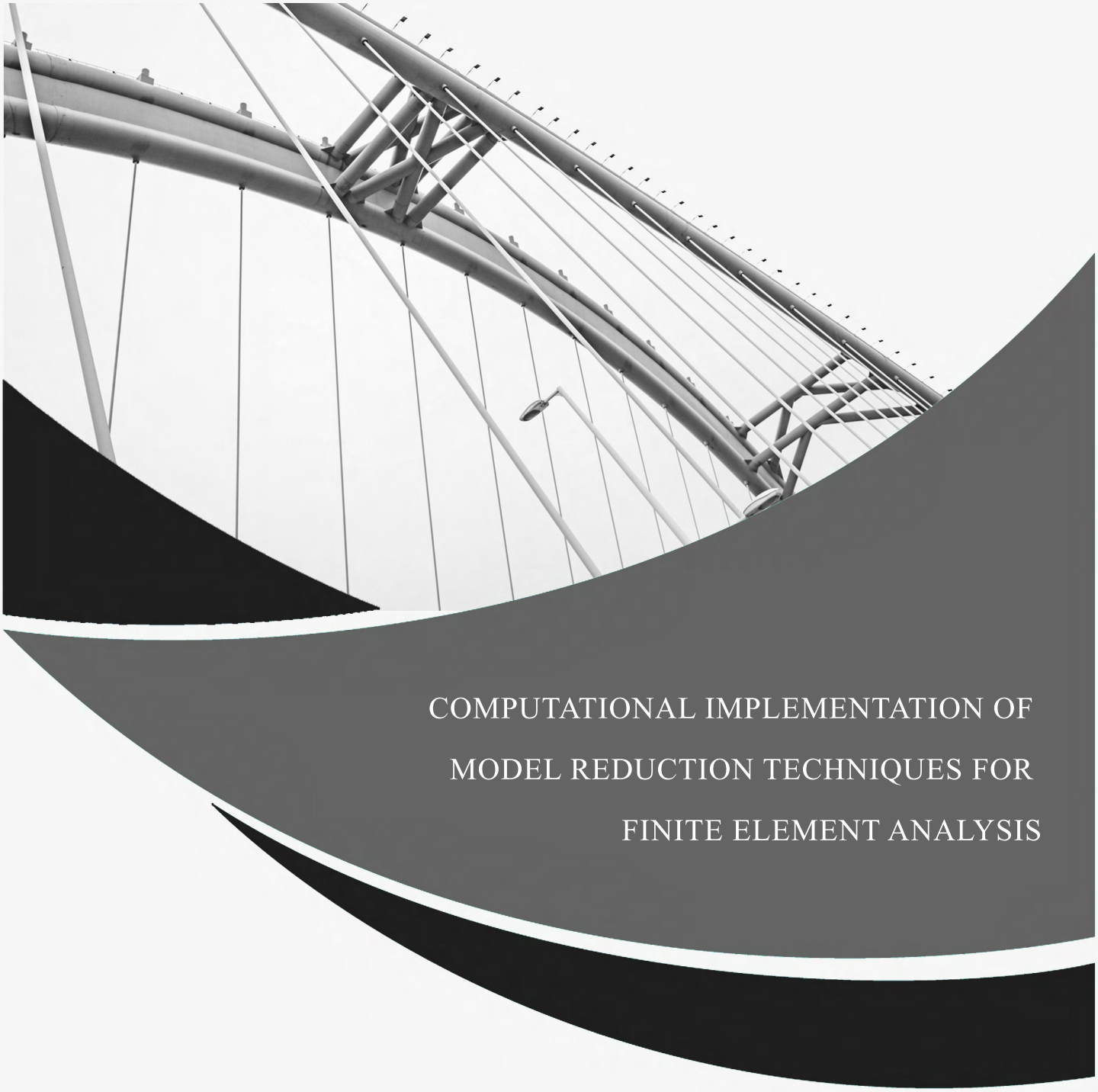


COMPUTATIONAL IMPLEMENTATION OF MODEL REDUCTION TECHNIQUES FOR FINITE ELEMENT ANALYSIS

Aspasia Giaouzi

Volos, July 2018

Univeristy of Thessaly, Faculty of Engineering
Department of Mechanical Engineering
Sector of Mechanics, Materials & Manufacturing Processses

Approved by the three members of the selection Committee

First Examiner: (Supervisor)	Dr. Kostantinos Papadimitriou Professor, Department of Mechanical Engineering, University of Thessaly
Second Examiner:	Dr. Anna Zervaki Adjunct Professor, Department of Mechanical Engineering, University of Thessaly
Third Examiner:	Dr. Spyridon Karamanos Professor, Department of Mechanical Engineering, University of Thessaly

Credits

First of all I would like to thank my thesis supervisor, Professor Mr. Konstantinos Papadimitriou, for his help and guidance during my project. Furthermore, I am thankful to the members of the selection committee of my thesis, Professors Mrs. Anna Zervaki and Mr. Spyros Karamanos for the meticulous examining of my project and their valuable suggestions. I also would like to thank my friends and especially my family for their constant moral support, love and understanding specifically the last few months of my endeavors.

Aspasia Giaouzi

COMPUTATIONAL IMPLEMENTATION AND INSPECTION OF MODEL REDUCTION TECHNIQUES FOR FINITE ELEMENT ANALYSIS

ASPASIA GIAOUZI

University of Thessaly, Department of Mechanical Engineering, 2018

Supervisor: Dr. Constantinos Papadimitriou, Professor of Structural Dynamics

Synopsis

The current work presents briefly, the methodology of Model Reduction process, utilizing firstly, the component mode synthesis techniques to carry out the re-analyses efficiently in a substantially reduced space of generalized coordinates using exact component modes and characteristic interface modes computed only once from a reference finite element model, and subsequently implementing an enhanced substructure coupling technique in the context of complex simulation based problems.

The rest of the thesis demonstrates the computational implementation of the aforementioned techniques and evaluates their practical accuracy and efficiency as far as computational effort, time and divergence in results are concerned.

Contents

Contents

1 Introduction

1.1 Motive and background of the current thesis

The first incentive of this work is to present in a summarized but inclusive way the proposed Model Reduction method, in the context of complex simulation based problems. For this purpose both the original formulation as well as an enhanced one are examined.

Component mode synthesis techniques are proposed to carry out the re-analyses efficiently in a substantially reduced space of generalized coordinates using exact component modes and characteristic interface modes computed only once from a reference FE mode.

The enhanced technique of Static Correction is derived by first considering explicitly the effect of higher order substructural modes in the definition of reduced-order models. Examining the static solution of those substructural modes, in other words, the fact that high frequency modes react essentially in a static manner when excited by low frequencies, produces a drastic reduction in computational effort. The model reduction is achieved efficiently, without compromising the accuracy of the results and computational efforts are reduced by more than two orders of magnitude.

The second incentive of the project is to explore and cite the effectiveness and accuracy in results, of those techniques while implemented computationally. For this purpose, two softwares, with the ability to interchange vital structural information, Matlab and Comsol, as well as a well-defined structure, were utilized.

1.2 Organization of the present thesis

As far as the rest of the project is concerned it is divided in three sections which occupy the chapters 2 to 5.

In Chapter 2 the basic structural model is introduced and a class of model reduction techniques known as component mode synthesis or substructure coupling for dynamic analysis is briefly reviewed.

In Chapter 3 the standard formulation of the model reduction method involving component mode synthesis technique using fixed normal interface modes and interface constrained modes, is demonstrated.

In Chapter 4 the enhanced formulation of model reduction method, using substructural coupling and utilizing the static correction is briefly cited and evaluated as far as computational saving and accuracy of results is concerned.

In Chapter 5 the results of the computational implementation of the above methods on a rudimentary as well as a well-defined model are presented with a brief evaluation of their accuracy and fluctuation according to a specific input.

The appendix includes a user's guide on how to successfully utilize the tools of Comsol Multiphysic© and the commands of MatLab© in order to create and study a structure and extract the data needed in the most effective way through model reduction.

1.3 Synoptic bibliographic overview

Structural model updating methods [1-3], are used to link mathematical models, usually discretized finite element (FE) models, with experimental data. Structural model parameter estimation problems based on identified modal characteristics (modal frequencies and mode shapes), are often formulated as weighted least-squares problems [2, 4-8] in which metrics, measuring the residuals between measured and model predicted modal characteristics, are build up into a single weighted residuals metric formed as a weighted average of the multiple individual metrics using weighting factors. Standard optimization techniques are then used to find the optimal values of the structural parameters that minimize the single weighted residuals metric. Due to model error and measurement noise, the results of the optimization are affected by the values assumed for the weighting factors. The model updating problem has also been formulated as a multi-objective optimization problem [9, 10] that allows the simultaneous minimization of the multiple metrics, eliminating the need for using arbitrary weighting factors for weighting the relative importance of each metric in the overall measure of fit.

The multi-objective parameter estimation methodology provides multiple Pareto optimal structural models. The Normal Boundary Intersection algorithm [11], is used to compute the Pareto optimal solutions.

Bayesian techniques [12,13] have also been proposed to quantify the uncertainty in the parameters of a FE model, select the best model class from a family of competitive model classes [14,15], as well as propagate uncertainties for robust response and reliability predictions [16].

The Bayesian tools for identifying uncertainty models as well as offering robust prediction analyses are Laplace methods of asymptotic approximation and more accurate stochastic simulation algorithms (SSA) such as Markov Chain Monte Carlo (MCMC) [17] , Transitional MCMC [18] and Delayed Rejection Adaptive Metropolis [19]. The optimal structural models and their uncertainties resulting from model updating methods can be used for improving the model response and reliability predictions [16, 20], for assessing structural health and identifying structural damage [5-8] and for improving effectiveness of structural control devices.

The optimization and SSA algorithms require a moderate to very large number of FE re-analyses to be performed over the space of model parameters. Consequently, the computational demands depend highly on the number of FE re-analyses and the time required for performing a FE analysis. In addition, gradient-based optimization algorithms require the estimation of the gradients of the residuals which may also add substantially to the computational effort. For high fidelity FE models involving hundreds of thousands or even million DOFs, the computational demands may be large or even excessive.

Specifically, component mode synthesis (CMS) techniques are widely used to carry out system analyses in a significantly reduced space of generalized coordinates. Such techniques have been incorporated in methods for uncertainty management in structural dynamics to efficiently handle the computational effort in system re-analyses that arise from FE model variations caused by variations in the values of the uncertain parameters. Such variations in the values of the model parameters require that the computation of the component and/or system modes be repeated in each re-analysis.

As a result, a computational overhead arises at component level which may be substantial.

The main objective in methods involving re-analyses of models with varying properties is to avoid, to the extent possible, the re-computation of the eigenproperties at the component or system level. Perturbation techniques provide accurate results locally for small variations of the model parameters about a reference structure. To improve the accuracy of the approximations for large variation of the model parameters, most efforts have been concentrated in approximating the modes at the component or system level in terms of the modes of a family of structures corresponding to support points in the parameter space. Linear and quadratic interpolations of the structural mass and stiffness matrix and the matrix of eigenvectors at the component and/or system level using support points in the larger region in the parameter space have been successfully used for model updating of large-order models of structures. Similar methods have been developed for damage detection at component level. Such techniques proved to be quite effective in substantially reducing the computational demands in problems requiring system re-analyses.

Finally, an appendix is created in order to comprehensively guide an average software user on how to exploit the model reduction technique for their own benefit, using the MatLab© software combined with the commercial software package COMSOL Multiphysics ©, which is used for developing and defining the structure and ultimately performing a finite element analysis on it in order to produce necessary data.

2 Model Reduction Techniques for Structural Dynamic Analyses

In this chapter the basic structural model defining a general class of dynamic systems and its parameters are presented. The convenience and efficiency of implementing a model reduction technique compared to an entire finite element analysis is also demonstrated and then further explained through the process of substructure coupling. Substructure coupling involves dividing the structure into a number of linear and nonlinear substructures, obtaining reduced-order models of the linear substructures and then assembling a reduced-order model of the entire structure.

2.1 Basic Structural Model

Attention is focused on a general class of structural dynamical systems with localized nonlinearities characterized by multi-degrees of freedom models satisfying the equation of motion:

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = f_{NL}\big(u(t), \dot{u}(t), y(t)\big) + f(t) \qquad (1)$$

where $u(t)$ denotes the displacement vector of dimension n, $\dot{u}(t)$ the velocity vector, $\ddot{u}(t)$ the acceleration vector, $f_{NL}\big(u(t), \dot{u}(t), y(t)\big)$ the vector of non-linear restoring forces, $y(t)$ the vector of a set of variables which describes the state of the nonlinear components, and $f(t)$ the external force vector. The matrices M, C, and K describe the mass, damping, and stiffness, respectively. The evolution of the set of variables $y(t)$ is described by an appropriate nonlinear model which depends on the nature of the nonlinearity. The equation of motion for the displacement vector $u(t)$ and the equation

for the evolution of the set of variables $y(t)$ constitute a system of coupled non-linear equations. This characterization of the dynamical system allows the modeling of different types of commonly used nonlinearities such as hysteresis, degradation, plasticity, and other types of nonlinearities.

For the dynamic characterization of this class of structural dynamical systems it is often inefficient to carry out a finite element analysis of the entire model. In fact, in many dynamic analysis problems the lower frequencies and the corresponding modes tend to dominate the dynamic behavior of the structure. It is also common for complex systems the component structures to be analyzed independently, which renders it more convenient to perform a dynamic analysis at substructure level.

In this framework, model reduction techniques have been developed as a practical and efficient tool for modeling and analyzing the dynamic of complex structural systems. The objective of model reduction techniques is to obtain reduced-order models that run significantly faster than the original high-fidelity model, incorporating the important dynamics of the system analyzed so that the analyses from the reduced-order models are sufficiently accurate.

2.2 Substructure modes and coupling

Substructure coupling involves dividing the structure into a number of linear and nonlinear substructures, obtaining reduced-order models of the linear substructures and then assembling a reduced-order model of the entire structure. Specifically, after the division of the structure into substructures the model reduction technique involves two basic steps: (a) definition of sets of substructure modes and (b) coupling of the

substructure modes models, to form a reduced-order system model. Substructure modes include normal, constraint, rigid-body and attachment modes.

Depending on the substructure modes considered, substructuring can be grouped into fixed-interface, free-interface and loaded-interface methods. Among the previous methods, the Craig-Bampton method, which is a fixed-interface technique, is widely used for its simplicity and computational stability. In the approach used in the present work, the substructure modes correspond to fixed-interface normal modes and interface constraint modes. In this manner, the dynamic behavior of the linear components of the structural system is described by a set of normal modes of individual substructures plus a set of constraint modes that account for the coupling at each interface where the substructures are connected.

The presented method below exploits the fact that in FE model parameterization schemes, the stiffness matrix of the structure often depends linearly on the parameters of the model and also that a parameter usually represents a global property (e.g. the modulus of elasticity, E) of a substructure. The division of the structure into components is then guided by the FE parameterization scheme so that the stiffness matrix that arise for each one of the introduced components to depend linearly on only one of the parameters to be estimated.

In this case the fixed-interface and constraint modes of the components for any value of the model parameters can be obtained directly from the fixed-interface and constraint modes corresponding to a single reference FE model, avoiding re-analyses at component level. Additional substantial reductions in computational effort are also introduced by reducing the number of interface DOFs, using characteristic interface modes through a Ritz coordinate transformation. The repeated solutions of the

component and interface eigen-problems are avoided, reducing drastically the computational demands in FE formulations, without compromising at a great extent the solution accuracy. It is also shown that the linear expansions of the original mass and stiffness matrices in terms of the structural parameters are preserved for the reduced mass and stiffness matrices. Thus, the re-assembling of the reduced system matrices from the original matrices is also avoided in the execution of the system re-analyses. The only time consuming operation left is the re-analysis of the eigenproblem of the reduced-order model.

3 Component mode synthesis (Analysis of the basic method implemented)

In this chapter component mode synthesis techniques are examined and proposed to carry out the re-analyses efficiently in a substantially reduced space of generalized coordinates using exact component modes and characteristic interface modes computed only once from a reference FE model. The re-assembling of the reduced-order system matrices from components and interface modes is avoided.

3.1 Formulation using fixed-interface and constrained interface modes

In CMS techniques, a structure is divided into several components. For each component, the unconstrained DOFs are partitioned into the boundary DOFs, denoted by the subscript b and the internal DOFs, denoted by the subscript i. The boundary DOFs of a component include only those that are common with the boundary DOFs of adjacent components, while the internal DOFs of a component are not shared with any adjacent component. The stiffness and mass matrices $K^{(s)} \in R^{n(s)\,x\,n(s)}$ and $M^{(s)} \in R^{n(s)\,x\,n(s)}$ of a component are partitioned to blocks related to the internal and boundary DOFs as follows:

$$M^{(s)} = \begin{bmatrix} M_{ii}^{(s)} & M_{ib}^{(s)} \\ M_{bi}^{(s)} & M_{bb}^{(s)} \end{bmatrix} \qquad (2)$$

$$K^{(s)} = \begin{bmatrix} K_{ii}^{(s)} & K_{ib}^{(s)} \\ K_{bi}^{(s)} & K_{bb}^{(s)} \end{bmatrix} \qquad (3)$$

where the indices i and b are sets containing the internal and boundary DOF of the component. According to the Craig–Bampton fixed-interface mode method, the Ritz coordinate transformation I mode method, the Ritz coordinate transformation $u^{(s)} = u_i{}^{(s)T},\ u_b{}^{(s)T} = \Psi^{(s)}\, p^{(s)}$, where

$$\Psi^{(s)} = \begin{bmatrix} \Phi_{ik}^{(s)} & \Psi_{ib}^{(s)} \\ 0_{bk}^{(s)} & I_{bb}^{(s)} \end{bmatrix} \qquad (4)$$

is used to relate the physical displacement coordinates $u^{(s)} \in R^{n(s)}$ of the component to the generalized coordinates $p^{(s)} = [\ p_k^{(s)T},\ \ p_b^{(s)T}] \in R^{n(s)}$, $\hat{n}^{(s)} = n_k^{(s)} + n_b^{(s)}$ using the kept fixed-interface normal modes $\Phi_{ik}^{(s)} \in R^{n_i(s)\ x\ n_k(s)}$ satisfying the eigen-problem:

$$K_{ii}^{(s)} \Phi_{ik}^{(s)} = M_{ii}^{(s)} \Phi_{ik}^{(s)} \Lambda_{kk}^{(s)} \qquad (5)$$

and the interface constrained modes $\Psi ib^{(s)} \in R^{n_i(s)\ x\ n_b(s)}$ given by

$$\Psi_{ib}^{(s)} = -[K_{ii}^{(s)}]^{-1} K_{ib}^{(s)} \qquad (6)$$

The matrix $\Lambda_{kk}^{(s)} = diag\left(\lambda_1^{(s)}, \dots, \lambda_{n_k^{(s)}}^{(s)}\right) \in R^{n_k(s)\ x\ n_k(s)}$ is diagonal containing the eigenvalues $\lambda_j^{(s)},\ j = 1, \dots, n_k^{(s)}$, of the fixed-interface normal modes.

The fixed interface modes $\Phi_{ik}^{(s)}$ are considered to be mass normalized satisfying

$$\Phi_{ik}^{(s)T} M_{ii}^{(s)} \Phi_{ik}^{(s)} = I_{kk}^{(s)} \qquad \Phi_{ik}^{(s)T} K_{ii}^{(s)} \Phi_{ik}^{(s)} = \Lambda_{kk}^{(s)} \qquad (7)$$

The components' mass and stiffness matrices $\widehat{M}^{(s)} \in R^{\hat{n}^{(s)} x \hat{n}^{(s)}}$ and $\widehat{K}^{(s)} \in R^{\hat{n}^{(s)} x \hat{n}^{(s)}}$ in the new reduced set of generalized coordinates $p^{(s)}$ are transformed as follows:

$$\widehat{M}^{(s)} = \Psi^{(s)T} M^{(s)} \Psi^{(s)} \qquad \widehat{K}^{(s)} = \Psi^{(s)T} K^{(s)} \Psi^{(s)} \qquad (8)$$

with the partitions for the component mass matrices:

$$\widehat{M}_{kk}^{(s)} \in R^{\widehat{n_k}^{(s)} x \widehat{n_k}^{(s)}}$$

$$\widehat{M}_{kb}^{(s)} \in R^{\widehat{n_k}^{(s)} x \widehat{n_b}^{(s)}}$$

$$\widehat{M}_{bb}^{(s)} \in R^{\widehat{n_b}^{(s)} x \widehat{n_b}^{(s)}}$$

and stiffness matrices:

$$\widehat{K}_{kk}^{(s)} \in R^{\widehat{n_k}^{(s)} x \widehat{n_k}^{(s)}}$$

$$K_{kb}^{(s)} \in R^{\widehat{n_k}^{(s)} x \widehat{n_b}^{(s)}}$$

$$K_{bb}^{(s)} \in R^{\widehat{n_b}^{(s)} x \widehat{n_b}^{(s)}}$$

Given respectively by:

$$\widehat{M}_{kk}^{(s)} = I_{kk}^{(s)}$$

$$\widehat{M}_{kb}^{(s)} = \widehat{M}_{bk}^{(s)T} = \Phi_{ik}^{(s)T} M_{ii}^{(s)} \Psi_{ib}^{(s)} + \Phi_{ik}^{(s)T} M_{ib}^{(s)}$$

$$\widehat{M}_{bb}^{(s)} = \left(\Psi_{ib}^{(s)T} M_{ii}^{(s)} + \widehat{M}_{bi}^{(s)} \right) \Psi_{ib}^{(s)} + \Psi_{ib}^{(s)T} M_{ib}^{(s)} + M_{bb}^{(s)} \tag{9}$$

And

$$\widehat{K}_{kk}^{(s)} = \Lambda_{kk}^{(s)}$$

$$\widehat{K}_{kb}^{(s)} = \widehat{K}_{bk}^{(s)T} = 0_{kb}^{(s)}$$

$$\widehat{K}_{bb}^{(s)} = K_{bb}^{(s)} - K_{bi}^{(s)} [K_{ii}^{(s)}]^{-1} K_{ib}^{(s)} = K_{bb}^{(s)} + \Psi_{ib}^{(s)T} K_{ib}^{(s)} \tag{10}$$

In the substructure assembly process, the vector

$p = [p^{(1)T}, \dots, p^{(Nc)T}]^T \in R^{n_p}, \quad n_p = \sum_{s=1}^{Nc} \hat{n}^{(s)}$, of the generalized coordinates for all Nc components is introduced.

Letting $q = \left[{p_k}^{(1)T}, \dots, {p_k}^{(Nc)T}, u_b^T \right]^T \in R^{n_q}$, be vector that contains the independent generalized coordinates consisting of the fixed-interface modal coordinates $p_k^{(s)}$ for

each component and the physical coordinates $u_b^T = \left[u_b^{(1)T}, \ldots, u_b^{(Nb)T}\right]^T$ at the Nb interfaces, where $u_b^{(l)}$ contains the displacements at the DOF of the interface l , the following transformation is introduced:

$$p = Sq \qquad (11)$$

where the component coupling matrix $S \in R^{n_p \, x \, n_q}$ is a matrix of zeros and ones that couples the independent generalized coordinates q of the reduced system with the generalized coordinates of each component.

The assembled Craig–Bampton stiffness matrix $\widehat{K}^{CB} \in R^{n_q \, x \, n_q}$ and mass matrix $\widehat{M}^{CB} \in R^{n_q \, x \, n_q}$ for the reduced vector q of generalized coordinates are given by:

$$\widehat{K}^{CB} = S^T \begin{bmatrix} \widehat{K}^{(1)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \widehat{K}^{(Nc)} \end{bmatrix} S = \sum_{s=1}^{Nc} F_s \left[\widehat{K}^{(s)}\right] \qquad (12)$$

$$\widehat{M}^{CB} = S^T \begin{bmatrix} \widehat{M}^{(1)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \widehat{M}^{(Nc)} \end{bmatrix} S = \sum_{s=1}^{Nc} F_s \left[\widehat{M}^{(s)}\right] \qquad (13)$$

where the new mathematical operator $F_s\left[\widehat{K}^{(s)}\right]$ is conveniently introduced by the second part of equation (12) as:

$$F_s\left[\widehat{K}^{(s)}\right] = S^T blockdiag[0_{\hat{n}^{(1)}\hat{n}^{(1)}}, \dots, 0_{\hat{n}^{(s-1)}\hat{n}^{(s-1)}}, \widehat{K}^{(s)}, 0_{\hat{n}^{(s+1)}\hat{n}^{(s+1)}}, \dots, 0_{\hat{n}^{(Nc)}\hat{n}^{(Nc)}}]S \quad (14)$$

Solving the reduced eigen-problem:

$$\widehat{K}^{CB} Q = \widehat{M}^{CB} Q \Lambda \quad (15)$$

associated with the reduced mass and stiffness matrices $\widehat{M}^{CB}$ and $\widehat{K}^{CB}$, respectively, one obtains the modal frequencies $\Lambda = diag(\omega^2) \in R^{n_q \, x \, n_q}$ and the corresponding mode shape matrix $Q = [\hat{q}_1, \dots, \hat{q}_{n_q}] \in R^{n_q x \, n_q}$ of the reduced system.

3.2 Reduction of the interface DOF using characteristic interface modes

Further reduction in the generalized coordinates can be achieved by replacing the interface DOF by a reduced number of characteristic interface modes. For this, the physical displacement coordinates $\underline{u}_b^{(l)} \in R^{m_b(l)}$ at an interface l between two components are represented in terms of the generalized coordinates $\underline{\zeta}^{(l)} \in R^{m_k(l)}$ of the interface by the Ritz coordinate transformation

$$\underline{u}_b^{(l)} = V^{(l)} \underline{\zeta}^{(l)} \quad (16)$$

$l = 1, \dots, Nb$, where the columns of $V^{(l)} \in R^{m_b(l) x m_k(l)}$ form the reduced basis of the $mb^{(l)}$-dimensional space and $mk^{(l)}$ is the number of elements in the basis.

The following transformation from the CMS generalized coordinates $\underline{q}$ to the reduced-order model generalized coordinates

$$\underline{u} = \left[\underline{p}_k^{(1)T}, \dots, \underline{p}_k^{(Ns)T}, \zeta^{(1)T}, \dots., \zeta^{(Nb)T}\right]^T \in R^{nr}$$

$$n_r = \sum_{s=1}^{Nc} n_k^{(s)} + \sum_{l=1}^{Nb} m_k^{(l)}$$

that contains the kept fixed interface modes and the kept characteristic interface modes, is introduced as:

$$\underline{q} = V\underline{u} \qquad (17)$$

where $V = blockdiag(I_{n_k^{(1)}}, \dots, I_{n_k^{(Nc)}}, V^{(1)}, \dots, V^{(Nb)}) \in R^{nq\ x\ nr}$ and I_n denotes the identity matrix of dimension n. Using (17), the final reduced mass and stiffness matrices take the form:

$$\widehat{K} = V^T \widehat{K}^{CB} V \quad \text{and} \quad \widehat{M} = V^T \widehat{M}^{CB} V \qquad (18)$$

and the resulting eigenvalue problem at the reduced system level becomes:

$$\widehat{K}\Gamma = \widehat{M}\Gamma\Lambda \qquad (19)$$

where the diagonal matrix K contains the modal frequencies and the matrix $\Gamma \in R^{nr\ x\ nr}$ contains the corresponding n_r mode shapes of the reduced system.

The kept characteristic interface modes of the matrix $V^{(l)}$ satisfy the eigen-problem:

$$\widehat{K}^{CB}_{\hat{b}_l\hat{b}_l} V^{(l)} = \widehat{M}^{CB}_{\hat{b}_l\hat{b}_l} V^{(l)} \Omega^{(l)} \qquad (20)$$

where b_l is the index set denoting the positions of the generalized coordinates $\underline{u}_b^{(l)} \in R^{mb^{(l)}}$ in the vector q corresponding to the interface, while the stiffness and mass matrices $\widehat{K}^{CB}_{\hat{b}_l\hat{b}_l} \in R^{m_b(l) x m_b(l)}$ and $\widehat{M}^{CB}_{\hat{b}_l\hat{b}_l} \in R^{m_b(l) x m_b(l)}$ in (20) are the partitions of the reduced stiffness and mass matrices $\widehat{K}^{CB}$ and $\widehat{M}^{CB}$ associated with the coordinates $\underline{u}_b^{(l)}$ at the *l*-th interface.

These partitions are easily obtained from the corresponding partitions of the stiffness and mass matrices of the components connecting to the interface *l* in the form:

$$\widehat{K}^{CB}_{\hat{b}_l\hat{b}_l} = \sum_{s\in C_l} \widehat{K}^{(s)}_{b_l b_l} \qquad \text{and} \qquad \widehat{M}^{CB}_{\hat{b}_l\hat{b}_l} = \sum_{s\in C_l} \widehat{M}^{(s)}_{b_l b_l} \qquad (21)$$

Where C_l is the integer set that contains the components that connect to the interface *l* and b_l , is the index set denoting the positions of the $\underline{u}_b^{(l)} \in R^{mb^{(l)}}$ corresponding to the interface *l* in the vector $\underline{u}^{(s)}$ of the component **s**.

3.3 Model updating using CMS

The CMS procedure is integrated into the FE model updating. The linear dependence of the mass and stiffness matrices on the parameter θ, implies that at the component level the mass and stiffness matrices as well as their partitions admit a similar representation, that is:

$$K^{(s)} = K_0^{(s)} + \sum_{J=1}^{N\theta} K_j^{(s)} \theta_j$$

$$M^{(s)} = M_0^{(s)} + \sum_{J=1}^{N\theta} M_j^{(s)} \theta_j \qquad (22)$$

Attention is focused on two special cases of the parameterization for a component s.

In the first case it is assumed that the mass and stiffness matrix of a component s do not depend on the model parameters in $\underline{\theta}$. In this case, $K^{(s)} = K_0^{(s)}$ and $M^{(s)} = M_0^{(s)}$. The component fixed-interface and constrained modes are independent of the parameter values. Only a single analysis is required to estimate the fixed-interface and constrained modes for the particular component s. These component modes are computed once for a reference model and are then used in the iterations or TMCMC sampling schemes involved in model updating.

The computational savings arise from the fact that the eigenvalue problem to compute the eigenvalues and mode shapes of the kept interface modes $\Phi_{ik}^{(s)}$ as well as the solution

of the linear system to compute the constrained interface modes $\Psi_{ib}^{(s)}$ for a component s are not repeated at each iteration or TMCMC sampling point.

In the second case the stiffness matrix of a structural component s depends only on one model parameter, say $\underline{\theta}_j$, in the parameter vector $\underline{\theta}$, while the mass matrix $M^{(s)} = M_0^{(s)}$ is constant independent of $\underline{\theta}$. This case is enforced by dividing the structure into components based on the parameters introduced in the FE model for each physical substructure.

Let Δ_j be the set of components that depend on the j-th variable θ_j.

The stiffness matrix of a component s $\in \Delta_j$ takes the form:

$$K^{(s)} = \overline{K}^{(s)}\theta_j \qquad (23)$$

Substituting the partitions $K_{ii}^{(s)} = \overline{K}_{ii}^{(s)}\theta_j$ and $M_{ii}^{(s)} = M_{0,ii}^{(s)}$ in (12), it is readily derived that the matrix of the kept eigenvalues and eigenvectors of the component fixed-interface modes are given with respect to the parameter $\underline{\theta}_j$ in the form:

$$\Lambda^{(s)} = \overline{\Lambda}^{(s)}\theta_j \quad \text{and} \quad \Phi_{ik}^{(s)} = \overline{\Phi}_{ik}^{(s)} \qquad (24)$$

where the matrices $\overline{\Lambda}^{(s)}$ and $\overline{\Phi}_{ik}^{(s)}$ are solutions of the following eigenproblem:

$$\overline{K}_{ii}^{(s)}\overline{\Phi}_{ik}^{(s)} = M_{0,ii}^{(s)}\overline{\Phi}_{ik}^{(s)}\overline{\Lambda}_{kk}^{(s)} \qquad (25)$$

and thus they are independent of the values of $\underline{\theta_j}$ or the FE model variations at the component level due to changes in the model parameter. Also using the stiffness matrix partitions $K_{ii}^{(s)} = \overline{K}_{ii}^{(s)}\theta_j$ and $K_{ib}^{(s)} = \overline{K}_{ib}^{(s)}\theta_j$, the constrained modes are given by the constant matrix:

$$\Psi_{ib}^{(s)} = -\left[K_{ii}^{(s)}\right]^{-1}K_{ib}^{(s)} = -\left[\overline{K}_{ii}^{(s)}\right]^{-1}\overline{K}_{ib}^{(s)} \qquad (26)$$

also independent of the values of the parameter θ_j or FE model variations at component level. Thus, a single component analysis is required to provide the exact estimate of the fixed-interface modes from (24) and the constrained modes from (26) for any value of the model parameter θ_j .

Substituting into the reduced mass and stiffness matrices (9) and (10) the partitions of the stiffness matrix (23), the eigenproperties (22) and the interface constraint modes (26) of the component s, it is straightforward to verify that **the reduced stiffness matrix of component s** takes the form:

$$K^{(s)} = \widehat{\overline{K}}^{(s)}\theta_j \qquad (27)$$

where the reduced stiffness matrix $\widehat{\overline{K}}^{(s)}$ and the reduced mass matrix $\widehat{M}^{(s)}$ are constant matrices, independent of the values of the model parameters $\underline{\theta}$.

Introduce next the index set R to contain the structural components s that depend on a parameter in the vector $\underline{\theta}$. Then the set $\bar{\Sigma} = \{1, \dots, Nc\} - \Sigma$ contains the component numbers for which their properties are constant and independent on the values of the parameter vector $\underline{\theta}$. Substituting (27) into (12), the stiffness matrix of the Craig–Bampton reduced system admits the representation:

$$\widehat{K}^{CB} = \widehat{K}_0^{CB} + \sum_{j=1}^{N\theta} \widehat{K}_j^{CB}\, \theta_j \qquad (28)$$

and the mass matrix is given by $\widehat{M}^{CB} = \widehat{M}_0^{CB}$, where the coefficient matrices $\widehat{K}_0^{CB}$ and $\widehat{K}_j^{CB}$ in the expansion (28) are assembled from the component stiffness matrices, defined in (27), by:

$$\widehat{K}_0^{CB} = \sum_{s\in\Sigma} F_s\, [\widehat{K}^{(s)}]$$

$$\widehat{K}_j^{CB} = \sum_{s\in\Delta j} F_s\, [\widehat{\overline{K}}^{(s)}] \qquad (29)$$

The sum in the second part of (29) takes into account that more than one components s ∈ Δj may depend on the parameter $\underline{\theta j}$.

It is important to note that the assembled matrices $\widehat{K}_0^{CB}$ and $\widehat{K}_j^{CB}$ of the Craig–Bampton reduced system in the expansion (28) are independent of the values of $\underline{\theta}$. In order to save computational time, these constant matrices are computed and assembled once and, therefore, there is no need this computation to be repeated during the iterations involved in optimization or TMCMC sampling algorithms for model updating due to the changes in the values of the parameter vector $\underline{\theta}$.

This aforementioned procedure results in substantial computational savings since it avoids

(a) re-computing the fixed-interface and constrained modes for each component,

(b) and assembling the reduced matrices from these components.

The formulation guarantees that the reduced system is based on the exact component modes for all values of the model parameters.

Special attention should be given when the size of the reduced mass and stiffness matrices are dominated by a large number of interface DOFs. In this case, the coordinate transformation (16) can be used to further reduce the number of interface DOFs for one or more interfaces. Using (21), it is clear that the stiffness matrix of the eigenvalue problem involved in (20) depends on the parameters associated with the

components that connect to the interface *l*. The variability of these parameters affects the characteristic interface modes $V^{(l)}$ which are functions of these parameters.

Exact estimates of the characteristic interface modes in iterative or TMCMC sampling algorithms can only be obtained by repeatedly solving (20) for each different value of the respective parameters. For large number of DOF at the interface, such re-analysis at the interface level may increase substantially the computational demands. Interpolation schemes can be used to approximate the characteristic interface modes at the interface level in terms of the characteristic interface modes at a number of support points in a significantly reduced space of model parameters associated with the components that connect to the interface *l*.

The choice of constant $V^{(l)}$ is critical in order to get accurate results with the least number of characteristic interface modes over the region of variation of the model parameters associated with the interface *l*. In FE model updating, the $V^{(l)}$ can be chosen as the eigenvectors of the lowest modes of the eigenvalue problem (20) corresponding to a reference model of the structure, avoiding the computational cost involved with the repetitive solution of (20) at each iteration or TMCMC sample. This, however, may deteriorate the accuracy of the predictions for large variations of the model parameters. To improve convergence and maintain the accuracy of the final optimal estimate in iterative optimization algorithms, the reduced basis forming $V^{(l)}$ can be updated every few iterations. Also, to maintain higher level of accuracy in the TMCMC sampling algorithm, the reduced basis forming $V^{(l)}$ can be kept constant within a TMCMC stage, with this basis selected to correspond to the most probable model predicted from the previous TMCMC stage. Such technique is expected to give sufficiently accurate results for the final TMCMC stage.

It should be pointed out that the significant savings arising partly from the reduction of the size of the eigenvalue problem from n to n_r in the proposed model reduction technique and partly from the fact that the estimation of the component fixed-interface modes and the characteristic interface modes need not to be repeated for each iteration involved in the optimization or TMCMC sampling algorithms.

Attention should also be paid on the optimal number of components that should be used to represent a substructure with stiffness that depends linearly on a single parameter. More components within such substructure introduce extra interface DOFs or characteristic interface modes which increase the size and affect the sparsity structure of the reduced matrices $\widehat{K}$ and $\widehat{M}$.

The total size of the reduced matrices is also affected by the number of the fixed interface modes for all components introduced for the substructure. From the computational point of view, the optimal choice of components for such a substructure would be to select the number of components and the optimal spatial division which will result in a reduced system that requires the least computational time for analysis. However, as the number of interface DOFs or characteristic interface modes increases by the introduction of more components per substructure, it is unlikely that the resulting increase in the size of the reduced matrices be effectively compensated by a decrease in the total number of fixed interface modes arising from the multiple components that represent the single substructure. Thus, in case where detailed optimal component selection studies are not available, the wisest choice is to select a single component per substructure.

As a final note, it is worth mentioning the treatment of a component in the CMS process for the general case for which the component stiffness and mass matrices depends on two or more parameters in the vector $\underline{\theta}$. In these cases, in order to obtain exact estimates of the component modes, the solution of the eigenvalue problems for such a component is not avoided. The fixed-interface and characteristic interface modes have to be recomputed in each iteration or TMCMC sample involved in the model updating procedure and used to form the reduced stiffness and mass matrices of the components. This repeated computation, however, is usually confined to a small number of components. Interpolation schemes can also be adopted to avoid re-analysis at the component or interface level. However, it should be pointed out that the use of interpolating schemes for approximating the fixed interface and the characteristic interface modes is an open issue and further analyses are required to evaluate the effectiveness of such techniques in the general case.

3.4 Conclusions regarding the CMS technique

Iterative optimization algorithms and stochastic simulation algorithms involved in both deterministic and Bayesian FE model updating formulations require a moderate to large number of FE model re-analyses. For large size FE models with hundreds of thousands or even million DOFs, the computational demands may be excessive. Exploiting certain stiffness-related parameterization schemes, often encountered in FE model updating formulations, to guide the division of the structure into components, results in exact linear representations of the Craig‑Bampton reduced stiffness matrix as a function of

the model parameters with coefficient matrices computed and assembled once from a single CMS analysis of a reference structure. Further significant reductions in the size of the reduced system are shown to be possible using characteristic interface modes estimated for each interface between components. Re-analyses required in FE model updating formulations are associated with the solution of the eigenproblem of the reduced-order system, completely avoiding the re-analyses of the component fixed-interface and characteristic interface modes as well as the re-assembling of the reduced system matrices.

4 Static Correction expansion

Expanding the aforementioned techniques known as substructure coupling for dynamic analysis, an improved type of reduction method is proposed below. Substructure coupling involves dividing the structure into a number of substructures obtaining reduced-order models of the substructures and then assembling a reduced order model for the entire structure. More specifically, after the division of the structure into substructures the model reduction technique involves two basic steps: definition of sets of substructure modes; and coupling the substructure-modes models to form a reduced-order model. Regarding the set of substructure modes, fixed-interface normal modes and interface constraint modes are used in the present formulation [21].

The standard approach for constructing reduced-order models is based on the consideration of the entire set of interface modes and only a small number of sub-structural fixed-interface normal modes for each substructure. If the residual or higher order fixed-interface normal modes are retained in the analysis the accuracy of the reduced-order models is expected to improve. In fact, the contribution of higher modes implies that reduced-order models are more precisely constructed and therefore their accuracy is enhanced. This idea has been explored in the past for estimating relative eigenvalue errors [22], and recently for improving the accuracy of eigenvalue estimation [23, 24]. A number of considerations and approximations were assumed in order to include the effect of residual sub-structural modes. The results reported in the previous references show that the accuracy of reduced-order models is significantly improved.

In the present formulation the contribution of residual sub-structural modes is based on physical grounds and it is derived by considering the so-called static solution or static correction. Such approximation is derived from the fact that high frequency modes react essentially in a static manner when excited by low frequencies. This feature is then combined with a particular finite element model parametrization scheme. When the division of the structure into substructures is guided by such parametrization scheme dramatic computational savings are achieved. Such drastic reduction in computational effort is obtained without compromising the accuracy of the results.

4.1 Reduced order model: original formulation (brief synopsis of the reduction methodology analyzed earlier)

As previously pointed out, the objective of model reduction techniques is to characterize the dynamic behavior of the system by a reduced number of generalized coordinates.

In the standard formulation of component mode synthesis, the dynamics of the system is described by a number of generalized coordinates which includes a fraction of the fixed-interface modal coordinates of each substructure and the physical interface coordinates. The derivation of the corresponding reduced-order model is presented in this section.

The fixed-interface normal modes and interface constraint modes are used to define the reduced transformation matrix T_{cb} (Craig–Bampton transformation matrix) that relates

the displacement vector of physical coordinates of all substructures u(t) with a set of generalized coordinates v(t). Such transformation is given by [21, 25]

$$\overline{u}(t) = T_{cb}\, v(t) \qquad (30)$$

where

$$\overline{u}(t) = \begin{Bmatrix} u_i(t) \\ u_b(t) \end{Bmatrix} \qquad v(t) = \begin{Bmatrix} v_k(t) \\ u_b(t) \end{Bmatrix} \qquad (31)$$

$$T_{cb} = \begin{bmatrix} [\Phi_{ik}^1, \dots, \Phi_{ik}^{Ns}] & [\Psi_{ib}^1, \dots, \Psi_{ib}^{Ns}]\tilde{T} \\ 0 & I \end{bmatrix}$$

* for the purpose of briefly re-examining the current method **expansion $\boldsymbol{T_{cb}}$ is in a way the equivalent of the $\boldsymbol{\Psi^{(s)}}$** matrix mentioned in Formulation using fixed-interface modes. **However it should be pointed out that $\boldsymbol{T_{cb}}$ has a different formulation than $\boldsymbol{\Psi^{(s)}}$, which involves the matrix $\boldsymbol{\tilde{T}}$.**

$\tilde{T}$ is a transformation matrix, consisting of zeros and ones, that maps the vector $u_b(t)$ to the vector $u_{bl}(t)^T = (u_{bl}^1(t)^T, \dots, u_{bl}^{Ns}(t)^T) \in R^{n_{bl}}$, $n_{bt} = \sum_{s=1}^{Ns} n_b^s$ of interface coordinates of all substructures. The particular structure of the transformation matrix $\tilde{T}$ depends on the definition of the independent interface coordinates $v(t)$.

$[\cdot, \dots, \cdot]$ indicates a block diagonal matrix having as diagonal blocks the matrices inside the square bracket.

The assembled mass matrix $\widehat{M} \in R^{n_v\, x\, n_v}$ and the stiffness matrix $\widehat{K} \in R^{n_v\, x\, n_v}$ ($n_v = n_k + n_b$) for the independent reduced set v(t) of generalized coordinates take the form:

$$\widehat{M} = T_{cb}^T M_g T_{cb} \qquad \text{and} \qquad \widehat{K} = T_{cb}^T K_g T_{cb} \qquad (32)$$

where M_g and K_g are the mass and stiffness matrices of the unreduced model referred to the vector of physical coordinates of all substructures u(t). They are defined in terms of the substructures mass and stiffness matrices as:

$$M_g = \begin{bmatrix} [M_{ii}^1, \dots, M_{ii}^{Ns}] & [M_{ib}^1, \dots, M_{ib}^{Ns}]\tilde{T} \\ \tilde{T}^T \left[M_{ib}^{1^T}, \dots, M_{ib}^{Ns^T}\right] & \tilde{T}^T [M_{bb}^1, \dots, M_{bb}^{Ns}]\tilde{T} \end{bmatrix}$$

$$K_g = \begin{bmatrix} [K_{ii}^1, \dots, K_{ii}^{Ns}] & [K_{ib}^1, \dots, K_{ib}^{Ns}]\tilde{T} \\ \tilde{T}^T \left[K_{ib}^{1^T}, \dots, K_{ib}^{Ns^T}\right] & \tilde{T}^T [K_{bb}^1, \dots, K_{bb}^{Ns}]\tilde{T} \end{bmatrix} \qquad (33)$$

from where it is easy to show that the reduced mass and stiffness matrices are given by:

$$\widehat{M} = \begin{bmatrix} I & \left[\widehat{M}_{ib}^1, \dots, \widehat{M}_{ib}^{Ns}\right]\tilde{T} \\ \tilde{T}^T \left[\widehat{M}_{ib}^{1^T}, \dots, \widehat{M}_{ib}^{Ns^T}\right] & \tilde{T}^T \left[\widehat{M}_{bb}^1, \dots, \widehat{M}_{bb}^{Ns}\right]\tilde{T} \end{bmatrix}$$

(34)

$$\widehat{K} = \begin{bmatrix} [\Lambda_{kk}^{1}, \dots, \Lambda_{kk}^{Ns}] & 0 \\ 0 & \tilde{T}^{T}[\widehat{K}_{bb}^{1}, \dots, \widehat{K}_{bb}^{Ns}]\tilde{T} \end{bmatrix}$$

With

$$\widehat{M}_{ib}^{s} = \Phi_{ik}^{s^T} M_{ii}^{s} \Psi_{ib}^{s} + \Phi_{ik}^{s^T} M_{ib}^{s} \qquad \widehat{K}_{bb}^{s} = K_{ib}^{s^T} \Psi_{ib}^{s} + K_{bb}^{s} \tag{35}$$

$$\widehat{M}_{bb}^{s} = \left(\Psi_{ib}^{s^T} M_{ii}^{s} + M_{ib}^{s^T}\right) \Psi_{ib}^{s} + \Psi_{ib}^{s^T} M_{ib}^{s} + M_{bb}^{s}, \quad s = 1, \dots, Ns$$

4.2 Reduced-order model: improved formulation

According to the transformation matrix T_{cb} the vector of physical coordinates at the internal degrees of freedom of all substructures is approximated as:

$$u_i(t) = [\Phi_{ik}^{1}, \dots, \Phi_{ik}^{Ns}] v_k(t) + [\Psi_{ib}^{1}, \dots, \Psi_{ib}^{Ns}] \tilde{T} u_b(t) \tag{36}$$

In order to consider the contribution of the residual modes it is first noted that the undamped equation of motion of the structural model referred to the set of generalized coordinates v(t) (reduced-order model) is given by:

$$\widehat{M} \begin{Bmatrix} \ddot{v}_k(t) \\ \ddot{u}_b(t) \end{Bmatrix} + \widehat{K} \begin{Bmatrix} v_k(t) \\ u_b(t) \end{Bmatrix} = 0 \tag{37}$$

from where the first block of this equation reads:

$$\ddot{v}_k(t) + [\Lambda_{kk}^1, \dots, \Lambda_{kk}^{Ns}] v_k(t) = -[\widehat{M}_{ib}^1, \dots, \widehat{M}_{ib}^{Ns}] \tilde{T} \ddot{u}_b(t) \qquad (38)$$

By considering the definition of $\widehat{M}_{ib}^s$, $s = 1, \dots, Ns$ given in (35), and the definition of the interface constraint modes matrix $\Psi_{ib,}^s$ $s = 1, \dots, Ns$, the previous equation can be rewritten as:

$$\ddot{v}_k(t) + [\Lambda_{kk}^1, \dots, \Lambda_{kk}^{Ns}] v_k(t) = -\left[\widehat{\Phi}_{ik}^1, \dots, \widehat{\Phi}_{ik}^{Ns^T}\right] M^* \tilde{T} \ddot{u}_b(t) \qquad (39)$$

Where

$$M^* = [M_{ib}^1 - M_{ii}^1 K_{ii}^{1^{-1}} K_{ib}^1, \dots, M_{ib}^{Ns} - M_{ii}^{Ns} K_{ii}^{Ns^{-1}} K_{ib}^1] \qquad (40)$$

The contribution of the residual fixed-interface normal modes to the response of the physical coordinates at the internal degrees of freedom of all substructures $u_i(t)$ is approximated by using the static solution or static correction. This approximation is reasonable due to the fact that high frequency modes react essentially in a static manner when excited by low frequencies [26]. Then, it can be shown that equation (36) becomes [27, 28]:

$$u_i(t) = [\Phi_{ik}^1, \dots, \Phi_{ik}^{Ns}]v_k(t) + [\Psi_{ib}^1, \dots, \Psi_{ib}^{Ns}]\tilde{T}u_b(t) - K^* M^* \tilde{T}\ddot{u}_b(t) \qquad (41)$$

where the term $K^* M^* \tilde{T}\ddot{u}_b(t)$ represents the residual modes contribution (static correction), and $K^*$is the static residual flexibility matrix given by [28]:

$$K^* = [K_{ii}^{1^{-1}} - \Phi_{ik}^1 \Lambda_{kk}^{1^{-1}} \Phi_{ik}^{1^T}, \dots, K_{ii}^{Ns^{-1}} - \Phi_{ik}^{Ns} \Lambda_{kk}^{Ns^{-1}} \Phi_{ik}^{Ns^T}] \qquad (42)$$

Using the previous approximation for $u_i(t)$ the vector of physical coordinates of all substructures can be expressed as:

$$\overline{u}(t) = \begin{Bmatrix} u_i(t) \\ u_b(t) \end{Bmatrix} = T_{cb} \begin{Bmatrix} v_k(t) \\ u_b(t) \end{Bmatrix} + \begin{bmatrix} 0 & -K^* M^* \tilde{T} \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{v}_k(t) \\ \ddot{u}_b(t) \end{Bmatrix} \qquad (43)$$

It is also seen that:

$$\begin{Bmatrix} \ddot{v}_k(t) \\ \ddot{u}_b(t) \end{Bmatrix} = -\widehat{M}^{-1}\widehat{K} \begin{Bmatrix} v_k(t) \\ u_b(t) \end{Bmatrix} \qquad (44)$$

Hence, $\bar{u}(t)$ can be represented by:

$$\bar{u}(t) = \begin{Bmatrix} u_i(t) \\ u_b(t) \end{Bmatrix} = T_{ecb} \begin{Bmatrix} v_k(t) \\ u_b(t) \end{Bmatrix} \qquad (45)$$

where T_{ecb} is the improved transformation matrix which takes the form, $T_{ecb} = T_{cb} + T_e$ with:

$$T_e = \begin{bmatrix} 0 & K^* M^* \tilde{T} \\ 0 & 0 \end{bmatrix} \widehat{M}^{-1} \widehat{K} \qquad (46)$$

Finally, by considering the definition of $\widehat{M}^{-1}$ and $\widehat{K}$, and by performing the product between the 2 by 2 partitioned matrix in (46) and the matrix $\widehat{M}^{-1}\widehat{K}$, it can be demonstrated that the transformation matrix T_e can be written as:

$$T_e = \begin{bmatrix} -K^* M^* \tilde{T} (\tilde{T}^T M^*_{bib} \tilde{T})^{-1} \tilde{T}^T \widehat{M}^T_{ib} \Lambda & K^* M^* \tilde{T} (\tilde{T}^T M^*_{bib} \tilde{T})^{-1} \tilde{T}^T \widehat{K}_{bb} \tilde{T} \\ 0 & 0 \end{bmatrix} \qquad (47)$$

Where

$$M^*_{bib} = [\widehat{M}^1_{bb} - \widehat{M}^{1^T}_{ib} \widehat{M}^1_{ib}, \dots, \widehat{M}^{Ns}_{bb} - \widehat{M}^{Ns^T}_{ib} \widehat{M}^{Ns}_{ib}$$

$$\widehat{M}_{ib} = [\widehat{M}^1_{ib}, \dots, \widehat{M}^{Ns}_{ib}] \qquad (48)$$

$$\Lambda = [\Lambda^1_{kk}, \dots \Lambda^{Ns}_{kk}]$$

$$\widehat{K}_{bb} = [\widehat{K}_{bb}^{1}, \dots, \widehat{K}_{bb}^{Ns}]$$

Using the redefined transformation matrix T_{ecb} instead of T_{cb} the improved reduced-order mass matrix $\widehat{M}_e \in R^{n_v \, x \, n_v}$ and the stiffness matrix $\widehat{K}_e \in R^{n_v \, x \, n_v}$ take the form

$$\widehat{M}_e = T_{ecb}^T M_g T_{ecb} = \widehat{M} + T_{cb}^T M_g T_e + T_e^T M_g T_{cb} + T_e^T M_g T_e$$

$$\widehat{K}_e = T_{ecb}^T K_g T_{ecb} = \widehat{K} + T_{cb}^T K_g T_e + T_e^T K_g T_{cb} + T_e^T K_g T_e \tag{49}$$

Because of the contribution of the residual fixed-interface normal modes in the transformation matrix T_{ecb}, it is expected that the reduced-order mass and stiffness matrices are more precisely constructed than the original reduced order matrices. So, it is anticipated that the improved formulation needs reduced-order models of much smaller dimension than the original formulation in order to obtain responses of similar accuracy. It is interesting to note that all the basic matrices involved in the definition of T_e are already available from the original formulation of the model reduction technique.

Considering the parametrization of the matrices M_g, K_g, T_{cb} (not presented here for simplicity in notation) and the parametrization of the different matrices involved in the definition of T_e (47), that is, K^*, M^*, M_{bib}^*, $\widehat{M}_{ib}$, Λ and $\widehat{K}_{bb}$, it is clear that the improved reduced mass matrix $\widehat{M}_e$ and the improved stiffness matrix $\widehat{K}_e$ in (49) can

be expressed in terms of a set of matrices, which are independent of the values of the vector of system parameters θ, and the parametrization functions $h_j(\theta_j)$ and $g_j(\theta_j)$. The set of matrices are computed and assembled once for a reference model. This feature results in substantial computational savings since it avoids the re-assembling of the various matrices involved in the definition of the reduced-order matrices for different values of the vector of system parameters θ. It is interesting to note that for problems in which the mass matrix is constant independent of θ, the block diagonal matrix M_{bib}^* is also independent of θ. Thus, the matrix $\tilde{T}^T M_{bib}^* \tilde{T}$ that appears in the definition of the transformation matrix T_e (47) needs to be inverted once.

It is noted that the independent physical coordinates $u(t)$ of the original unreduced structural model can be written directly in terms of the set of generalized coordinates $v(t)$ as:

$$u(t) = \overline{T} T_{ecb} v(t) \qquad (50)$$

Where $\overline{T} \in R^{n\,x\,n_{\overline{u}}}$ is a constant matrix that maps the vector of physical coordinates of all substructures $\overline{u}(t)$ to $u(t)$, and T_{ecb} is the enhanced transformation matrix. Based on this transformation, the equation of motion of the reduced-order system can be written as:

$$\hat{M}_e \ddot{v}(t) + \hat{C}_e \dot{v}(t) + \hat{K}_e v(t) = T_{ecb}^T \overline{T}^T f(t) \qquad (51)$$

where $\hat{C}_e \in R^{n_v \, x \, n_v}$ is the assembled damping matrix for the independent reduced set v(t) which can be defined in terms of the damping matrix C_g, of the unreduced model referred to the vector $\overline{u}(t)$. The assembled damping matrix has a similar structure as

$\widehat{M}_e$ and $\widehat{K}_e$ (49). It is noted that the dimension of the matrices involved in the equation of motion of the reduced-order model can be substantially smaller (one or more orders of magnitude) than the dimension of the unreduced matrices, e.g. $n_v \ll$ n. The equation of motion (51) can be integrated efficiently by an appropriate step-by-step integration scheme [27] or by modal analysis [29, 30, 31].

Finally, it is stressed that all matrices involved in the enhanced formulation of the model reduction technique are computed and assembled once before the corresponding complex-simulation based problem is solved. In other words, the characterization of those matrices corresponds to off-line calculations. This in turn implies a drastic reduction in computational efforts.

4.3 Conclusions regarding the enhanced formulation

An enhanced model reduction technique for dealing with a class of complex simulation-based problems involving medium/large finite element models has been presented. The analyses are performed in a reduced space of generalized coordinates. Specifically, a method based on substructure coupling is considered in the present implementation. Usually a small number of dominant fixed-interface normal modes is used to define reduced-order models. In this regard, the contribution of the residual fixed-interface normal modes is considered explicitly in the present formulation.

This contribution corresponds to the so-called static solution or static correction of the higher order modes. The model reduction technique is then combined with a particular substructure parametrization scheme. In such scheme all matrices involved in the characterization of the reduced-order model can be expressed in terms of a set matrices independent of the values of the system parameters and a set of parametrization functions. Thus, the different substructure and reduced-order matrices are computed and assembled once for a reference model. Results show that the enhanced formulation of the model reduction technique requires models of dimension significantly smaller than of the model of the original formulation in order to get reduced-order models of the same level of accuracy. Thus, the accuracy of the reduced-order model is dramatically improved by the enhanced formulation. The corresponding computational cost of the enhanced formulation is a fraction of the cost involved in the original formulation for similar levels of precision. Additionally, the use of the enhanced formulation shows that the computational effort involved in complex simulation-based problems is decreased drastically by two or more orders of magnitude with respect to the full finite element model. Furthermore, the drastic reduction in computational efforts is achieved without compromising the accuracy of the results. Future research directions aim at increasing even further the computational savings by considering for example not only generalized coordinates in terms of the fixed-interface normal modes but in terms of interface constraint modes as well. Finally, the extension of the proposed enhanced model reduction technique to a class of nonlinear models and the application of the technique to a more variety of complex simulation-based problems are additional topics for future research.

5 Observation of Results

In order to evaluate the credibility fluctuation and accuracy of the aforementioned model reduction techniques some basic tests were run. Those tests provided valuable information on how this methods operate on different conditions and showcased the efficiency of those methods in each condition.

5.1 Original Formulation

Below are demonstrated observations regarding the results of the original formulation of Model Reduction technique without considering the enhanced formulation of static correction.

- It is imperative for the model to be as well-defined as possible. That involves the proper definition and selection of the material of each component of the structure whether the model is created in Comsol or imported from another software. Defining a material also includes the definition of certain structural parameters such as the young modulus **E**, the Poisson ratio **ν**, the density of the material **ρ**, etc… It is also important to correctly define the domains, interfaces and fixed boundaries of each component and the relation of it with the other components of the structure. For example if two or more components are part of the same group it is vital to choose the definition of Group Union and select those components accordingly. Furthermore, attention should be paid to the definition of fixed boundaries both in the Definitions section and the Physics section (Solid Mechanics) where boundaries are concerned. That needs to happen because especially in Solid Mechanics all boundaries are by default free.

Free boundaries affect the accuracy of the the results in Eigenfrequency Study, providing in the form of computational noise, complex numbers for the first 6-10 eigenfrequencies.

- The Eigenfrequency values given in Comsol are affected at a certain extent by the size of the mesh chosen. For each mesh selection a new Study must be run and the MatLab code must be updated by closing the program and reopen it from the start.

- When Eigenfrequencies are arithmetically quite large, usually even larger than 1 , the divergence between the values given through Comsol and the ones given through MatLab is quite noticeable, especially for smaller structures.

- Structures of small size have higher Eigenfrequencies, as it is expected. Hence the Model Reduction Method in MatLab will not be able to provide extremely accurate results.

- The error resulting from the difference between the eigenfrequencies in Comsol and in Matlab does not follow a certain rate of divergence or convergence. Therefore its rate of growth depending on variables like the size and structural difference of each model, cannot extremely credibly be predicted.

- In order for the model reduction technique to retain a certain number of modes to proceed with the model reduction, eigenfrequencies are compared to a

predefined cutoff frequency. The modes that are eventually kept are the ones that correspond to smaller frequencies than the cutoff frequency. More kept modes provide more accurate results. The number of those modes depends on the comparison between Eigenfrequencies and the cutoff frequency.

E.g.: MatLab code (component reduction):

Angular frequency: vc

Frequency derived from the solution of the eigenfrequency problem: D3

Frequency compared with eigenfrequency: par = PAR2*vc

Kept mode if: D3 ≤ par

Cutoff frequency = the eigenfrequency D3 that is bigger than par

All the previous modes until this value of frequency are kept.

The interface modes are kept accordingly.

Specifically if the parameter vc is kept the same, whereas the parameter PAR2 changes, the accuracy of results changes accordingly.

If PAR2 increases→par does too→more modes kept→ more accurate results.

If PAR2 decreases→ par does too→less modes are kept→ less accurate results.

To examine the efficiency of the model reduction method and the correctness of the above observations, a couple of different case scenarios are presented. Firstly, the method is applied to a rudimentary 4-component model, presented below, created in Comsol. Secondly it is applied to a 22-component, 16-group model of a Bridge for a point of reference as far as the accuracy of the method for a fully defined large structure is concerned. Some indicative results are cited below:

Four Components “Beam”

1st Component: 1st block w=500m d=400m h=500m

2nd Component: 2nd block w=700m d=400m h=500m

3rd Component: 3rd block w=800m d=400m h=500m

4th Component: 4th block w=300m d=400m h=500m

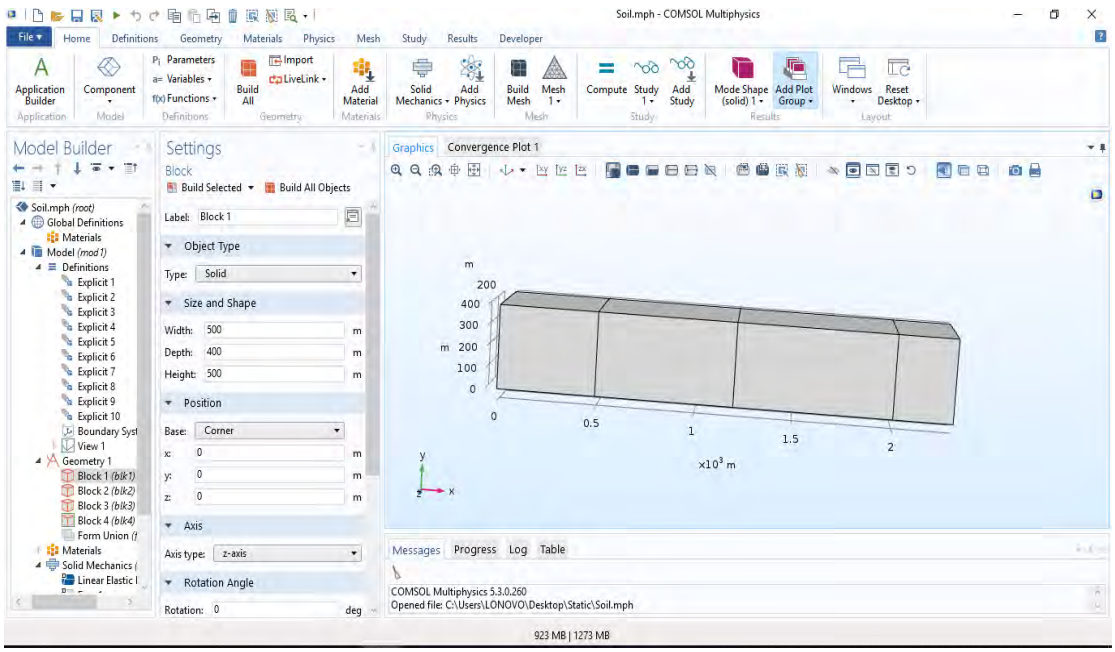
3 interfaces

Elastic Material: Cast iron

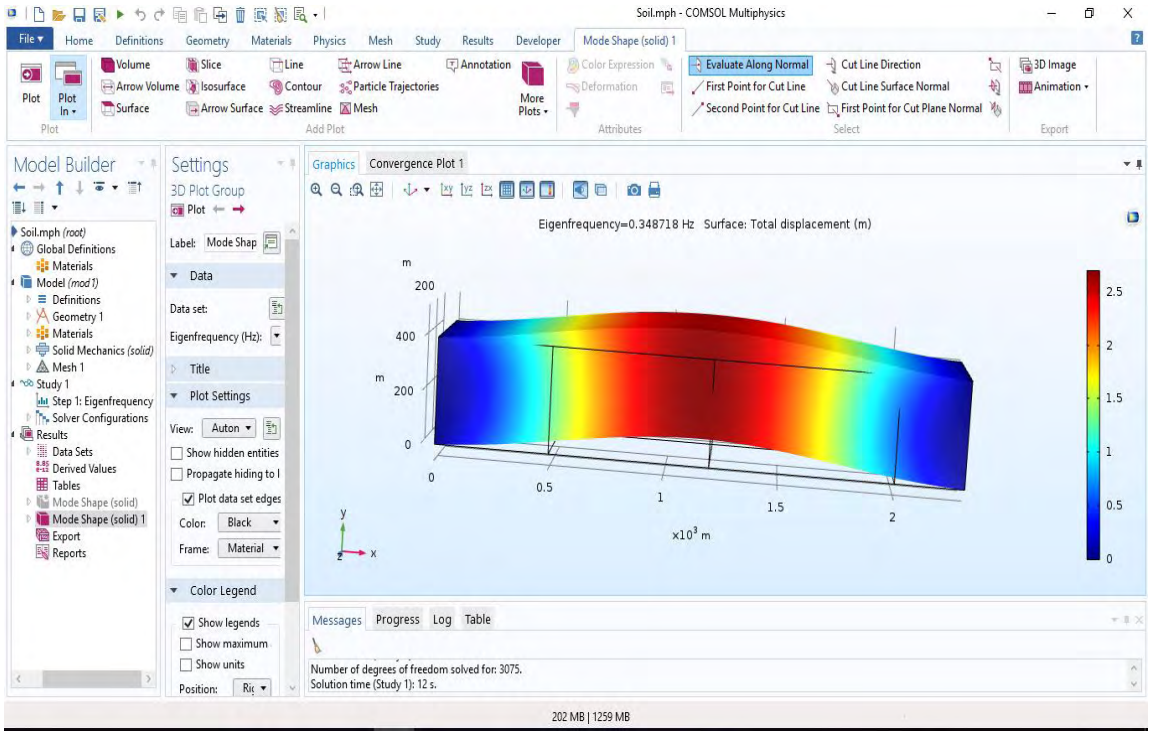
Fixed boundaries: The first most-left and the last most-right

(the constant parameters such as E, ρ, ν are defined through the material, so no separate inputs and definitions are required)

Geometry of the beam



The picture below depicts the deformation of the structure measured by its spacial displacement after the application of the **Eigenfrequency Study**. Essentially this study performs a Finite Element Analysis returning the eigenfrequencies of the structure and the displacements generated by the activation of each eigenfrequency.



1st Case

Extra Coarse mesh

- DOFs=387 (max k=27)
- Vc = 5 PAR2 = [2 2 2 2] → par = 10

- Stiff_final = Mass_final = 146x146 (size)

10 indicative Eigenfrequencies chosen from Comsol & MatLab

COMSOL	MATLAB	Error1
1,57036	1,5966	0,016709544
2,28784	2,295	0,003129589
3,40608	3,3773	0,008449596
4,11584	4,1127	0,000762906
4,54977	4,5362	0,002982568
4,7473	4,7338	0,002843722
4,95911	4,9579	0,000243995
5,01639	5,0062	0,002031341
5,25408	5,288	0,006455935
5,36804	5,3588	0,001721299

Sum Error1	0,045330497
Mean Error1	0,00453305

2nd Case

Extra Coarse mesh

- DOFs=387 (max k=27)
- Vc = 5 PAR2 = [1 1 1 1] → par = 5

- Stiff_final = Mass_final = 99x99 (size)

10 indicative Eigenfrequencies chosen from Comsol & MatLab

COMSOL	MATLAB	Error2
1,57036	1,5968	0,016836904
2,28784	2,2961	0,003610392
3,40608	3,4278	0,006376832
4,11584	4,1221	0,001520953
4,54977	4,6723	0,026931032
4,7473	4,7916	0,00933162

4,95911	4,8564	0,020711378
5,01639	5,0088	0,00151304
5,25408	5,2799	0,004914276
5,36804	5,303	0,012116154

Sum Error2	0,103862581
Mean Error2	0,010386258

3rd Case

Extra Coarse mesh

- DOFs=387 (max k=27)
- Vc = 3 PAR2 = [1 1 1 1] → par = 3

- Stiff_final = Mass_final = 91x91 (size)

10 indicative Eigenfrequencies chosen from Comsol & MatLab

COMSOL	MATLAB	Error3
1,57036	1,5972	0,017091622
2,28784	2,3061	0,007981327
3,40608	3,3684	0,01106257
4,11584	4,0412	0,018134816
4,54977	4,2536	0,065095598
4,7473	4,4372	0,065321341
4,95911	4,9765	0,003506678
5,01639	5,1668	0,029983713
5,25408	5,3123	0,011080912
5,36804	5,6585	0,054109135

Sum Error3	0,283367713
Mean Error3	0,028336771

4th Case

Normal mesh

- DOFs=3075
- Vc = 5 PAR2 = [2 2 2 2] → par = 10

- Stiff_final = Mass_final = 416x416 (size)

10 indicative Eigenfrequencies chosen from Comsol & MatLab

COMSOL	MATLAB	Error4
2,02249	2,0336	0,005493229
2,818	2,8152	0,000993612
3,5758	3,5947	0,005285531
4,07388	4,1039	0,007368896
4,43965	4,46164	0,004953093
4,5596	4,559	0,00013159
4,57296	4,6164	0,009499318
4,65657	4,6421	0,003107437
5,02254	5,0287	0,001226471
5,11157	5,1008	0,002106985

Sum Error4	0,040166163
Mean Error4	0,004016616

5th Case

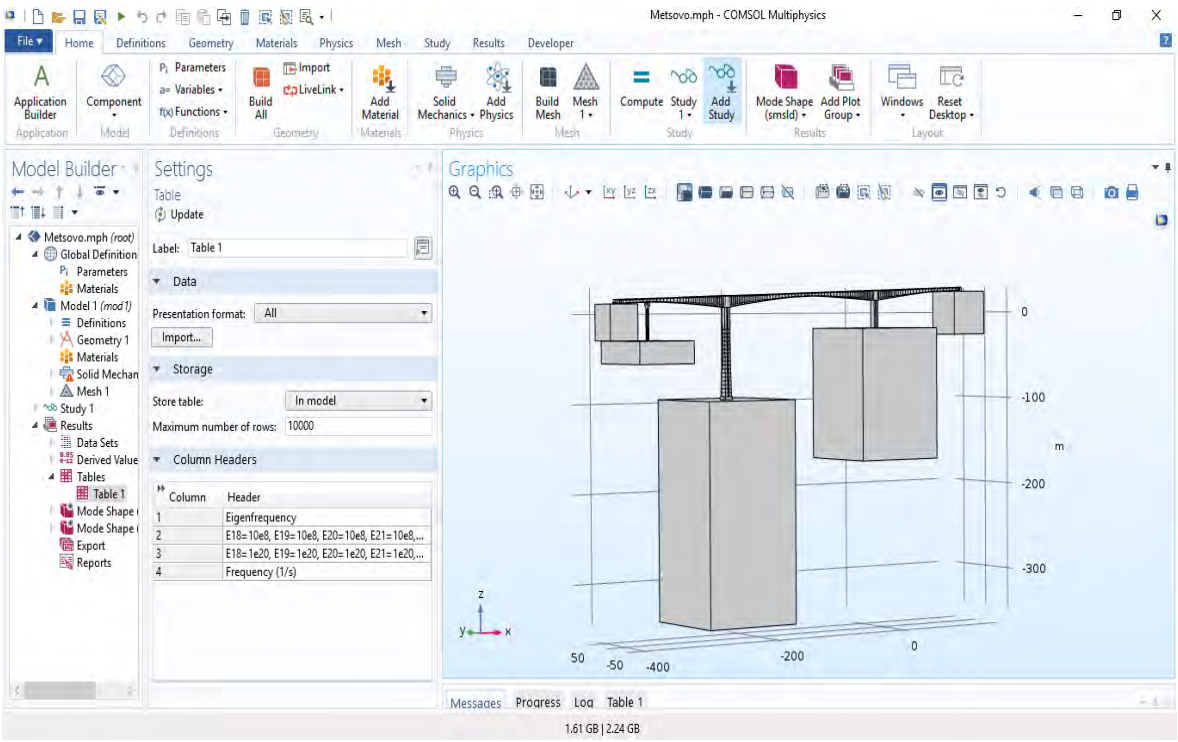
Metsovo Bridge

Soil nominal values 1.10^9 Pa

Deck nominal values 37.10^9 Pa

Pier nominal values 34.10^9 Pa

Geometry of the Metsovo Bridge



20 indicative Eigenfrequencies chosen from Comsol & MatLab

COMSOL	MATLAB	Error5
0,29229	0,2923	3,42126E-05
0,57285	0,5729	8,72829E-05
0,61889	0,6189	1,6158E-05
0,84749	0,8475	1,17995E-05
1,04847	1,0485	2,86131E-05
1,06912	1,0692	7,48279E-05
1,38448	1,3848	0,000231134
1,57685	1,5771	0,000158544
1,68234	1,6828	0,000273429
1,96412	1,9647	0,000295298
2,15487	2,1553	0,000199548
2,31451	2,315	0,000211708
2,49529	2,4994	0,001647103
2,74198	2,7429	0,000335524

2,81053	2,8141	0,001270223
2,87191	2,8753	0,001180399
2,94499	2,9463	0,000444823
3,31186	3,318	0,001853943
3,37427	3,38	0,001698145
3,5128	3,5205	0,002191984

Sum Error5	0,012244698
Mean Error5	0,000612235

5.2 Enhanced formulation

As a second stage, there was an attempt to perform the Static correction expansion to the original formulation, starting from the 4 components simplistic model. However, some problems surfaced during the process of generating results due to the different formulation that was needed to be done from the beginning. This is probably due to the fact that the model reduction technique without static correction was based on an initial formulation which even though it produces most of the required matrices for static correction, it follows a relatively altered methodology which uses differently constructed transformation matrices. Therefore when Static Correction with regeneration of the appropriate transformation matrices is implemented, certain problems appear, which at the current time frame they could not be solved.

5.3 General Conclusion

Exploiting certain stiffness-related parameterization schemes, often encountered in FE model updating formulations, to guide the division of the structure into components results inexact linear representations of the Craig–Bampton reduced stiffness matrix as a function of the model parameters with coefficient matrices computed and assembled once from a single CMS analysis of a reference structure. Further significant reductions in the size of the reduced system are shown to be possible using characteristic interface modes estimated for each interface between components. Re-analyses required in FE model updating formulations are associated with the solution of the eigenproblem of the reduced-order system, completely avoiding the re-analyses of the component Fixed-interface and characteristic interface modes as well as the re-assembling of the reduced system matrices. The results of the aforementioned methodology have been examined both on a rudimentary model as well as on a complex structure, providing the expected data as far as the effectiveness and the accuracy of the CMS method and the error fluctuations regarding Eigenfrequencies it generates in various case scenarios. Hence, valuable information were provided for future enhancements and/or expansions regarding the computational implementation of the model reduction techniques.

Finally, an enhanced model reduction technique for dealing with a class of complex simulation-based problems involving medium to large finite element models has been presented. The analyses are performed in a reduced space of generalized coordinates. Specifically, a method based on substructure coupling is considered in the present implementation. Usually a small number of dominant fixed-interface normal modes is used to define reduced-order models. In this regard, the contribution of the residual fixed-interface normal modes is considered explicitly in the present formulation. This

contribution corresponds to the so-called static solution or static correction of the higher order modes. The enhanced formulation of the model reduction technique requires models of dimension significantly smaller than of the model of the original formulation in order to get reduced-order models of the same level of accuracy. Thus, the accuracy of the reduced-order model is dramatically improved by the enhanced formulation. The corresponding computational cost of the enhanced formulation is a fraction of the cost involved in the original formulation for similar levels of precision. However, it should be noted that the enhance formulation in order to be successful, even though it utilizes the outputs of the original formulation, requires a different computational approach which may prove to be slightly challenging, especially for large structures.

APPENDIX

Computational application and exploitation of the Model reduction techniques

As far as the computational aspect is concerned, the above methods were implemented with the exact order of the mathematical relations in order to produce the reduction model. There was an effort to enhance an already existent code by making some commands in all files involved, simpler and/or faster with the goal of making the code more efficient and applicable on various different case scenarios which do not involve the same inputs. Moreover, part of the endeavor to make the code and generally the process of model reduction more comprehensive and obtainable for an average user, was to examine some of the basic capabilities of Comsol Multiphysics© and how its GUI interacts with a user with no former experience and with the need to perform the model reduction technique on a very simplistic structure. This process of creating a model, defining it, extracting the appropriate information from it and utilizing it through the MatLab code, are briefly analyzed below.

(Simple user's guide to handling from start to finish, the programs necessary for CMS analysis)

Create a structure in Comsol Multiphysics©, either by utilizing its toolkit or by importing an already existing structure designed by another program e.g. Solidworks, CAD, e.tc… Comsol is essentially a package software that is capable of either designing on its own, from the beginning a structure or simply importing it from another software, and analyze its structural and mechanical behavior under specified user defined conditions.

Analyzing in Comsol involves basically, running an embedded Finite element analysis and providing on its GUI and/or through MatLab the appropriate matrices (K, M), elements, eigenfrequencies and structural selections in order to continue with the model reduction process.

HOW TO CREATE A VERY SIMPLISTIC MODEL IN COMSOL in order to test the model reduction method

Open the Comsol application. Once opened select Model Wizard and choose the appropriate dimension for your study (3D, 2D,…,etc).

If 3D designing is chosen, go to the top toolbar and select **Geometry** in order to create the basic shape of your construct.(In the example the selection **Block** is used for the 4 components.)

After the blocks are designed, some Definitions must be made in order to examine accurately the model in the end. Those definitions are also vitas far as the import of data from Comsol to MatLab, through Livelink, are concerned.

- **Once your structure is loaded it is imperative to:**

1. Make sure to define the exact components of your structure.

2. Firstly change the Label and Name of the Structure from Component1 to Model.

3. Go to **Definitions** on the **Model builder bar** and add for each component the appropriate selection. The most common one as far as the definition of domains, interfaces and fixed selections is concerned is **Explicit** found in the Definitions top toolbar.

4. Each explicit selection corresponds to a predefined substructure. Every substructure is comprised of one or multiple **Components**, their **Interface boundaries** and their **Fixed boundaries** that need to be accurately defined, both in the Comsol model as well as in the Input Code mat. file. In case where detailed optimal component selection studies are not available, the wisest choice is to select a single component per substructure

5. You can change the name of the explicit selections to refer directly to the kind of selection that is made. However, **always keep the serial number that refers to the sequence of each selection** (e.g. Explicit **3**) because this is the number that is going to be needed in the input code in order to form the necessary matrices with the correspondent names.

6. **To define fully** a **Component** you need to select the exact **Domains** that is comprised of. **Interface selections** require **Boundaries**. **Fixed selections** require **Boundaries** as well.

7. **Add Material** to the structure if needed by selecting the appropriate command on the top toolbar. If there is no material selected it is important to choose one, otherwise the Study selection will present an error.

8. **Add Physics** to define the boundary conditions and the exact equations that describe accurately your study. (Solid Mechanics is commonly used selection with a wide range of implementation).

9. **Add Study** to specify the nature of the solution you expect after the examination of the structure. In the simple example presented, the study that has been chosen in order to test the accuracy of the reduction method is Eigenfrequency. (If the eigenfrequencies given by the Comsol model approximate the eigenfrequencies given by MatLab after reduction then the reduction method is accurate).

   !! Important Observation: If fixed boundaries are not defined to the Physics section (Solid Mechanics), apart from the Definitions section, then the provided Eigenfrequencies, especially the first ones will be given as small complex numbers due to the fact that all the boundaries of the structure are free.

The model is ready for any extra computation.

- **<u>Once an already fully defined Comsol model exists:</u>**

1. Open the Comsol application.

2. Open the Comsol with Matlab application (Livelink).

3. Once Matlab is fully loaded and synced with Comsol, check the directory of both the editor file and the command window that has been opened. It should be the directory of the file where all the archives concerning the structure are placed. Both mat. as well as mph. files.

4. Open and examine the Input Code.mat file in case you want to proceed in manual changes as far as a specific structure is concerned. Make sure the changes in the Matlab code correspond precisely to the model in Comsol. If you want to alternate your structure this must be done firstly in Comsol, by saving the changes, and subsequently in Matlab. After a certain change has been made, Comsol should be closed and MatLab Livelink should be launched anew to assure that the MatLab code is processing the newly altered model.

5. The only mat.file that requires user interaction is the Input file. Therefore no changes should be required in the other mat.files.

6. If the changes in the Input code are completed the Main code is ready to run in Matlab's editor. No further editing is needed in the Main code.

The Matlab results of the CMS analysis of the specific model are now saved with distinctive names in the same file as all the other documents.

➢ MatLab commands extracting vital information from the Comsol model

Input.m file: specific to each model	
model = mphload(fname); (line 73)	Loading the COMSOL model to obtain the information.
a = mphgetselection(model.selection(['sel' num2str(domain_selections(i))])); (line 85) The same command is again used to extract the interface_selections and fixed_selections of the Comsol model (In lines 95 and 130 respectively.)	Use the function mphgetselection to retrieve the model selection information: **str = mphgetselection(model.selection(<*seltag*>))** where **<*seltag*>** is the tag/number a selection node was defined as, in the model. Here the ***<seltag> is (['sel' num2str(domain_selections(i))])*** The output str is a MatLab structure with the following fields: • **dimension**, the space dimension of the geometry entity selected, • **geom**, the tag of the geometry node used in the selection, • **entities**, the list of the entity indexes listed in the selection, and • **isGlobal**, Boolean value to indicate if the selection is global or not.

MReduction.m file : general file executing the model reduction process. It is used independently of the Input file		
xmeshinfo = mphxmeshinfo(model); (line 67)	Use the function **info = mphxmeshinfo(model,...)** to extract extended mesh information from the active solution object. The extended mesh information provide information about the numbering of elements, nodes, and degrees of freedom (DOFs) in the extended mesh and in the matrices returned by **mphmatrix**.	
	FIELD	**DESCRIPTION**
	soltag	Tag of the solution object
	ndofs	Number of DOFs
nod1 = rmfield(nod1,'dofnames'); nod1.names = {'u' 'v' 'w'};	fieldnames	Names of the field variables
	fieldndofs	Number of DOFs per field name
elements1{1}.type='vtx'; elements1{2}.type='edg'; elements1{3}.type='tri'; elements1{4}.type='tet';	meshtypes	Types of mesh element
nod1.dofs=nod1.dofs+1;	dofs	Structure with information about the degrees of freedom
nod1 = xmeshinfo.nodes;	nodes	Structure with information about the nodes
elements2 = xmeshinfo.elements;	elements	Structure with information about each element type

[~,meshdata] = mphmeshstats(model); (line 93)	The function mphmeshstats also returns the mesh data such as element coordinates. Use the function with two output variables to get the mesh data. Enter: ***[meshstats,meshdata] = mphmeshstats(model)*** where meshdata is a MATLAB structure with the following fields:
	• **vertex** which contains the mesh vertex coordinates
el1{1}.elem=meshdata.elem{4}+1;	• **elem** which contains the element data information
el1{1}.dom=meshdata.elementity{4}';	• **elementity** which contains the element entity information for each element type.
matrices = mphmatrix(model,'sol1','Out',{'K' 'E'}, **'initmethod','init');** (line 215)	Extract the matrices of the COMSOL linearized system with the function **mphmatrix**. To call the function mphmatrix, specify a solver node and the list of the system matrices to extract: ***str = mphmatrix(model, <soltag>, 'out', out)*** where **<*soltag*>** is the solver node tag used to assemble the system matrices and ***out*** is a cell array containing the list of the matrices to evaluate. The output data str returned by mphmatrix is a MatLab structure, and the fields correspond to the assembled system matrices. Use the **initmethod** property as in this command: ***str = mphmatrix(model,<soltag>,'out',out,'initmethod', method)*** where ***method*** corresponds to the type of linearization point—the initial value expression **('init')** or a solution **('sol')**. Without the **'initmethod','init'** the above command returns error for too many zero elements. With the 'initmethod','init' it successfully returns the matrices.

The system matrices that can be extracted with mphmatrix :

EXPRESSION	DESCRIPTION
K	Stiffness matrix
L	Load vector
M	Constraint vector
N	Constraint Jacobian
D	Damping matrix
E	Mass matrix
NF	Constraint force Jacobian
NP	Optimization constraint Jacobian (*)
MP	Optimization constraint vector (*)
MLB	Lower bound constraint vector (*)
MUB	Upper bound constraint vector (*)
Kc	Eliminated stiffness matrix
Lc	Eliminated load vector
Dc	Eliminated damping matrix
Ec	Eliminated mass matrix
Null	Constraint null-space basis
Nullf	Constraint force null-space matrix
ud	Particular solution ud
uscale	Scale vector
(*) Requires the Optimization Module.	

Bibliographic References

[1] Yuen KV, Kuok SC. Bayesian methods for updating dynamic models. Appl Mech Rev 2011; 64(1).

[2] Mottershead JE, Friswell MI. Model updating in structural dynamics: a survey. J Sound Vib 1993; 167:347‑75.

[3] Marwala T. Finite element model updating using computational intelligence techniques: applications to structural dynamics. Springer; 2010.

[4] Moaveni B, He X, Conte JP, De Callafon RA. Damage identification of a composite beam using finite element model updating. Comput Aided Civil Infrastruct Eng 2008; 23(5):339‑59.

[5] Christodoulou K, Papadimitriou C. Structural identification based on optimally weighted modal residuals. Mech Syst Signal Process 2007; 21:4‑23.

[6] Katafygiotis LS, Papadimitriou C, Lam HF. A probabilistic approach to structural model updating. Int J Soil Dynam Earthquake Eng 1998; 17:495‑507.

[7] Fritzen CP, Jennewein D, Kiefer T. Damage detection based on model updating methods. Mech Syst Signal Process 1998; 12(1):163‑86.

[8] Yuen K-V, Beck JL, Katafygiotis LS. Efficient model updating and health monitoring methodology using incomplete modal data without mode matching. Struct Control Health Monit 2006; 13:91‑107.

[9] Haralampidis Y, Papadimitriou C, Pavlidou M. Multi-objective framework for structural model identification. Earthquake Eng Struct Dynam 2005;34(6):665‑85.

[10] Christodoulou K, Ntotsios E, Papadimitriou C, Panetsos P. Structural model updating and prediction variability using Pareto optimal models. Comput Methods Appl Mech Eng 2008; 198(1):138‑49.

[11] Das I, Dennis Jr JE. Normal-boundary intersection: a new method for generating the Pareto surface in nonlinear multi-criteria optimization problems. SIAM J Optim 1998; 8:631‑57.

[12] Beck JL, Katafygiotis LS. Updating models and their uncertainties ‑ I: Bayesian statistical framework. ASCE J Eng Mech 1998; 124(4):455‑61.

[13] Yuen K-V. Bayesian methods for structural dynamics and civil engineering. John Wiley & Sons; 2010.

[14] Beck JL, Yuen KV. Model selection using response measurements: Bayesian probabilistic approach. ASCE J Eng Mech 2004; 130(2):192‑203.

[15] Yuen KV. Recent developments of Bayesian model class selection and applications in civil engineering. Struct Safety 2010; 32(5):338‑46.

[16] Papadimitriou C, Beck JL, Katafygiotis LS. Updating robust reliability using structural test data. Probab Eng Mech 2001; 16:103‑13.

[17] Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. Equation of state calculations by fast computing machines. J Chem Phys 1953;21(6): 1087‑92.

[18] Ching J, Chen YC. Transitional Markov Chain Monte Carlo method for Bayesian updating, model class selection, and model averaging. ASCE J Eng Mech 2007; 133:816‑32.

[19] Haario H, Laine M, Mira A, Saksman E. DRAM: efficient adaptive MCMC. Stat Comput 2006; 16:339‑54.

[20] Beck JL, Au SK. Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation. ASCE J Eng Mech 2002; 128(4):380‑91.

[21] R.R Craig Jr., Structural Dynamics, An Introduction to Computer Methods, John Wiley & Sons, New York, 1981.

[22] K.C. Park, Y.H. Park, Partitioned component mode synthesis via flexibility approach, AIAA J. 42 (6) (2004) 1236‑1245.

[23] J.G. Kim, K.H. Lee, P.S. Lee, Estimating relative eigenvalue errors in the Craig-Bammpton method, Comput. Struct. 139 (2014) 54‑64.

[24] J.G. Kim, P.S. Lee, An enhanced Craig-Bampton method, Internat. J. Numer. Methods Engrg. 103 (2015) 79‑93.

[25] R.R Craig Jr., M.C.C. Bampton, Coupling of substructures for dynamic analysis, AIAA J. 6 (5) (1965) 678‑685.

[26] K.V. Yuen, S.C. Kuok, Bayesian methods for updating dynamic models, Appl. Mech. Rev. 64 (2011) 010802.

[27] K.J. Bathe, Finite Element Procedures, Prentice Hall, New Jersey, 2006.

[28] C.A. Schenk, H.J. Pradlwarter, G.I. Schüeller, Non-stationary response of large, non-linear finite element systems under stochastic loading,

Comput. Struct. 83 (2005) 1086‑1102. 15‑28.

[29] HA. Jensen, E. Millas, D. Kusanovic, C. Papadimitriou, Model-reduction techniques for Bayesian finite element model updating using

dynamic response data, Comput. Methods Appl. Mech. Engrg. 279 (2014) 301‑324.

[30] F. Tisseur, K. Meerbergen, The quadratic eigenvalue problem, SIAM Rev. 34 (2) (2001) 235‑286.

[31] F. Ma, M. Morzfeld, A general methodology for decoupling damped linear systems, Procedia Eng. 314 (2011) 2498‑2502.

[32] C. Papadimitriou, D.Ch. Papadioti, Component mode synthesis techniques for finite element model updating, Comput. Struct. 126 (2013) 15‑28.

[33] H.A. Jensen, A. Muñoz, C. Papadimitriou, C. Vergara, An enhanced substructure coupling technique for dynamic re-analyses: Application to simulation-based problems, Comput. Methods Appl. Mech. Engrg. 307 (2016) 215‑234

[34] C. Papadimitriou, Bayesian uncertainty quantification and propagation in structural dynamics simulations, in: 9th International Conference on Structural Dynamics. EURODYN2014, Porto, Portugal, 30 June‑2 July, 2014.