x}) = \mathbf{z}(\mathbf{x})^{\mathrm{T}}\mathbf{W}\mathbf{z}(\mathbf{x})$ thus requires $d = 2t$. Testing whether a given polynomial $p(\mathbf{x})$ is SOS can be formulated as a SDP problem:

$$\begin{aligned} \text{find} \quad & \mathbf{W} \\ \text{subject to} \quad & p(\mathbf{x}) = \mathbf{z}(\mathbf{x})^{\mathrm{T}}\mathbf{W}\mathbf{z}(\mathbf{x}) \\ & \mathbf{W} \succcurlyeq 0 \end{aligned} \tag{9}$$

The identity in Eq. (9) is an equality constraint that holds for arbitrary $\mathbf{x}$, which essentially says two sides of the equation should have the same coefficient c_k for the same base monomial $m_k(\mathbf{x}) = x_1^{\alpha_{k,1}} x_2^{\alpha_{k,2}} \cdots x_n^{\alpha_{k,n}}$. We use n_p to represent the number of monomials in n variables of degree $d = 2t$ or lower, i.e. $n_p = \binom{n+d}{n}$. Thus, the equality constraint is effectively a group of n_p affine equality constraints on the entries of $\mathbf{W}$. We use $\langle \cdot, \cdot \rangle$ to represent the matrix inner product and denote $\mathbf{z}(\mathbf{x})^{\mathrm{T}}\mathbf{W}\mathbf{z}(\mathbf{x}) = \langle \mathbf{z}(\mathbf{x})\mathbf{z}(\mathbf{x})^{\mathrm{T}}, \mathbf{W} \rangle$. These equality constraints can then be explicitly expressed using constant selection matrices $\mathbf{A}_k \in \mathbb{S}^{n_{\mathbf{z}}}$, which has one in entries where $m_k(\mathbf{x})$ appears in matrix $\mathbf{z}(\mathbf{x})\mathbf{z}(\mathbf{x})^{\mathrm{T}}$ and zero otherwise. In other words, $\mathbf{A}_k$ selects $m_k(\mathbf{x})$ out from the matrix $\mathbf{z}(\mathbf{x})\mathbf{z}(\mathbf{x})^{\mathrm{T}}$. Using the selection matrices, the feasibility problem in Eq. (9) can be equivalently rewritten as:

$$\begin{aligned} \text{find} \quad & \mathbf{W} \\ \text{subject to} \quad & \langle \mathbf{A}_k, \mathbf{W} \rangle = c_k, k = 1, 2, \cdots, n_p \\ & \mathbf{W} \succcurlyeq 0 \end{aligned} \tag{10}$$

Illustration: An example is provided here to better illustrate SOS decomposition. Consider polynomials $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ with $n = 2$. The following SOS polynomial $p(\mathbf{x})$ has an even degree $d = 2$:

$$p(\mathbf{x}) = 1 - 4x_1 + 6x_2 + 8x_1^2 - 8x_1x_2 + 10x_2^2$$

This polynomial contains $n_p = \binom{n+d}{n}$=6 monomials. The monomials and corresponding coefficients are shown below:

$m_1(\mathbf{x}) = 1$	$m_2(\mathbf{x}) = x_1$	$m_3(\mathbf{x}) = x_2$	$m_4(\mathbf{x}) = x_1^2$	$m_5(\mathbf{x}) = x_1 x_2$	$m_6(\mathbf{x}) = x_2^2$
$c_1 = 1$	$c_2 = -4$	$c_3 = 6$	$c_4 = 8$	$c_5 = -8$	$c_6 = 10$

To express $p(\mathbf{x})$ in a quadratic form, the vector $\mathbf{z}(\mathbf{x})$ including the base monomials is defined following Eq. (6). The highest degree of monomials in $\mathbf{z}(\mathbf{x})$ is $t = d/2 = 1$, and the length $n_{\mathbf{z}} = \binom{n+t}{n} = 3$.

$$\mathbf{z}(\mathbf{x}) = (1, x_1, x_2)^{\mathrm{T}}$$

For illustrating the equality constraint in Eq. (10), take $k = 5$ and the monomial $m_5(\mathbf{x}) = x_1 x_2$ as an example. The constant selection matrix which selects $m_5(\mathbf{x}) = x_1 x_2$ out from the matrix $\mathbf{z}(\mathbf{x})\mathbf{z}(\mathbf{x})^{\mathrm{T}}$ is shown below:

$$\mathbf{A}_5 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Solving the feasibility problem in Eq. (10), a particular solution is found as:

$$\mathbf{W} = \begin{bmatrix} 1 & -2 & 3 \\ -2 & 8 & -4 \\ 3 & -4 & 10 \end{bmatrix}$$

The entries $W_{2,3} = W_{3,2} = -4$ correspond to the coefficient c_5 of monomial $m_5(\mathbf{x}) = x_1 x_2$, which is why $c_5 = \langle \mathbf{A}_5, \mathbf{W} \rangle = W_{2,3} + W_{3,2} = -8$. The positive semidefinite matrix $\mathbf{W}$ can be decomposed as $\mathbf{L}^{\mathrm{T}}\mathbf{L}$ by many decomposition methods, such as eigen-decomposition or Cholesky decomposition. For example, Cholesky decomposition provides

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

Finally, the polynomial can be written as the sum of squared polynomials:

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

3.2 Polynomial optimization problem

By the means of SOS decomposition, many difficult polynomial optimization problems can be relaxed to more solvable ones. Now consider a constrained polynomial optimization problem:

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) = \sum_{k=1}^{n_f} c_k m_k(\mathbf{x}) \\ &\text{subject to} \quad g_i(\mathbf{x}) = \sum_{j=1}^{n_{g_i}} h_{i,j} m_{i,j}(\mathbf{x}) \geq 0, i = 1,2,\cdots,l \end{aligned} \tag{11}$$

where $f(\mathbf{x}) \colon \mathbb{R}^n \to \mathbb{R}$ and $g_i(\mathbf{x}) \colon \mathbb{R}^n \to \mathbb{R}$ are polynomials with degree d and $e_i \in \mathbb{Z}_{++}$, respectively; $m_k(\mathbf{x})$ is the *k*-th monomial in $f(\mathbf{x})$ and $m_{i,j}(\mathbf{x})$ is the *j*-th monomial in $g_i(\mathbf{x})$. We denote the optimal objective function value of the problem Eq. (11) as f^*. In general, the optimization problem in Eq. (11) is a nonconvex problem. To cast this optimization problem to a convex one, we search for the best (maximum possible) lower bound γ of the objective function $f(\mathbf{x})$ over the feasible set $\mathbf{\Omega} = \{\mathbf{x} \in \mathbb{R}^n | g_i(\mathbf{x}) \geq 0, i = 1, 2, \cdots, l\}$:

$$\begin{aligned} &\underset{\gamma}{\text{maximize}} \quad \gamma \\ &\text{subject to} \quad f(\mathbf{x}) - \gamma \geq 0, \forall \mathbf{x} \in \mathbf{\Omega} \end{aligned} \tag{12}$$

Note that $\mathbf{x}$ is no longer an optimization variable for the problem in Eq. (12) but acts as a constraint on γ. For each $\mathbf{x} \in \mathbf{\Omega}$, $f(\mathbf{x}) - \gamma \geq 0$ is an affine, and thus convex constraint of γ. Because the feasible set of γ in Eq. (12) is the intersection of infinite number of convex sets on γ, this optimization problem is convex on γ (Boyd and Vandenberghe 2004). Although the optimization problem has been converted to a convex one, it is still yet to implement the constraint that the polynomial $f(\mathbf{x}) - \gamma$ is nonnegative for all $\mathbf{x} \in \mathbf{\Omega}$. To make the constraint easier to implement, the SOS decomposition is utilized. With the feasible set $\mathbf{\Omega}$ involved, the sufficient condition for $f(\mathbf{x}) - \gamma \geq 0$ over $\mathbf{\Omega}$ is that there exist SOS polynomials $p_0(\mathbf{x}) = \mathbf{z}_0(\mathbf{x})^{\mathrm{T}} \mathbf{W} \mathbf{z}_0(\mathbf{x})$, and $p_i(\mathbf{x}) = \mathbf{z}_i(\mathbf{x})^{\mathrm{T}} \mathbf{Q}_i \mathbf{z}_i(\mathbf{x})$, $i = 1, 2, \cdots, l$, that satisfy the following condition:

$$f(\mathbf{x}) - \gamma = p_0(\mathbf{x}) + \sum_{i=1}^{l} p_i(\mathbf{x}) g_i(\mathbf{x}) \tag{13}$$

where $\mathbf{W} \succcurlyeq 0 \in \mathbb{S}_+^{n_{\mathbf{z}_0}}$ and $\mathbf{Q}_i \succcurlyeq 0 \in \mathbb{S}_+^{n_{\mathbf{z}_i}}$ are positive semidefinite matrices. To make sure the equality in Eq. (13) hold, we express both sides of the equation as polynomials with degree of $2t \geq \max(d, e_1, \cdots, e_l)$. Recall that d is the degree of $f(\mathbf{x})$ and e_i is the degree of $g_i(\mathbf{x})$. On the left-hand side, if the degree d of

$f(\mathbf{x})-\gamma$ is smaller than $2t$, the monomials with degree larger than d are simply assigned as zero coefficients. Thus, the total number of monomials from both sides of Eq. (13) is regarded as $n_f = \binom{n+2t}{n}$. On the right-hand side of Eq. (13), to ensure the degree of $p_0(\mathbf{x})$ is no more than $2t$, we define the vector $\mathbf{z}_0(\mathbf{x}) = (1, x_1, x_2, \cdots, x_n, x_1^2, x_1x_2, \cdots, x_{n-1}x_n^{t-1}, x_n^t)^{\mathrm{T}} \in \mathbb{R}^{n_{\mathbf{z}_0}}$ to represent all the base monomials of degree $t \in \mathbb{R}$ or lower. The length of $\mathbf{z}_0(\mathbf{x})$ is $n_{\mathbf{z}_0} = \binom{n+t}{n}$. To ensure the degree of each product $p_i(\mathbf{x})g_i(\mathbf{x})$ is no more than $2t$, $\mathbf{z}_i(\mathbf{x}), i = 1, 2, \cdots, l$, is defined as the vector including all the base monomials of degree $t - \tilde{e}_i$ or lower, where $\tilde{e}_i = \lceil e_i/2 \rceil$ represents the smallest integer larger than or equal to $e_i/2$. The length of $\mathbf{z}_i(\mathbf{x})$ is $n_{\mathbf{z}_i} = \binom{n+t-\tilde{e}_i}{n}$. In this way, the optimization problem described in Eq. (12) can be relaxed to:

$$\begin{aligned} &\underset{\gamma,\mathbf{W},\mathbf{Q}_i}{\text{maximize}} && \gamma \\ &\text{subject to} && f(\mathbf{x}) - \gamma = \mathbf{z}_0(\mathbf{x})^{\mathrm{T}}\mathbf{W}\mathbf{z}_0(\mathbf{x}) + \sum_{i=1}^{l} \left(\mathbf{z}_i(\mathbf{x})^{\mathrm{T}}\mathbf{Q}_i\mathbf{z}_i(\mathbf{x})\right) g_i(\mathbf{x}) \\ & && \mathbf{W} \succcurlyeq 0, \mathbf{Q}_i \succcurlyeq 0,\ i = 1, 2, \cdots, l \end{aligned} \tag{14}$$

To express the equality constraints explicitly, we introduce the selection matrices $\mathbf{A}_k$ and $\mathbf{B}_{i,k}$ $(i = 1, 2, \cdots, l)$. $\mathbf{A}_k \in \mathbb{S}^{n_{\mathbf{z}_0}}$ has one in entries where $m_k(\mathbf{x})$ appears in matrix $\mathbf{z}_0(\mathbf{x})\mathbf{z}_0(\mathbf{x})^{\mathrm{T}}$ and zero otherwise; $\mathbf{B}_{i,k} \in \mathbb{S}^{n_{\mathbf{z}_i}}$ has $h_{i,j}$ in entries where $m_k(\mathbf{x})$ appears in matrix $m_{i,j}(\mathbf{x})\mathbf{z}_i(\mathbf{x})\mathbf{z}_i(\mathbf{x})^{\mathrm{T}}$ and zero otherwise. Using the selection matrices, the optimization problem in (14) can be equivalently rewritten as:

$$\begin{aligned} &\underset{\gamma,\mathbf{W},\mathbf{Q}_i}{\text{maximize}} && \gamma \\ &\text{subject to} && \langle \mathbf{A}_1, \mathbf{W} \rangle + \sum\nolimits_{i=1}^{l} \langle \mathbf{B}_{i,1}, \mathbf{Q}_i \rangle = c_1 - \gamma \\ & && \langle \mathbf{A}_k, \mathbf{W} \rangle + \sum\nolimits_{i=1}^{l} \langle \mathbf{B}_{i,k}, \mathbf{Q}_i \rangle = c_k \qquad k = 2, 3, \cdots, n_f \\ & && \mathbf{W} \succcurlyeq 0, \mathbf{Q}_i \succcurlyeq 0 \qquad\qquad\qquad\quad i = 1, 2, \cdots, l \end{aligned} \tag{15}$$

where γ, $\mathbf{W}$ and $\mathbf{Q}_i$ are optimization variables; $n_f = \binom{n+2t}{n}$ from Eq. (13) is the number of monomials in $\mathbf{x} \in \mathbb{R}^n$ of degree less than or equal to $2t$. Thus, the original nonconvex polynomial optimization problem has been recast to a convex SDP problem. By solving the optimization problem in Eq. (15) formulated by the SOS method, the best (maximum possible) lower bound, i.e. the largest γ^* such that $\gamma^* \leq f^*$, of the objective function in Eq. (11) is obtained. Although there may be cases that a suboptimal value of the objective function, i.e. $\gamma^* < f^*$, is attained, in practice the lower bound obtained by the SOS method usually coincides with the optimal value of the objective function, i.e. $\gamma^* = f^*$ (Parrilo 2003).

The solution of the optimization problem formulated by SOS method in Eq. (15) provides the information on γ^*, the best (maximum possible) lower bound of objective function $f(\mathbf{x})$ of original polynomial optimization problem. The optimal solution $\mathbf{x}^*$ of the original polynomial optimization problem can be computed by solving the dual problem of the SOS formulation. The Lagrange dual function of problem in Eq. (15) is $(\mathbf{y}, \mathbf{V}, \mathbf{U}_i) = \sup\limits_{\gamma, \mathbf{W}, \mathbf{Q}_i} \mathcal{L}(\gamma, \mathbf{W}, \mathbf{Q}_i,\ \mathbf{y}, \mathbf{V}, \mathbf{U}_i)$, where $\mathbf{y}$, $\mathbf{V}$ and $\mathbf{U}_i$ are dual variables. It has been shown that if the optimal value of the original problem (Eq. (11)) and the SOS primal problem (Eq. (15)) coincide with each other, the optimal solution of the SOS dual problem can be calculated as:

$$\mathbf{y}^* = (1, x_1^*, \cdots, x_n^*, (x_1^*)^2, x_1^* x_2^* \cdots, x_{n-1}^* (x_n^*)^{2t-1}, (x_n^*)^{2t})^{\mathrm{T}} \tag{16}$$

In this way, the optimal solution $\mathbf{x}^*$ of the original problem in Eq. (11) can be extracted as the second term through the $(n+1)$-th term in $\mathbf{y}^*$. We refer the interested readers to Lasserre (2001) and Henrion (2005) for details of the optimal solution extracting technique. Since practical SDP solvers, such as SeDuMi (Sturm 1999), simultaneously solve both primal and dual problems, the optimal point $\mathbf{x}^*$ can be computed efficiently. As all the functions in modal dynamic residual approach (Eq. (3)) are polynomials, the SOS optimization method can be directly implemented. In this way, the modal dynamic residual approach is recast as a convex problem.

4 Reduce the size of SDP problem in SOS optimization

Although the SOS optimization method is powerful for solving polynomial optimization problems, the formulated SDP problem can be very expensive when n or t is large. To this end, the polynomial sparsity in the optimization problem can be utilized to reduce computation load. A sparse polynomial function means the polynomial function contains a relatively small number of monomials, i.e. many monomials with the same or a lower degree have zero coefficient. This paper examines a specific sparsity pattern that the objective function consists of several polynomials only involving a small number of variables. Take the model updating formulation in Eq. (3) as an example. The objective function consists of n_{modes} number of polynomials. Each polynomial involves only $\boldsymbol{\theta}$ and one $\boldsymbol{\psi}_{i,\mathrm{u}}$, rather than $\boldsymbol{\theta}$ and entire $\boldsymbol{\psi}_{\mathrm{u}} = \left(\boldsymbol{\psi}_{1,\mathrm{u}}, \boldsymbol{\psi}_{2,\mathrm{u}}, \cdots, \boldsymbol{\psi}_{n_{\mathrm{modes}},\mathrm{u}}\right)^{\mathrm{T}}$. As a result, we can represent each polynomial in SOS form, so that coefficients of the cross terms between $\boldsymbol{\psi}_{i,\mathrm{u}}$ and $\boldsymbol{\psi}_{j,\mathrm{u}}$, $i \neq j$, need not be considered. In this way, the number of optimization variables in SOS method can be significantly reduced.

Now consider a constrained polynomial optimization problem, in which the objective function consists of several polynomials:

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) = \sum_{q=1}^{m} f_q(\mathbf{x}) = \sum_{q=1}^{m} \sum_{k=1}^{n_{f_q}} c_{q,k} m_{q,k}(\mathbf{x}) \\ &\text{subject to} \quad g_i(\mathbf{x}) = \sum_{j=1}^{n_{g_i}} h_{i,j} m_{i,j}(\mathbf{x}) \geq 0, i = 1, 2, \cdots, l \end{aligned} \tag{17}$$

Each polynomial $f_q(\mathbf{x}) = \sum_{k=1}^{n_{f_q}} c_{q,k} m_{q,k}(\mathbf{x})$ has the quadratic form $f_q(\mathbf{x}) = \mathbf{z}_q(\mathbf{x})^{\mathrm{T}} \mathbf{W}_q \mathbf{z}_q(\mathbf{x})$. Instead of representing $f(\mathbf{x}) = \sum_{k=1}^{n_f} c_k m_k(\mathbf{x})$ as SOS directly, each $f_q(\mathbf{x})$ is represented as SOS. In this way, the redundant cross terms are excluded between variables in different $f_q(\mathbf{x})$. The degree of SOS polynomial $p_i(\mathbf{x}) = \mathbf{z}_i(\mathbf{x})^{\mathrm{T}} \mathbf{Q}_i \mathbf{z}_i(\mathbf{x})$ corresponding to $g_i(\mathbf{x})$ is usually small and utilizing the sparsity of $g_i(\mathbf{x})$ is not as advantageous as the sparsity of $f(\mathbf{x})$. Thus, this paper does not consider the sparsity in $g_i(\mathbf{x})$. Using this sparse SOS method, the SDP problem then can be formulated as:

$$\begin{aligned} &\underset{\gamma,\mathbf{W}_q,\mathbf{Q}_i}{\text{maximize}} && \gamma \\ &\text{subject to} && f(\mathbf{x})-\gamma=\sum_{q=1}^{m}\mathbf{z}_q(\mathbf{x})^{\mathrm{T}}\mathbf{W}_q\mathbf{z}_q(\mathbf{x})+\sum_{i=1}^{l}\left(\mathbf{z}_i(\mathbf{x})^{\mathrm{T}}\mathbf{Q}_i\mathbf{z}_i(\mathbf{x})\right)g_i(\mathbf{x}) \\ & && \mathbf{W}_q \succcurlyeq 0,\ q=1,2,\cdots,m \\ & && \mathbf{Q}_i \succcurlyeq 0,\ i=1,2,\cdots,l \end{aligned} \tag{18}$$

Note that although we represent each $f_q(\mathbf{x})$ as SOS separately, the equality constraint on coefficient $c_k = \sum_{q=1}^{m} c_{q,k}$, $k = 1,2,\dots,n_f$, should hold for every monomial $m_k(\mathbf{x})$ in $f(\mathbf{x})-\gamma$.To express the optimization problem explicitly, selection matrices can be utilized. Similar to selection matrix $\mathbf{A}_k$ defined in Section 3, for each polynomial $f_q(\mathbf{x})=\sum_{k=1}^{n_{f_q}} c_{q,k}m_{q,k}(\mathbf{x})$, $\mathbf{A}_{q,k}\in\mathbb{S}^{n_{\mathbf{z}_q}}$ has 1 in entries where $m_{q,k}(\mathbf{x})$ appears in matrix $\mathbf{z}_q(\mathbf{x})\mathbf{z}_q(\mathbf{x})^{\mathrm{T}}$ and 0 otherwise. As we do not utilize the sparsity of polynomials $g_i(\mathbf{x})$, selection matrix $\mathbf{B}_{i,k}$ remains the same. The SDP problem formulated by sparse SOS method can be written as:

$$\begin{aligned} &\underset{\gamma,\mathbf{W}_q,\mathbf{Q}_i}{\text{maximize}} && \gamma \\ &\text{subject to} && \sum\nolimits_{q=1}^{m}\langle\mathbf{A}_{q,1},\mathbf{W}_q\rangle+\sum\nolimits_{i=1}^{l}\langle\mathbf{B}_{i,1},\mathbf{Q}_i\rangle=c_1-\gamma \\ & && \sum\nolimits_{q=1}^{m}\langle\mathbf{A}_{q,k},\mathbf{W}_q\rangle+\sum\nolimits_{i=1}^{l}\langle\mathbf{B}_{i,k},\mathbf{Q}_i\rangle=c_k \qquad k=2,3,\cdots,n_f \\ & && \mathbf{W}_q\succcurlyeq 0 \qquad q=1,2,\cdots,m \\ & && \mathbf{Q}_i\succcurlyeq 0 \qquad i=1,2,\cdots,l \end{aligned} \tag{19}$$

Similar to the SOS method, the optimal solution $\mathbf{x}^*$ of the original polynomial optimization problem can be computed by solving the dual problem of the sparse SOS formulation in Eq. (19). In addition, the dual problem of the sparse SOS formulation can be simultaneously solved by practical SDP solvers, and the optimal solution can be obtained using the same strategy described in Section 3.2.

5 Numerical Simulation

5.1 Plane truss with dense measurement

To validate the proposed sparse SOS method for model updating, a plane truss structure is simulated (Figure 1). All member sections are set as 8×10^{-5} m^2, and material density is set as 7,849 $\mathrm{kg/m^3}$. The truss model has 10 nodes, and each node has a vertical and a horizontal DOF. 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 group, 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 1, including the initial/nominal values and the "as-built"/actual values.

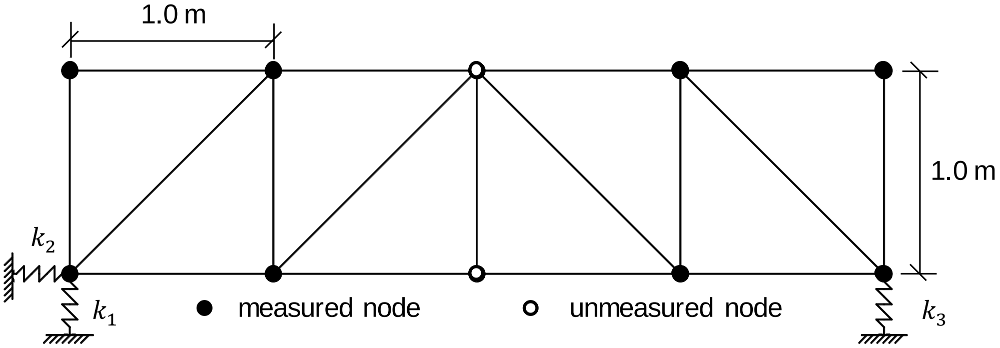


Figure 1. Plane truss structure with 8 nodes (16 DOFs) instrumented/measured

Table 1. Model updating parameters

Property		Initial/Nominal	"As-built"/Actual	Ideal updating result for θ_i
Young's moduli (×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 (×10^6 N/m)	k_1	6	7	0.167
	k_2	6	3	−0.500
	k_3	6	5	−0.167

In this study, modal properties of the "as-built" structure are directly used as "experimental" properties. It is assumed that all the nodes except the middle nodes are installed with sensors measuring both vertical and horizontal DOFs. Mode shapes extracted from the "experimental" data are only available at the measured DOFs. Considering practicality, it is assumed that only the first three modes are available for model updating. For each mode, the norm of the mode shape vector at measured DOFs, $\boldsymbol{\psi}_{i,\mathrm{m}}$, is normalized to be 1.

The stiffness updating variables $\boldsymbol{\theta} \in \mathbb{R}^6$ correspond 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). The last column in Table 1 shows the ideal updating result for each θ_i. Each unmeasured mode shape vector $\boldsymbol{\psi}_{q,\mathrm{u}} \in \mathbb{R}^4$, $q = 1,2,3$, contains the entries for the four unmeasured DOFs. All unmeasured entries in three mode shapes, $\boldsymbol{\psi}_{\mathrm{u}} = \left(\boldsymbol{\psi}_{1,\mathrm{u}}, \boldsymbol{\psi}_{2,\mathrm{u}}, \boldsymbol{\psi}_{3,\mathrm{u}}\right)^{\mathrm{T}} \in \mathbb{R}^{12}$, are the optimization variables together with $\boldsymbol{\theta}$. The total number of optimization variables is $n = n_{\boldsymbol{\theta}} + n_{\boldsymbol{\psi}_{\mathrm{u}}} = 18$. The lower bound for $\boldsymbol{\theta}$ is $\mathbf{L} = -\mathbf{1}_{6\times1}$ and the upper bound is $\mathbf{U} = \mathbf{1}_{6\times1}$. This means the relative change to each stiffness updating parameter is allowed to be ±100%. To minimize modal dynamic residual r, the model updating problem can be formulated as follows with optimization variables $\mathbf{x} = (\boldsymbol{\theta}, \boldsymbol{\psi}_{\mathrm{u}})$. Note that here we equivalently rewrite the inequality constraints $\mathbf{L} \leq \boldsymbol{\theta} \leq \mathbf{U}$ into polynomials of θ_i for directly applying SOS optimization method.

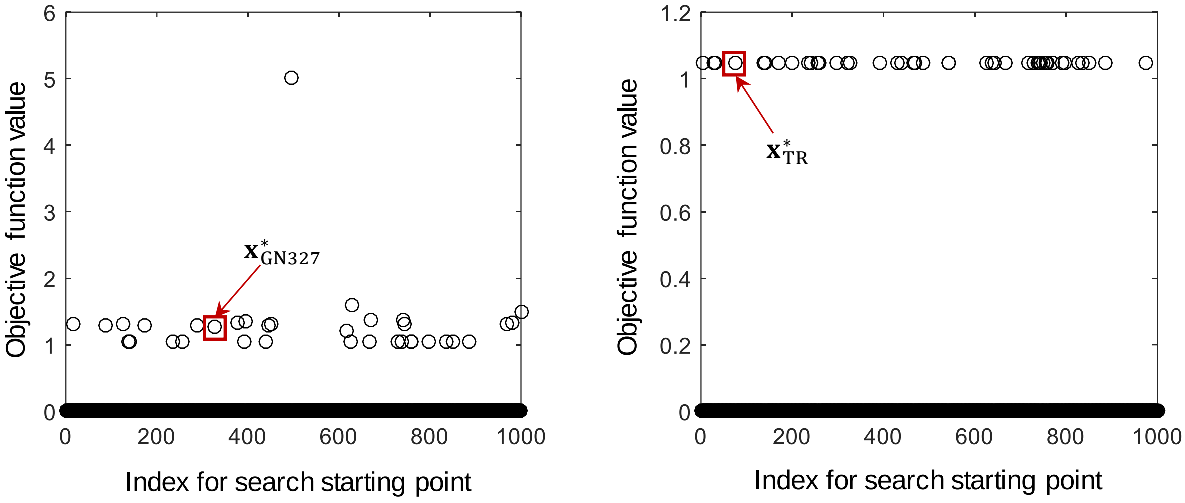
$$\begin{aligned} \underset{\mathbf{x}=(\boldsymbol{\theta},\boldsymbol{\psi}_{\mathrm{u}})}{\text{minimize}} \quad & f(\mathbf{x}) = r = \sum_{q=1}^{3} \left\| \left[\mathbf{K}(\boldsymbol{\theta}) - \omega_q^2 \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{q,\mathrm{m}} \\ \boldsymbol{\psi}_{q,\mathrm{u}} \end{Bmatrix} \right\|_2^2 \\ \text{subject to} \quad & 1 - \theta_i^2 \geq 0,\ i = 1, 2, \cdots, 6 \end{aligned} \tag{20}$$

As shown in Eq.(20), the objective function $f(\mathbf{x})$ consists of three polynomials $f_q(\mathbf{x})$, $q = 1,2,3$. Each polynomial $f_q(\mathbf{x}) = \left\| \left[\mathbf{K}(\boldsymbol{\theta}) - \omega_q^2 \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{q,\mathrm{m}} \\ \boldsymbol{\psi}_{q,\mathrm{u}} \end{Bmatrix} \right\|_2^2$ represents the modal dynamic residual from the *q*-th mode, with $n_q = n_{\boldsymbol{\theta}} + n_{\boldsymbol{\psi}_{q,\mathrm{u}}} = 10$ variables and degree of $d_q = 2t_q = 4$. Each inequality constraint $g_i(\mathbf{x}) = 1 - \theta_i^2 \geq 0$ is a polynomial with one variable θ_i and degree of $e_i = 2$.

To compare with the SOS optimization method, two local optimization algorithms are adopted to solve the optimization problem. The first local optimization algorithm is Gauss-Newton algorithm for nonlinear least squares problems (Nocedal and Wright 2006). Gauss-Newton algorithm is a modified version of Newton algorithm with an approximation of the Hessian matrix by omitting the higher order term. Through the MATLAB command '`lsqnonlin`' (MathWorks Inc. 2016), the second algorithm is the trust-region-reflective algorithm (Coleman and Li 1996). The algorithm heuristically minimizes the objective function by solving a sequence of quadratic subproblems subject to ellipsoidal constraints.

For a nonconvex problem, depending on different search starting points, a local optimization algorithm may converge to different locally optimal points. To show this phenomenon, 1,000 search starting points of the updating variables $\mathbf{x} = (\boldsymbol{\theta}, \boldsymbol{\psi}_{\mathrm{u}}) \in \mathbb{R}^{18}$ are uniformly randomly generated in the feasible space $\mathbf{L} \leq \boldsymbol{\theta} \leq \mathbf{U}$. Starting from each of the 1,000 points, both local optimization algorithms are used to search the optimal solution. The optimization problem is solved on a laptop PC with Intel® Core™ i7-6700HQ (2.60 GHz) and 8 GB RAM memory. Solving the optimization problem from 1,000 starting points by Gauss-Newton algorithm takes 15.865 seconds. On the other side, solving the optimization problem from the same 1,000 starting points by trust-region-reflective algorithm takes 57.070 seconds.

Figure 2 plots the optimized objective function values from 1,000 starting points by each local optimization algorithm. Figure 2(a) plots the performance of Gauss-Newton algorithm. The plot shows that many of the final solutions (965 out of 1,000) converge at the optimal point, with the value of objective function close to 0. However, some local optimal points are far away from the optimal point, and the achieved values of objective function are much higher than 0. For example, search from starting point #327 converges at a local minimum $\mathbf{x}_{\mathrm{GN327}}^{*}$ with the achieved objective function value of 1.269 as demonstrated in Figure 2(a). Figure 2(b) shows the performance of trust-region-reflective algorithm. Similar to Gauss-Newton algorithm, it turns out that the majority of the searches (955 out of 1,000) converge at the optimal point with the values of objective function close to 0. However, all the other 45 solutions end at a (same) local minimum $\mathbf{x}_{\mathrm{TR}}^{*}$ with the values of objective function higher than 1.0.



(a) Gauss-Newton

(b) Trust-region-reflective

Figure 2. Optimized objective function value (i.e. optimal residual $r^* = f(\mathbf{x}^*)$) for all search starting points

Still using search No. 327 in the Gauss-Newton method as an example, we obtain the optimal $\boldsymbol{\theta}^*_{\mathrm{GN327}}$ from $\mathbf{x}^*_{\mathrm{GN327}}$. From $\boldsymbol{\theta}^*_{\mathrm{GN327}}$, the updated 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 calculated and shown in Table 2. Similarly, $\boldsymbol{\theta}^*_{\mathrm{TR}}$ is obtained from $\mathbf{x}^*_{\mathrm{TR}}$ to calculate the updated stiffness parameters. The results show that the updated stiffness parameters at local minima are far away from the actual values. For example, the Young's modulus E_1 from $\mathbf{x}^*_{\mathrm{GN327}}$ is zero, meaning the gradient search stopped at a boundary point of the feasible set. Meanwhile, the E_3 from $\mathbf{x}^*_{\mathrm{TR}}$ is also close to zero. The table also lists the achieved objective function values, i.e. residual r for these two cases, both higher than 1.0.

Table 2. Updating results of different optimization methods

	Residual r	Young's moduli ($\times 10^{11}$ N/m^2)			Springs ($\times 10^6$ N/m)		
		Top (E_1)	Diagonal & Vertical (E_2)	Bottom (E_3)	k_1	k_2	k_3
Actual Value	0	2.2	1.8	1.9	7	3	5
Gauss-Newton ($\mathbf{x}^*_{\mathrm{GN327}}$)	1.269	0.000	0.210	0.065	2.534	0.464	2.123
Trust-region-reflective ($\mathbf{x}^*_{\mathrm{TR}}$)	1.047	1.433	1.043	0.060	4.874	0.876	3.631
Regular SOS	5.09×10^{-9}	2.200	1.800	1.900	7.000	3.000	5.000
Sparse SOS	8.69×10^{-8}	2.200	1.800	1.900	7.000	3.000	5.000

To further illustrate nonconvexity of the model updating problem in Eq. (20), the objective function value is evaluated along a line segment determined by the global minimum $\mathbf{x}^*$ and the local minimum $\mathbf{x}_{\mathrm{TR}}^*$ calculated by trust-region-reflective algorithm. Figure 3 plots the objective function value along this line segment $(1-\alpha)\mathbf{x}_{\mathrm{TR}}^* + \alpha\mathbf{x}^*$, which is parameterized on $\alpha \in [-0.1, 1.1]$. The plot clearly shows that the linearly interpolated value between $\left(\mathbf{x}_{\mathrm{TR}}^*, f(\mathbf{x}_{\mathrm{TR}}^*)\right)$ and $\left(\mathbf{x}^*, f(\mathbf{x}^*)\right)$ lies below the graph of $f(\mathbf{x})$, which confirms that the function $f(\mathbf{x})$ is nonconvex.

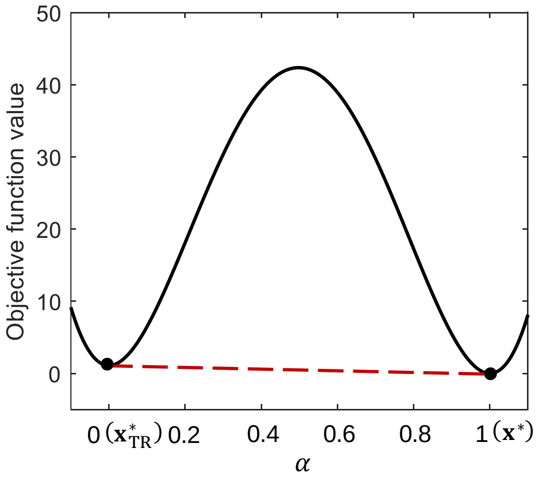


Figure 3. Objective function value (i.e. residual $r = f(\mathbf{x})$) on a line segment between a local minimum $\mathbf{x}_{\mathrm{TR}}^*$ and the global minimum $\mathbf{x}^*$

Using SOS optimization method, the nonconvex problem in Eq. (20) is recast into a convex SDP problem (Eq. (15)). By solving the optimization problem in Eq. (15) and its dual problem, the optimal solutions can be calculated as $\gamma^* = 0.000$ for the primal problem and $\mathbf{y}^* = (1, 0.100, -0.100, -0.050, 0.167, -0.5, -0.167, \cdots)$ for the dual problem (Eq. (16)). The optimal solution $\boldsymbol{\theta}^*$ for the original problem in Eq. (3) is now easily extracted as $(0.100, -0.100, -0.050, 0.167, -0.5, -0.167)$. Using the calculated $\boldsymbol{\theta}^*$, the updated 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) can be calculated and are shown in Table 2. The SOS optimization method recasts the original problem as a convex SDP problem and can reliably find the lowest minimum point, without searching from a large quantity of randomized starting points. Similarly, the stiffness values updated by sparse SOS method are also calculated and listed in Table 2. Both the regular and sparse SOS methods accurately identify the stiffness values to more than three digits after the decimal point. The table also shows both SOS methods achieve a residual value (i.e. objective function value) of nearly zero, which is much lower than these from $\mathbf{x}^*_{\mathrm{GN327}}$ and $\mathbf{x}^*_{\mathrm{TR}}$.

While achieving similar accuracy, the sparse SOS method saves a great amount of computation effort. For the problem in Eq. (20), there are $n = 18$ optimization variables in total, and the degree of the objective function is $d = 2t = 4$. The problem has $l = 6$ inequality constraints on $\boldsymbol{\theta}$ in total. The degree of each inequality constraint is $e_i = 2, i = 1, \ldots, 6$. To apply SOS optimization method, optimization variables γ, $\mathbf{W}$, $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ are introduced (Eq. (15)). With $d = 2t = 4$ and $n = 18$, the length of base monomial vector $\mathbf{z}_0$ is $n_{\mathbf{z}_0} = \binom{n+t}{n} = \binom{18+2}{18}$=190. For the symmetric matrix $\mathbf{W}$, the number of optimization variables in $\mathbf{W}$ is $n_{\mathbf{z}_0}(n_{\mathbf{z}_0} + 1)/2 = 190 \times (190 + 1)/2 = 18{,}145$. Similarly, with $\tilde{e}_i = 1$, the length of base monomial variables $\mathbf{z}_i$ is $n_{\mathbf{z}_i} = \binom{n+t-\tilde{e}_i}{n} = \binom{18+2-1}{18} = 19$. For each symmetric matrix $\mathbf{Q}_i$, the number of optimization variables is $n_{\mathbf{z}_i}(n_{\mathbf{z}_i} + 1)/2 = 19 \times (19 + 1)/2 = 190$. Recalling we have $\mathbf{Q}_i$ $(i = 1, \ldots, 6)$, the total number of optimization variables for regular SOS method is thus $1 + 18{,}145 + 6 \times 190 = 19{,}286$ (the first number 1 corresponds to scalar γ). Solving the SDP problem formulated by regular SOS method consumes 4,107 seconds on a laptop PC with Intel® Core™ i7-6700HQ (2.60 GHz) and 8 GB RAM memory.

On the other hand, sparse SOS method can reduce the computation load by eliminating those unnecessary monomials. The objective function in problem Eq. (20) consists of three polynomials, each of which contains $n_q = 10$ variables and has degree of $d_q = 2t_q = 4$. To apply sparse SOS optimization method, optimization variables γ, $\mathbf{W}_q$ $(q = 1, 2, 3)$, $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ are introduced (Eq. (19)). The variables γ and $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ share the same size as those in regular SOS method. With $d_q = 2t_q = 4$ and $n_q = 10$, the length of $\mathbf{z}_q$ is $n_{\mathbf{z}_q} = \binom{n_q + t_q}{n_q} = \binom{10+2}{10} = 66$. For each symmetric matrix $\mathbf{W}_q$, the number of optimization variables is $n_{\mathbf{z}_q}\left(n_{\mathbf{z}_q}+1\right)/2 = 66 \times (66+1)/2 = 2{,}211$. Thus, the total number of optimization variables for sparse SOS method is $1 + 3 \times 2{,}211 + 6 \times 190 = 7{,}774$, which is approximately one third of regular SOS method. Furthermore, solving the SDP problem from sparse SOS method consumes only 15 seconds on the same PC. Note the reduction in computing time is exponentially less with the reduction of optimization variables. Table 3 briefly summarizes the comparison between regular and sparse SOS methods applied on this model updating problem.

Table 3. Computation loads of regular SOS method and sparse SOS method

	Size of $\mathbf{W}_q$	# of $\mathbf{W}_q$	Size of $\mathbf{Q}_i$	# of $\mathbf{Q}_i$	# of opt. variables	Computation time (s)
Regular SOS	190×190	1	19×19	6	19,286	4,107
Sparse SOS	66×66	3	19×19	6	7,774	15

5.2 Plane truss with sparse measurement

To further validate the performance of sparse SOS method, the same plane truss structure but with less sensor measurement is studied. The dimensions, material properties, and boundary conditions of the structure are the same as those described in Section 5.2. However, it is now assumed that only eight DOFs are measured by sensors and the measurement layout is illustrated in Figure 4. Mode shapes extracted from the “experimental” data are only available at these eight measured DOFs. Furthermore, it is also assumed that only the first two modes (associated with the two lowest resonance frequencies) are available for model updating.

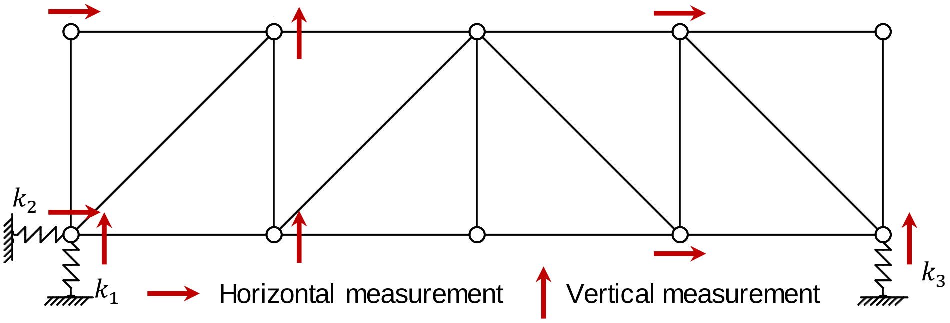


Figure 4. Plane truss structure with 8 DOFs measured

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 using the "experimental" modal properties. To formulate the optimization problem, all unmeasured entries in the two available mode shapes, $\boldsymbol{\psi}_{\mathrm{u}} = \left(\boldsymbol{\psi}_{1,\mathrm{u}}, \boldsymbol{\psi}_{2,\mathrm{u}}\right)^{\mathrm{T}} \in \mathbb{R}^{24}$, are the optimization variables together with $\boldsymbol{\theta}$. The total number of optimization variables is $n = n_{\boldsymbol{\theta}} + n_{\boldsymbol{\psi}_{\mathrm{u}}} = 30$, which is notably higher than the dense measurement example in previous Section 5.1. The same lower bound and upper bound for $\boldsymbol{\theta}$ are adopted here, and the optimization problem can be formulated as follow:

$$\begin{aligned} &\underset{\mathbf{x}=(\boldsymbol{\theta},\boldsymbol{\psi}_{\mathrm{u}})}{\text{minimize}} \quad f(\mathbf{x}) = r = \sum_{q=1}^{2} \left\| \left[\mathbf{K}(\boldsymbol{\theta}) - \omega_q^2 \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{q,\mathrm{m}} \\ \boldsymbol{\psi}_{q,\mathrm{u}} \end{Bmatrix} \right\|_2^2 \\ &\text{subject to} \quad 1 - \theta_i^2 \geq 0,\ i = 1, 2, \cdots, 6 \end{aligned} \tag{21}$$

Using SOS optimization method, the nonconvex problem in Eq. (21) is recast into an equivalent convex SDP problem. In the SDP problem, optimization variables γ, $\mathbf{W}$, $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ are introduced. With $d = 2t = 4$ and $n = 30$, the length of base monomial vector $\mathbf{z}_0$ is $n_{\mathbf{z}_0} = \binom{n+t}{n} = \binom{30+2}{30}$=496. For the symmetric matrix $\mathbf{W}$, the number of optimization variables in $\mathbf{W}$ is $n_{\mathbf{z}_0}\left(n_{\mathbf{z}_0} + 1\right)/2 = 496 \times (496 + 1)/2 = 123{,}256$. Similarly, with $\tilde{e}_i = 1$, the length of base monomial vector $\mathbf{z}_i$ is $n_{\mathbf{z}_i} = \binom{n + t - \tilde{e}_i}{n} = \binom{30 + 2 - 1}{30} = 31$. For each symmetric matrix $\mathbf{Q}_i$, the number of optimization variables is $n_{\mathbf{z}_i}\left(n_{\mathbf{z}_i} + 1\right)/2 = 31 \times (31 + 1)/2 = 496$. Recalling we have $\mathbf{Q}_i$ $(i = 1, \dots, 6)$, the total number of optimization variables for regular SOS method is thus $1 + 123{,}256 + 6 \times 496 = 126{,}213$. Due to more unmeasured mode shape entries, this number is significantly higher than the 19,286 SOS variables in previous Section 5.1. Solving the SDP problem formulated by regular SOS method consumes 457 hours 16 minutes and 49 seconds on computing clusters using 16 CPUs and 84.56 GB RAM memory. Previous PC with 8 GB memory cannot support the high memory requirement needed by so many optimization variables. On the other hand, the proposed sparse SOS method can reduce the computation load significantly. The objective function in problem Eq. (21) consists of two polynomials, each of which contains $n_q = 18$

variables and has degree of $d_q = 2t_q = 4$. To apply sparse SOS optimization method, optimization variables γ, $\mathbf{W}_q$ $(q = 1, 2)$, $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ are introduced. The variables γ and $\mathbf{Q}_i$ $(i = 1, \cdots, 6)$ share the same size as those in regular SOS method. With $d_q = 2t_q = 4$ and $n_q = 18$, the length of $\mathbf{z}_q$ is $n_{\mathbf{z}_q} = \binom{n_q + t_q}{n_q} = \binom{18 + 2}{18} = 190$. For each symmetric matrix $\mathbf{W}_q$, the number of optimization variables is $n_{\mathbf{z}_q}\left(n_{\mathbf{z}_q} + 1\right)/2 = 190 \times (190 + 1)/2 = 18{,}145$. Thus, the total number of optimization variables for sparse SOS method is $1 + 2 \times 18{,}145 + 6 \times 496 = 39{,}267$, which is approximately one third of the number 123,256 from regular SOS method. Furthermore, solving the SDP problem from sparse SOS method consumes only 3 hours 13 minutes and 14 seconds on computing clusters using the same 16 CPUs, but requiring only 4.75 GB RAM memory. Table 4 briefly summarizes the comparison between regular and sparse SOS methods applied on this model updating problem. Sparse SOS method is again shown to significantly reduce the computation load.

Table 4. Computation load of regular SOS method and sparse SOS method

	Size of $\mathbf{W}_q$	# of $\mathbf{W}_q$	Size of $\mathbf{Q}_i$	# of $\mathbf{Q}_i$	# of optimization variables	Computation time
Regular SOS	496×496	1	31×31	6	123,256	457 h 16 m 49 s
Sparse SOS	190×190	2	31×31	6	39,267	3 h 13 m 14 s

Table 5 summarizes the updating results obtained from SOS optimization and sparse SOS optimization methods. Both methods can solve the model updating problem with less sensor measurement at acceptable accuracy.

Table 5. Updating results for the structure with 8 DOFs measured

Variables	Ideal updating result for θ_i	Regular SOS method	Sparse SOS method
θ_1	0.100	0.099	0.099
θ_2	−0.100	−0.101	−0.101
θ_3	−0.050	−0.051	−0.051
θ_4	0.167	0.165	0.166
θ_5	−0.500	−0.501	−0.500
θ_6	−0.167	−0.168	−0.167

6 Conclusion

This paper investigates sparse SOS optimization method for FE model updating with modal dynamic residual formulation. The formulation entails an optimization problem with a polynomial objective function and polynomial inequality constraints. The SOS optimization method can recast such a nonconvex polynomial optimization problem into a convex SDP problem, which makes the optimization process tractable and efficient. In this paper, the sparsity in SOS optimization method is discussed and proposed for significantly reducing the computation load for FE model updating. Numerical simulation on a plane truss structure is conducted to validate the proposed approach. It is shown that the proposed sparse SOS optimization method can reliably reach the global optimum while significantly reducing computation effort compared with regular SOS method.

ACKNOWLEDGEMENTS

This research was partially funded by the National Science Foundation (CMMI-1634483 and 1150700). The first author received partial support from the China Scholarship Council (#201406260201). Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the view of the sponsors.

REFERENCE

Astroza, R., Ebrahimian, H., Li, Y., and Conte, J. P. (2017). "Bayesian nonlinear structural FE model and seismic input identification for damage assessment of civil structures." *Mechanical Systems and Signal Processing*, 93, 661-687.

Basu, S., Pollack, R., and Roy, M.-F. (2003). *Algorithms in real algebraic geometry*, Springer Heidelberg.

Boyd, S. P., and Vandenberghe, L. (2004). *Convex Optimization*, Cambridge University Press, Cambridge, UK ; New York.

Coleman, T. F., and Li, Y. (1996). "An interior trust region approach for nonlinear minimization subject to bounds." *SIAM Journal on optimization*, 6(2), 418-445.

Ebrahimian, H., Astroza, R., and Conte, J. P. (2015). "Extended Kalman filter for material parameter estimation in nonlinear structural finite element models using direct differentiation method." *Earthquake Engineering & Structural Dynamics*, 44(10), 1495-1522.

Farhat, C., and Hemez, F. M. (1993). "Updating finite element dynamic models using an element-by-element sensitivity methodology." *AIAA Journal*, 31(9), 1702-1711.

Henrion, D., and Lasserre, J. B. (2005). "Detecting global optimality and extracting solutions in GloptiPoly." Positive polynomials in control, Springer, 293-310.

Hoshiya, M., and Saito, E. (1984). "Structural identification by extended kalman filter." *Journal of Engineering Mechanics*, 110(12), 1757-1770.

Jaishi, B., and Ren, W. X. (2006). "Damage detection by finite element model updating using modal flexibility residual." *Journal of Sound and Vibration*, 290(1-2), 369-387.

Koh, C., and Shankar, K. (2003). "Substructural identification method without interface measurement." *Journal of engineering mechanics*, 129(7), 769-776.

Kosmatka, J. B., and Ricles, J. M. (1999). "Damage detection in structures by modal vibration characterization." *Journal of Structural Engineering-Asce*, 125(12), 1384-1392.

Lall, S. (2011). "Sums of Squares." Technical report, Lecture slides.

Lasserre, J. B. (2001). "Global optimization with polynomials and the problem of moments." *SIAM Journal on Optimization*, 11(3), 796-817.

Laurent, M. (2009). "Sums of squares, moment matrices and optimization over polynomials." Emerging applications of algebraic geometry, Springer, 157-270.

Li, D., Dong, X., and Wang, Y. (2018). "Model updating using sum of squares (SOS) optimization to minimize modal dynamic residuals." *Structural Control and Health Monitoring*, accepted (arXiv: 1709.02499).

MathWorks Inc. (2016). *Optimization Toolbox™ User's Guide*, MathWorks Inc., Natick, MA.

Moaveni, B., Stavridis, A., Lombaert, G., Conte, J. P., and Shing, P. B. (2013). "Finite-element model updating for assessment of progressive damage in a three-story infilled RC frame." *Journal of Structural Engineering, ASCE*, 139(10), 1665-1674.

Nesterov, Y. (2000). "Squared functional systems and optimization problems." High performance optimization, Springer, 405-440.

Nie, J., and Demmel, J. (2008). "Sparse SOS relaxations for minimizing functions that are summations of small polynomials." *SIAM Journal on Optimization*, 19(4), 1534-1558.

Nie, J., Demmel, J., and Sturmfels, B. (2006). "Minimizing polynomials via sum of squares over the gradient ideal." *Mathematical programming*, 106(3), 587-606.

Nocedal, J., and Wright, S. (2006). *Numerical optimization*, Springer Science & Business Media.

Nozari, A., Behmanesh, I., Yousefianmoghadam, S., Moaveni, B., and Stavridis, A. (2017). "Effects of variability in ambient vibration data on model updating and damage identification of a 10-story building." *Engineering Structures*, 151, 540-553.

Parrilo, P. A. (2000). "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," California Institute of Technology.

Parrilo, P. A. (2003). "Semidefinite programming relaxations for semialgebraic problems." *Mathematical programming*, 96(2), 293-320.

Salawu, O. S. (1997). "Detection of structural damage through changes in frequency: A review." *Engineering Structures*, 19(9), 718-723.

Sanayei, M., Arya, B., Santini, E. M., and Wadia‐Fascetti, S. (2001). "Significance of modeling error in structural parameter estimation." *Computer‐Aided Civil and Infrastructure Engineering*, 16(1), 12-27.

Sturm, J. F. (1999). "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones." *Optimization Methods and Software*, 11(1-4), 625-653.

Wu, M., and Smyth, A. W. (2007). "Application of the unscented Kalman filter for real‐time nonlinear structural system identification." *Structural Control and Health Monitoring*, 14(7), 971-990.

Yang, J. N., Lin, S., Huang, H., and Zhou, L. (2006). "An adaptive extended Kalman filter for structural damage identification." *Structural Control and Health Monitoring*, 13(4), 849-867.

Zhang, D., and Johnson, E. A. (2013). "Substructure identification for shear structures I: Substructure identification method." *Structural Control and Health Monitoring*, 20(5), 804-820.

Zhu, D., Dong, X., and Wang, Y. (2016). "Substructure stiffness and mass updating through minimization of modal dynamic residuals." *Journal of Engineering Mechanics*, 142(5), 04016013.