Computers and Structures 196 (2018) 63–75

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

Towards improving the enhanced Craig-Bampton method

Seung-Hwan Boo, Jeong-Ho Kim, Phill-Seung Lee *

Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

ARTICLE INFO

Article history:
Received 11 May 2017
Accepted 31 October 2017

Keywords:
Structural dynamics
Finite element method
Craig-Bampton method
Algebraic substructuring
Interface boundary reduction

ABSTRACT

In this study, we improve the performance of the enhanced Craig-Bampton (ECB) method. The improved ECB method is derived by employing the algebraic substructuring and interface boundary reduction. Unlike for the original method, the residual substructural modes are compensated only for the reduced mass matrix, and this is the most attractive feature of the proposed method to reduce the computation time significantly. In addition, for effective implementation and computer memory management, we give a computer-aided formulation of the reduced mass, stiffness, and transformation matrices. Several large structural FE models are used to illustrate the significantly improved solution accuracy and computational efficiency of the improved method.

1. Introduction

Over the last half-century, component mode synthesis (CMS) [1–18] has frequently been employed in structural dynamics as an efficient and powerful tool to analyze the dynamic response of large finite element (FE) models with small computational effort. The primary concept of CMS is substructuring, and because of this, CMS is often called a dynamic substructuring method [19–21]. The first CMS method was proposed by Hurty in 1965 [1], and shortly thereafter, the Craig-Bampton (CB) method [2] was developed, and is the most popular CMS method at present. Later, various CMS methods were developed based on the CB method, and have been applied in many engineering fields [22–25].

In the CB method [2], using substructural eigenvalue problems, the substructural normal modes are computed, and those are classified into dominant and residual substructural modes using mode selection methods [22,26–28]. Then, the constraint modes are computed to define the static deformation between the substructures and the interface boundary [2]. Finally, a reduced model is constructed by synthesizing the dominant substructural modes, a very small portion of the total substructural modes, and the constraint modes.

In general, the error between the global (original) and reduced models is caused by the residual substructural modes that are neglected. Based on this fact, the residual substructural modes can be regarded as a crucial ingredient for improving the solution accuracy of the reduced model. Recently, using the residual flexibility matrix to compensate for the residual substructural modes, the enhanced Craig-Bampton (ECB) method [29,30] was developed. The ECB method results in a greatly improved reduced model in aspect of the solution accuracy.

However, the ECB method has limitations for solving large FE models involving more than hundreds of thousands of degrees of freedom (DOFs). In the ECB method, the substructuring is accomplished by using the physical domain-based substructuring considering the geometrical characteristics of the structure. Therefore, it is not easy to make a large number of substructures. In such cases, each substructure may contain relatively large DOFs and thus the computation time for calculating the constraint modes, which requires the computation of the inverse of a matrix, would be considerably expensive. In addition, because the residual flexibility matrix, a key to the ECB method, is highly populated, thus it has substantial memory requirements and requires restricting computational work during the reduction procedure. For these reasons, the original ECB method is not appropriate for dealing with large FE models. Given the recent trend of increase in the size of FE models, these limitations should be resolved.

In this study, to resolve the aforementioned limitations, we improve the performance of the ECB method. We first identify sources deteriorating the computational efficiency in the original method, and focus on managing them effectively in the proposed method. To increase the computational efficiency and reduce the requirement for computer memory, we use algebraic substructuring [31–38], giving many small substructures, instead of physical domain-based substructuring. To reduce the size of the interface boundary, inevitably increased by algebraic substructuring, interface boundary reduction [39] is employed. In the improved ECB

* Corresponding author.
E-mail address: phillseung@kaist.edu (P.-S. Lee).

https://doi.org/10.1016/j.compstruc.2017.10.017
0045-7949/

method, the residual flexibility matrix, reflecting the residual substructural modes, is applied only to the reduced mass matrix. This is a significant feature for reducing the computation time. We also demonstrate a computer-aided formulation for efficient implementation and computer memory management of the proposed method.

In Section 2 of this paper, the original ECB method is reviewed in brief. In Section 3, the improved ECB method is derived, and the computer-aided formulation is presented in Section 4. In Section 5, we verify the performance of the proposed method through several large structural FE models, and finally, conclusions are drawn in Section 6.

2. Original ECB method

In this section, we briefly review the formulation of the original ECB method. The detailed derivation procedure is described in Refs. [29,30].

The generalized eigenvalue problem for the non-partitioned global (original) structural FE model is defined by

$$\mathbf{K}_g\mathbf{u}_g = \lambda\mathbf{M}_g\mathbf{u}_g, \tag{1}$$

where $\mathbf{M}_g$ and $\mathbf{K}_g$ denote the mass and stiffness matrices for the global structure non-partitioned, respectively, and $\mathbf{u}_g$ and λ denote the global displacement vector and the eigenvalue of the global structure, respectively.

Through the substructuring shown in Fig. 1, the global structure is partitioned into n substructures that are fixed to its interface boundary. Then, Eq. (1) can be represented in a partitioned matrix form as

$$\begin{bmatrix} \mathbf{K}_s & \mathbf{K}_c \\ \mathbf{K}_c^T & \mathbf{K}_b \end{bmatrix}\begin{bmatrix} \mathbf{u}_s \\ \mathbf{u}_b \end{bmatrix} = \lambda\begin{bmatrix} \mathbf{M}_s & \mathbf{M}_c \\ \mathbf{M}_c^T & \mathbf{M}_b \end{bmatrix}\begin{bmatrix} \mathbf{u}_s \\ \mathbf{u}_b \end{bmatrix}, \tag{2}$$

in which the subscripts s, b, and c denote the substructural, interface boundary, and coupled quantities, respectively. $\mathbf{M}_s$ and $\mathbf{K}_s$ are block-diagonal mass and stiffness matrices of which diagonal terms consist of substructural mass and stiffness matrices, $\mathbf{M}_s^{(i)}$ and $\mathbf{K}_s^{(i)}$ (for $i = 1, 2, \ldots, n$), respectively.

In the CB method, the transformation matrix is defined by

$$\mathbf{T}_0 = \begin{bmatrix} \boldsymbol{\Phi}_s & \boldsymbol{\Psi}_c \\ \mathbf{0} & \mathbf{I}_b \end{bmatrix} \quad \text{with} \quad \boldsymbol{\Phi}_s = \begin{bmatrix} \boldsymbol{\Phi}_s^d & \boldsymbol{\Phi}_s^r \end{bmatrix}, \quad \boldsymbol{\Psi}_c = -\mathbf{K}_s^{-1}\mathbf{K}_c, \tag{3}$$

where $\boldsymbol{\Phi}_s$ denotes the substructural eigenvector matrix containing all substructural modes, and it is decomposed into $\boldsymbol{\Phi}_s^d$ and $\boldsymbol{\Phi}_s^r$, which are corresponding to the dominant and residual substructural modes, respectively. Here, $\boldsymbol{\Psi}_c$ and $\mathbf{I}_b$ denote the constraint mode matrix and the identity matrix for the interface boundary, respectively.

In Eq. (3), $\boldsymbol{\Phi}_s$ is a block-diagonal matrix, of which the diagonal terms consist of the substructural eigenvector matrices $\boldsymbol{\Phi}_s^{(i)}$ (for $i = 1, 2, \ldots, n$), and $\boldsymbol{\Phi}_s^{(i)}$ are computed from the following substructural eigenvalue problems

$$\begin{aligned} &\mathbf{K}_s^{(i)}\boldsymbol{\Phi}_s^{(i)} = \mathbf{M}_s^{(i)}\boldsymbol{\Phi}_s^{(i)}\boldsymbol{\Lambda}_s^{(i)} \quad \text{with} \quad \boldsymbol{\Phi}_s^{(i)} = [\boldsymbol{\Phi}_d^{(i)}\boldsymbol{\Phi}_r^{(i)}], \\ &\boldsymbol{\Lambda}_s^{(i)} = \begin{bmatrix} \boldsymbol{\Lambda}_d^{(i)} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_r^{(i)} \end{bmatrix} \quad \text{for} \quad i = 1, 2, \ldots, n, \end{aligned} \tag{4}$$

in which $\boldsymbol{\Lambda}_s^{(i)}$ denotes the substructural eigenvalue matrix corresponding to the i^{th} substructure, and $\boldsymbol{\Phi}_s^{(i)}$ and $\boldsymbol{\Lambda}_s^{(i)}$ are decomposed into dominant terms ($\boldsymbol{\Phi}_d^{(i)}$ and $\boldsymbol{\Lambda}_d^{(i)}$) and residual terms ($\boldsymbol{\Phi}_r^{(i)}$ and $\boldsymbol{\Lambda}_r^{(i)}$).

The constraint mode matrix $\boldsymbol{\Psi}_c$ in Eq. (3) is computed by

$$\boldsymbol{\Psi}_c = \begin{bmatrix} \boldsymbol{\Psi}_c^{(1)} \\ \vdots \\ \boldsymbol{\Psi}_c^{(n)} \end{bmatrix} \quad \text{with} \quad \boldsymbol{\Psi}_c^{(i)} = -\left(\mathbf{K}_s^{(i)}\right)^{-1}\mathbf{K}_c^{(i)} \quad \text{for} \quad i = 1, 2, \ldots, n, \tag{5}$$

where $\boldsymbol{\Psi}_c^{(i)}$ denotes the i^{th} substructural constraint mode matrix. Here, the inverse matrix $\left(\mathbf{K}_s^{(i)}\right)^{-1}$ can be effectively computed using the Cholesky factorization of $\mathbf{K}_s^{(i)}$.

The global displacement vector $\mathbf{u}_g$ is transformed using the transformation matrix $\mathbf{T}_0$ in Eq. (3) as follows

$$\mathbf{u}_g = \begin{bmatrix} \mathbf{u}_s \\ \mathbf{u}_b \end{bmatrix} = \mathbf{T}_0\mathbf{u} \quad \text{with} \quad \mathbf{T}_0 = \begin{bmatrix} \boldsymbol{\Phi}_s^d & \boldsymbol{\Phi}_s^r & \boldsymbol{\Psi}_c \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_b \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \mathbf{q}_s^d \\ \mathbf{q}_s^r \\ \mathbf{u}_b \end{bmatrix}, \tag{6}$$

in which $\mathbf{u}$ denotes the generalized coordinate vector, and $\mathbf{q}_s^d$ and $\mathbf{q}_s^r$ denote the modal coordinate vectors corresponding to the dominant and residual substructural eigenvector matrices, $\boldsymbol{\Phi}_s^d$ and $\boldsymbol{\Phi}_s^r$, respectively.

In Eq. (6), selecting the dominant terms, $\boldsymbol{\Phi}_s^d$ and $\mathbf{q}_s^d$, we can obtain the approximated global displacement vector $\overline{\mathbf{u}}_g$ as

$$\mathbf{u}_g \approx \overline{\mathbf{u}}_g = \overline{\mathbf{T}}_0\overline{\mathbf{u}} \quad \text{with} \quad \overline{\mathbf{T}}_0 = \begin{bmatrix} \boldsymbol{\Phi}_s^d & \boldsymbol{\Psi}_c \\ \mathbf{0} & \mathbf{I}_b \end{bmatrix}, \quad \overline{\mathbf{u}} = \begin{bmatrix} \mathbf{q}_s^d \\ \mathbf{u}_b \end{bmatrix}, \tag{7}$$

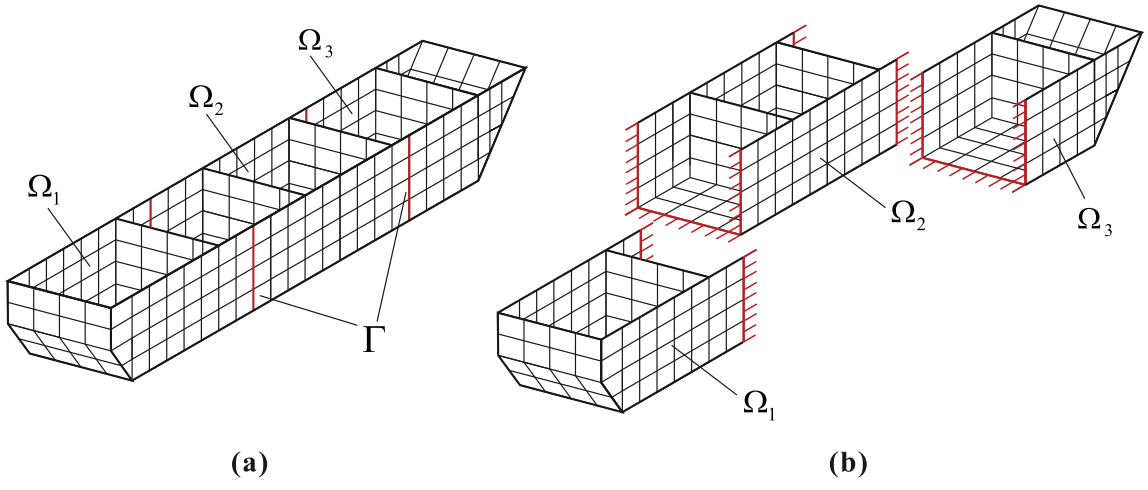

Fig. 1. Substructuring for the global structure: (a) Partitioned structure, where Ω_i and Γ denote the i^{th} substructure and the interface boundary, respectively, (b) fixed interface boundary.

where $\overline{\mathbf{T}}_0$ is the reduced transformation matrix in the CB method, and $\overline{\mathbf{u}}$ is the corresponding generalized coordinate vector.

By using the reduced transformation matrix $\overline{\mathbf{T}}_0$ in Eq. (7), the reduced mass and stiffness matrices in the CB method are obtained as

$$\overline{\mathbf{M}} = \overline{\mathbf{T}}_0^T \mathbf{M}_g \overline{\mathbf{T}}_0 = \begin{bmatrix} \mathbf{I}_s^d & (\boldsymbol{\Phi}_s^d)^T \hat{\mathbf{M}}_c \\ \hat{\mathbf{M}}_c^T (\boldsymbol{\Phi}_s^d) & \hat{\mathbf{M}}_b \end{bmatrix}, \quad \overline{\mathbf{K}} = \overline{\mathbf{T}}_0^T \mathbf{K}_g \overline{\mathbf{T}}_0 = \begin{bmatrix} \boldsymbol{\Lambda}_s^d & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{K}}_b \end{bmatrix}, \tag{8}$$

with

$$\mathbf{I}_s^d = (\boldsymbol{\Phi}_s^d)^T \mathbf{M}_s (\boldsymbol{\Phi}_s^d), \quad \hat{\mathbf{M}}_c = \mathbf{M}_c + \mathbf{M}_s \boldsymbol{\Psi}_c, \quad \hat{\mathbf{M}}_b = \mathbf{M}_b + \mathbf{M}_c^T \boldsymbol{\Psi}_c + \boldsymbol{\Psi}_c^T \hat{\mathbf{M}}_c, \tag{9a}$$

$$\boldsymbol{\Lambda}_s^d = (\boldsymbol{\Phi}_s^d)^T \mathbf{K}_s (\boldsymbol{\Phi}_s^d), \quad \hat{\mathbf{K}}_b = \mathbf{K}_b + \mathbf{K}_c^T \boldsymbol{\Psi}_c, \tag{9b}$$

in which $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$ are $\overline{N}_0 \times \overline{N}_0$ matrices ($\overline{N}_0 = \overline{N}_d + N_b$, where $\overline{N}_d$ and N_b are the numbers of dominant substructural modes and interface boundary DOFs, respectively).

Note that, in Eq. (9a), $\mathbf{I}_s^d$ is an identity matrix, and it is simply obtained without computing $(\boldsymbol{\Phi}_s^d)^T \mathbf{M}_s (\boldsymbol{\Phi}_s^d)$. Also, in Eq. (9b), there is no need to compute $(\boldsymbol{\Phi}_s^d)^T \mathbf{K}_s (\boldsymbol{\Phi}_s^d)$ to obtain $\boldsymbol{\Lambda}_s^d$. Because $\boldsymbol{\Lambda}_s^d$ is a block-diagonal matrix of which the diagonal terms consist of the substructural eigenvalue matrices $\boldsymbol{\Lambda}_d^{(i)}$ (for $i = 1, 2, \ldots, n$), $\boldsymbol{\Lambda}_s^d$ is easily obtained by reusing $\boldsymbol{\Lambda}_d^{(i)}$, already calculated in Eq. (4).

In the ECB method [29,30], the residual substructural eigenvector matrix $\boldsymbol{\Phi}_s^r$ in Eq. (6), which is not considered in the CB method, is reflected to the transformation matrix $\overline{\mathbf{T}}_0$ by considering the static part of the residual flexibility for substructures. Then, the enhanced transformation matrix is derived, which improves the solution accuracy of the reduced model significantly.

The static residual flexibility matrix for substructures $\mathbf{F}_s^{rs}$ [29,30] is defined by

$$\mathbf{F}_s^{rs} = \begin{bmatrix} \mathbf{F}_s^{(1)} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{F}_s^{(n)} \end{bmatrix} \quad \text{with} \quad \mathbf{F}_s^{(i)} = (\mathbf{K}_s^{(i)})^{-1} - (\boldsymbol{\Phi}_d^{(i)})(\boldsymbol{\Lambda}_d^{(i)})^{-1}(\boldsymbol{\Phi}_d^{(i)})^T \quad \text{for} \quad i = 1, 2, \ldots, n, \tag{10}$$

where $(\mathbf{K}_s^{(i)})^{-1}$ and $(\boldsymbol{\Phi}_d^{(i)})(\boldsymbol{\Lambda}_d^{(i)})^{-1}(\boldsymbol{\Phi}_d^{(i)})^T$ are the full and dominant flexibility matrices for the i^{th} substructure. Note that we reuse the inverse matrix $(\mathbf{K}_s^{(i)})^{-1}$ already computed in Eq. (5) for the matrix $\boldsymbol{\Psi}_c^{(i)}$, and the inverse matrix $(\boldsymbol{\Lambda}_d^{(i)})^{-1}$ is easily computed requiring decreased computation time, because it is a diagonal matrix.

After constructing the reduced matrices $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$ in Eq. (8), the enhanced transformation matrix $\overline{\mathbf{T}}_1$ [29,30] is defined as

$$\overline{\mathbf{T}}_1 = \overline{\mathbf{T}}_0 + \mathbf{T}_a \mathbf{R} \quad \text{with} \quad \mathbf{T}_a = \begin{bmatrix} \mathbf{0} & \mathbf{F}_s^{rs} \hat{\mathbf{M}}_c \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{R} = \overline{\mathbf{M}}^{-1} \overline{\mathbf{K}}, \tag{11}$$

in which $\mathbf{T}_a$ contains the effect of the residual substructural modes $\boldsymbol{\Phi}_s^r$ through $\mathbf{F}_s^{rs}$. It is very important to note that, the enhanced transformation matrix $\overline{\mathbf{T}}_1$ is computed without directly calculating the residual substructural eigenvector matrix $\boldsymbol{\Phi}_s^r$.

Thus, conducting the Rayleigh-Ritz procedure with the enhanced transformation matrix $\overline{\mathbf{T}}_1$, the reduced mass and stiffness matrices in the ECB method [29,30] are defined by

$$\overline{\mathbf{M}}_e = \overline{\mathbf{M}} + \mathbf{R}^T \mathbf{T}_a^T \mathbf{M}_g \overline{\mathbf{T}}_0 + \overline{\mathbf{T}}_0^T \mathbf{M}_g \mathbf{T}_a \mathbf{R} + \mathbf{R}^T \mathbf{T}_a^T \mathbf{M}_g \mathbf{T}_a \mathbf{R}, \tag{12a}$$

$$\overline{\mathbf{K}}_e = \overline{\mathbf{K}} + \mathbf{R}^T \mathbf{T}_a^T \mathbf{K}_g \overline{\mathbf{T}}_0 + \overline{\mathbf{T}}_0^T \mathbf{K}_g \mathbf{T}_a \mathbf{R} + \mathbf{R}^T \mathbf{T}_a^T \mathbf{K}_g \mathbf{T}_a \mathbf{R}. \tag{12b}$$

Substituting $\overline{\mathbf{T}}_0$ in Eq. (7) and $\mathbf{T}_a$ in Eq. (11) into Eq. (12), and using the orthogonality property between $\boldsymbol{\Phi}_s^d$ and $\mathbf{F}_s^{rs}$ for the mass and stiffness matrices [16,17], the four terms in Eq. (12) are represented as follows

$$\mathbf{T}_a^T \mathbf{M}_g \overline{\mathbf{T}}_0 = \mathbf{A}, \quad \mathbf{T}_a^T \mathbf{M}_g \mathbf{T}_a = \mathbf{E}, \quad \mathbf{T}_a^T \mathbf{K}_g \overline{\mathbf{T}}_0 = \mathbf{0}, \quad \mathbf{T}_a^T \mathbf{K}_g \mathbf{T}_a = \mathbf{A}, \tag{13a}$$

$$\text{with} \quad \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{M}}_c^T \mathbf{F}_s^{rs} \hat{\mathbf{M}}_c \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{M}}_c^T \mathbf{F}_s^{rs} \mathbf{M}_s \mathbf{F}_s^{rs} \hat{\mathbf{M}}_c \end{bmatrix}. \tag{13b}$$

Finally, using Eqs. (12) and (13), the reduced mass and stiffness matrices of the ECB method, $\overline{\mathbf{M}}_e$ and $\overline{\mathbf{K}}_e$, are rewritten as

$$\overline{\mathbf{M}}_e = \overline{\mathbf{M}} + \mathbf{R}^T \mathbf{A} + \mathbf{A}^T \mathbf{R} + \mathbf{R}^T \mathbf{E} \mathbf{R}, \tag{14a}$$

$$\overline{\mathbf{K}}_e = \overline{\mathbf{K}} + \mathbf{R}^T \mathbf{A} \mathbf{R}, \tag{14b}$$

where $\overline{\mathbf{M}}_e$ and $\overline{\mathbf{K}}_e$ are $\overline{N}_0 \times \overline{N}_0$ matrices, that is, the ECB method gives the same size of reduced model as the CB method does.

Then, the reduced eigenvalue problem in the ECB method is given by

$$\overline{\mathbf{K}}_e \overline{\boldsymbol{\varphi}}_j = \overline{\lambda}_j \overline{\mathbf{M}}_e \overline{\boldsymbol{\varphi}}_j \quad \text{for} \quad j = 1, 2, \ldots, \overline{N}_0, \tag{15}$$

in which $\overline{\lambda}_j$ and $\overline{\boldsymbol{\varphi}}_j$ are the j^{th} approximated eigenvalues and eigenvectors, respectively.

The ECB method produces more accurate reduced models than the CB method. The numerical results have been well demonstrated [29,30]. However, with the original ECB method, it is difficult to solve large FE models containing over 10^5 DOFs.

For large FE models, if the number of defined substructures is small, each substructure contains relatively large DOFs. For this case, the computation time to calculate the constraint mode matrix $\boldsymbol{\Psi}_c^{(i)}$ in Eq. (5) becomes expensive due to the computation of the inverse matrix $(\mathbf{K}_s^{(i)})^{-1}$. In addition, because the diagonal component matrices of $\mathbf{F}_s^{rs}$ ($\mathbf{F}_s^{(i)}$ for $i = 1, 2, \ldots, n$) are fully populated, they require large computer memory to save, and $\mathbf{F}_s^{rs}$ causes long computation time to calculate the matrices $\mathbf{A}$ and $\mathbf{E}$ in Eq. (13a). These will be investigated using numerical examples in Section 5.

3. Improved ECB method

In this section, we present the improved ECB method created using the following three procedures: algebraic substructuring, interface boundary reduction, and residual substructural mode compensation.

3.1. Algebraic substructuring

In the proposed method, using algebraic substructuring [31–38], the global mass and stiffness matrices $\mathbf{M}_g$ and $\mathbf{K}_g$ (generally sparse) are divided into many small submatrices automatically. Then, these submatrices are designated as substructures and as interface boundary in the algebraic perspective. For this, $\mathbf{M}_g$ and $\mathbf{K}_g$ are permuted automatically by the node renumbering process [36–38]. The procedure for algebraic substructuring is described in Fig. 2.

Then, the automatically partitioned global mass and stiffness matrices, $\mathbf{M}_g$ and $\mathbf{K}_g$, are represented as follows

$$\mathbf{M}_g = \left[\begin{array}{ccc|c} \mathbf{M}_s^{(1)} & & \mathbf{0} & \mathbf{M}_c^{(1)} \\ & \ddots & & \vdots \\ \mathbf{0} & & \mathbf{M}_s^{(n)} & \mathbf{M}_c^{(n)} \\ \hline \multicolumn{3}{c|}{sym.} & \mathbf{M}_b \end{array}\right], \quad \mathbf{K}_g = \left[\begin{array}{ccc|c} \mathbf{K}_s^{(1)} & & \mathbf{0} & \mathbf{K}_c^{(1)} \\ & \ddots & & \vdots \\ \mathbf{0} & & \mathbf{K}_s^{(n)} & \mathbf{K}_c^{(n)} \\ \hline \multicolumn{3}{c|}{sym.} & \mathbf{K}_b \end{array}\right]. \tag{16}$$

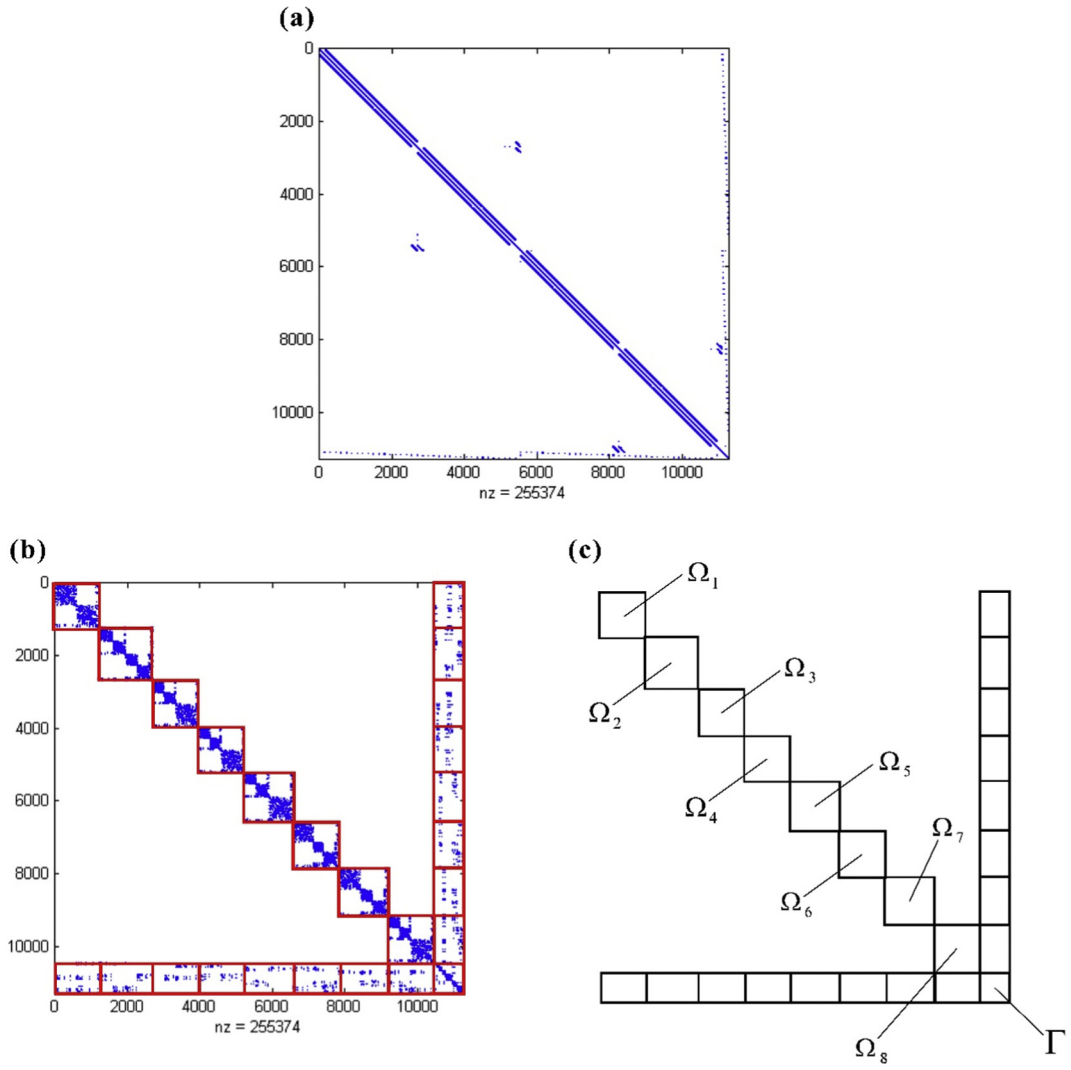

Fig. 2. Algebraic substructuring procedure: (a) Non-zero pattern of an original large sparse matrix, (b) matrix permutation and partitioning, (c) definition of the substructures and interface boundary in the algebraic perspective (8 substructures are considered).

Because the algebraic substructuring [31–38] provides a large number of submatrices (= substructures) of small size, the computation time and computer memory problems induced by $\boldsymbol{\Psi}_c^{(i)}$ and $\mathbf{F}_s^{(i)}$ mentioned in the previous section, can be managed effectively. This makes it possible to compute reduced models even for large FE models.

3.2. Interface boundary reduction

Employing algebraic substructuring [31–38], the size of the interface boundary inevitably grows because the number of substructures increases. This causes escalation of the computation time and of the size of the reduced model. Therefore, it is essential to reduce the interface boundary.

After constructing $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$ in Eq. (8), considering the mass and stiffness matrices for the interface boundary, $\hat{\mathbf{M}}_b$ and $\hat{\mathbf{K}}_b$, the eigenvalue problem for the interface boundary is given by

$$\hat{\mathbf{K}}_b\boldsymbol{\Phi}_b = \hat{\mathbf{M}}_b\boldsymbol{\Phi}_b\boldsymbol{\Lambda}_b \quad \text{with} \quad \boldsymbol{\Phi}_b = [\boldsymbol{\Phi}_b^d\boldsymbol{\Phi}_b^r], \quad \boldsymbol{\Lambda}_b = \begin{bmatrix} \boldsymbol{\Lambda}_b^d & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_b^r \end{bmatrix}, \tag{17}$$

where $\boldsymbol{\Phi}_b$ and $\boldsymbol{\Lambda}_b$ are the eigenvector and eigenvalue matrices for the interface boundary, and those are decomposed into dominant terms ($\boldsymbol{\Phi}_b^d$ and $\boldsymbol{\Lambda}_b^d$) and residual terms ($\boldsymbol{\Phi}_b^r$ and $\boldsymbol{\Lambda}_b^r$).

The eigenvalue problem in Eq. (17) is solved and the interface displacement vector $\mathbf{u}_b$ is expressed as

$$\mathbf{u}_b = \boldsymbol{\Phi}_b\mathbf{q}_b = [\boldsymbol{\Phi}_b^d\boldsymbol{\Phi}_b^r]\begin{bmatrix} \mathbf{q}_b^d \\ \mathbf{q}_b^r \end{bmatrix}, \tag{18}$$

in which $\mathbf{q}_b$ is the modal coordinate vector corresponding to $\boldsymbol{\Phi}_b$, and $\mathbf{q}_b$ is decomposed into dominant and residual terms, $\mathbf{q}_b^d$ and $\mathbf{q}_b^r$.

Using Eq. (18), the generalized coordinate vector $\mathbf{u}$ in Eq. (6) is rewritten as

$$\mathbf{u} = \begin{bmatrix} \mathbf{q}_s^d \\ \mathbf{q}_s^r \\ \mathbf{u}_b \end{bmatrix} = \mathbf{T}_b\mathbf{q} \quad \text{with} \quad \mathbf{T}_b = \begin{bmatrix} \mathbf{I}_s^d & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_s^r & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_b^d & \boldsymbol{\Phi}_b^r \end{bmatrix}, \quad \mathbf{q} = \begin{bmatrix} \mathbf{q}_s^d \\ \mathbf{q}_s^r \\ \mathbf{q}_b^d \\ \mathbf{q}_b^r \end{bmatrix}, \tag{19}$$

and applying $\mathbf{u} = \mathbf{T}_b\mathbf{q}$ into Eq. (6), the global displacement vector $\mathbf{u}_g$ is represented as

$$\mathbf{u}_g = \mathbf{T}_0\mathbf{u} = \mathbf{T}\mathbf{q} \quad \text{with} \quad \mathbf{T} = \mathbf{T}_0\mathbf{T}_b = \begin{bmatrix} \boldsymbol{\Phi}_s^d & \boldsymbol{\Phi}_s^r & \boldsymbol{\Psi}_c\boldsymbol{\Phi}_b^d & \boldsymbol{\Psi}_c\boldsymbol{\Phi}_b^r \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_b^d & \boldsymbol{\Phi}_b^r \end{bmatrix}, \tag{20}$$

where $\mathbf{T}$ is the transformation matrix for the CB method considering the interface reduction [15], $\mathbf{T}_b$ is the interface transformation matrix, and $\mathbf{q}$ is the modal coordinate vector corresponding to the transformation matrix $\mathbf{T}$.

After column reordering for the $\mathbf{T}$ matrix according to the dominant and residual terms, the global displacement vector $\mathbf{u}_g$ in Eq. (20) is rewritten as

$$\mathbf{u}_g = \mathbf{T}\mathbf{q} = [\mathbf{T}_d \quad \mathbf{T}_r]\begin{bmatrix}\mathbf{q}_d \\ \mathbf{q}_r\end{bmatrix} \tag{21}$$

$$\text{with} \quad \mathbf{T}_d = \begin{bmatrix}\mathbf{\Phi}_s^d & \mathbf{\Psi}_c\mathbf{\Phi}_b^d \\ \mathbf{0} & \mathbf{\Phi}_b^d\end{bmatrix}, \quad \mathbf{T}_r = \begin{bmatrix}\mathbf{\Phi}_s^r & \mathbf{\Psi}_c\mathbf{\Phi}_b^r \\ \mathbf{0} & \mathbf{\Phi}_b^r\end{bmatrix}, \quad \mathbf{q}_d = \begin{bmatrix}\mathbf{q}_s^d \\ \mathbf{q}_b^d\end{bmatrix}, \quad \mathbf{q}_r = \begin{bmatrix}\mathbf{q}_s^r \\ \mathbf{q}_b^r\end{bmatrix}, \tag{22}$$

in which $\mathbf{T}_d$ and $\mathbf{T}_r$ are the dominant and residual parts of the transformation matrix $\mathbf{T}$, and $\mathbf{q}_d$ and $\mathbf{q}_r$ are the modal coordinate vectors corresponding to $\mathbf{T}_d$ and $\mathbf{T}_r$, respectively.Considering only the dominant part $\mathbf{T}_d$, the global displacement vector $\mathbf{u}_g$ could be approximated as

$$\mathbf{u}_g \approx \overline{\mathbf{u}}_g = \mathbf{T}_d\mathbf{q}_d. \tag{23}$$

Then, using $\overline{\mathbf{u}}_g = \mathbf{T}_d\mathbf{q}_d$ in Eq. (23) in Eq. (1), the following reduced eigenvalue problem is obtained

$$\widetilde{\mathbf{K}}\mathbf{q}_d = \overline{\lambda}\widetilde{\mathbf{M}}\mathbf{q}_d \quad \text{with} \quad \widetilde{\mathbf{M}} = \mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_d = \begin{bmatrix}\mathbf{I}_s^d & \widetilde{\mathbf{M}}_c \\ \widetilde{\mathbf{M}}_c^T & \mathbf{I}_b^d\end{bmatrix}, \quad \widetilde{\mathbf{K}} = \mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_d = \begin{bmatrix}\mathbf{\Lambda}_s^d & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_b^d\end{bmatrix}, \tag{24}$$

$$\mathbf{I}_b^d = (\mathbf{\Phi}_b^d)^T\hat{\mathbf{M}}_b(\mathbf{\Phi}_b^d), \quad \widetilde{\mathbf{M}}_c = (\mathbf{\Phi}_s^d)^T\hat{\mathbf{M}}_c(\mathbf{\Phi}_b^d), \quad \mathbf{\Lambda}_b^d = (\mathbf{\Phi}_b^d)^T\hat{\mathbf{K}}_b(\mathbf{\Phi}_b^d), \tag{25}$$

where $\widetilde{\mathbf{M}}$ and $\widetilde{\mathbf{K}}$ are the reduced mass and stiffness matrices in the CB method considering the interface reduction [15], and $\overline{\lambda}$ is the approximated eigenvalue. Note that $\mathbf{I}_b^d$ is an identity matrix, and $\mathbf{\Lambda}_b^d$ was obtained previously in Eq. (17).

3.3. Residual substructural mode compensation

Substituting Eq. (21) into Eq. (1), and pre-multiplying $\mathbf{T}^T$, we obtain the following equation

$$\begin{bmatrix}\mathbf{\Lambda}_d - \lambda\mathbf{M}_d & -\lambda\mathbf{M}_{dr} \\ -\lambda\mathbf{M}_{dr}^T & \mathbf{\Lambda}_r - \lambda\mathbf{M}_r\end{bmatrix}\begin{bmatrix}\mathbf{q}_d \\ \mathbf{q}_r\end{bmatrix} = \begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \tag{26a}$$

$$\mathbf{\Lambda}_d = \mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_d, \quad \mathbf{\Lambda}_r = \mathbf{T}_r^T\mathbf{K}_g\mathbf{T}_r, \tag{26b}$$

$$\mathbf{M}_d = \mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_d, \quad \mathbf{M}_{dr} = \mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_r, \quad \mathbf{M}_r = \mathbf{T}_r^T\mathbf{M}_g\mathbf{T}_r. \tag{26c}$$

Expanding the linear equation for the second row, $\mathbf{q}_r$ is given by

$$\mathbf{q}_r = \lambda(\mathbf{\Lambda}_r - \lambda\mathbf{M}_r)^{-1}\mathbf{M}_{dr}^T\mathbf{q}_d, \tag{27}$$

and substituting Eq. (27) into Eq. (21), the global displacement vector $\mathbf{u}_g$ is represented by

$$\mathbf{u}_g = [\mathbf{T}_d + \lambda\mathbf{T}_r(\mathbf{\Lambda}_r - \lambda\mathbf{M}_r)^{-1}\mathbf{M}_{dr}^T]\mathbf{q}_d. \tag{28}$$

In Eq. (22), the residual part of the transformation matrix ($\mathbf{T}_r$) is represented as

$$\mathbf{T}_r = \mathbf{\Psi}\mathbf{\Phi}_r \quad \text{with} \quad \mathbf{\Psi} = \begin{bmatrix}\mathbf{I} & \mathbf{\Psi}_c \\ \mathbf{0} & \mathbf{I}\end{bmatrix}, \quad \mathbf{\Phi}_r = \begin{bmatrix}\mathbf{\Phi}_s^r & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_b^r\end{bmatrix}. \tag{29}$$

We then use $\mathbf{T}_r = \mathbf{\Psi}\mathbf{\Phi}_r$ in Eq. (29) and $\mathbf{M}_{dr} = \mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_r$ in Eq. (26c), and the global displacement vector $\mathbf{u}_g$ in Eq. (28) is rewritten by

$$\mathbf{u}_g = [\mathbf{T}_d + \lambda\mathbf{\Psi}\mathbf{F}_r\mathbf{\Psi}^T\mathbf{M}_g\mathbf{T}_d]\mathbf{q}_d \quad \text{with} \quad \mathbf{F}_r = \mathbf{\Phi}_r(\mathbf{\Lambda}_r - \lambda\mathbf{M}_r)^{-1}\mathbf{\Phi}_r^T, \tag{30}$$

in which $\mathbf{F}_r$ denotes the residual flexibility matrix [29,30].

Based on Neumann series expansion [40], the residual flexibility matrix $\mathbf{F}_r$ in Eq. (30) can be expanded as follows

$$\mathbf{F}_r = \mathbf{\Phi}_r(\mathbf{\Lambda}_r - \lambda\mathbf{M}_r)^{-1}\mathbf{\Phi}_r^T = \mathbf{\Phi}_r\mathbf{\Lambda}_r^{-1}\mathbf{\Phi}_r^T + \mathrm{O}(\lambda) + \mathrm{O}(\lambda^2) + \mathrm{O}(\lambda^3) + \cdots. \tag{31}$$

Neglecting the terms related with unknown λ in Eq. (31), we obtain the static part of the residual flexibility matrix $\mathbf{F}_{rs}$ [29,30] as follows

$$\mathbf{F}_{rs} = \mathbf{\Phi}_r\mathbf{\Lambda}_r^{-1}\mathbf{\Phi}_r^T = \begin{bmatrix}\mathbf{F}_s^{rs} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_b^{rs}\end{bmatrix}, \tag{32}$$

where $\mathbf{F}_s^{rs}$ and $\mathbf{F}_b^{rs}$ denote the static residual flexibility matrix for the substructures and interface boundary, respectively. Note that, the detailed formulation of $\mathbf{F}_s^{rs}$ is already given in Eq. (10).

Considering only the static residual flexibility matrix for substructures $\mathbf{F}_s^{rs}$, the following equation is obtained

$$\hat{\mathbf{F}}_{rs} = \begin{bmatrix}\mathbf{F}_s^{rs} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}. \tag{33}$$

In Eq. (30), using $\hat{\mathbf{F}}_{rs}$ and $\overline{\lambda}$ instead of $\mathbf{F}_r$ and λ, respectively, the global displacement vector $\mathbf{u}_g$ is approximated by

$$\mathbf{u}_g \approx \overline{\mathbf{u}}_g = \mathbf{T}_e\mathbf{q}_d \quad \text{with} \quad \mathbf{T}_e = \mathbf{T}_d + \overline{\lambda}\mathbf{T}_a, \quad \mathbf{T}_a = \mathbf{\Psi}\hat{\mathbf{F}}_{rs}\mathbf{\Psi}^T\mathbf{M}_g\mathbf{T}_d, \tag{34}$$

in which $\mathbf{T}_e$ is a new transformation matrix for the proposed method, which is enhanced by $\mathbf{T}_a$, which contains the effect of residual modes.

Applying $\overline{\mathbf{u}}_g = \mathbf{T}_e\mathbf{q}_d$ in Eq. (34) into Eq. (1), the following reduced eigenvalue problem is obtained

$$\widetilde{\mathbf{K}}_e\mathbf{q}_d = \overline{\lambda}\widetilde{\mathbf{M}}_e\mathbf{q}_d, \tag{35}$$

and the reduced mass and stiffness matrices, $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$, are defined as

$$\widetilde{\mathbf{M}}_e = \mathbf{T}_e^T\mathbf{M}_g\mathbf{T}_e = \widetilde{\mathbf{M}} + \overline{\lambda}\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_d + \overline{\lambda}\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a + \overline{\lambda}^2\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_a, \tag{36a}$$

$$\widetilde{\mathbf{K}}_e = \mathbf{T}_e^T\mathbf{K}_g\mathbf{T}_e = \widetilde{\mathbf{K}} + \overline{\lambda}\mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_d + \overline{\lambda}\mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_a + \overline{\lambda}^2\mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_a. \tag{36b}$$

Using $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$ in Eq. (36), the reduced eigenvalue problem in Eq. (35) is rewritten as

$$[\widetilde{\mathbf{K}} + \overline{\lambda}\mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_d + \overline{\lambda}\mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_a + \overline{\lambda}^2\mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_a]\mathbf{q}_d = \overline{\lambda}[\widetilde{\mathbf{M}} + \overline{\lambda}\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_d + \overline{\lambda}\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a + \overline{\lambda}^2\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_a]\mathbf{q}_d. \tag{37}$$

Substituting $\mathbf{\Psi}$ in Eq. (29) and $\hat{\mathbf{F}}_{rs}$ in Eq. (33) into $\mathbf{T}_a$ in Eq. (34), the additional transformation matrix $\mathbf{T}_a$ is represented as

$$\mathbf{T}_a = \mathbf{\Psi}\hat{\mathbf{F}}_{rs}\mathbf{\Psi}^T\mathbf{M}_g\mathbf{T}_d = \begin{bmatrix}\mathbf{F}_s^{rs}\mathbf{M}_s\mathbf{\Phi}_s^d & \mathbf{F}_s^{rs}\hat{\mathbf{M}}_c\mathbf{\Phi}_b^d \\ \mathbf{0} & \mathbf{0}\end{bmatrix}. \tag{38}$$

Because the static residual flexibility matrix $\mathbf{F}_s^{rs}$ is mass-orthogonal to the substructural eigenvector matrix $\mathbf{\Phi}_s^d$, we can identify that $\mathbf{F}_s^{rs}\mathbf{M}_s\mathbf{\Phi}_s^d = \mathbf{0}$ in $\mathbf{T}_a$ and thus, Eq. (38) can be rewritten as

$$\mathbf{T}_a = \begin{bmatrix}\mathbf{0} & \mathbf{F}_s^{rs}\hat{\mathbf{M}}_c\mathbf{\Phi}_b^d \\ \mathbf{0} & \mathbf{0}\end{bmatrix}. \tag{39}$$

Then, substituting $\mathbf{T}_d$ in Eq. (22) and $\mathbf{T}_a$ in Eq. (39) into $\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a$, $\mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_a$ and $\mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_a$ in Eq. (37), we have following relations:

$$\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a=\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{A}_b\end{bmatrix} \quad \text{with} \quad \mathbf{A}_b=(\boldsymbol{\Phi}_b^d)^T\hat{\mathbf{M}}_c^T\mathbf{F}_s^{rs}\hat{\mathbf{M}}_c\boldsymbol{\Phi}_b^d, \tag{40a}$$

$$\mathbf{T}_d^T\mathbf{K}_g\mathbf{T}_a=\mathbf{0}, \mathbf{T}_a^T\mathbf{K}_g\mathbf{T}_a=\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a. \tag{40b}$$

Pre-multiplying $\widetilde{\mathbf{M}}^{-1}$ in Eq. (24), the following equation is obtained

$$\overline{\lambda}\mathbf{q}_d=\mathbf{R}\mathbf{q}_d \quad \text{with} \quad \mathbf{R}=\widetilde{\mathbf{M}}^{-1}\widetilde{\mathbf{K}}. \tag{41}$$

Note that the matrix $\mathbf{R}$ is not a symmetric matrix.

Applying Eq. (40) into Eq. (37), the reduced eigenvalue problem is rewritten as

$$\widetilde{\mathbf{K}}\mathbf{q}_d=\overline{\lambda}[\widetilde{\mathbf{M}}+\overline{\lambda}\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a+\overline{\lambda}^2\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_a]\mathbf{q}_d, \tag{42}$$

and neglecting the high-order term ($\overline{\lambda}^2\mathbf{T}_a^T\mathbf{M}_g\mathbf{T}_a$), and using the relation $\overline{\lambda}\mathbf{q}_d=\mathbf{R}\mathbf{q}_d$ in Eq. (41) in $\overline{\lambda}\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a$, we can obtain a new reduced eigenvalue problem as follows

$$\widetilde{\mathbf{K}}\mathbf{q}_d=\overline{\lambda}[\widetilde{\mathbf{M}}+\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a\mathbf{R}]\mathbf{q}_d. \tag{43}$$

Finally, substituting $\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a$ in Eq. (40a) into Eq. (43), the reduced mass and stiffness matrices of the proposed method, $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$, are obtained as follows:

$$\widetilde{\mathbf{M}}_e=\widetilde{\mathbf{M}}+\mathbf{Y} \quad \text{with} \quad \mathbf{Y}=\mathbf{T}_d^T\mathbf{M}_g\mathbf{T}_a\mathbf{R}=\begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{A}_b\end{bmatrix}\mathbf{R}, \tag{44a}$$

$$\widetilde{\mathbf{K}}_e=\widetilde{\mathbf{K}}, \tag{44b}$$

in which $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$ are matrices of size $\overline{N}_1\times\overline{N}_1$, and $\overline{N}_1$ is the number of DOFs in the reduced model obtained using the proposed method ($\overline{N}_1=\overline{N}_d+\overline{N}_b$, where $\overline{N}_b$ is the number of dominant interface boundary modes selected).

Note that, unlike the original ECB method, the matrix $\mathbf{T}_a$, which contains the effect of residual substructural modes, is considered only for the mass matrix in the proposed method, and this is a highly advantageous feature for reducing computation time.

In the proposed method, the reduced matrices in Eq. (44) are computed without conducting the Rayleigh-Ritz procedure using the transformation matrix $\mathbf{T}_e$. Therefore, to obtain the approximated global eigenvectors, we need to compute the transformation matrix $\mathbf{T}_e$. Applying the relation $\overline{\lambda}\mathbf{q}_d=\mathbf{R}\mathbf{q}_d$ in Eq. (41) to Eq. (34), the transformation matrix $\mathbf{T}_e$ is redefined by

$$\mathbf{T}_e=\mathbf{T}_d+\mathbf{T}_a\mathbf{R}, \tag{45}$$

and then, the approximated global eigenvectors are calculated as

$$(\overline{\boldsymbol{\varphi}}_g)_j=\mathbf{T}_e\tilde{\boldsymbol{\varphi}}_j \quad \text{for} \quad j=1,2,\ldots,\overline{N}_1, \tag{46}$$

where $\tilde{\boldsymbol{\varphi}}_j$ is the j^{th} approximated eigenvector calculated from the reduced eigenvalue problem by the proposed method, $\widetilde{\mathbf{K}}_e\tilde{\boldsymbol{\varphi}}_j=\overline{\lambda}_j\widetilde{\mathbf{M}}_e\tilde{\boldsymbol{\varphi}}_j$.

4. Computer-aided formulation of the improved ECB method

In this section, we give a computer-aided formulation of the proposed method that provides computationally efficient implementation and reduces the computer memory required. For these purposes, the formulations are represented in a substructural matrix form. The detailed computational procedure of the proposed method is described in Table 1.

In Eq. (8), the matrices $\hat{\mathbf{M}}_c$, $\hat{\mathbf{M}}_b$, and $\hat{\mathbf{K}}_b$ in the reduced matrices $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$ are computed using

$$\hat{\mathbf{M}}_c=\begin{bmatrix}\hat{\mathbf{M}}_c^{(1)}\\ \vdots\\ \hat{\mathbf{M}}_c^{(n)}\end{bmatrix} \quad \text{with} \quad \hat{\mathbf{M}}_c^{(i)}=\mathbf{M}_c^{(i)}+\mathbf{M}_s^{(i)}\boldsymbol{\Psi}_c^{(i)} \quad \text{for} \quad i=1,2,\ldots,n, \tag{47}$$

$$\hat{\mathbf{M}}_b=\mathbf{M}_b+\sum_{i=1}^{n}[(\mathbf{M}_c^{(i)})^T\boldsymbol{\Psi}_c^{(i)}+(\boldsymbol{\Psi}_c^{(i)})^T\hat{\mathbf{M}}_c^{(i)}], \quad \hat{\mathbf{K}}_b=\mathbf{K}_b+\sum_{i=1}^{n}(\mathbf{K}_c^{(i)})^T\boldsymbol{\Psi}_c^{(i)}. \tag{48}$$

Then, the reduced mass and stiffness matrices in the CB method, $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$, are computed using the following computer-aided formulation

$$\overline{\mathbf{M}}=\left[\begin{array}{ccc:c}\mathbf{I}_d^{(1)} & & \mathbf{0} & (\boldsymbol{\Phi}_d^{(1)})^T\hat{\mathbf{M}}_c^{(1)}\\ & \ddots & & \vdots\\ \mathbf{0} & & \mathbf{I}_d^{(n)} & (\boldsymbol{\Phi}_d^{(n)})^T\hat{\mathbf{M}}_c^{(n)}\\ \hdashline \multicolumn{3}{c:}{sym.} & \hat{\mathbf{M}}_b\end{array}\right], \quad \overline{\mathbf{K}}=\left[\begin{array}{ccc:c}\boldsymbol{\Lambda}_d^{(1)} & & \mathbf{0} & \mathbf{0}\\ & \ddots & & \vdots\\ \mathbf{0} & & \boldsymbol{\Lambda}_d^{(n)} & \mathbf{0}\\ \hdashline \multicolumn{3}{c:}{sym.} & \hat{\mathbf{K}}_b\end{array}\right], \tag{49}$$

in which $\mathbf{I}_d^{(i)}$ (for $i=1,2,\ldots,n$) are the identity matrices requiring very small computing effort, and in which $\boldsymbol{\Phi}_d^{(i)}$ and $\boldsymbol{\Lambda}_d^{(i)}$ (for $i=1,2,\ldots,n$) are calculated from the substructural eigenvalue problem in Eq. (4).

After calculating the eigenvalue problem for the interface boundary in Eq. (17), the matrix $\widetilde{\mathbf{M}}_c$ in Eq. (24) is computed using

$$\widetilde{\mathbf{M}}_c=\begin{bmatrix}\widetilde{\mathbf{M}}_c^{(1)}\\ \vdots\\ \widetilde{\mathbf{M}}_c^{(n)}\end{bmatrix} \quad \text{with} \quad \widetilde{\mathbf{M}}_c^{(i)}=(\boldsymbol{\Phi}_d^{(i)})^T\hat{\mathbf{M}}_c^{(i)}(\boldsymbol{\Phi}_b^d) \quad \text{for} \quad i=1,2,\ldots,n, \tag{50}$$

and the reduced matrices $\widetilde{\mathbf{M}}$ and $\widetilde{\mathbf{K}}$ in Eq. (24) can be constructed as follows

Table 1
Computational procedure of the proposed method.

Computational procedure	Related equation numbers
Step 1. Algebraic substructuring	16
Step 2. Calculation of the constraint mode matrix	
a. Calculating $(\mathbf{K}_s^{(i)})^{-1}$ and $\boldsymbol{\Psi}_c^{(i)}$	5
Step 3. Substructural eigenvalue problems	
a. Solving $\mathbf{K}_s^{(i)}\boldsymbol{\Phi}_s^{(i)}=\boldsymbol{\Lambda}_s^{(i)}\mathbf{M}_s^{(i)}\boldsymbol{\Phi}_s^{(i)}$	4
b. Calculating $\boldsymbol{\Lambda}_d^{(i)}$ and $\boldsymbol{\Phi}_d^{(i)}$	4
c. Calculating $\mathbf{F}_s^{(i)}$ and factoring into $\mathbf{F}_d^{(i)}$ and $\mathbf{F}_u^{(i)}$	10, 52
Step 4. Construction of the reduced matrices of the CB method	
a. Calculating $\mathbf{I}_d^{(i)}, \hat{\mathbf{M}}_c^{(i)}, \hat{\mathbf{M}}_b$, and $\hat{\mathbf{K}}_b$	47, 48, 49
b. Constructing $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$	49
Step 5. Interface boundary reduction	
a. Solving $\hat{\mathbf{K}}_b\boldsymbol{\Phi}_b=\boldsymbol{\Lambda}_b\hat{\mathbf{M}}_b\boldsymbol{\Phi}_b$	17
b. Calculating $\boldsymbol{\Lambda}_b^d, \boldsymbol{\Phi}_b^d$, and $\widetilde{\mathbf{M}}_c^{(i)}$	17, 50
c. Constructing $\widetilde{\mathbf{M}}$ and $\widetilde{\mathbf{K}}$	51
Step 6. Residual substructural mode compensation	
a. Calculating $\mathbf{R}$ and partitioning	41, 55
b. Calculating $\mathbf{H}^{(i)}, \mathbf{Q}_d^{(i)}, \mathbf{Q}_u^{(i)}, \mathbf{A}_b$, and $\mathbf{Y}$	53, 54, 55
c. Constructing $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$	44, 51, 56
Step 7. Reduced eigenvalue problem	43

$$\widetilde{\mathbf{M}} = \begin{bmatrix} \mathbf{I}_d^{(1)} & & \mathbf{0} & \widetilde{\mathbf{M}}_c^{(1)} \\ & \ddots & & \vdots \\ \mathbf{0} & & \mathbf{I}_d^{(n)} & \widetilde{\mathbf{M}}_c^{(n)} \\ \hline & sym. & & \mathbf{I}_b^d \end{bmatrix}, \quad \widetilde{\mathbf{K}} = \begin{bmatrix} \mathbf{\Lambda}_d^{(1)} & & \mathbf{0} & \mathbf{0} \\ & \ddots & & \vdots \\ \mathbf{0} & & \mathbf{\Lambda}_d^{(n)} & \mathbf{0} \\ \hline & sym. & & \mathbf{\Lambda}_b^d \end{bmatrix}, \tag{51}$$

where $\mathbf{I}_b^d$ is an identity matrix, and $\mathbf{\Lambda}_b^d$ has already been calculated from Eq. (17).In Eqs. (10) and (33), the matrix $\mathbf{F}_s^{(i)}$, which is a diagonal substructural matrix of $\mathbf{F}_s^{rs}$ for the i^{th} substructure, is symmetric, and it can be factored into three parts as follows

$$\mathbf{F}_s^{(i)} = \mathbf{F}_d^{(i)} + \mathbf{F}_u^{(i)} + \left(\mathbf{F}_u^{(i)}\right)^T, \tag{52}$$

in which $\mathbf{F}_d^{(i)}$ and $\mathbf{F}_u^{(i)}$ are the diagonal and upper triangular parts of $\mathbf{F}_s^{(i)}$, respectively. Note that, as mentioned before, $\mathbf{F}_s^{(i)}$ requires large computer memory and increases the computation time. Therefore, it is beneficial to save $\mathbf{F}_d^{(i)}$ and $\mathbf{F}_u^{(i)}$ instead of saving $\mathbf{F}_s^{(i)}$ in computer memory.

Using Eqs. (10), (47), and (52), the matrix $\mathbf{A}_b$ in Eq. (40a) is computed as

$$\mathbf{A}_b = \mathbf{A}_d + \mathbf{A}_u + \mathbf{A}_u^T \tag{53}$$

with

$$\mathbf{A}_d = \sum_{i=1}^{n} \left(\mathbf{H}^{(i)}\right)^T \mathbf{Q}_d^{(i)}, \quad \mathbf{A}_u = \sum_{i=1}^{n} \left(\mathbf{H}^{(i)}\right)^T \mathbf{Q}_u^{(i)}, \tag{54a}$$

$$\mathbf{H}^{(i)} = \hat{\mathbf{M}}_c^{(i)} \mathbf{\Phi}_b^d, \quad \mathbf{Q}_d^{(i)} = \mathbf{F}_d^{(i)} \mathbf{H}^{(i)}, \quad \mathbf{Q}_u^{(i)} = \mathbf{F}_u^{(i)} \mathbf{H}^{(i)}. \tag{54b}$$

From Eq. (41), $\mathbf{R}$ is obtained and partitioned into substructural and interface boundary parts as follows

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_s & \mathbf{R}_c \\ \hat{\mathbf{R}}_c & \mathbf{R}_b \end{bmatrix}. \tag{55}$$

Note that, because the matrix $\mathbf{R}$ is non-symmetric, the upper triangular part matrix $\mathbf{R}_c$ is not equal to the lower triangular part matrix $\hat{\mathbf{R}}_c$.

Then, using $\mathbf{R}$ in Eq. (55) and $\mathbf{A}_b$ in Eq. (53), the matrix $\mathbf{Y}$ in Eq. (44a) is obtained

$$\mathbf{Y} = \begin{bmatrix} \mathbf{0} & \mathbf{Y}_c \\ \mathbf{0} & \mathbf{Y}_b \end{bmatrix} \quad \text{with} \quad \mathbf{Y}_c = \hat{\mathbf{R}}_c^T \mathbf{A}_b, \quad \mathbf{Y}_b = \mathbf{R}_b^T \mathbf{A}_b. \tag{56}$$

Finally, applying the computed matrices $\widetilde{\mathbf{M}}$ and $\widetilde{\mathbf{K}}$ in Eq. (51) and $\mathbf{Y}$ in Eq. (56) to Eq. (44), we can efficiently calculate the reduced mass and stiffness matrices for the proposed method, $\widetilde{\mathbf{M}}_e$ and $\widetilde{\mathbf{K}}_e$.

To obtain the transformation matrix of the proposed method $\mathbf{T}_e$ in Eq. (45), the dominant part of the transformation matrix, $\mathbf{T}_d$ in

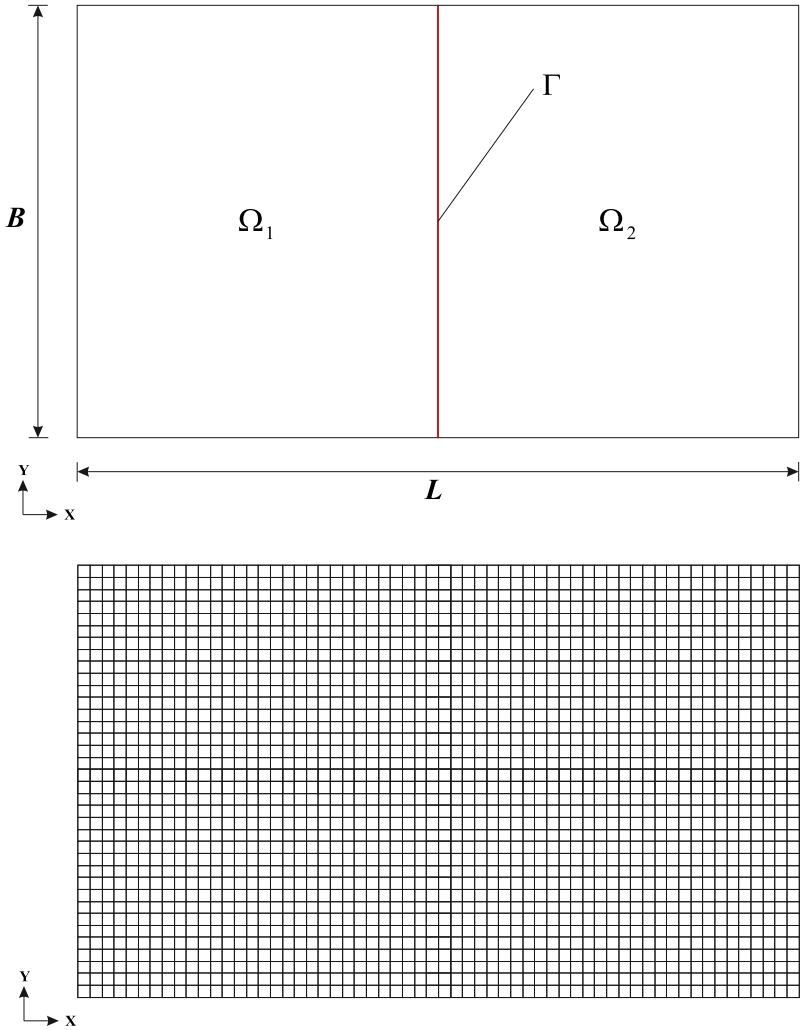

Fig. 3. Rectangular plate problem (60 × 36 mesh, 11285 DOFs, length $L = 20$ m, width $B = 12$ m, thickness $t = 0.025$ m).

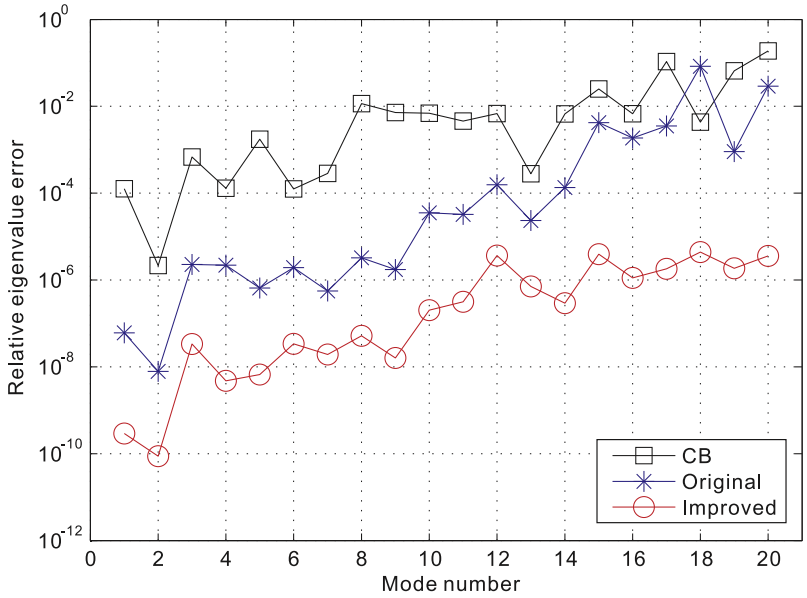

Fig. 4. Relative eigenvalue errors for the rectangular plate problem: $\overline{N}_0 = \overline{N}_1 = 205$.

Table 2
Relative eigenvalue errors for the rectangular plate problem.

Mode number	CB	ECB	
		Original	Improved
1	1.26018E−04	6.04433E−08	2.92022E−10
2	2.15755E−06	7.87076E−09	8.77754E−11
3	6.78399E−04	2.27441E−06	3.34367E−08
4	1.30336E−04	2.19235E−06	4.78053E−09
5	1.74384E−03	6.54263E−07	6.61991E−09
6	1.25321E−04	1.93101E−06	3.37899E−08
7	2.84494E−04	5.58958E−07	1.92922E−08
8	1.15526E−02	3.24691E−06	5.20070E−08
9	7.21621E−03	1.74406E−06	1.60505E−08
10	6.94656E−03	3.52401E−05	2.02076E−07
11	4.56213E−03	3.22845E−05	3.14789E−07
12	6.77084E−03	1.56672E−04	3.65606E−06
13	2.78424E−04	2.34888E−05	7.10430E−07
14	6.65006E−03	1.35011E−04	2.92710E−07
15	2.50601E−02	4.21868E−03	3.93829E−06
16	6.73162E−03	1.85850E−03	1.11335E−06
17	1.07488E−01	3.51637E−03	1.81602E−06
18	4.33276E−03	8.33458E−02	4.38059E−06
19	6.56255E−02	9.04742E−04	1.86634E−06
20	1.87902E−01	2.89952E−02	3.57041E−06

Table 3
Computation time for the rectangular plate problem.

Methods	Computation time	
	[sec]	Ratio [%]
CB	12.46	100.00
Original ECB	44.39	356.26
Improved ECB	2.94	23.60

Eq. (22), and the additional transformation matrix $\mathbf{T}_a$ in Eq. (39), should both be calculated. Fortunately, these matrices are easily obtained by using submatrices already computed, as follows

$$\mathbf{T}_d = \left[\begin{array}{ccc|c} \boldsymbol{\Phi}_d^{(1)} & & \mathbf{0} & \boldsymbol{\Psi}_c^{(1)}\boldsymbol{\Phi}_b^d \\ & \ddots & & \vdots \\ \mathbf{0} & & \boldsymbol{\Phi}_d^{(n)} & \boldsymbol{\Psi}_c^{(n)}\boldsymbol{\Phi}_b^d \\ \hline \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{\Phi}_b^d \end{array}\right], \quad \mathbf{T}_a = \left[\begin{array}{ccc|c} \mathbf{0} & & \mathbf{0} & \mathbf{Q}_d^{(1)} + \mathbf{Q}_u^{(1)} + (\mathbf{F}_u^{(1)})^T \mathbf{H}^{(1)} \\ & \ddots & & \vdots \\ \mathbf{0} & & \mathbf{0} & \mathbf{Q}_d^{(n)} + \mathbf{Q}_u^{(n)} + (\mathbf{F}_u^{(n)})^T \mathbf{H}^{(n)} \\ \hline \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \end{array}\right]. \tag{57}$$

It is reasonable to expect that the proposed method would be very computationally efficient, because the reduced model is constructed using substructural matrix computations without the global matrix computations that result in long computation times. The computational efficiency of the proposed method will be verified in the following section.

5. Numerical examples

In this section, to verify the performance of the proposed method, we solve four practical engineering problems involving a rectangular plate, a wheel structure, a jacket structure, and a stiffened plate. For FE models, the MITC shell elements [41–44] and four-node tetrahedral solid elements are used. A free boundary condition is imposed, and the material properties of a mild steel are used (Young's modulus E = 206 GPa, Poisson's ratio $\upsilon = 0.3$, and density ρ = 7850 kg/m^3).

For algebraic substructuring, we use the "*ndmetis*", which is the one of the subroutine programs of METIS [33]. The subroutine provides optimized permutation and partitioning for a large sparse matrix very fast. METIS has been widely used as an efficient matrix permutation and partitioning software package. We adopt the frequency cut-off technique [2] to select the dominant substructural and interface boundary modes. The reduced models are calculated using the CB, original and improved (proposed) ECB methods, and the numerical results are compared in terms of solution accuracy and computation time. MATLAB [45] is used for implementation of the three methods, via sparse matrix computation using a personal computer (Intel core (TM) i7-3770, 3.40 GHz CPU, and 32 GB RAM).

To evaluate the solution accuracy, the following relative eigenvalue error is used

$$\xi_j = \frac{\overline{\lambda}_j - \lambda_j}{\lambda_j}, \tag{58}$$

in which ξ_j, λ_j, and $\overline{\lambda}_j$ are the relative eigenvalue error, the exact eigenvalue obtained in the global FE model, and the approximated eigenvalue obtained in the reduced model for the j^{th} mode, respectively.

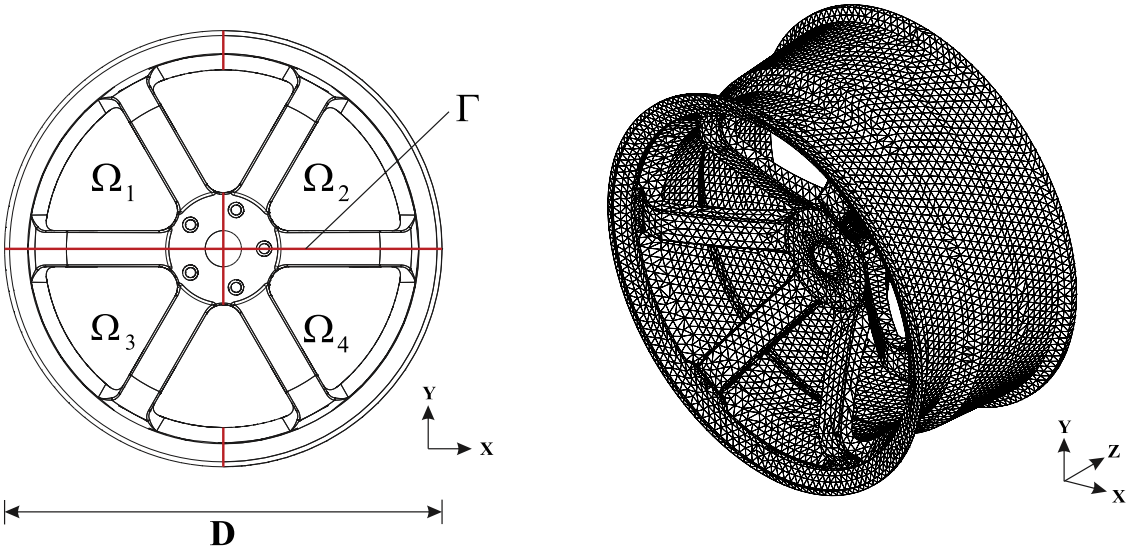

Fig. 5. Wheel structure problem (83754 four-node solid elements, 69714 DOFs, diameter $D = 0.483$ m).

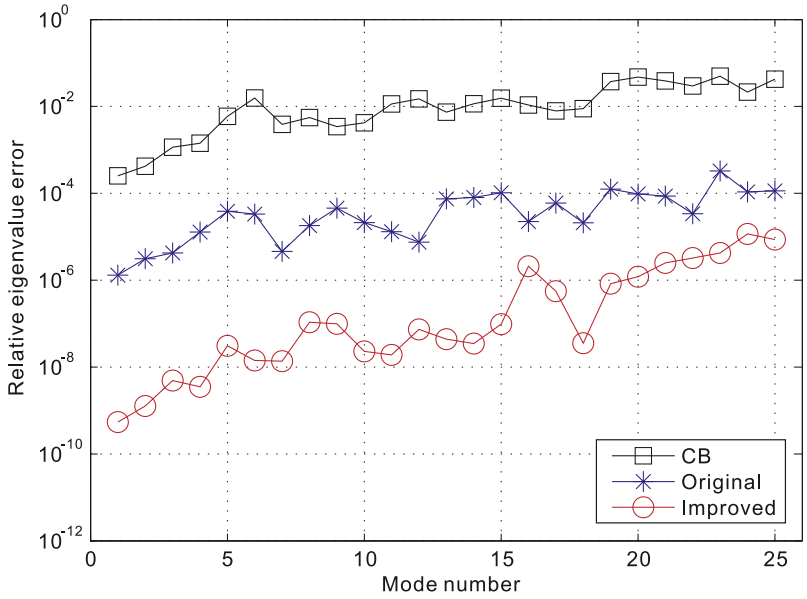

Fig. 6. Relative eigenvalue errors for the wheel problem: $\overline{N}_0 = \overline{N}_1 = 1387$.

5.1. Rectangular plate problem

Let us consider a rectangular plate as shown in Fig. 3. The length L, width B, and thickness are 20, 12, and 0.025 m, respectively. The plate is modeled by a 60×36 mesh of 4-node MITC shell finite elements, and the total number of DOFs is 11285. The same sized reduced models are constructed using the CB method, and the original and improved ECB methods.

For the CB and original ECB methods, the plate is partitioned into two substructures through physical domain-based substructuring as shown in Fig. 3. The number of the interface boundary DOFs is 185 ($N_b = 185$), and we select 20 dominant substructural modes ($\overline{N}_d = 20$). Thus, the size of the reduced models obtained by two methods $\overline{N}_0$ is 205 ($\overline{N}_0 = \overline{N}_d + N_b$). For the improved ECB method, algebraic substructuring is used to automatically partition the plate into 16 substructures. We select 80 dominant substructural modes and 125 dominant interface boundary modes ($\overline{N}_d = 80$ and $\overline{N}_b = 125$), and the size of the reduced model obtained by the proposed method $\overline{N}_1$ is 205 ($\overline{N}_1 = \overline{N}_d + \overline{N}_b$).

Note that, for the CB and original ECB methods, a small number of substructures (two) is considered to avoid increasing the number of interface boundary DOFs affecting the size of the reduced model. In contrast, for the proposed method, a relatively large number of substructures (16 substructures in this problem) could be considered, because the size of the reduced model can be controlled by adjusting the number of dominant substructural and interface boundary modes selected. That is, in the proposed method, we can control the size of the reduced model regardless of the number of substructures considered.

Fig. 4 and Table 2 demonstrate the relative eigenvalue errors, obtained by the CB, original ECB, and improved ECB methods, corresponding to the 1^{st}–20^{th} modes. The improved ECB method produces a significantly more accurate reduced model than do the CB and original ECB methods. Table 3 shows the computation time.

Table 4
Specific computation time for the wheel structure problem.

Methods	Items	Computation time	
		[sec]	Ratio [%]
CB	Transformation procedure	1233.00	99.87
	Reduced eigenvalue problem	1.62	0.13
	Total	1234.62	100.00
Original ECB	Calculation of $(\mathbf{K}_s^{(i)})^{-1}$ and $\mathbf{\Psi}_c^{(i)}$	1148.30	93.01
	Solving $\mathbf{K}_s^{(i)}\mathbf{\Phi}_s^{(i)} = \mathbf{\Lambda}_s^{(i)}\mathbf{M}_s^{(i)}\mathbf{\Phi}_s^{(i)}$	4.25	0.34
	Construction of $\overline{\mathbf{M}}$ and $\overline{\mathbf{K}}$	69.05	5.59
	Calculation of $\mathbf{F}_s^{rs}$ and $\mathbf{R}$	34.53	2.80
	Calculation of $\mathbf{A}$ and $\mathbf{E}$	5413.84	438.50
	Construction of $\overline{\mathbf{M}}_e$ and $\overline{\mathbf{K}}_e$	19.34	1.57
	Reduced eigenvalue problem	3.88	0.31
	Total	6693.19	542.13
Improved ECB	Step 1. Algebraic substructuring	0.76	0.06
	Step 2. Calculation of the constraint mode matrix	5.57	0.45
	Step 3. Substructural eigenvalue problems	5.15	0.42
	Step 4. Construction of the reduced matrices of the CB method	11.01	0.89
	Step 5. Interface boundary reduction	9.62	0.78
	Step 6. Residual substructural mode compensation	2.44	0.20
	Step 7. Reduced eigenvalue problem	1.04	0.08
	Total	35.58	2.88

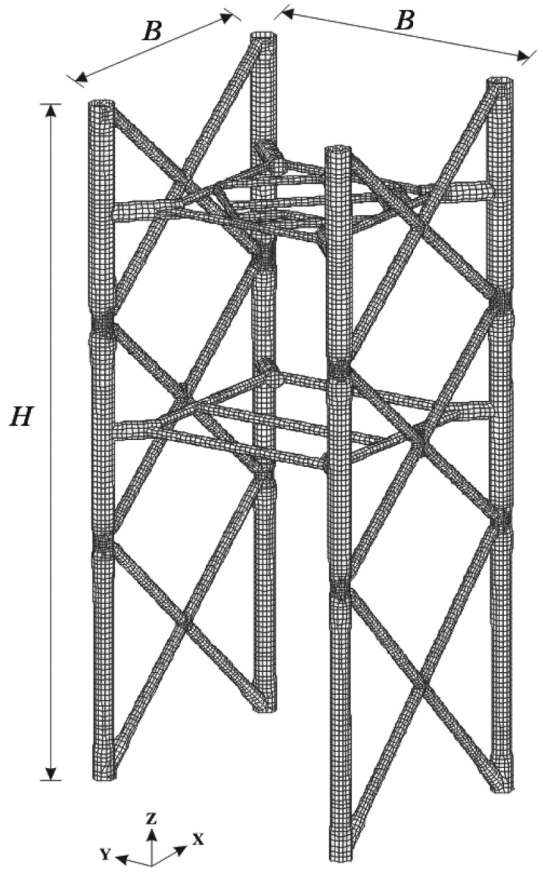

Fig. 7. Jacket structure problem (26484 shell elements, 155766 DOFs, height $H = 87$ m, breadth $B = 37$ m, thickness $t = 0.025$ m).

The improved ECB method only requires 2.94 s to construct the reduced model, while the CB and original ECB methods require 12.46 and 44.39 s, respectively. From these results, we can identify the improved performance of the improved ECB method compared to the CB and original ECB methods in both accuracy and computational efficiency.

5.2. Wheel structure problem

A wheel structure of diameter $D = 0.483$ m is considered, as shown in Fig. 5. We use 83754 four-node tetrahedral solid elements and 23238 nodes for finite element modeling. The number of total DOFs is 69714. The same size of the reduced models is obtained with the CB, original ECB, and improved ECB methods.

Through physical domain-based substructuring, the global structure is partitioned into four substructures for the CB and original ECB methods, as shown in Fig. 5. The numbers of the dominant substructural modes and the interface boundary DOFs, $\overline{N}_d$ and N_b, are 40 and 1347, respectively, and the size of the reduced models $\overline{N}_0$ is 1387. For the proposed method, the global mass and stiffness matrices are automatically partitioned using algebraic substructuring, and the 128 substructures are defined in the algebraic perspective. To construct a reduced model of the same size ($\overline{N}_0 = \overline{N}_1 = 1387$), we select 1267 dominant substructural modes ($\overline{N}_d = 1267$) and 120 dominant interface boundary modes ($\overline{N}_b = 120$).

Fig. 6 presents the relative eigenvalue errors, obtained by the CB, original ECB, and improved ECB methods corresponding to the 1^{st}–25^{th} modes. Table 4 presents the required computation time. As mentioned in Section 2, we can identify that $\mathbf{\Psi}_c^{(i)}$ and $(\mathbf{K}_s^{(i)})^{-1}$ require relatively long computation time with the original ECB method. Furthermore, the matrices $\mathbf{F}_s^{(i)}$, almost fully populated matrices, require 20.12 GB of computer memory, and induce 5413.84 s to calculate matrices $\mathbf{A}$ and $\mathbf{E}$. On the other hand, the proposed method needs only 35.58 s to calculate the reduced model, with higher accuracy than with the CB and original ECB methods.

5.3. Jacket structure problem

We consider a jacket structure, as shown in Fig. 7. The height H and the width B are 87 and 37 m, respectively, and the pipe thickness is 0.025 m. The structure is modeled using 26484 shell elements, and the total number of DOFs is 155766. For this problem, we cannot obtain the reduced model by using the original ECB method, due to insufficient computer memory for $\mathbf{F}_s^{rs}$, as explained in Eq. (10). This requires almost 20.5 GB of computer memory, when we use 16 substructures.

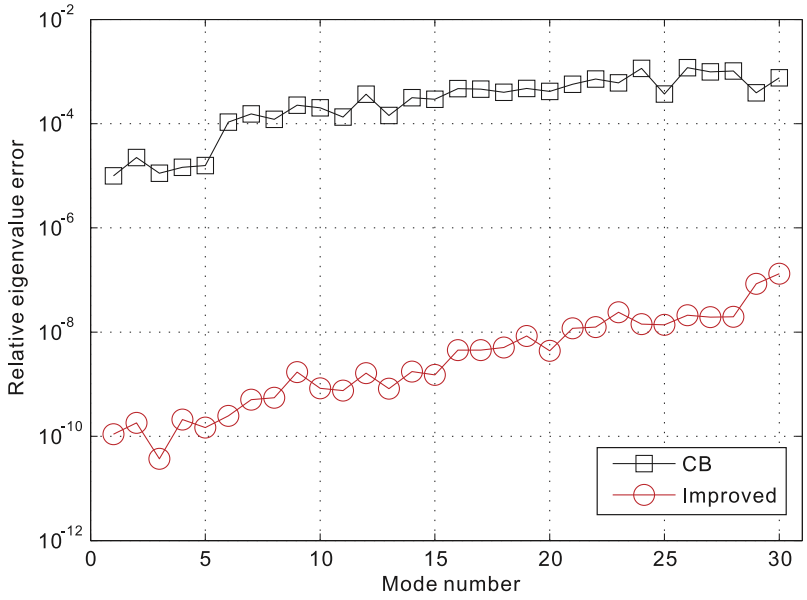

Fig. 8. Relative eigenvalue errors for the jacket structure problem: $\overline{N}_0 = \overline{N}_1 = 3560$.

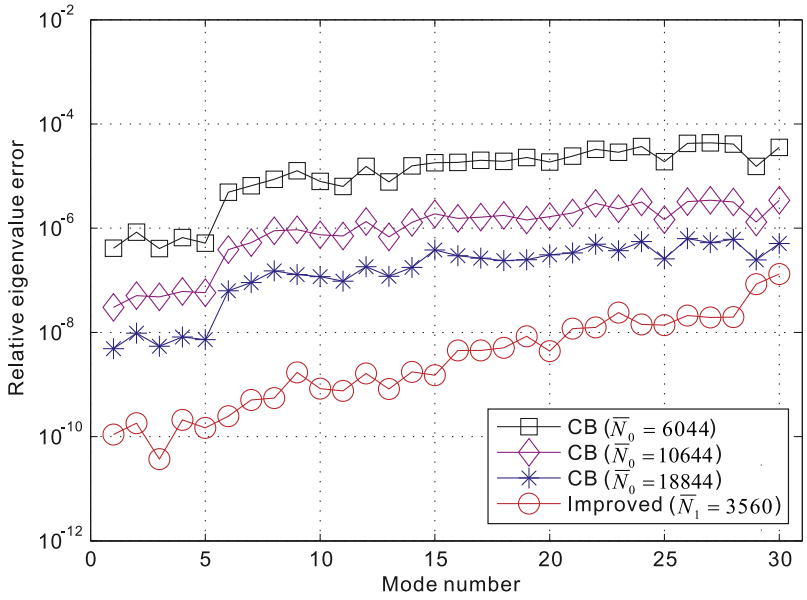

Fig. 9. Relative eigenvalue errors for the jacket structure problem: $\overline{N}_0 = 6044, \overline{N}_0 = 10644, \overline{N}_0 = 18844$, and $\overline{N}_1 = 3560$.

For the CB method, the global structure is partitioned into 16 substructures, and the numbers of dominant substructural modes and the interface boundary DOFs considered are 716 and 2844 ($\overline{N}_d = 716$ and $N_b = 2844$). For the proposed method, algebraic substructuring is used to partition the global matrices into 256 submatrices (or substructures). Then, 3360 dominant substructural modes, and 200 dominant interface boundary modes ($\overline{N}_d = 3360$ and $\overline{N}_b = 200$) are used to construct the reduced model. Fig. 8 presents the relative eigenvalue errors corresponding to the 1^{st}–30^{th} modes, when $\overline{N}_0 = \overline{N}_1 = 3560$. The proposed method shows excellent solution accuracy compared to the CB method.

In addition, we construct larger sized reduced models using the CB method ($\overline{N}_0 = 6044, \overline{N}_0 = 10644$, and $\overline{N}_0 = 18844$). As shown in Fig. 9 and Table 5, although the reduced model calculated by the proposed method is five times smaller than that from the CB method, its solution accuracy and computational efficiency are much better than from the CB method.

Table 5
Computation time for the jacket structure problem when $\overline{N}_0 = 18844$ and $\overline{N}_1 = 3560$.

Methods	Computation time	
	[sec]	Ratio [%]
CB	1687.60	100.00
Original ECB	N/A	–
Improved ECB	100.94	5.98

5.4. Stiffened plate problem

Let us consider a stiffened plate, which is an important structural unit of offshore structures, as shown in Fig. 10. The length

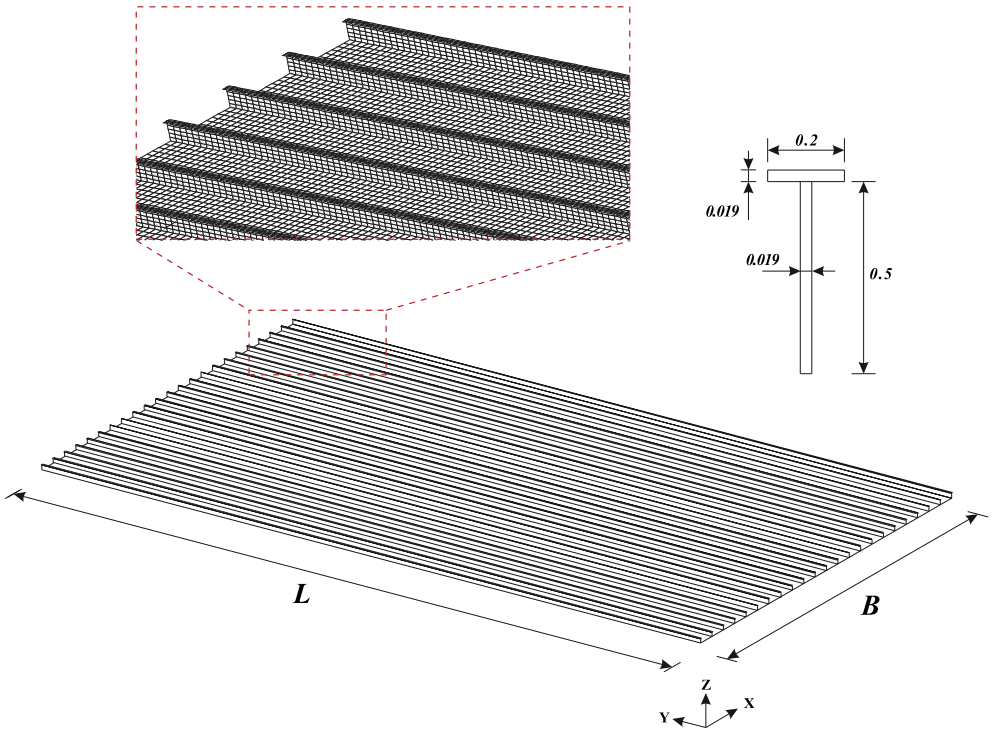

Fig. 10. Stiffened plate problem (166530 shell elements, 1004088 DOFs, length $L = 78.0$ m, breadth $B = 44.0$ m, stiffener spacing $S = 2.0$ m, thickness $t = 0.019$ m).

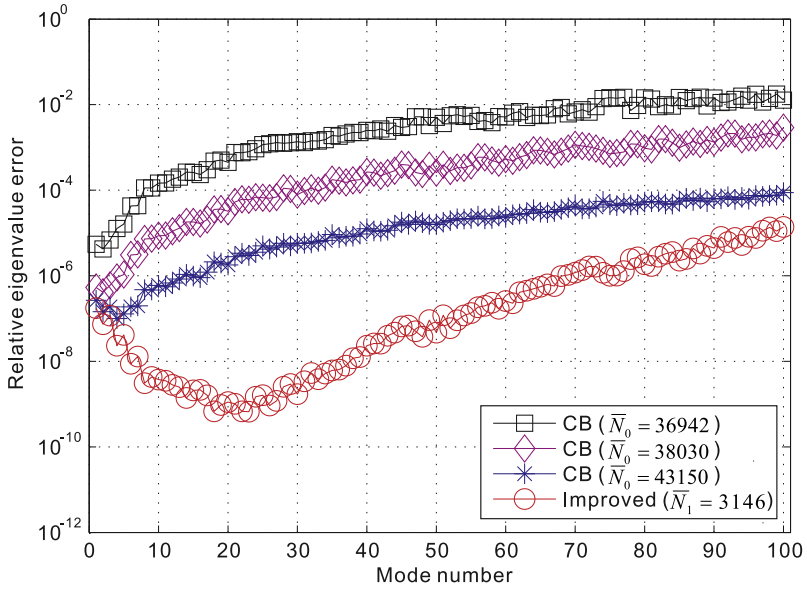

Fig. 11. Relative eigenvalue errors for the stiffened plate problem: $\overline{N}_0 = 36942, \overline{N}_0 = 38030, \overline{N}_0 = 43150$, and $\overline{N}_1 = 3146$.

Table 6
Computation time for the stiffened plate problem when $\overline{N}_0 = 43150$ and $\overline{N}_1 = 3146$.

Methods	Computation time	
	[sec]	Ratio [%]
CB	18994.70	100.00
Original ECB	N/A	–
Improved ECB	934.09	4.92

L, breadth B, and stiffener spacing are 78.0, 44.0, and 2.0 m, respectively, and the thickness is 0.019 m. The stiffener consists of a vertical web of height 0.5 m and a flange of breadth 0.2 m. For FE modeling, 166530 shell finite elements are used, and the total number of DOFs is 1004088. In this problem, the original ECB method fails to calculate reduced models of any size due to the large amount of computer memory required to store $\mathbf{F}_s^{rs}$.

Through physical domain-based substructuring, the global FE model is uniformly partitioned into 64 substructures in the CB method. The algebraic substructuring is performed for the proposed method, and the global mass and stiffness matrices are divided into 1024 substructures.

Fig. 11 shows the relative eigenvalue errors obtained using the CB and proposed (improved ECB) methods for four different sizes of reduced models: $\overline{N}_0 = 36942, \overline{N}_0 = 38030, \overline{N}_0 = 43150$, and $\overline{N}_1 = 3146$. The computation time is described in Table 6, when

$\overline{N}_0 = 43150$ and $\overline{N}_1 = 3146$. From Fig. 11 and Table 6, we can clearly verify that the proposed method produces a small, accurate reduced model with excellent computational efficiency.

6. Conclusions

In this study, we improved the enhanced Craig-Bampton (ECB) method. The improved ECB method was created by employing algebraic substructuring and interface boundary reduction. An important feature of the proposed method is that the residual substructural modes were compensated only for the reduced mass matrix, and this led to significantly reduce computation time. We also presented a computer-aided formulation to provide computationally efficient implementation and to decrease the amount of computer memory required. We demonstrated the excellent computational efficiency and accuracy of the proposed method by solving several practical structural problems. The proposed method could handle relatively large FE models that the original ECB method fails to solve.

In future work, it would be very valuable to develop a more efficient formulation of the enhanced automated multi-level substructuring (EAMLS) method. This could then be used as an alternative to the popularly used LANCZOS eigensolver, when large FE models (containing several millions of DOFs) are solved.

Acknowledgements

This research was supported by the Climate Change Research Hub of KAIST (Grant No. N11170061).

References

[1] Hurty W. Dynamic analysis of structural systems using component modes. AIAA J 1965;3(4):678–85.
[2] Craig RR, Bampton MCC. Coupling of substructures for dynamic analysis. AIAA J 1968;6(7):1313–9.
[3] MacNeal RH. Hybrid method of component mode synthesis. Comput Struct 1971;1(4):581–601.
[4] Benfield WA, Hruda RF. Vibration analysis of structures by component mode substitution. AIAA J 1971;9:1255–61.
[5] Kuhar EJ, Stahle CV. Dynamic transformation method for modal synthesis. AIAA J 1974;12(5):672–8.
[6] Rubin S. Improved component-mode representation for structural dynamic analysis. AIAA J 1975;13(8):995–1006.
[7] Craig RR, Hale AL. Block-Krylov component synthesis method for structural model reduction. AIAA J 1988;11(6):562–70.
[8] Ibrahimbegovic A, Chen HC, Wilson EL, Taylor RL. Ritz method for dynamic analysis of linear systems with non-proportional damping. Int J Earthq Eng Struct Dyn 1990;19:877–89.
[9] Ibrahimbegovic A, Wilson EL. Automated truncation of Ritz vector basis in modal transformation. ASCE J Engrg Mech Div 1990;116:2506–20.
[10] Bourquin F. Component mode synthesis and eigenvalues of second order operators: discretization and algorithm. Math Model Numer Anal 1992;26(3):385–423.
[11] Bourquin F, d'Hennezel F. Numerical study of an intrinsic component mode synthesis method. Comput Methods Appl Mech Eng 1992;97(1):49–76.
[12] Craig RR. Substructure method in vibration. J Vib Acoust 1995;117(B):207–13.
[13] Ibrahimbegovic A, Taylor RL, Lim H. Nonlinear dynamics of flexible multibody systems. Comput Struct 2003;81:1113–32.
[14] Rixen DJ. A dual Craig-Bampton method for dynamic substructuring. J Comput Appl Math 2004;168(1–2):383–91.
[15] Junge M, Brunner D, Becker J, Gaul L. Interface-reduction for the Craig-Bampton and Rubin method applied to FE-BE coupling with a large fluid-structure interface. Intl J Numer Methods Eng 2009;77(12):1731–52.
[16] Boo SH, Kim JG, Lee PS. A simplified error estimator for the Craig-Bampton method and its application to error control. Comput Struct 2016;164:53–62.
[17] Boo SH, Kim JG, Lee PS. Error estimation method for automated multi-level substructuring method. Intl J Numer Methods Eng 2016;106:927–50.
[18] Kim JH, Kim JM, Lee PS. Improving the accuracy of the dual Craig-Bampton method. Comput Struct 2017;191:22–32.
[19] Leung YT. An accurate method of dynamic substructuring with simplified computation. Intl J Numer Methods Eng 1979;14(8):1241–56.
[20] Soize C, Mziou S. Dynamic substructuring in the medium-frequency range. AIAA J 2003;41(6):1113–8.
[21] Klerk DD, Rixen DJ, Voormeeren SN. General framework for dynamic substructuring: history, review and classification of techniques. AIAA J 2008;46(5):1169–81.
[22] John TS, Walter ST. Selection of component modes for flexible multibody simulation. AIAA J 1991;14(2):278–86.
[23] Papadimiriou C, Papadioti DC. Component mode synthesis technique for finite element model updating. Comput Struct 2013;126:15–28.
[24] Hou G, Maroju V. Component mode synthesis-based design optimization method for local structural modification. Struct Optimiz 1995;10:128–36.
[25] Lall S, Marsden JE, Glavaski S. A subspace approach to balanced truncation for model reduction of nonlinear control system. Int J Robust Nonlinear Control 2002;12(6):519–35.
[26] Kammer DC, Triller MJ. Selection of component modes for Craig-Bampton substructure representations. AIAA J 1996;118(2):264–70.
[27] Givoli D, Barbone PE, Patlashenko I. Which are the important modes of a subsystem? Intl J Numer Methods Eng 2004;59(2):1657–78.
[28] Liao BS, Bai Z, Gao W. The important modes of subsystems: a moment-matching approach. Intl J Numer Methods Eng 2007;70(13):1581–97.
[29] Kim JG, Boo SH, Lee PS. Performance of the enhanced Craig-Bampton method. In: The 2015 world congress on advances in civil, environmental, and materials research (ACEM15), Incheon, Korea; August, 2015.
[30] Kim JG, Lee PS. An enhanced Craig-Bampton method. Intl J Numer Methods Eng 2015;103(2):79–93.
[31] George A. Nested dissection of a rectangular finite element mesh. SIAM J Numer Anal 1973;10(2):345–63.
[32] Hendrickson B, Rothberg E. Effective sparse matrix ordering: just around the bend. In: Proceedings of eighth SIAM conference on parallel processing for scientific computing, Minneapolis, Minnesota; March 1997.
[33] Karypis G, Kumar V. METIS v4.0, a software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Technical report. Minneapolis, MN: University of Minnesota; 1998.
[34] Bennighof JK, Lehoucq RB. An automated multilevel substructuring method for eigenspace computation in linear elastodynamics. SIAM J Sci Comput 2004;25(6):2084–106.
[35] Yang C, Gao W, Bai Z, Li XS, Lee LQ, Husbands P, et al. An algebraic substructuring method for large-scale eigenvalue calculation. SIAM J Sci Comput 2005;27(3):873–92.
[36] Boo SH, Lee PS. A dynamic condensation method using algebraic substructuring. Intl J Numer Methods Eng 2017;109(12):1701–20.
[37] Boo SH, Lee PS. An iterative algebraic dynamic condensation method and its performance. Comput Struct 2017;182:419–29.
[38] Boo SH, Oh MH. Automated static condensation method for local analysis of large finite element models. Struct Eng Mech 2017;61(6):807–16.
[39] Rixen DJ. Interface reduction in the Dual Craig-Bampton method based on dual interface modes. Linking Models Exp 2011;2:311–28.
[40] Stewart GW. Matrix algorithms: basic decompositions. SIAM J 1998;1.
[41] Bathe KJ. Finite element procedures. USA: Prentice Hall; 1996.
[42] Dvorkin EN, Bathe KJ. A continuum mechanics based four-node shell element for general nonlinear analysis. Eng Comput 1984;1(1):77–88.
[43] Lee PS, Bathe KJ. Development of MITC isotropic triangular shell finite elements. Comput Struct 2004;82(11–12):945–62.
[44] Lee YG, Lee PS, Bathe KJ. The MITC3+ shell finite element and its performance. Comput Struct 2014;138:12–23.
[45] MATLAB R2013a. The MathWorks Inc, Natick, Massachusetts, United States.