Institut für
Kontinuumsmechanik

Model Reduction in Nonlinear Finite Element Method for Engineering Structures

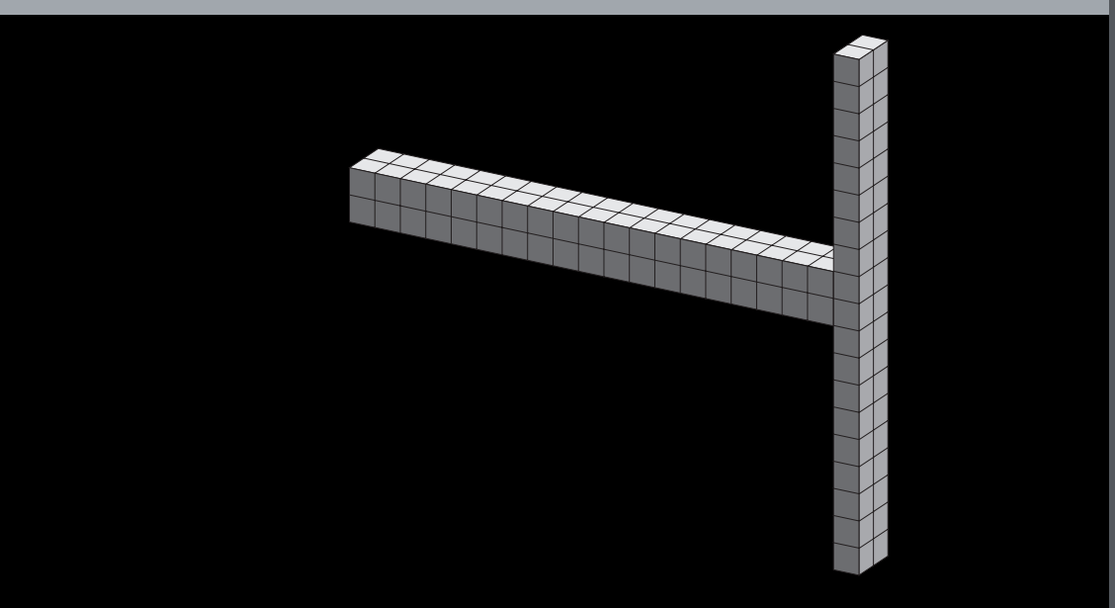
Mohamed A. A. Anndif

B15/2

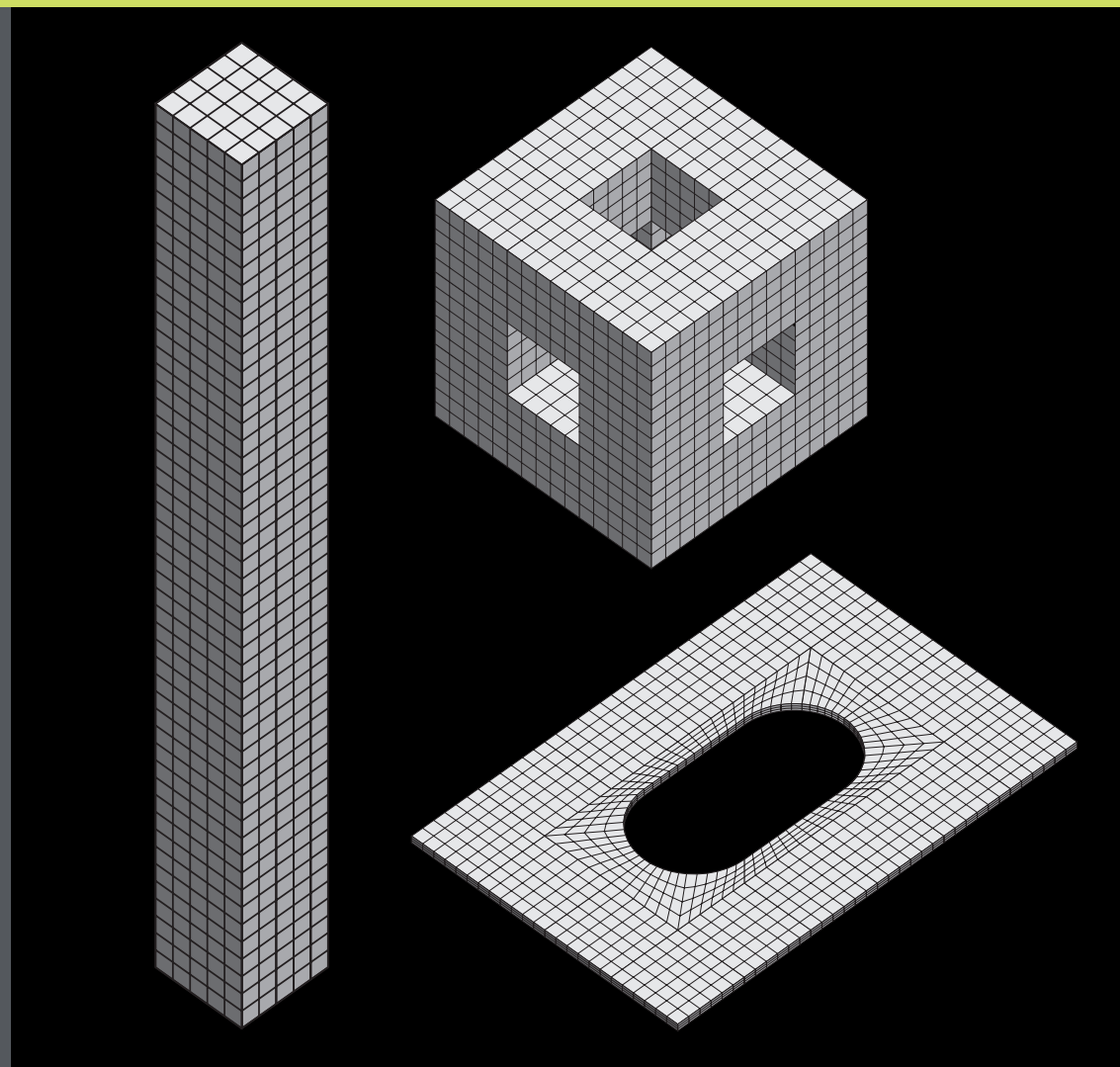
Leibniz
Universität
Hannover

ISBN 978-3-941302-15-0

The approximate solutions for complex nonlinear mechanical systems by using standard approaches (finite element, finite volume...etc.) are expensive with respect to both storage and CPU costs. Therefore, Reduced-order model (ROM) is usually thought as computationally inexpensive mathematical representations that offer potential for near real-time analysis such as systems of nonlinear structural mechanics. Nevertheless, ROM usually lacks robustness with respect to parameter changes and therefore must often be rebuilt for each parameter variation. Together, these two issues underline the need for a fast and robust method for adapting recomputed ROMs to new sets of physical or modeling parameters. ROM is based on eliminating degrees of freedom from the computational problem as appropriate to attain required computational efficiency.



Mohamed A. A. Anndif | Model Reduction in Nonlinear F E M for Engineering structures



Model Reduction in Nonlinear Finite Element Method for Engineering Structures

B15/2

Mohamed A. A. Anndif

Leibniz
Universität
Hannover

Institut für
Kontinuumsmechanik

Model Reduction in Nonlinear Finite Element Method for Engineering Structures

Von der Fakultät für Maschinenbau
der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des akademischen Grades
Doktor-Ingenieur

genehmigte Dissertation
von

M. Sc. Mohamed Ahmed Annour Anndif

geboren am 29.03.1984 in Alghrifa, Libyen

2015

Herausgeber:
Prof. Dr.-Ing. habil. Dr.h.c. mult. Peter Wriggers

Verwaltung:
Institut für Kontinuumsmechanik
Gottfried Wilhelm Leibniz Universität Hannover
Appelstraße 11
30167 Hannover

Tel: +49 511 762 3220
Fax: +49 511 762 5496
Web: www.ikm.uni-hannover.de

ISBN 978-3-941302-15-0

1. Referent: Prof. Dr.-Ing. Peter Wriggers
2. Referent: Prof. Dr.-Ing. Udo Nackenhorst

Tag der Promotion: 09.03.2015

"mechanics is the paradise of the mathematical science, because by means of it one comes to the fruits of mathematics "

Leonardo da Vinci

"if people do not believe that mathematics is simple, it is only because they do not realize how complicated life is"

John von Neumann

Abstract

The approximate solutions for complex nonlinear mechanical systems by using standard approaches (finite element, finite volume...etc.) are expensive with respect to both storage and CPU costs. Therefore, Reduced-order model (ROM) is usually thought as computationally inexpensive mathematical representations that offer potential for near real-time analysis such as systems of nonlinear structural mechanics. Nevertheless, ROM usually lacks robustness with respect to parameter changes and therefore must often be rebuilt for each parameter variation. Together, these two issues underline the need for a fast and robust method for adapting recomputed ROMs to new sets of physical or modeling parameters. ROM is based on eliminating degrees of freedom from the computational problem as appropriate to attain required computational efficiency. In this work, different approaches are introduced to reduce nonlinear models. These approaches are adaptive ROM based on proper orthogonal decomposition combined with BFGS method to decrease the computational cost, adaptive ROM based on the technique called proper snapshots selection, adaptive hyper-ROM based on Missing point estimation, and machine learning approach based on multi support vector regression.

Zusammenfassung

Die Näherungslösung für komplexe, nichtlineare mechanische Systeme unter Verwendung der standard Methoden (Finite Elemente, Finite Volumen, usw.) ist aufwändig in Bezug auf Speicher und CPU Leistung. Deshalb werden Modelle reduzierter Ordnung (ROM) als weniger rechenintensive mathematische Beschreibung gewählt, die das Potential für eine nahezuEchtzeit-Analyse von beispielsweise Systemen nichtlineare Strukturmechanik bieten. Allerdings fehlt den ROM im Allgemeinen die Robustheit in Bezug auf Parameteränderungen und die Modelle müssen deshalb häufig für jede Änderung der Parameter neu aufgebaut werden. Zusammen heben diese zwei Probleme den Bedarf für eine schnelle und robuste Methode zur Adaption von neu berechneten ROMs an neue Sätze physikalischer oder modellierender Parameter hervor. Die ROM Methode basiert auf der zulässigen Eliminierung von Freiheitsgraden aus der

Berechnung zur Erreichung der angestrebten Recheneffizienz. In dieser Arbeit werden verschiedene Ansätze zur Reduktion nichtlinearer Modelle vorgestellt. Diese Ansätze sind adaptive ROM unter Nutzung einer passenden orthogonalen Zerlegung kombiniert mit der BFGS Methode, um den Rechenaufwand zu verkleinern, adaptive ROM basierend auf der Methode der Auswahl passender Zustandsaufnahmen, adaptive hyper-ROM basierend auf einer Abschätzung fehlender Punkte und der Ansatz des maschinellen Lernens basierend auf einer Multi-Stützvektor-Regression.

Acknowledgements

I would like to thank my advisor **Prof. Dr.-Ing. habil. Dr.h.c. mult. Peter Wriggers** for giving me the opportunity to do this research under his supervision and also for his encouragements to make collaboration with the group of **Prof. Dr.-Ing. Paulo Pimenta**. My thanks also go to **Prof. Dr.-Ing. habil. Udo Nackenhorst** for volunteering to give his experience as a co-reference and for his constructive comments.

This dissertation is a result of a research work that carried out mainly at the Iinstitute of Continuum Mechanics at Leibniz University of Hannover, where I have benefited a lot from the inspiring and friendly atmosphere. I never see face in the institute look like the face of a child when he loose his mother. Thanks are due to all members of the institute. I, especially, acknowledge support I got from Ms. Vera Halfar during my hard time in Germany when I lost my scholarship. I would like to thank my friends from Brazil for their advice and help Paulo Nigro, Alfredo Gay Neto and his wife Juliana, and Jorge Costa.

Especially, I can not thank you enough Aisha and Ahmed for being good and loving parents to me. If I possess any good qualities, it is because of you. My gratitude goes also to my brothers for their useful advice.

I express my gratitude to my classmate from Libya and also friends in Germany (Ibrahim Al tarhouni (Braunschweig), Saleh Aboazom (Hamburg), and Ahmed Kalel (Hannover)). I should not finish without expressing my gratitude to my Libyan friends in Braunschweig when we spend most of our weekends talking about medical doctors! (Muhanad Abod, Ibrahim al Tarhouni, Mohamed Alhadad, Wael ben Dala, Seraj Al azrag, Altaher fatoh, Faqri, and Fathi Al aguri).

List of Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Motivation

The demand of the analysis of complex problems in the field of nonlinear structural mechanics has increased in both the academic and the industrial world. Therefore, a robust numerical tool which provides a systematic framework for analysis of such problems is required such as the finite element method. In general, the analysis of nonlinear deformation in solids and structures by using the finite element method can be expensive, especially for large systems or in situations when one requires many repeated trials such as structural optimization and design.

Reduced order models (ROMs) based on projection techniques give a solution to analysis and computation of such high dimensional finite element models. The basic idea of model reduction (dimensional reduction) is to find a small number of generalized coordinates which express the system with some error (Krysl et al. (2001)).

The projection based techniques were successfully applied in different computational engineering areas such as frequency response (Avery et al. (2007) & Hetmaniuk et al. (2013)), aeroelasticity (Thomas et al. (2003)), control (Bergmann et al. (2005)), structural dynamics (Amsallem et al. (2009) & Amabili et al. (2003)), and aerodynamics (Epureanu (2003).

During the last decades, different approaches and techniques of reduced order models came to existence. Unfortunately, most of these approaches pay a little attention to the stability of the reduced model compared to the high attention on the accuracy and the approximation of the original system (Farhat et al. (2014)).

1.2 State of the art

(Nickell (1976)) is the first who found a modal reduction based on projection for nonlinear systems. His method is known as modal basis. At each time step, an

eigenvalue problem is solved for a linearized system to find reduced basis. But, this will lead to the repetition of solving an eigenvalue problem (computationally expensive). Furthermore, the change of approximate basis is very important in the case of time–dependent. Almroth et al. (1978)) suggested to use a correct displacement in the first iteration to enhance the current basis for nonlinear static analysis. (Noor & Peters (1980), (1983)) used the path derivative which is the derivative of a displacement (nonlinear solution) with respect to step-size control parameter as basis. The later basis has good performance for geometrical nonlinearity problems but it is difficult to use it for material nonlinearities

To avoid the previous problem, (Wilson et al. (1982)) introduces a superposition of load-dependent Ritz vectors for linear analysis which he extended later to nonlinear analysis. In the context of Ritz vector, (Nour-Omid & Clough (1984)) used a Lanczos algorithm to generate basis vectors. (Idelson & Cardona (1985)) extended load-dependent Ritz vectors with derivatives respect to modal coefficients for nonlinear dynamic analysis. (Chan & Hsiao (1985)) used the orthogonality of the current and the previous displacement vectors with one selected equilibrium displacement vectors as the basis for nonlinear static analysis. (Kapania & Byun (1993)) used a combination between eigenvectors and Ritz vectors and they found that the updating or reduced basis is essential to obtain the accurate result.

The error appears as the result of few number of basis dimension was studied by different researchers. Kline (1986) found two sources of the error, the first one is due to inability of reduced basis to reproduce the loading, and the second is due to inability of reduced system to produce the eigenvibration of the full system. For that, he used a combination between eigenvectors and Ritz vectors as basis vectors to decrease the error. (Ibrahimbegovic & Wilson (1990)) suggested that eliminating of frequency effect during the process of generating the Ritz vectors for better approximation. (Cabos (1994)) introduced error bounds for the Krylov subspace.

Different extended approaches have been introduced to calculate the optimal basis. But, it is still challenging. Sirovich (1987) introduces the snapshots

method called Proper Orthogonal Decomposition (POD) which is based on Karhunen-Loéve decomposition. The method is considered as the best technique to generate optimal basis compared to previous techniques especially for nonlinear systems. Proper Orthogonal Decomposition was used in the first time by (Kreuzer & Kust(1996)) for the dynamic structures.

A comparison of the previous reduction techniques can be seen in (Spiess & Wriggers (2005), Spiess (2006), Radermacher & Reese (2013)). Recently, different approaches based on the interpolation and proper orthogonal decomposition where introduced such as, a missing point estimation (Astridet al.(2008)), (Vendl & Faβbender (2010, 2011)), a priori hyper-reduction (Ryckelynck (2005), Kerfriden et al. (2011)), a discrete empirical interpolation (Chaturantabut & Sorensen (2010)) and a Gauss-Newton approximate tensor method (Carlberg et al.(2011)). These techniques show dramatically decreases of computational possessing time.

1.3 Objectives

The objective of this study is thus to evaluate the robustness of the projection-based reduction techniques in order to carry out better adjustment for both accuracy and processing time. At the end, this would result in obtaining a more controlled system which lead to better accuracy.

1.4 The organization of the thesis

This thesis consists of six chapters. The first chapter presents an introduction. The second chapter gives a brief overview of non-liner finite element method. The third chapter presents reduced order model based on projection with some reduction techniques with emphasis on Proper orthogonal decomposition (POD) and Missing point estimation Technique. The fourth chapter presents different proposed approaches of an adaptive reduced order model with numerical results. The fifth chapter introduces a support vector machine and numerical result. Finally, the sixth chapter includes conclusion, limitations and recommendations for future research.

Chapter 2 Introduction to Nonlinear Finite Element Method

2.1 Solid and Structural Nonlinearities

In practical applications, the linear elasticity has the limitation due to the presence of nonlinearity effects or due to the geometry of the thin structure in one or two dimensions. The simple material nonlinearity is the elasticity when the relation between stress and strain is nonlinear. In other side, there is a case when the deformation of the structure reach a point leads to that the deformable and undeformable shape are highly different and it is not possible to find a linear strain-displacement or equilibrium equations on the undeformed geometry (finite deformation). Before this point (finite deformation), it may be possible to observe buckling or load bifurcations in some solids and nonlinear equilibrium should be considered (Zienkiewicz & Taylor (2000)).

In general, different sources of nonlinearities may exist in the analysis of nonlinear structural mechanics systems such as Geometrical nonlinearity (eg. large displacements, rotations), physical nonlinearity (nonlinear material) and nonlinearity due to boundary conditions (contact problem, deformation dependent loading). These kinds of nonlinearity exist from most real structural mechanics systems.

In fact, the small strain problems can be solved accurately by using standard finite element method but there are still some challenges to solve some nonlinearity due to finite deformation, contact problem and also material instabilities by using standard routine codes and software (Wriggers (2008)).

2.2 Principles of nonlinear FEM

In this section, a brief overview of the nonlinear finite element method is presented. The detailed description of it can be found in different references such as (Zienkiewicz& Taylor (2000), Wriggers (2008)). The system of nonlinear structural mechanics is described as to satisfy three issues: ***kinematic***

relations, constitutive law, and balance of momentum. Starting from the balance momentum equation:

$$\text{Div}\ \boldsymbol{P} + \rho_0\ \boldsymbol{b} = \rho_0 \ddot{\boldsymbol{u}} \tag{2.1}$$

where $\boldsymbol{P}$ is the first Piola–Kirchhoff stress tensor, ρ_0 the density of undeformed configuration, $\ddot{\boldsymbol{u}}$ is the acceleration, and $\boldsymbol{b}$ is the body force.

Taken into account stress and kinematic relations, the first Piola–Kirchhoff stress tensor is defined as:

$$\boldsymbol{P} = \boldsymbol{F}\ \boldsymbol{S} \tag{2.2}$$

where $\boldsymbol{S}$ is the second Piola–Kirchhoff stress tensor, $\boldsymbol{F}$ is the deformation gradient.

In case of inelastic constitutive material, the second Piola–Kirchhoff stress tensor is a function of Green-Lagrange strain tensor ($\boldsymbol{E}$) and internal variables ($\boldsymbol{Q}_i$) which depend on the material (rate-dependence or rate-independence) and can be found from evolution equation as:

$$\boldsymbol{S} = \boldsymbol{S}_{function}\left(\boldsymbol{E}, \boldsymbol{Q}_i\right) \tag{2.3}$$

The kinematic relation between displacements and strain is expressed by:

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{C} - \boldsymbol{I}\right) = \frac{1}{2}\left(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}\right)$$

where $\boldsymbol{C}$ is the Right Cauchy-Green tensor and the $\boldsymbol{F}$ is the deformation gradient.

In this work, a simple Poly-convex Neo-Hookean material is used which it is often called Neo-Hookean constitutive model of Simo-Ciarlet. The strain Energy Function ($\left(\psi\right)$) can be given as:

$$\psi\left(\boldsymbol{J};\boldsymbol{I}_1\right) = \frac{1}{2}\boldsymbol{\lambda}\left(\frac{1}{2}\left(\boldsymbol{J}^2 - \boldsymbol{1}\right) - \ln \boldsymbol{J}\right) + \frac{1}{2}\boldsymbol{\mu}\left(\boldsymbol{I}_1 - 3 - 2\ln \boldsymbol{J}\right) \tag{2.4}$$

where $\boldsymbol{J}$ is the third invariant of deformation gradient $\boldsymbol{F}$, $\boldsymbol{I}_1$ is the first invariant of Cauchy tensor, $\boldsymbol{\lambda}$ and μ are the Lammé constants.

Inserting both equations (2.3) and (2.2) into equation (2.1) leads to a new form of balance momentum equation as:

$$\mathrm{Div}\left(\boldsymbol{F}\left(\boldsymbol{u}\right)\left(\boldsymbol{E};\boldsymbol{Q}_i\right)\right)+\rho_0\,\boldsymbol{b}=\rho_0\ddot{\boldsymbol{u}} \tag{2.5}$$

This equation (balance of momentum) should be fulfilled in every point of the body at any time. Furthermore, the initial condition and boundary condition should be satisfied. Given a domain ($\boldsymbol{B}$) enclosed by boundary ($\partial\boldsymbol{B}$), this boundary can be split intoparts Displacements are imposed on $\partial\boldsymbol{B}_u$ and Traction is imposed on $\partial\boldsymbol{B}_t$.

Thus, the initial boundary value problem is fully stated by:

- Balance of momentum

$$\mathrm{Div}\left(\boldsymbol{F}\left(\boldsymbol{u}\right)\left(\boldsymbol{E};\boldsymbol{Q}_i\right)\right)+\rho_0\,\boldsymbol{b}=\rho_0\ddot{\boldsymbol{u}}$$

- Initial conditions

$$\boldsymbol{u}\left(t=0\right)=\boldsymbol{u}_0\;on\;\boldsymbol{B}$$
$$\dot{\boldsymbol{u}}\left(t=0\right)=\dot{\boldsymbol{u}}_0\;on\;\boldsymbol{B}$$

- Boundary condition

$$\boldsymbol{u}=\boldsymbol{u}^*\qquad on\;\partial\boldsymbol{B}_u$$
$$\boldsymbol{t}^*=\boldsymbol{P}\,\boldsymbol{n}^*=\boldsymbol{F}\,\boldsymbol{S}\,\boldsymbol{n}^*\quad on\;\partial\boldsymbol{B}_t$$
$$\partial\boldsymbol{B}_u\cup\partial\boldsymbol{B}_t=\partial\boldsymbol{B}$$
$$\partial\boldsymbol{B}_u\cap\partial\boldsymbol{B}_t=\boldsymbol{\phi} \tag{2.6}$$

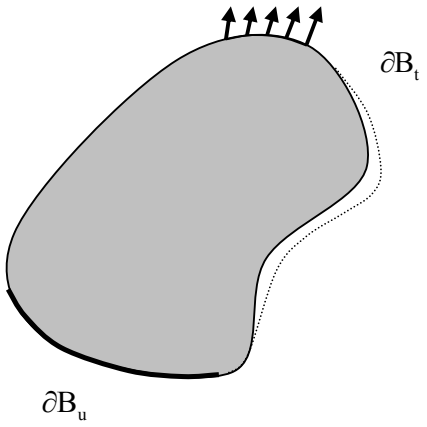


Figure 2.1: Boundary condition of initial value problem.

To solve an initial value problem, a numerical tool such as finite element method is required. The main idea of finite element method is to descritize the entire domain ($\boldsymbol{B}$) into finite elements, size (Ω_e). It is possible that the finite elements does not fit the entire domain ($\boldsymbol{B}$) which will lead to discritization errors.

The finite element method is based on weak solution of the balance of the momentum. This means that, the integration of balance of the momentum multiplied by test function ($\boldsymbol{\eta}$) over all domains should be zero for any test function.

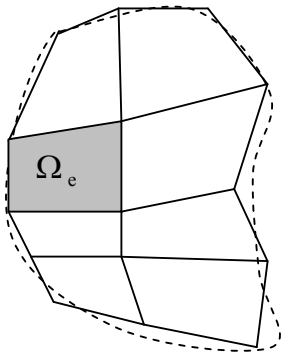


Figure 2.2: Discretization of finite element method

$$\int_B \left[\mathrm{Div}\boldsymbol{P} + \rho\,\boldsymbol{b} - \rho\ddot{\boldsymbol{u}} \right] \cdot \boldsymbol{\eta}\, dV = 0, \forall \boldsymbol{\eta} \tag{2.7}$$

After applying Green's theorem and considering ($\boldsymbol{\eta} = \boldsymbol{0}$ *on* $\partial \boldsymbol{B}_u$), a weak form of equation of motion is presented in (2.8). The first term internal virtual work and last two terms describe respectively the external virtual work and inertia contributions.

$$\int_B \boldsymbol{P} : Grad\ \boldsymbol{\eta}\ \ dV\ \ -\ \int_B \rho\ (\boldsymbol{b} - \ddot{\boldsymbol{u}})\ \ \boldsymbol{\eta}\ dV\ \ -\int_{B_t} \boldsymbol{t}\ \boldsymbol{\eta}\ dA = 0 \tag{2.8}$$

A finite element describes the structure deformation by the nodes. So, the finite element approximation for displacements is given as:

$$\boldsymbol{u}\left(\boldsymbol{x},t\right) \approx \tilde{\boldsymbol{u}}^{h}\left(\boldsymbol{x},t\right) = \sum_{b} \boldsymbol{N}_{b}\left(\boldsymbol{x}\right)\tilde{\boldsymbol{u}}_{b}\left(t\right) = \boldsymbol{N}\left(\boldsymbol{x}\right)\tilde{\boldsymbol{u}}\left(t\right) \tag{2.9}$$

where $\boldsymbol{N}_{b}$ are element shape functions, $\tilde{\boldsymbol{u}}_{b}$ are time-dependent nodal displacements, and the sum is over the number of nodes of an element. The previous approximation can be expressed alternatively by Isoparametric form as:

$$\begin{aligned} \boldsymbol{u}\left(\zeta,t\right) &\approx \tilde{\boldsymbol{u}}^{h}\left(\zeta,t\right) = \sum_{b} \boldsymbol{N}_{b}\left(\zeta\right)\tilde{\boldsymbol{u}}_{b}\left(t\right) = \boldsymbol{N}\left(\zeta\right)\tilde{\boldsymbol{u}}\left(t\right) \\ \boldsymbol{x}\left(\zeta\right) &= \sum_{b} \boldsymbol{N}_{b}\left(\zeta\right)\tilde{\boldsymbol{x}}_{b}\left(t\right) = \boldsymbol{N}\left(\zeta\right)\tilde{\boldsymbol{x}}\left(t\right) \end{aligned} \tag{2.10}$$

where $\tilde{\boldsymbol{x}}$ represent nodal coordinate parameters and ζ are the parametric coordinates for each element.

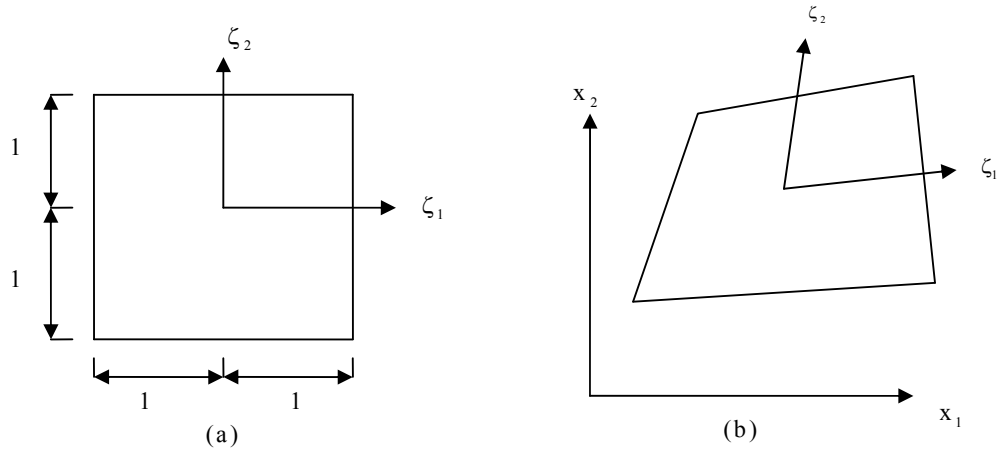


Figure 2.3: Isoperimetric map for four-node two-dimensional quadrilateral: (a) element in ζ coordinates and (b) element in x coordinates.

The gradient of the approximated field can be obtained from the derivative of the shape functions as

$$Grad\ \tilde{\boldsymbol{u}}^{h}\left(\boldsymbol{x},t\right) = \sum_{b} Grad\ \boldsymbol{N}_{b}\left(\boldsymbol{x}\right)\tilde{\boldsymbol{u}}_{b}\left(t\right) \tag{2.11}$$

If we gather the nodal values in a vector ($\tilde{\boldsymbol{u}}$) and the shape function in the matrix ($\boldsymbol{N}$), we can express $\tilde{\boldsymbol{u}}^{h}\left(\boldsymbol{x},t\right) = \boldsymbol{N}\left(\boldsymbol{x}\right)\tilde{\boldsymbol{u}}\left(t\right)$ and $Grad\ \tilde{\boldsymbol{u}}^{h} = \boldsymbol{B}\left(\boldsymbol{x}\right)\boldsymbol{u}\left(t\right)$, where $\boldsymbol{B}$ gathers $Grad\ \boldsymbol{N}_{b}$.

The same shape functions are also used for the test function ($\boldsymbol{\eta}$) as:

$$\boldsymbol{\eta}\left(\zeta\right) = \sum_{b} \boldsymbol{N}_b\left(\zeta\right) \cdot \boldsymbol{\eta}_b \tag{2.12}$$

The strong form which split into sub-domains (2.1) can also be weakly expressed in each element's domain as:

$$\underset{e=1}{\overset{ne}{\mathbf{A}}} \left(\int_{\Omega_e} \boldsymbol{P} : Grad\ \boldsymbol{\eta}\ \ dV\ \ - \int_{\Omega_e} \rho\ (\boldsymbol{b}\text{ - }\ddot{\boldsymbol{u}})\ \ \boldsymbol{\eta}\ dV\ \ - \int_{\partial\Omega_e} \boldsymbol{t}\ \boldsymbol{\eta}\ dA \right) = 0 \tag{2.13}$$

After discretization, this can be expressed as:

$$\boldsymbol{\eta}^T \int_{\Omega_e} \boldsymbol{B}^T\ \boldsymbol{P}\ d\Omega_e - \boldsymbol{\eta}^T \int_{\Omega_e} \rho\ \boldsymbol{N}\ \boldsymbol{b}\ d\Omega_e + \boldsymbol{\eta}^T \int_{\Omega_e} \rho\ \boldsymbol{N}^T\boldsymbol{N}\ d\Omega_e \ddot{\boldsymbol{u}} - \boldsymbol{\eta}^T \int_{\partial\Omega_e} \boldsymbol{N}^T\boldsymbol{t}\ d\Omega_e \tag{2.14}$$

where $\ddot{\boldsymbol{u}}$ collects the acceleration of this element's nodes. If we define:

$$\begin{aligned} \boldsymbol{M}_e &= \int_{\Omega_e} \rho\ \boldsymbol{N}^T\boldsymbol{N}\ d\Omega_e \\ \boldsymbol{R}_e &= \int_{\Omega_e} \boldsymbol{B}^T\ \boldsymbol{P}\ d\Omega_e \\ P_e &= \int_{\Omega_e} \rho\ \boldsymbol{N}^T\ \boldsymbol{b}\ d\Omega_e + \int_{\partial\Omega_e} \boldsymbol{N}^T\boldsymbol{t}\ d\Omega_e \end{aligned} \tag{2.15}$$

and take into account the arbitrary of $\boldsymbol{\eta}$, we can express the element's equilibrium as

$$\underset{e=1}{\overset{ne}{\mathbf{A}}} \left(\boldsymbol{\eta}_e^T \left(\boldsymbol{M}_e\left(\ddot{u}\right) + \boldsymbol{R}_e\ \left(u\right) - \boldsymbol{P}_e \right) \right) = 0 \tag{2.16}$$

The global equilibrium can be expressed after the assembling for the entire domain as

$$\begin{aligned} &\boldsymbol{M}\ \ddot{\boldsymbol{u}}(t) + \boldsymbol{R}\ (\boldsymbol{u}(t)) = \boldsymbol{P}(t) \\ &\boldsymbol{M} = \bigcup_{e=1}^{n_e} \boldsymbol{M}_e \\ &\boldsymbol{R} = \bigcup_{e=1}^{n_e} \boldsymbol{R}_e \end{aligned} \tag{2.17}$$

where $\boldsymbol{M}$ is the mass matrix, $\ddot{\boldsymbol{u}}$ is acceleration, $\boldsymbol{R}$ the internal force vector, $\boldsymbol{P}$ is eternal force vector, e is the expression of an element.

The nonlinear structural mechanics system in equation (2.17) is in the form of a dynamic system. For the dynamic case, a damping term should be added whose real behavior is still discussed, which is very difficult to find its behavior of the system in reality. So, the system (2.17) will be:

$$\boldsymbol{M}\ \ddot{\boldsymbol{u}}(t) + \boldsymbol{C}\ \dot{\boldsymbol{u}}(t) + \boldsymbol{R}\ \left(\boldsymbol{u}(t)\right) = \boldsymbol{P}(t) \tag{2.18}$$

where $\mathbf{C}$ is the damping matrix, $\dot{\mathbf{u}}$ is velocity.

In this work, a simple damping called Rayleigh damping is used. It is assumed to be proportional to the mass and stiffness matrices as follows:

$$\begin{aligned} \boldsymbol{C} &= d_1\ \boldsymbol{M} + \mathrm{d}_2\ \boldsymbol{K} \\ \xi &= \frac{d_1}{2\omega} + \frac{d_2\omega}{2} \end{aligned} \tag{2.19}$$

where d_1 is the mass-proportional damping coefficient, d_2 is the stiffness-proportional damping coefficient, ξ is damping ratio, and ω is natural frequency.

The value of the variables (d_1, d_2) can be found from solving the system of equations for two different values of natural frequency (ω) with fixed value of the damping ratio (ξ). The damping matrix was computed based on the incremental procedure because stiffness changes with the time.

2.3 Time integration methods:

The equation (2.17) can be reformulated to be in the first order differential equation form by introducing the independent variables as:

$$\begin{aligned} \dot{\boldsymbol{u}} &= \boldsymbol{v}, \text{ for velocity} \\ \ddot{\boldsymbol{u}} &= \boldsymbol{v} = \boldsymbol{M}^{-1}\left[\boldsymbol{P} - \boldsymbol{C}\,v - \boldsymbol{R}(u)\right], \text{ for acceleration} \end{aligned} \tag{2.20}$$

There are two ways to solve this equation numerically which are well known as implicit or explicit time integration. The explicit time integration has some disadvantages such as limitation of time step and instability due to the fact that solution at time (t_{i+1}) depends only on the solution at time (t_i). In other hand, the explicit time integration is easy to implement and it is suitable for problems

such as impact problem which require very small time steps for handling high frequencies.

The implicit time integration does not has instability problems due to that the solution at time (t_{i+1}) does not depend only on the solution at time (t_i). Furthermore, the equilibrium is checked simultaneously, especially if it is combined with Newton method. For that, it can be used with much bigger time steps.

In this work, the implicit time integration method called Newmark method is used. It is commonly used in structural dynamic. A brief overview of its formulations will be presented in this work based on (Wriggers (2008)). Its approximation of the displacement at time (t_{i+1}) is:

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i + \Delta t\ \dot{\boldsymbol{u}}_i + \frac{(\Delta t)^2}{2}\left[(1-2\beta)\ddot{\boldsymbol{u}}_i + 2\beta\ \ddot{\boldsymbol{u}}_{i+1}\right] \tag{2.21}$$

and for velocity

$$\dot{\boldsymbol{u}}_{i+1} = \dot{\boldsymbol{u}}_i + \Delta t\ \left[(1-\gamma)\ \ddot{\boldsymbol{u}}_i + \gamma\ \ddot{\boldsymbol{u}}_{i+1}\right] \tag{2.22}$$

where γ & β are Newmark's parameters

Thus, the equations (2.21) and (2.22) are not dependent only on (t_i). But, it depends also on acceleration at (t_{i+1}). The acceleration at time (t_{i+1}) can be found from equilibrium equation(2.17). Finally equations (2.21) and (2.22) can be formulated to be that acceleration and velocity are depends only on displacements at (t_{i+1}) and other quantities at (t_i).

$$\ddot{\boldsymbol{u}}_{i+1} = \alpha_1(\boldsymbol{u}_{i+1} - \boldsymbol{u}_i) - \alpha_2\ \dot{\boldsymbol{u}}_i - \alpha_3\ddot{\boldsymbol{u}}_i \tag{2.23}$$

$$\dot{\boldsymbol{u}}_{i+1} = \alpha_4(\boldsymbol{u}_{i+1} - \boldsymbol{u}_i) + \alpha_5\ \dot{\boldsymbol{u}}_i + \alpha_6\ddot{\boldsymbol{u}}_i \tag{2.24}$$

$$a_1 = \frac{1}{\beta(\Delta t)^2},\quad a_2 = \frac{1}{\beta\Delta t},\quad a_3 = \frac{1-2\beta}{2\beta},$$

$$\alpha_4 = \frac{\gamma}{\beta\Delta t},\quad \alpha_5 = (1-\frac{\gamma}{\beta}),\quad \alpha_6 = (1-\frac{\gamma}{2\beta})\ \Delta t,$$

Common values of Newmark's parameters are $(\gamma = 0.5)$ and $(\beta = 0.25)$. The system in equation (2.17) becomes:

$$\begin{aligned}&\boldsymbol{M}\ [\alpha_1(\boldsymbol{u}_{i+1}-\boldsymbol{u}_i)-\alpha_2\dot{\boldsymbol{u}}_i-\alpha_3\ddot{\boldsymbol{u}}_i]\\&+\boldsymbol{C}\ [\alpha_4(\boldsymbol{u}_{i+1}-\boldsymbol{u}_i)-\alpha_5\dot{\boldsymbol{u}}_i-\alpha_6\ddot{\boldsymbol{u}}_i]\\&+\boldsymbol{R}\ (\mathrm{u}_{i+1})=\boldsymbol{P}_{i+1}\end{aligned}\tag{2.25}$$

As all quantities are known at (t_i), the only unknown in (t_{i+1}) is $(\boldsymbol{u}_{i+1})$ which is the primary variables of the system.

2.4 Solution methods of nonlinear structural mechanics systems

There are different numerical methods which can be used to solve nonlinear systems. An extensive overview of these methods was mentioned in (Wriggers (2008) & Zienkiewicz et al (2000)). In this context, a short description of the used method is presented.

2.4.1 Newton-Raphson method

Newton-Raphson is an iterative method to solve nonlinear algebraic system of equations such as the system in structural mechanics:

$$\begin{aligned}&\boldsymbol{G}(\boldsymbol{u})=0\\&\boldsymbol{G}(\boldsymbol{u})=\boldsymbol{M}\ \ddot{\boldsymbol{u}}(t)+\boldsymbol{C}\ \dot{\boldsymbol{u}}(t)+\boldsymbol{R}\ (\boldsymbol{u}(t))-\boldsymbol{P}(t)=0\end{aligned}\tag{2.26}$$

Newton-Raphson is based on Taylor expansion of the residual $\boldsymbol{G}(\boldsymbol{u})$. It neglects higher order terms (HOT) and thus yields

$$\boldsymbol{G}(\boldsymbol{u}_{i+1}^{j})=\boldsymbol{G}(\boldsymbol{u}_i^{j})\ +\ D\boldsymbol{G}(\boldsymbol{u}_i^{j})\,\Delta\boldsymbol{u}\ +\ HOT=0\tag{2.27}$$

$D\boldsymbol{G}(\boldsymbol{u})$ is called Hesse-, Jacobi-, Tangent- matrix which is a linearization of the residual vector $\boldsymbol{G}(\boldsymbol{u})$. The derivative can be done for the Newmark time integration (2.25)as:

$$\begin{aligned}&D\boldsymbol{G}(\boldsymbol{u}_i^{j})=\boldsymbol{K}_{eff}(\boldsymbol{u}_i^{j})=\boldsymbol{M}\ \frac{\partial\ddot{\boldsymbol{u}}\left(t\right)}{\partial\boldsymbol{u}}+\boldsymbol{C}\ \frac{\partial\dot{\boldsymbol{u}}\left(t\right)}{\partial\boldsymbol{u}}+D\boldsymbol{R}\ \left(\boldsymbol{u}(t)\right)-D\boldsymbol{P}\left(t\right)\\&\boldsymbol{K}_{eff}(\boldsymbol{u}_i^{j})=\frac{1}{\Delta t^2\beta}\boldsymbol{M}+\frac{\gamma}{\Delta t\beta}\boldsymbol{C}+\boldsymbol{K}_T\left(\boldsymbol{u}_i^{j}\right)\\&\boldsymbol{K}_T(\boldsymbol{u}_i^{j})=\frac{\partial\boldsymbol{R}}{\partial\boldsymbol{u}_i}\end{aligned}\tag{2.28}$$

$\boldsymbol{K}_T$ is called tangent stiffness matrix which is a linearization of internal force vector which depends on the displacement.

Algorithm 2.1: Standard Newton-Raphson with Newmark integration method

Input $\boldsymbol{u}_0$, $i = 0$

repeat

$$\Delta \boldsymbol{u}_{i+1}^{j} = [\boldsymbol{K}_{eff}(\boldsymbol{u}_i^j)]^{-1} \, (-\boldsymbol{G}(\boldsymbol{u}_i^j))$$

$$\boldsymbol{u}_{i+1}^{j} = \boldsymbol{u}_i^j + \Delta \boldsymbol{u}_{i+1}^{j}$$

$$\dot{\boldsymbol{u}}_{i+1}^{j} = \dot{\boldsymbol{u}}_i^j + \frac{\gamma}{\Delta t \beta} \Delta \boldsymbol{u}_{i+1}^{j}$$

$$\ddot{\boldsymbol{u}}_{i+1}^{j} = \ddot{\boldsymbol{u}}_i^j + \frac{1}{\Delta t^2 \beta} \Delta \boldsymbol{u}_{i+1}^{j}$$

$$i = i + 1$$

until $\left\| \boldsymbol{G}(\boldsymbol{u}_i) \right\| \leq \boldsymbol{tol}$

In fact, computing of the tangent stiffness matrix is computational expensive, especially for the big systems. Finally, the Newtown-Raphson method solves the problem iteratively as shown in algorithm (2.1). The main advantage of the Newton-Raphson method is that, its convergence is quadratic. This will lead to reach the solution with small number of iterations. The illustration of Newton-Raphson can be seen in Figure 2.4.

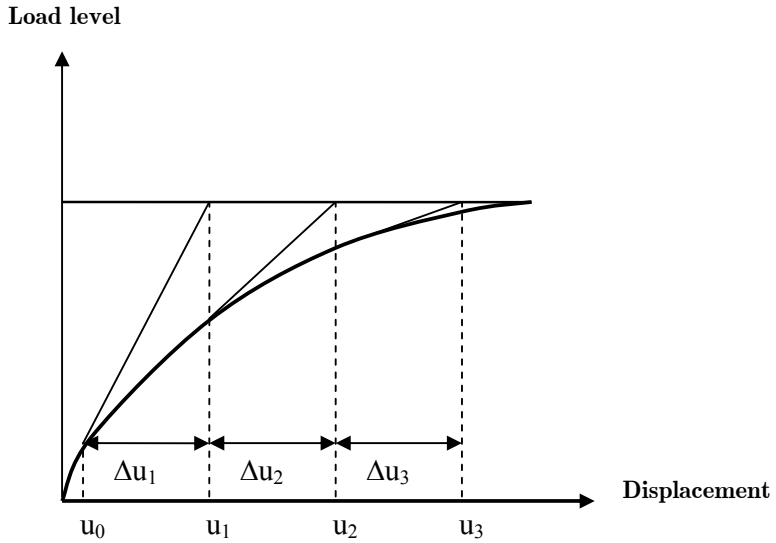


Figure 2.4: Standard Newton-Raphson (NR) method

2.4.2 Quasi-Newton method

The main idea of Quasi-Newton method is that using a secant slope formulation instead of the derivative (tangent stiffness matrix). One of the popular method

of Quasi-Newton method is Broyden-Fletcher-Goldfarb-Shanno (BFGS) ((Matthies & Strang(1979)), (Yuan G. & Yao S.(2011)). The BFGS computes a second derivative as:

$$\begin{aligned}
&\Delta \boldsymbol{u} = \boldsymbol{u}_i - \boldsymbol{u}_{i-1} \\
&\Delta \boldsymbol{G} = \boldsymbol{G}(\boldsymbol{u}_{i-1}) - \boldsymbol{G}(\boldsymbol{u}_i) \\
&\Delta \boldsymbol{G}_{\Delta k} = \boldsymbol{K}_{eff}(\boldsymbol{u}_{i\text{-}1})\ \Delta \boldsymbol{u} \\
&\boldsymbol{K}_{eff}(\boldsymbol{u}_i) = \boldsymbol{K}_{eff}(\boldsymbol{u}_{i\text{-}1}) + \frac{\Delta \boldsymbol{G}\ \Delta \boldsymbol{G}^T}{\Delta \boldsymbol{G}^T \Delta \boldsymbol{u}} - \frac{\Delta \boldsymbol{G}_{\Delta k}\ \Delta \boldsymbol{G}_{\Delta k}^T}{\Delta \boldsymbol{G}_{\Delta k}^T\ \Delta \boldsymbol{u}}
\end{aligned} \tag{2.29}$$

The main advantage of BFGS is that avoiding a direct computation of global stiffness matrix which is computationally expensive. But, it gives a superliner convergence compared with standard Newton method which gives a quadratic convergence.

2.4.3 Arc-Length method

The arc-length control method is a path following method to solve some problem of stability in nonlinear structural mechanics such as critical points, bifurcation...etc for static cases. This technique is required when there is no unique solution of the equilibrium solution path for each load parameter. The following of the path is very important for different problems in structural mechanics, especially in the field of thin structure and material instability. The path follow is possible even when singular points are present (determination of stiffness matrix equal zero) (Wriggers (2008)).

It is a control of an equilibrium equation in form of the residual $\boldsymbol{G}$ by introducing the load parameter (λ) as:

$$\boldsymbol{G}\left(\boldsymbol{u},\lambda\right) = \boldsymbol{F}_{\text{int}}\ \left(\boldsymbol{u}\left(t\right)\right) - \lambda \boldsymbol{P}\left(t\right) = \boldsymbol{0} \tag{2.30}$$

Then, the control can be done by an additional constraint $(\boldsymbol{f}(\boldsymbol{u},\lambda) = 0)$ which leads to found a load parameter (λ)as:

$$\boldsymbol{f}\left(\boldsymbol{u},\lambda\right) = \boldsymbol{0} \tag{2.31}$$

where $\boldsymbol{F}_{\text{int}}$is the internal force vector, $\boldsymbol{P}\left(t\right)$is external force vector, and λ is load level parameter.

A linearization is required of this system of equations to be solved by the Newton-Raphson method. The linearization is done with respect to the main variables (displacement $\left(\boldsymbol{u}\right)$ and load parameter (λ)) as:

$$\begin{aligned}
&\boldsymbol{G}(\boldsymbol{u}_k,\lambda_k) \;+\; \overbrace{\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{u}}\Big|_{\boldsymbol{u}_k,\lambda_k}}^{\boldsymbol{K}_{T_k}} \cdot\; \Delta \boldsymbol{u}_{\,k+1} \;+\; \overbrace{\frac{\partial \boldsymbol{G}}{\partial \lambda}\Big|_{\boldsymbol{u}_k,\lambda_k}}^{-\boldsymbol{P}} \cdot\; \Delta\lambda_{\,k+1} = 0 \\
&\boldsymbol{f}(u_k,\lambda_k) \;+\; \underbrace{\frac{\partial \boldsymbol{f}}{\partial u}\Big|_{\boldsymbol{u}_k,\lambda_k}}_{\boldsymbol{q}_k^T} \cdot\; \Delta \boldsymbol{u}_{\,k+1} \;+\; \underbrace{\frac{\partial \boldsymbol{f}}{\partial \lambda}\Big|_{\boldsymbol{u}_k,\lambda_k}}_{\boldsymbol{f},\lambda_k} \cdot\; \Delta\lambda_{\,k+1} = 0
\end{aligned} \tag{2.32}$$

The system of equations (2.33) is non-symmetric. So, the partition technique will be used for solving such kind of system. The system of equation (2.33) can not be singular in the limit point (determination of tangent matrix equal zero). The partition will lead to two equations of increments (displacement increment $\Delta \boldsymbol{u}_{k+1}$ and load increment $\Delta\lambda_{k+1}$).

$$\begin{bmatrix} \boldsymbol{K}_{T_k} & -\boldsymbol{P} \\ \boldsymbol{q}_k^T & \boldsymbol{f},\lambda_k \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{u}_{k+1} \\ \Delta\lambda_{k+1} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{G}_k \\ \boldsymbol{f}_k \end{bmatrix} \tag{2.33}$$

Those increments are determined as:

$$\Delta \boldsymbol{u}_{k+1} = \Delta\lambda_{k+1}\; \Delta \boldsymbol{u}_{\boldsymbol{P}_{k+1}} + \Delta \boldsymbol{u}_{\boldsymbol{G}_{k+1}} \tag{2.34}$$

where :

$$\begin{aligned}
\Delta \boldsymbol{u}_{\boldsymbol{P}_{k+1}} &= (\boldsymbol{K}_{T_k})^{-1}\ \mathrm{P} \\
\Delta \boldsymbol{u}_{\boldsymbol{G}_{k+1}} &= -(\boldsymbol{K}_{T_k})^{-1}\ \boldsymbol{G}_k \\
\Delta\lambda_{k+1} &= \frac{-\boldsymbol{f}_k - \boldsymbol{q}_k^T \Delta \boldsymbol{u}_{\boldsymbol{G}_{k+1}}}{\boldsymbol{f},\lambda_k + \ \boldsymbol{q}_k^T\ \Delta \boldsymbol{u}_{P_{k+1}}}
\end{aligned}$$

In the literature, there are different options of choosing the constraint function ($\boldsymbol{f}$). Riks (1972) suggested that a constraint function is linear. But, Crisfield (1981) suggested being a spherical function. A more detail description of Arc-length method is presented by (Wriggers(2008)).

Chapter 3 Reduced Order Model based on Projection

3.1 Introduction

Simulation of real nonlinear problems in mechanics requires a solution of large systems with up to millions of unknowns. Such a high fidelity system requires powerful computers with high storage capabilities. In several cases, the repetition of computation with slightly changing inputs is required in different fields such as control design, optimization, and simulation of surgery process…etc.

Thus, the simulation time becomes an important issue in order to improve the assessment and design of products. Model order reduction for both linear and nonlinear system has received a significant attention over the past decades. Its goal is to reduce the number of equations to be solved and thereby decrease the time to be spent on solving the system. The solution s of high fidelity models of order N lies in low dimensional subspaces of order k <<N.

3.2 Reduced order model

In general, the description of any system starts from a partial differential equation (PDE). The spatial discretization of PDE will lead to an ordinary differential equation whose dimension is governed by the spatial mesh size. The dimension can range from hundreds to millions.

The Model Order Reduction is a reduction of the dimension of an ordinary differential equation. An overview of the system includes model order reduction in Figure 3.1. A typical system of the ordinary differential equation systems in structural dynamic is:

$$\boldsymbol{M}\ \ddot{\boldsymbol{u}}\big(t\big) + \boldsymbol{C}\ \dot{\boldsymbol{u}}\big(t\big) + \boldsymbol{R}\ \Big(\boldsymbol{u}\big(t\big)\Big) = \boldsymbol{P}\big(t\big) \tag{3.1}$$

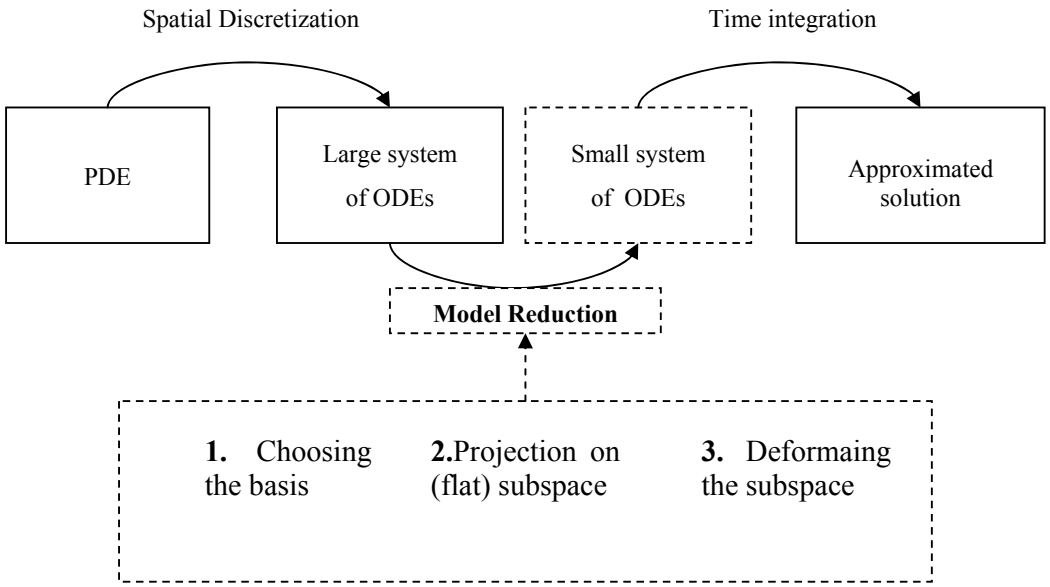


Figure 3.1: Projection based Model reduction (Meyer (2006)).

3.3 Dimension reduction

The main idea of reduced order model is to project the full system space of the ODEs onto a subspace of small dimension. In this way, the coordinates of the full spaced used for approximate the displacements $\boldsymbol{u}$ can be obtained from smaller set of generalized coordinates $\boldsymbol{q}$ by projection matrix $\boldsymbol{\Phi}$ as:

$$\boldsymbol{u}(t) = \boldsymbol{\Phi}\ \boldsymbol{q}(t) \tag{3.2}$$

where $\boldsymbol{u}$ is a displacement vector with size $N \times 1$, $\boldsymbol{\Phi}$ is a subspace matrix with size $N \times m$, $\boldsymbol{q}$ is a vector of generalized coordinates with size $m \times 1$. N is the dimension of the full space, and m is the dimension of the reduced space.

Using projection(3.2), the structural dynamic system in equation (3.1) becomes:

$$\boldsymbol{M}\ \boldsymbol{\Phi}\ \ddot{\boldsymbol{q}}(t) + \boldsymbol{C}\ \boldsymbol{\Phi}\ \dot{\boldsymbol{q}}(t) + \boldsymbol{R}\ \left(\boldsymbol{\Phi}\ \boldsymbol{q}(t)\right) = \mathbf{P}(t) \tag{3.3}$$

A pre-multiplication with $\boldsymbol{\psi}^T$ (size $N \times m$) *yields:*

$$\begin{aligned} &\boldsymbol{\psi}^T\ \boldsymbol{M}\ \boldsymbol{\Phi}\ \ddot{\boldsymbol{q}}(t) + \boldsymbol{\psi}^T\ \boldsymbol{C}\ \boldsymbol{\Phi}\ \dot{\boldsymbol{q}}(t) + \boldsymbol{\psi}^T\ \boldsymbol{R}\left(\boldsymbol{\Phi}\ \boldsymbol{q}(t)\right) = \boldsymbol{\psi}^T\ \boldsymbol{P}(t) \\ &\boldsymbol{M}_r\ \ddot{\boldsymbol{q}}(t) + \boldsymbol{C}_r\ \dot{\boldsymbol{q}}(t) + \boldsymbol{R}_r\left(\boldsymbol{\Phi}\ \boldsymbol{q}(t)\right) = \boldsymbol{P}_r(t) \end{aligned} \tag{3.4}$$

- $\boldsymbol{\psi}$: the projection matrix of the trial space
- $\boldsymbol{\Phi}$: the projection matrix of the test space

The previous projection is called Galerkin projection when $\boldsymbol{\psi} = \boldsymbol{\Phi}$, otherwise called Petrov-Galerkin projections ($\boldsymbol{\psi} \neq \boldsymbol{\Phi}$). A Galerkin projection is suitable for the symmetric algebraic system. An algorithm (3.1) illustrates the use of a reduced order system for solving nonlinear structural dynamic with implicit time integration method (Newmark).

The projection approach is successfully applied in different real problems such as control (Bergmann et al. (2005)), structural dynamics (Amsallem et al. (2009) & Amabili et al.(2003)), aerodynamics (Epureanu(2003)),....*etc.*

Algorithm 3.1: Reduced order model

$$\ddot{\boldsymbol{u}} = \boldsymbol{M}^{-1}\left(\boldsymbol{P}(t) - \left(\boldsymbol{C}\,\dot{\boldsymbol{u}}(t) + \boldsymbol{R}\,\left(\boldsymbol{u}(t)\right)\right)\right), \quad i = 0,$$

$$\textbf{\textit{Time step}}\left(j = 1,\ldots\right)$$

$$\left.\begin{aligned} \ddot{\boldsymbol{u}}_i^j &= \ddot{\boldsymbol{u}}^j \\ \dot{\boldsymbol{u}}_i^j &= \dot{\boldsymbol{u}}^j + \Delta t\ddot{\boldsymbol{u}}^j \\ \boldsymbol{u}_i^j &= \boldsymbol{u}^j + \Delta t\dot{\boldsymbol{u}}^j + \frac{1}{2}\Delta t^2\Delta\boldsymbol{u}^j \end{aligned}\right\} \textbf{\textit{Predictor}}$$

$$repeat\,(iteration)$$

$$\boldsymbol{\Phi}^T\boldsymbol{K}_{eff}\left(\boldsymbol{u}_i^j\right)\boldsymbol{\Phi} = \sum_{e=1}^{E}\boldsymbol{\Phi}_e^T\left(\frac{1}{\Delta t^2\beta}\boldsymbol{M}_e + \frac{\gamma}{\Delta t\beta}\boldsymbol{C}_e + \boldsymbol{K}_{Te}\left(\boldsymbol{u}_i^j\right)\right)\boldsymbol{\Phi}_e$$

$$\left.\Delta u_{i+1}^j = \boldsymbol{\Phi}\left(\left[\boldsymbol{\Phi}^T\boldsymbol{K}_{eff}\left(\boldsymbol{u}_i^j\right)\boldsymbol{\Phi}\right]^{-1}\left(-\boldsymbol{\Phi}^T\,\boldsymbol{G}\left(\boldsymbol{u}_i^j\right)\right)\right)\right\} \textbf{\textit{Disp. increment}}$$

$$\left.\begin{aligned} \boldsymbol{u}_{i+1}^j &= \boldsymbol{u}_i^j + \Delta\boldsymbol{u}_{i+1}^j \\ \dot{\boldsymbol{u}}_{i+1}^j &= \dot{\boldsymbol{u}}_i^j + \frac{\gamma}{\Delta t\beta}\Delta\boldsymbol{u}_{i+1}^j \\ \ddot{\boldsymbol{u}}_{i+1}^j &= \ddot{\boldsymbol{u}}_i^j + \frac{1}{\Delta t^2\beta}\Delta\boldsymbol{u}_{i+1}^j \end{aligned}\right\} \textbf{\textit{Corrector}}$$

$$i = i + 1$$

$$until \quad \left\|\boldsymbol{G}(\boldsymbol{u}_i)\right\| \leq tol$$

$$j = j + 1$$

$$end$$

3.4 Computational complexity of dimensionality reduction

Based on (Krysl et al. (2001) and Spiess (2006)), the computational cost of finite element system can be seen in the Table 3.1, where N number of unknowns (DOF), and b (bandwidth of the system). It is clear that the most expensive operations are tangent stiffness assembly and the solution for the displacement increment. The form of the function s($N\ b^2$) depends on the solver used even with sparse tangent stiffness matrix. For a reduced order model, the computation is drastically decreased and it depends on the number of basis (m) of the subspace (Φ).

Table 3.1 computational cost

Operation	Full system	Reduced system
Predictor	$O(N)$	$O(m)$
Compute residual	$O(N)$	$O(m)$
Compute effective Tangent	$O(Nb)$	$O(m^2)$
Solve for Dis. increment	$O[\mathrm{s}(Nb^2)]$	$O(m^3)$
Configuration update	$O(N)$	$O(m)$

3.5 An error estimation of dimensionality reduction

The projection of the system onto a flat subspace ($\boldsymbol{\Phi}$) introduces an error ($\mathbf{e}$) between an original system $\boldsymbol{u}(t)$ and approximate system $\tilde{\boldsymbol{u}}(t)$as:

$$\boldsymbol{e}(t) = \boldsymbol{u}(t) - \tilde{\boldsymbol{u}}(t) \tag{3.5}$$

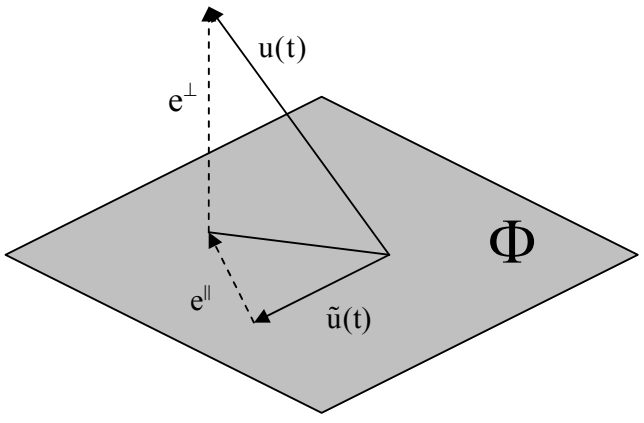


Figure 3.2: An error estimation of dimensionality reduction.

This error can be split into two components. The first component ($\boldsymbol{e}^{\perp}$)is orthogonal to subspace ($\boldsymbol{\Phi}$) and it's caused by the projection onto the low-dimension subspace. The second one ($\boldsymbol{e}^{\parallel}$) is caused by integration in the subspace ($\boldsymbol{\Phi}$) ((Meyer(2006)). This is shown in Figure 3.2.

$$\boldsymbol{e}\big(t\big) = \boldsymbol{e}^{\perp} + \boldsymbol{e}^{\parallel} \tag{3.6}$$

In the literature, there are some works on the error estimation such as calculating the error bounds for the Krylov subspace (Cabos (1994)), a selection of a Lanczos basis based on error estimation (Joo et al. (1989)). The quality of a subspace is the main issue for the quality of the reduced order model.

Actually, the goal of all works related to reduce order model based on projection is to find a projection that minimizes the error between two spaces (full space and projection subspace). Thus, a projection of a system to optimal subspace ($\boldsymbol{\Phi}$) is needed (Spiess (2006)).

3.6 Subspace basis

The best choice for a subspace basis from a numerical point of view is that the basis should be orthogonal and leads to a well-conditioned system of equation after projection (Eriksson et al. (1996)). Based on (Noor (1994)), the basis should also satisfy the following criteria:

- They should by linearly independent.
- Easy to generate with optimal number of basis.
- Gives high accuracy and approximation of the system of interest.

More practical requirements were introduced by (Spiess (2006)):

- The basis should ensure an orthogonality to get a unique projection ($\boldsymbol{\Phi}^T\ \boldsymbol{\Phi} = \boldsymbol{I}$).
- The basis should satisfy the essential boundary conditions (basis vector components equal to zero at the degrees of freedoms where displacements are prescribed). This can be done in practice by eliminating those components before computing the basis.

3.7 Reduced order model techniques

There are different techniques to reduce the model based on projection. In this work, the reduction techniques are divided into two groups. The first group is called dimensional reduction techniques such as modal basis reduction (Nickell (1976)), load-dependent Ritz method (Krysl(2001) & (Wilson et al. (1982)), Proper orthogonal decomposition POD (Sirovich (1987), Lenaerts et al.(2001), Kerschen et al. (2005)). The second group is called Spatial-complexity reduction techniques such as the missing point estimation (Astridet al.(2008)), (Vendl & Faβbender (2010, 2011), a priori hyper-reduction (Ryckelynck (2005), Kerfriden et al. (2011)), discrete empirical interpolation (Chaturantabut & Sorensen (2010)).

3.7.1 Dimensional reduction techniques

These techniques are responsible for calculating the basis of the projection subspace. Once the projection matrix $\boldsymbol{\Phi}$ is calculated, it is used in a Galerkin fashion to solve the incremental linear system in a reduced dimension as:

$$\boldsymbol{\delta u}_{i+1} = [\boldsymbol{\Phi}^T \boldsymbol{K}_{eff}\left(\boldsymbol{u}_i^j\right) \boldsymbol{\Phi}]^{-1} \left(-\boldsymbol{\Phi}^T \boldsymbol{G}\left(\boldsymbol{u}_i^j\right)\right) \tag{3.7}$$

Then, the solution of the full system will be:

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i + \boldsymbol{\Phi}\boldsymbol{\delta u}_{i+1} \tag{3.8}$$

3.7.1.1 Modal method

This method is also known as the modal superposition or modal analysis. It uses selected eigenmodes of the system as the basis of the subspace matrix ($\boldsymbol{\Phi}$). The main idea behind that is a few number of eigenmodes can give a good approximation of the system. So, the eigenvalue problem is solved at the beginning of each time step as equation(3.7).

$$(\boldsymbol{K}_T(\boldsymbol{u}) - \lambda \boldsymbol{I})\boldsymbol{V} = \boldsymbol{0} \tag{3.9}$$

The eigen vectors $\boldsymbol{v}_i$ corresponding to the smallest eigenvalues (more energy) are taken to form the basis of the projection matrix as in (3.10).

$$\boldsymbol{\Phi} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ...\boldsymbol{v}_m] \tag{3.10}$$

The extension of this approach was done by (Slaats et al.1995) who used a combination of modes with second order term. The disadvantage of this method is that a solution of eigenvalue problem is computationally expensive especially at each time step because the tangent matrix changes for nonlinear system. In this case, two problems should be solved simultaneously (equilibrium of the motion equation and eigenvalue problem).

3.7.1.2 Load dependent RITZ method

The Load Dependent RITZ method is introduced by (Wilson et al. (1982)). It is a method to calculate the subspace basis based on Ritz principle. There are many approaches to find the starting vector (first basis) while the rest of the basis can be calculated by ensuring an orthogonality condition (Gram-Schmidt). The simplest starting vector is a solution (vector of displacement ($\boldsymbol{u}$)) of first time step considered as a first basis after normalized as equation(3.11).

$$\overline{\boldsymbol{u}}_1 = \frac{\boldsymbol{u}_1}{\left\|\boldsymbol{u}_1\right\|} \tag{3.11}$$

Then, solutions of sequent steps are added to the projection matrix after being made orthogonal to the current base vectors by Gram-Schmidt. This means that all base vectors should be orthogonal as equation(3.12). The advantage of this method is to avoid the solution of the eigenvalue problem at each time step such as in the modal basis method.

$$\boldsymbol{\Phi} = \left(\overline{\boldsymbol{u}}_1 \perp \overline{\boldsymbol{u}}_2 \perp \overline{\boldsymbol{u}}_3 .....\right) \tag{3.12}$$

3.7.1.3 Proper Orthogonal Decomposition (POD)

It is one of the most widely used techniques to construct a reduced subspace, also called Karhunen-Loéve decomposition. It uses full problem solutions (snapshots) to analyze the redundancy level among the degrees of freedom, through singular value decomposition (SVD).

The advantage of this method is that it does not require a theory which describes the problem. This means that it does not require adaptation of algorithm for different sort of application. The disadvantage is that solving of a full problem must be done at least for a considerable part of the total steps (off-

line). Then, the on-line part is reduced by using the parameters (basis) which are previously acquired. POD is well known in different fields such as weather forecast (Shao et al. (2009)), biology (Price et al. (2003)), fluid dynamic ((Kunisch & Volkwein (2002), (Du et al. (2013)), and bio-mechanics (Reese & Radermacher et al. (2012)).

To find a subspace based on POD, the set of m snapshots obtained from the solution of the system at different instants (displacement vectors) is collected.

$$\boldsymbol{u} \;=\; \left(\boldsymbol{u}_1, \boldsymbol{u}_2, ...\boldsymbol{u}_m\right) \tag{3.13}$$

POD is an optimization tool which is used to find a set of orthogonal basis vectors $\left\{\varphi_i\right\}_{i=1}^{k}$ with $k \le m$ such that:

$$\arg\min_{\left\{\varphi_i\right\}_{i=1}^{N}} \sum_{j=0}^{m} \left\| \boldsymbol{u}_j - \sum\nolimits_{i=0}^{N} \left(\boldsymbol{u}_j^k . \varphi_i\right) \varphi_i \right\|_2^2 \tag{3.14}$$

The solution of this optimization problem is given by left singular vectors of snapshot matrix $\boldsymbol{u} = \left(\boldsymbol{u}_1, \boldsymbol{u}_2, ...\boldsymbol{u}_m\right)$. The left singular vectors are found from singular value decomposition (SVD) as $\boldsymbol{u} = \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Psi}^T$ where $\boldsymbol{\Sigma}$ is diagonal matrix called singular value ($\left\{\sigma_1, ...\sigma_r\right\}$ $\left\{\sigma_1 \ge .... \ge \sigma_r\right\}$), other matrices whose columns are orthogonal ($\boldsymbol{\Phi}$ & $\boldsymbol{\Psi}^T$) are (left and right singular vectors sequentially). The projection subspace matrix is first (k) vectors $\left\{\varphi_i\right\}_{i=1}^{k}$ of left singular matrix $\boldsymbol{\Phi}$ which satisfy equation (3.7) (Vendl & Faßbender (2010), Carlberg & Farhat (2009, 2011)). The value of (P) depends on user (accuracy requirement).

$$\varepsilon_{POD} = \left(\sum\nolimits_{i=1}^{k} \sigma_i \;/\; \sum\nolimits_{i=1}^{r} \sigma_i\right), \quad 1 - \varepsilon_{POD} \le P \tag{3.15}$$

A comparison between previous methods (modal basis reduction, load-dependent Ritz method, and Proper orthogonal decomposition) was done by (Spiess(2006), Radermacher & Reese (2013)). *From this comparison, the author decided to choose POD technique to construct the projection subspace.*

3.7.2 Spatial-complexity reduction techniques

Recently, several approaches have been proposed to deal with nonlinearities (most of them are empirical approaches) such as the empirical interpolation (Grepl et al. (2007), Gappy-POD (Vendl & Faßbender (2010), hyper-reduction techniques which evaluate a nonlinear system and their Jacobian at few locations (points) (Astrid et al. (2008), Vendl & Faßbender (2011), Ryckelynck (2005)).

But, most of these techniques used a Petrov-Galerkin projection which is convenient for this kind of approximation (Carlberg et al. (2011)). In this projection, left and right subspaces are not the same. Thus, equation (3.7) becomes (3.16).

$$\boldsymbol{\delta u}_{i+1} = [\boldsymbol{\psi}^T \boldsymbol{K}_{eff}\left(\boldsymbol{u}_i^j\right)\ \boldsymbol{\Phi}]^{-1}\ \left(\text{-}\boldsymbol{\psi}^T\ \ \boldsymbol{G}\left(\boldsymbol{u}_i^j\right)\right) \tag{3.16}$$

where;

$$\boldsymbol{\psi}^T = \boldsymbol{\Phi}^T\ \boldsymbol{D} \tag{3.17}$$

In this context, three methods for spatial complexity reduction will be mentioned (Discrete Empirical Interpolation, Hyper-Reduction, and Missing Point Estimation). The matrix ($\boldsymbol{D}$) will be found based on each technique with the emphases on Missing Point Estimation method.

3.7.2.1 Discrete Empirical Interpolation Method (DEIM)

It is a method to approximate nonlinear forces by a projection on subspace which has small dimension m<n. The subspace is spanned by bases $\boldsymbol{Y} = \left[\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots\ldots \boldsymbol{y}_m\right]$ which are obtained through POD by collecting snapshots for nonlinear forces ($\boldsymbol{R}$). This leads to:

$$\boldsymbol{R} \approx \boldsymbol{Y}c \tag{3.18}$$

The parameter (c) is a vector of unknown. To solve an over-determined system of equations(3.18), a pre-multiplying with a transpose of a Boolean matrix $\boldsymbol{P} = \left[\boldsymbol{e}_{\wp_1}, \ldots\ldots \boldsymbol{e}_{\wp_m}\right]$ such as:

$$\boldsymbol{P}^T\boldsymbol{R} = \boldsymbol{P}^T\boldsymbol{Y}c \tag{3.19}$$

This leads to:

$$c = (\boldsymbol{P}^T\boldsymbol{Y})^{-1}\boldsymbol{P}^T\boldsymbol{R} \tag{3.20}$$

Then, the approximation of the force becomes:

$$\boldsymbol{R} \approx \boldsymbol{Y}(\boldsymbol{P}^T\boldsymbol{Y})^{-1}\boldsymbol{P}^T\ \boldsymbol{R} \tag{3.21}$$

The advantage of multiplication with transpose of the Boolean matrix is to calculate only the specific entries which are selected. The selection in DEIM is done depending on the basis of subspace $\boldsymbol{Y} = \left[\boldsymbol{y}_1, \boldsymbol{y}_2, ...\boldsymbol{y}_m\right]$ and Interpolation indices $\left\{\wp_1, ...\wp_m\right\}$. The details of this method are presented in (Chaturantabut & Sorensen (2010)). Then, the ($\boldsymbol{D}$) matrix in equation (3.17) becomes:

$$\boldsymbol{D} = \boldsymbol{Y}(\boldsymbol{P}^T\boldsymbol{Y})^{-1}\boldsymbol{P}^T \tag{3.22}$$

3.7.2.2 A priori hyper-reduction

It is based on the selection of some equations (points) as control equations. These equations are selected based on a Boolean matrix $\boldsymbol{P} = \left[\boldsymbol{e}_{\wp_1}, ...\boldsymbol{e}_{\wp_m}\right]$ which satisfies a balance condition in equation(3.23).

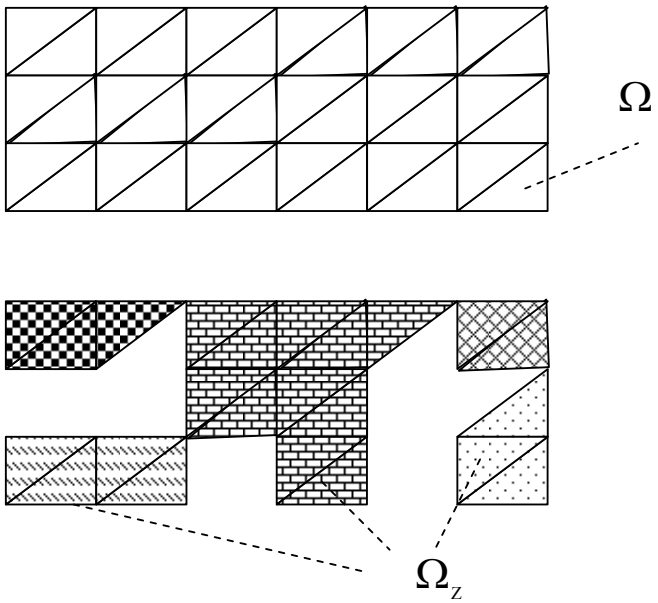


Figure 3.3: Illustration of the full ($\boldsymbol{\Omega}$) and reduced integration domain ($\boldsymbol{\Omega}_Z$).

$$\boldsymbol{\Phi}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{f}_{\text{int}} = \boldsymbol{\Phi}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{f}_{ext} \tag{3.23}$$

Thus, only few elements are connected to control nodes which are the nodes in the middle of the reduced integration domain ($\boldsymbol{\Omega}_Z$ RID(Figure 3.3)). The details of this method are presented in (Ryckelynck (2005) & Kerfriden et al.(2011)). Finally, a ($\boldsymbol{D}$) matrix in equation (3.17) is:

$$\boldsymbol{D} = \boldsymbol{P}\,\boldsymbol{P}^T \tag{3.24}$$

3.7.2.3 Missing point estimation (MPE)

This method is based on the Gappy-POD, which was introduced by (Everson & Sirovich (1995)). Gappy-POD or gappy-data is a method to approximate the data that lie in the gaps. If there is a vector (g) including some missing components, and also on the other hand the POD's subspace ($\Phi \in R^{n\times k}$) of this data is available. Then the approximation of the missing vector can be done by introducing a boolean matrix $P = (e_{j1}.....e_{j\tilde{n}}) \in R^{n\times\tilde{n}}$ whose columns contain the known components ($\tilde{n}$) and its rows presents all components (n).

$$\boldsymbol{g} = \begin{bmatrix} g_1 \\ g_2 \\ \cdots \\ g_4 \\ \cdots \\ g_6 \\ g_7 \\ g_8 \\ \cdots \\ g_{10} \end{bmatrix} \qquad \boldsymbol{g}_{org} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \\ g_6 \\ g_7 \\ g_8 \\ g_9 \\ g_{10} \end{bmatrix}$$

Thus, the missing vector and its missing subspace are respectively:

$$\boldsymbol{g} = \boldsymbol{P}^T\, \boldsymbol{g}_{org} \tag{3.25}$$

$$\tilde{\boldsymbol{\Phi}} = \boldsymbol{P}^T\, \boldsymbol{\Phi} \in R^{\tilde{n}\times k} \tag{3.26}$$

The approximation of ($\boldsymbol{g}_{org}$) is ($\tilde{\boldsymbol{g}}$)

$$\tilde{\boldsymbol{g}} = \boldsymbol{\Phi}\, a \tag{3.27}$$

Furthermore, the approximation of ($\boldsymbol{g}$) is ($\tilde{\boldsymbol{g}}_{mask}$).

$$\tilde{\boldsymbol{g}}_{mask} = \tilde{\boldsymbol{\Phi}}\, a \tag{3.28}$$

This problem will be solved by a least-square optimization to find (a):

$$\begin{aligned} &\min \left\| \tilde{\boldsymbol{g}}_{mask} - \boldsymbol{g} \right\|_n^2 \\ &\min_{a \in R^k} \left\| \tilde{\boldsymbol{\Phi}}\, a - \boldsymbol{g} \right\|_n^2 \end{aligned} \tag{3.29}$$

The solution of the least square optimization (linear regression problem) is given by a linear system:

$$\boldsymbol{K} a = \boldsymbol{R} \tag{3.30}$$

where

$$\boldsymbol{K} = \tilde{\boldsymbol{\Phi}}^T\, \tilde{\boldsymbol{\Phi}}, \qquad \boldsymbol{R} = \tilde{\boldsymbol{\Phi}}^T\, \boldsymbol{g}$$

Finally, the approximation ($\tilde{\boldsymbol{g}}$) can be found from equation(3.27).

Based on Gappy-Data, different criteria were introduced to achieve hyper-reduction such as the ones presented in (Vendl & Faßbender (2010), Astrid et al. (2008)) and others. Some of the criteria are based on experience of the system (empirical). In this work, the author selects the criterion which has fully mathematical meaning with modification to become more efficient. This criterion based on the orthogonality of a subspace is given as:

$$\tilde{\boldsymbol{\Phi}}^T\, \tilde{\boldsymbol{\Phi}} \approx \boldsymbol{I} \tag{3.31}$$

where;

$$\tilde{\boldsymbol{\Phi}} \approx \boldsymbol{P}^T\, \boldsymbol{\Phi} \tag{3.32}$$

To ensure the criterion(3.31), the condition number of ($\tilde{\boldsymbol{\Phi}}^T \times \tilde{\boldsymbol{\Phi}}$) should be close to one (tolerance) or fall into small tolerance.

$$\textit{Condition number}\ (\boldsymbol{\Phi}^T\ \boldsymbol{P}\,\boldsymbol{P}^T\ \boldsymbol{\Phi}) \leq tol \tag{3.33}$$

The author suggests that $\boldsymbol{P}\ \boldsymbol{P}^T$ is unit matrix when there are no missing points (components). Then, a sensitivity analysis is done by calculating the condition number for each missing point. This can be done by forcing a diagonal element in a unit matrix ($\boldsymbol{P}\ \boldsymbol{P}^T$) which belong to this point to be zero. Finally, all points with low impact are eliminated. A ($\boldsymbol{D}$) matrix in equation (3.17) is:

$$\boldsymbol{D} = \boldsymbol{P}\,\boldsymbol{P}^T \tag{3.34}$$

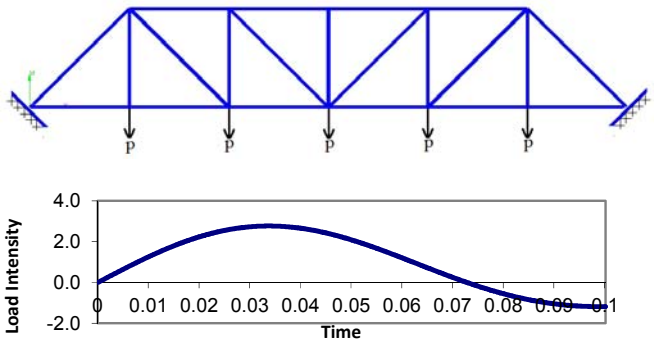
Chapter 4 Proposed Approaches of Reduced Order Model

4.1 Introduction

In this chapter, the different proposed approaches of reduced order model are presented with numerical results. It is organized to be three parts. First, simple numerical examples are introduced to address the necessity of an adaptive reduced order model. Second, the proposed adaptive strategy is introduced and later combined with two approaches of construct reduced basis (POD approach and PSS approach). Third, the adaptive hyper-reduction approach is introduced to increase both performance and efficiency based on POD. The later approach uses an adaptivity based on Dynamic-POD to avoid some problems of instability.

4.2 Reduced order model and the necessity of an adaptivity

Reduced order modeling of the nonlinear structural mechanics based on projection is implemented on a simple system such as Truss (restriction in the z direction) in Figure 4.1. The nonlinearity comes from geometrical behavior.

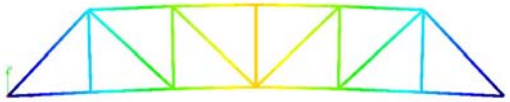
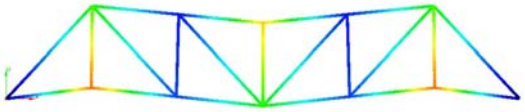
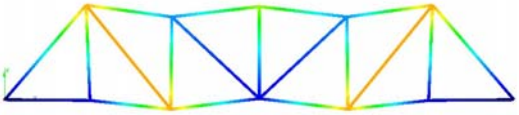
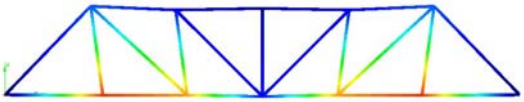


The properties of Truss system :	
E(Young's modulus)=2e11Pa	ζ (Damping ratio) = 0.01,
A(Cross section)=31.3e-4 m^2	Δt=0.0001
ρ=(Density)=7850 kg/m^3	T (Simulation time) =0.1s
L (Length of horizontal members)=7.5m,	β (Newmark's parameter)=0.25
P (Load) = -14e5N	γ(Newmark's parameter)=0.5

Figure 4.1: Truss for dynamic analysis.

The first five modes obtained from POD analysis are presented in Table 4.1 along with the (ε_{POD}) which indicates a significance of each mode based on the proper orthogonal decomposition. By using five snapshots only, one response frequency can be capture depending on the time step size of Newark scheme.

Table 4.1: First five modes of Truss

ε_{POD} (%)	$\sum \varepsilon_{POD}$ (%)	Mode (POD-Basis)
96.6382	96.6382	
2.4499	99.0881	
0.3543	99.4424	
0.2042	99.6466	
0.1632	99.8098	

In fact, the subspace constructed by the POD contains the most significant modes. Nevertheless, no reduced order base can approximate a system without some error which have been investigated in the truss system. Figure 4.2 shows that the error decreases with the number of POD-basis, but on other hand, adding new vectors to the base leads to increase the computational time. The truss system is considered as an ideal system for reduced order model because the mode does not change (the movement is in one direction).

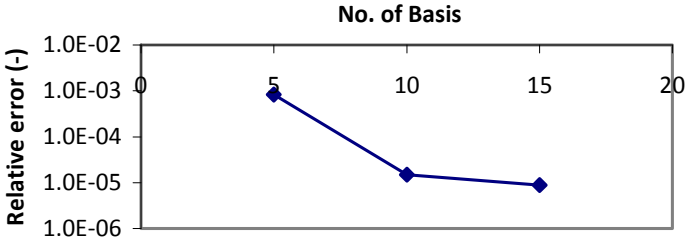
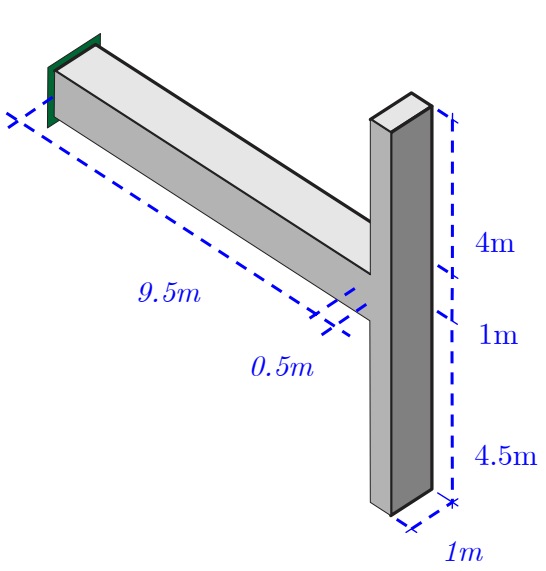
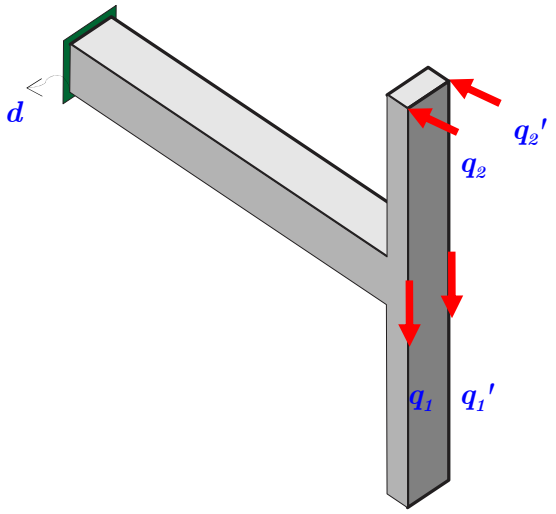


Figure 4.2: Relations between POD basis and a relative error.





The properties of T-Beam system :

E(Young's modulus)=0.5e9Pa	β (Newmark parameter)=0.25
v(Poisson's ratio)=0.45	γ(Newmark parameter)=0.5
ρ(Density)=1600 kg/m^3	q_1 (Load) = 5e6N
ζ (Damping ratio) = 0.01,	q_2 (Load) = 5e6N
Δt=0.005s	d(Fixed support)=restriction for all directions
T (Simulation time) =2s	

Figure 4.3: T-Beam model.

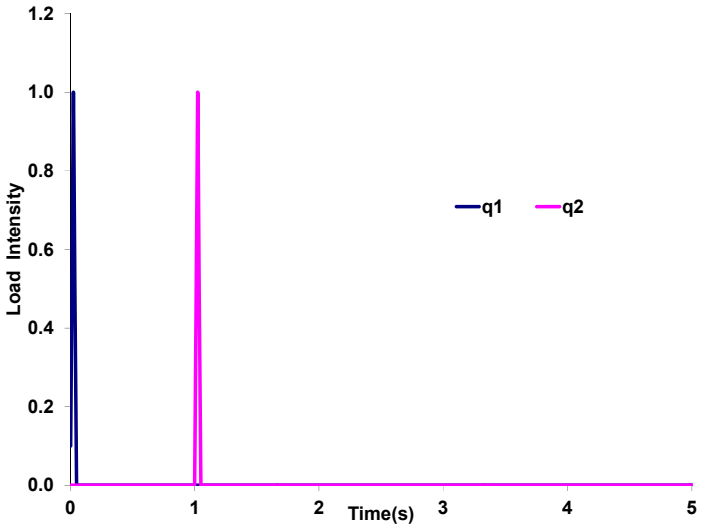
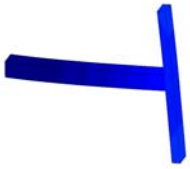
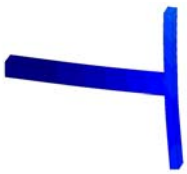
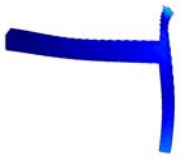


Figure 4.4: load intensity vs. time for T-Beam model

In reality, more complex structures have a different behavior as POD modes are configuration dependent and thus change throughout the simulation. An example of such structures can be seen in Figure 4.3, which depicts a T-beam of nonlinear Neo-Hookean hyper-elastic material. Furthermore, the load intensity with the load can be found in Figure 4.4

Table 4.2: First ten modes of T-Beam

ε_{POD} (%)	$\sum \varepsilon_{POD}$ (%)	Mode (POD-Basis)	ε_{POD} (%)	$\sum \varepsilon_{POD}$ (%)	Mode (POD-Basis)
60.01	60.01		2.48	94.34	
14.04	74.05		1.85	96.19	
7.80	81.85		1.16	97.35	
5.90	87.75		0.58	97.93	
4.11	91.86		0.30	98.23	

The POD modes can be found in Table 4.2. The computing of POD-modes for this structure using snapshots over all time domain leads to an average subspace. While, proper orthogonal decomposition is based on least square optimization. This leads to that a subspace is not able to reflect the approximation of the system at different configurations. For this reason, an adaptive strategy is introduced in the next section.

4.3 An adaptive reduced order model

4.3.1 An adaptive strategy

An adaptive strategy is introduced to perform corrections on the reduced space during the process. The adaptivity is independent from the method of constructing the subspace (Φ). The overview of the system is illustrated in Figure 4.5. The full system is solved for a certain number of time steps (5 steps) then the projection subspace is constructed based on the snapshots (point A). Second, the adaptive reduced order model is performed for the rest of simulation. In fact, the adaptivity leads to increase subspace dimension. For that reason, a new subspace should be constructed when the subspace dimension reaches a certain size ($n_{\Phi_{stable}}$) as in the point (B).

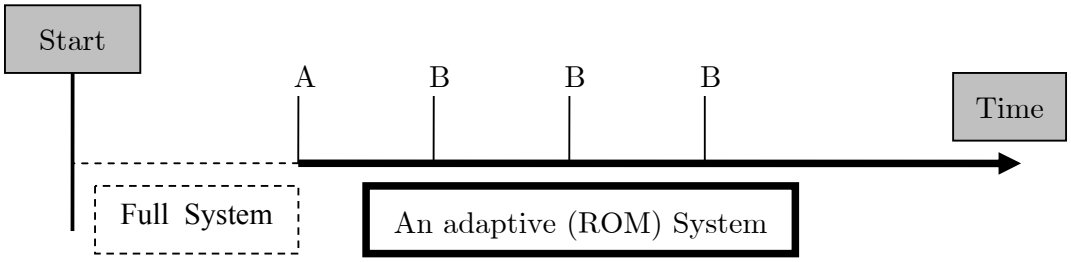


Figure 4.5: An overview of the system.

- A: the first subspace ($\Phi_{increment}$) is computed based on all snapshots of processing time which are converged solutions of full system.
- (A-B or B-B): the adaptivity is done when it is necessary (see notice).
- B: A subspace ($\Phi_{increment}$) reaches a maximum size ($n_{\Phi_{stable}}$). This is due to an adaptivity which leads to increase the subspace. For that, a new subspace should be computed based on (POD or Proper snapshots selection (PSS)) approach.

Notice: *In general, there are two positions to check the requirement for adaptivity. The first one is inside Newton iteration (section 4.3.1.1) and the second one is at the end of each time step (section (4.3.1.2).*

4.3.1.1 An iterative adaptivity-Error analysis

The adaptivity inside iteration is called ***iterative adaptivity-Error analysis***. It is necessary when the error between full and reduced system is high. This error can be checked from the convergence criteria ((4.1), (4.2), and (4.3)) and other criteria ((4.4), (4.5)) as:

$$error(\boldsymbol{G}) = \frac{\left\|\boldsymbol{G}\right\|}{\left\|\boldsymbol{G}_0\right\|}, \quad error(\boldsymbol{G}) \le tol \qquad where \qquad \boldsymbol{G}_0 \in \mathbb{R}^n \tag{4.1}$$

$$error(\boldsymbol{G}_R) = \frac{\left\|\boldsymbol{G}_R\right\|}{\left\|\boldsymbol{G}_{R0}\right\|}, \quad error(\boldsymbol{G}_R) \le tol_R \tag{4.2}$$

$$where \quad \boldsymbol{G}_{R0} = \boldsymbol{\Phi}^T \boldsymbol{G}_0 \quad and \quad G_{R0} \in \mathbb{R}^{n_\Phi}$$

$$error\left(\boldsymbol{u}\right) = \frac{\left\|\Delta\boldsymbol{u}\right\|}{\left\|\boldsymbol{u}_0\right\|}, \quad error\left(\boldsymbol{u}\right) \le tol_u \qquad where \qquad \Delta\boldsymbol{u}, \boldsymbol{u}_0 \in \mathbb{R}^n \tag{4.3}$$

where $\boldsymbol{G}$ is the residual of the full space, $\boldsymbol{G}_R$ is the residual of the reduced space, $\boldsymbol{u}$ is a displacement.

$$ratio\left(\boldsymbol{G}_R \,/\, \boldsymbol{G}\right) = \frac{error\left(\boldsymbol{G}_R\right)}{error\left(\boldsymbol{G}\right)}, \quad ratio\left(\boldsymbol{G}_R \,/\, \boldsymbol{G}\right) \le k_\Phi \quad and \quad 0 < k_\Phi \le 1 \tag{4.4}$$

$$ratio\left(\boldsymbol{G} \,/\, \boldsymbol{G}_{old}\right) = \frac{error\left(\boldsymbol{G}\right)}{error\left(\boldsymbol{G}_{old}\right)}, \quad ratio\left(\boldsymbol{G} \,/\, \boldsymbol{G}_{old}\right) \le k_r \quad and \quad 0 < k_r \le 1 \tag{4.5}$$

The adaptivity is necessary, if one of the previous criteria is not satisfing after certain number of iterations (2 iterations). Then, the adaptivity will be done by adding an error between a full and a reduced space to a projection subspace (Φ) by using Gram-Schmidt. The main idea is to split the solution of the full space into two complementary subspaces as:

$$\boldsymbol{u} = \boldsymbol{u}_\Phi + \boldsymbol{u}_K \quad where \quad \boldsymbol{u}_\Phi,\ \boldsymbol{u}_K \ \in \ \mathbb{R}^{\,n} \tag{4.6}$$

where $\boldsymbol{u}$ *(solution of a full space),* $\boldsymbol{u}_\Phi$ *(solution of reduced space), and* $\boldsymbol{u}_K$ *(error)*

$$\boldsymbol{u}_K = \boldsymbol{u} - \boldsymbol{u}_\Phi = \boldsymbol{K}_{eff}^{-1}\boldsymbol{G} - (\boldsymbol{\Phi}(\boldsymbol{K}_{eff\,R}^{-1}\boldsymbol{G}_R)) \tag{4.7}$$

Then, this error should be added to a subspace as in Figure 4.6.

4.3.1.2 An incremental adaptivity

The adaptivity at the end of the time step called *an incremental adaptivity*. It is done based on the information which is obtained from iterations of previous step. This information helps to assess and improve $\Phi_{\text{increment}}$ for the next time step. The incremental adaptivity is done based on three cases:

1- If the number of corrections inside iteration $\text{n}_{\text{iteration}}$ is a greater than a certain number ($\text{n}_{\text{k}} = 2$) then:

- The $\Phi_{\text{increment}}$ is increased by adding one basis. This basis will be the orthogonal of the converged solution ($U^{\perp}$).

Time step 1, 2……

$\Phi = \left[\Phi_{\text{increment}} \right]$

Iterations 1,2,3,.....

If criteria (4.1, 4.2, 4.3, 4.4,and 4.5) are not satisfied then:

- compute $\boldsymbol{u}_k$ & make it orthogonal $\boldsymbol{u}_k^{\perp}$ with all bases in Φ and added to Φ as

$$\begin{bmatrix} \Phi_{increment(1)\ldots} & \Phi_{increment(\text{n})} & \Phi_{k\,(1)\ldots}^{\perp} & \Phi_{k\,(n)}^{\perp} \\ \Phi_{increment(1)\ldots} & \Phi_{increment(\text{n})} & \Phi_{k\,(1)\ldots}^{\perp} & \Phi_{k\,(n)}^{\perp} \\ \Phi_{increment(1)\ldots} & \Phi_{increment(\text{n})} & \Phi_{k\,(1)\ldots}^{\perp} & \Phi_{k\,(n)}^{\perp} \\ \Phi_{increment(1)\ldots} & \Phi_{increment(\text{n})} & \Phi_{k\,(1)\ldots}^{\perp} & \Phi_{k\,(n)}^{\perp} \end{bmatrix}$$

End

- Reject all $\Phi_k^{\perp}$ columns from Φ
- niteration = Number of $\Phi_k^{\perp}$ columns
- U= a converged solution

$\Phi = \left[\Phi_{\text{increment}} \right]$ check the necessity of an incremental adaptivity (section 4.3.1.2)

End

Figure 4.6: Illustration of an adaptivity inside the iteration's loop.

2- If the number of corrections inside iteration $\text{n}_{\text{iteration}}$ is equal or less than a certain number ($\text{n}_{\text{k}} = 2$) then:

- The $\Phi_{\text{increment}}$ is enriched without adding any basis vector (replace the last vector with the orthogonal of the converged solution ($U^{\perp}$)).

3 - If the step 2 was repeated sequentially over a certain number of steps (n_{stable}=25 or if the dimension of $\Phi_{increment}$ reaches a certain size n_{max} = 50).Then:

- The $\Phi_{increment}$ is reconstructed.

4.3.2 An adaptive reduced order model based on POD & BFGS method

The previous adaptive strategy was combined with BFGS method to avoid computing the tangent stiffness matrix several times. Some extra criteria were added to avoid a precision problem with the BFGS formula when the displacement or residual vectors reach small values (close to zero) in the denominator. This would lead to high component values of the new secant matrix. The BFGS formula in reduced space will be:

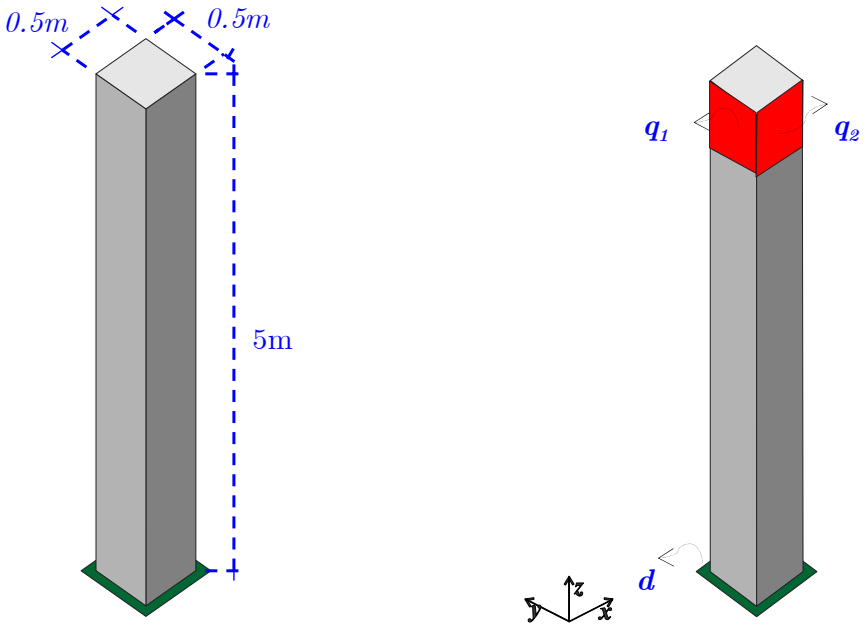
$$\begin{aligned}
&\delta \boldsymbol{u} = \boldsymbol{u}_{R_i} - \boldsymbol{u}_{R_{i-1}} \\
&\Delta \boldsymbol{G}_R = \boldsymbol{G}_R\left(\boldsymbol{u}_{R_{i-1}}\right) - \boldsymbol{G}_R\left(\boldsymbol{u}_{R_i}\right) \\
&\Delta \boldsymbol{K}_{eff\,R_{\Delta k}} = \Delta \boldsymbol{K}_{eff\,R}\left(\boldsymbol{u}_{R_{i-1}}\right)\delta \boldsymbol{u} \\
&\Delta \boldsymbol{K}_{eff\,R}\left(\boldsymbol{u}_i\right) = \Delta \boldsymbol{K}_{eff\,R}\left(\boldsymbol{u}_{R_{i-1}}\right) + \frac{\Delta \boldsymbol{G}_R\,\Delta \boldsymbol{G}_R^T}{\Delta \boldsymbol{G}_R^T\,\delta \boldsymbol{u}} - \frac{\Delta \boldsymbol{K}_{eff\,R_{\Delta k}}\,\Delta \boldsymbol{K}_{eff\,R_{\Delta k}}^T}{\Delta \boldsymbol{K}_{eff\,R_{\Delta k}}^T\,\delta \boldsymbol{u}}
\end{aligned} \tag{4.8}$$

To ensure a correct update of the tangent matrix, the following conditions are applied:

$$error\left(\boldsymbol{G}_R\right) \le tol_{BFGS} \quad \text{and} \quad error\left(\boldsymbol{u}\right) \le tol_{BFGS} \tag{4.9}$$

$$\Delta \boldsymbol{G}_R^T \Delta \delta \boldsymbol{u} > 0 \quad \text{and} \quad \Delta \boldsymbol{K}_{eff\,R_{\Delta k}}^T \delta \boldsymbol{u} > 0 \tag{4.10}$$

Conditions (4.9) represent the relative error in the reduced residual vector and the displacement vector. These conditions are more stable than a direct denominator analysis. Furthermore they are non dimensional conditions, which avoid a tolerance change due to problem dimension. Conditions (4.10) are used to keep the new secant matrix positive definite.



The properties of column system :

E(Young's modulus)=0.5e9Pa	β (Newmark parameter)=0.25
v(Poisson's ratio)=0.45	γ(Newmark parameter)=0.5
ρ(Density)=1600 kg/m^3	q_1 (Load) = 3.63e6N (X-direction)
ζ (Damping ratio) = 0.01,	q_2 (Load) = 3.63e6N (Y-direction)
Δt=0.005s	d(Fixed support)=restriction for all
T (Simulation time) =5s	directions

Figure 4.7: Column model

To study the convergence and accuracy of this approach for reducing the computational effort and time of the system, two models are simulated. The first model is a column of Neo-Hookean material as seen in Figure 4.7 which is discretized by a mesh with 12302 nodes and 36906 degrees of freedom. The different loads intensitis with the time are presented in Figure 4.8.

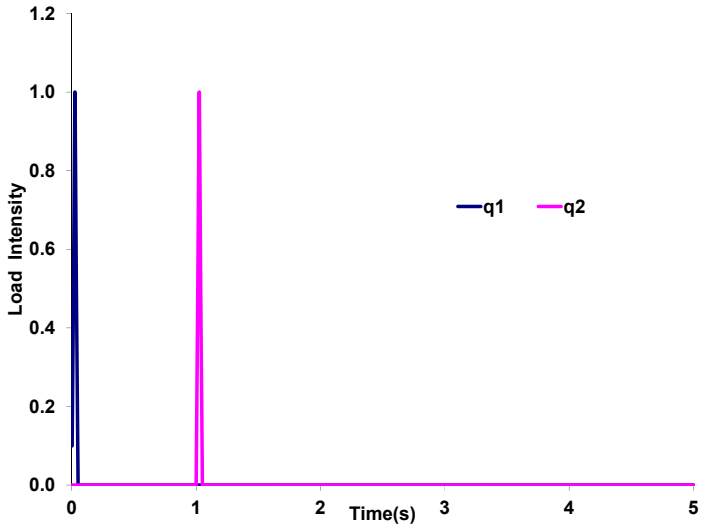


Figure 4.8: load intensity vs. time for column model

(a)

(b)

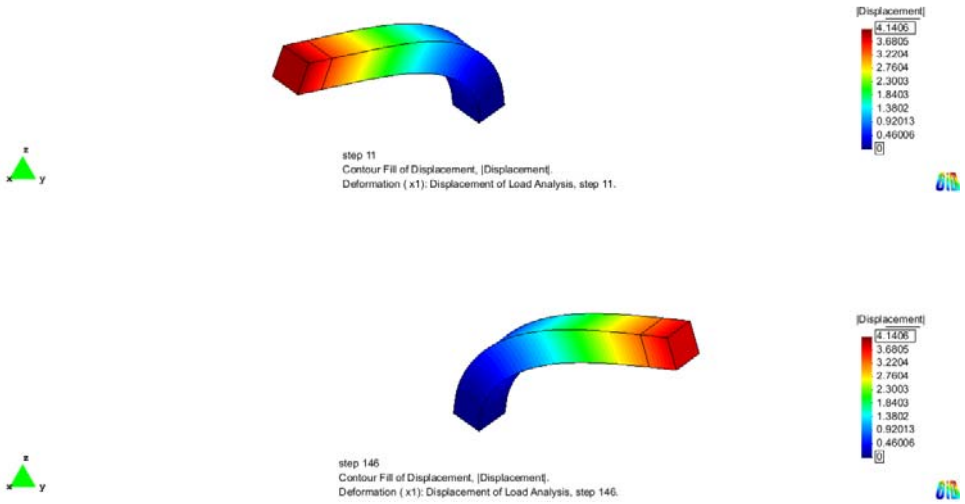
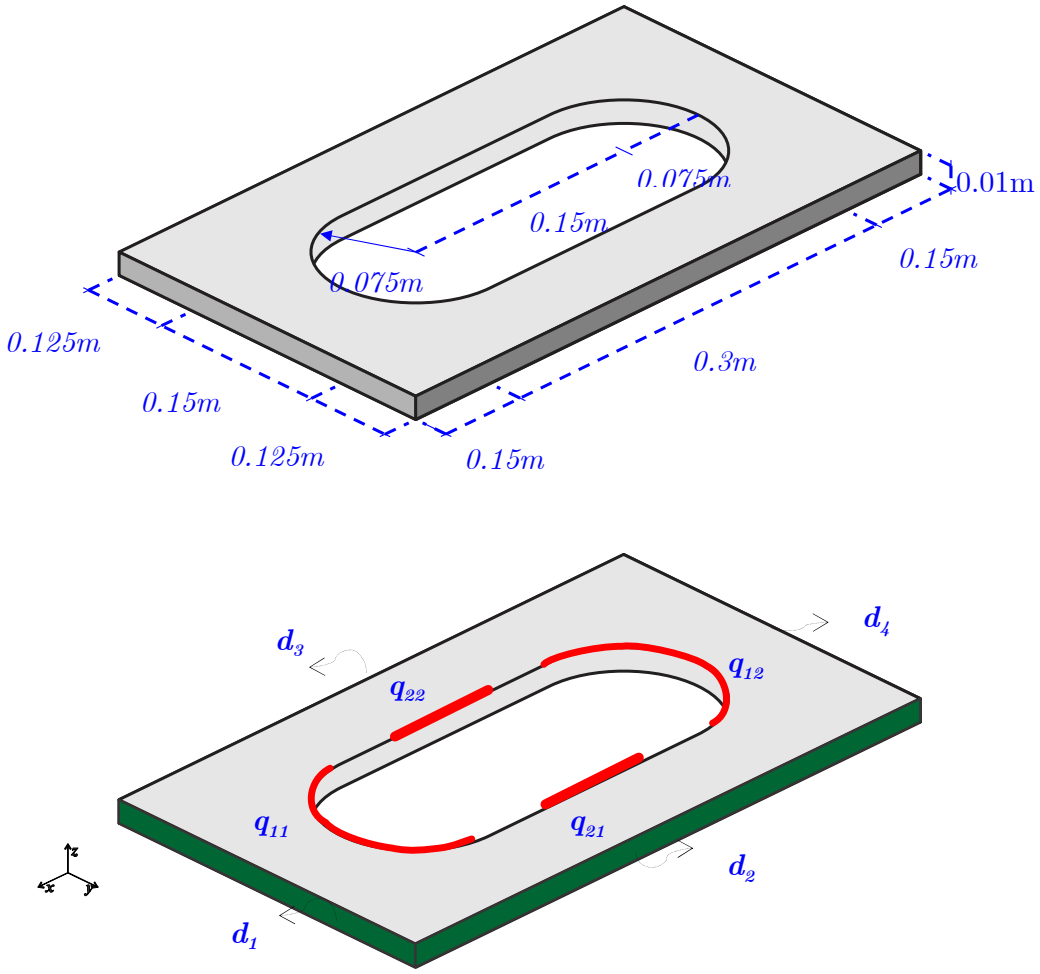


Figure 4.9: Column model (a): Displacements (m) at t=0.0825s. (b): Displacements (m) at t=1.0950s.

The second one is a plate of Neo-Hookean material as seen in Figure 4.10 which is discretized by a mesh with 25536 nodes and 76608 degrees of freedom. The different loads intensity with the time is presented in Figure 4.11. The comparison is made for each model among solving the full system and reduced order system.



The properties of plate system :

E(Young's modulus)=0.5e9Pa	q_{11} (Load) = -9.8e6N (X-direction & Z-direction)
v(Poisson's ratio)=0.45	q_{12} (Load) = 9.8e6N (Y-direction)
ρ(Density)=1600 kg/m^3	q_{12} (Load) = -9.8e6N (Z-direction)
ζ (Damping ratio) = 0.01,	q_{21} (Load) = -9.0e6N (Y-direction & Z-direction)
Δt=0.005s	q_{22} (Load) = 9.0e6N (Y-direction)
T (Simulation time) =5s	q_{22} (Load) = -9.0e6N (Z-direction)
β(Newmark parameter)=0.25	d_1(Fixed support)=restriction for directions (Y&Z)
γ(Newmark parameter)=0.5	d_2(Fixed support)=restriction for all directions(X&Z)
	d_3(Fixed support)=restriction for all direction(Z)
	d_4(Fixed support)=restriction for all direction(Z)

Figure 4.10: Plate model

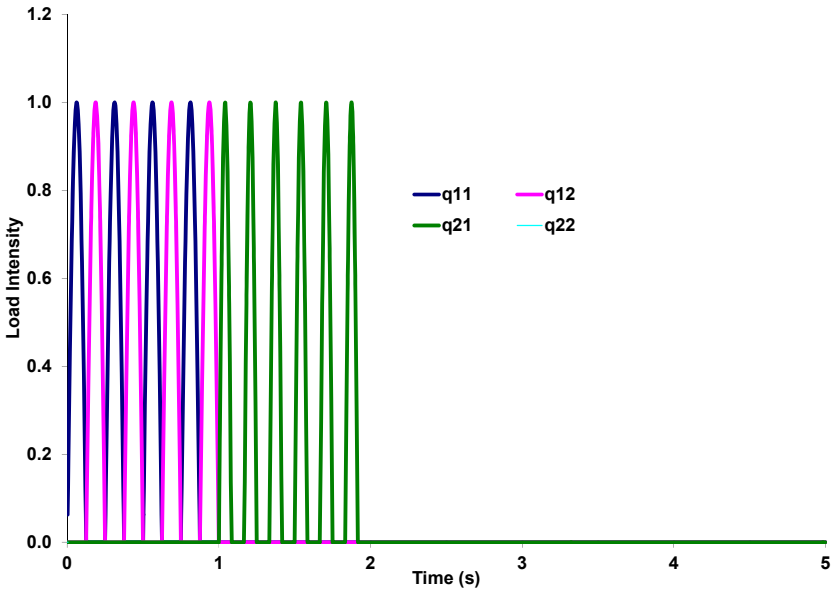


Figure 4.11: load intensity vs. time for Plate model

(a)

(b)

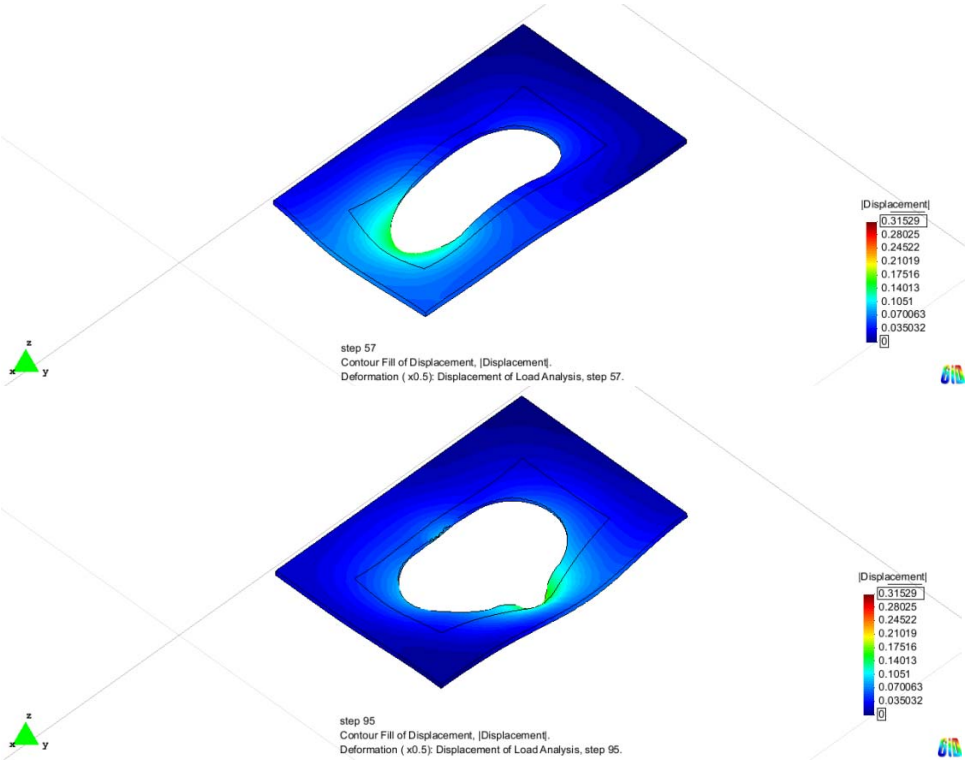


Figure 4.12: Plate model (a): Displacements (m) at t=0.8550s. (b): Displacements (m) at t=1.4250s.

The reduced system is obtained by Galerkin projection with and without reduced BFGS. Furthermore, the convergence of all systems is shown below.

Tables 4.3 and 4.4 present the processing time, iterations and corrections related to different residual vector tolerances, the POD times and reduced BFGS updates.

Table 4.3.a: An overview of the column model results (POD)

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance - tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	7412	18231	14976	14334
No. Corrections	-	3816	2602	2316
No. POD	-	5	3	1
Total Time(s)	30730.84	17717.83	11874.87	10899.09
Time per step (s)	15.36	8.85	5.93	5.446165
Total Time (%)	100.00	57.65	38.64	35.47

Table 4.3.b: An overview of the column model results (BFGS)

	Full	**Reduced BFGS**	**Reduced BFGS**	**Reduced BFGS**
Tolerance - tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	7412	17784	14724	14534
No. Corrections	-	3816	2601	2298
No. POD	-	5	3	0
No. BFGS	-	7996	8017	8089
Total Time(s)	30730.84	17653.48	11865.32	10853.64
Time per step (s)	15.36	8.823316	5.929536	5.423401
Total Time (%)	100.00	57.45	38.61	35.32

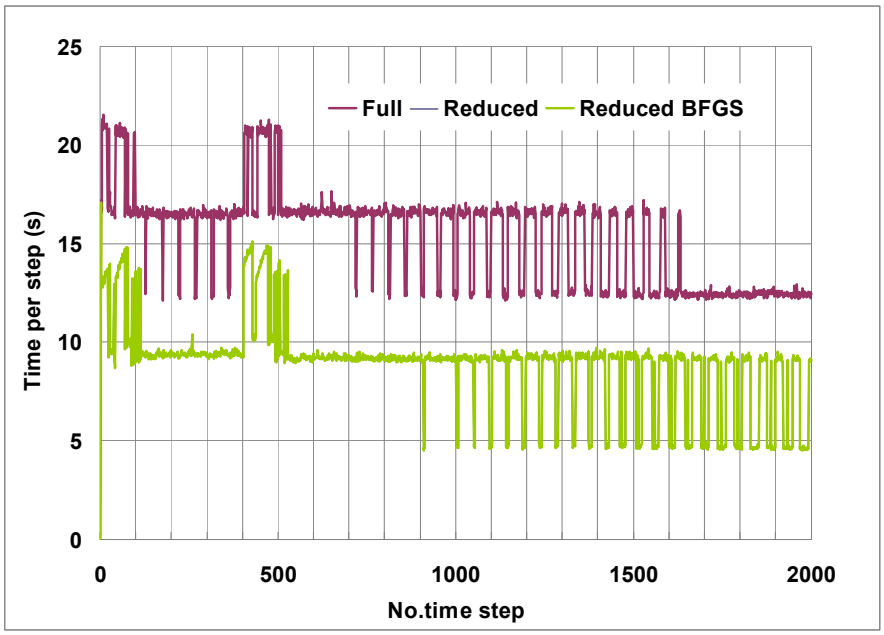
Table 4.4a: An overview of the Plate model results (POD)

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance - tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	6627	16502	16158	13710
No. Corrections	-	4052	3929	2609
No. POD	-	4	1	0
Total Time(s)	99051.08	66350.37	65442.26	43719.16
Time per step (s)	49.52	33.16	32.71	21.85
Total Time (%)	100	66.99	66.07	44.14

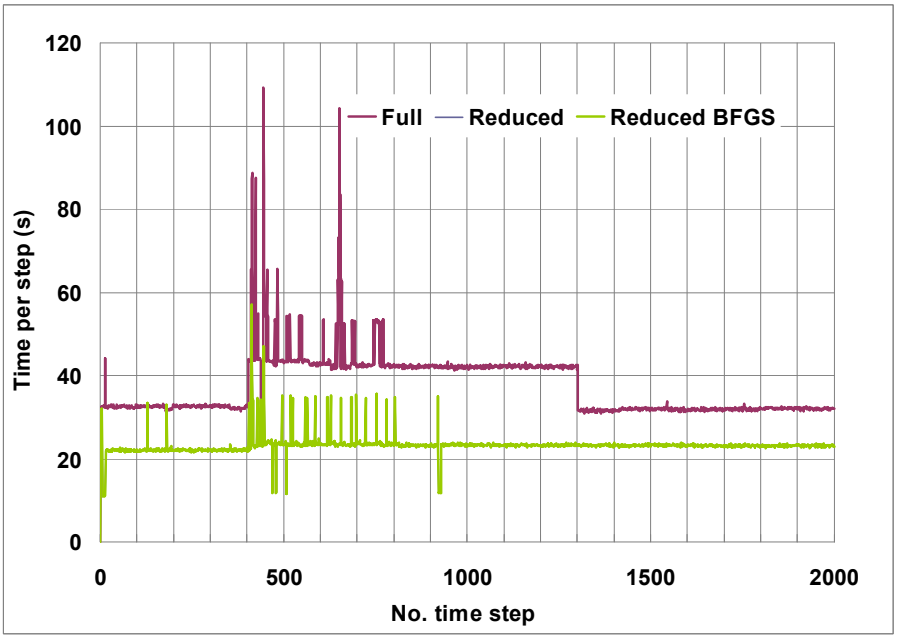
Table 4.4.b.: An overview of the Plate model results (BFGS)

	Full	**Reduced BFGS**	**Reduced BFGS**	**Reduced BFGS**
Tolerance - tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	6627	16416	16028	13569
No. Corrections	-	4052	3929	2642
No. POD	-	4	1	0
No. BFGS	-	6540	6917	6598
Total Time(s)	99051.08	65898.83	65157.22	43728.29
Time per step (s)	49.52	33.09	32.57	21.85
Total Time (%)	100	66.53	65.78	44.15

In both models (column and plate), the processing time of each time step in the case of the full system is greater than the reduced system as seen in Figure 4.13. The dimension of the subspace changes during the computational process as seen in Figure 4.14. This is due to the change of implicit modes in snapshots. In the same way, the number of corrections related to change the configuration is given in Figure 4.15. Therefore, the correction increases when the configuration changes significantly in a short period due to the large displacements.

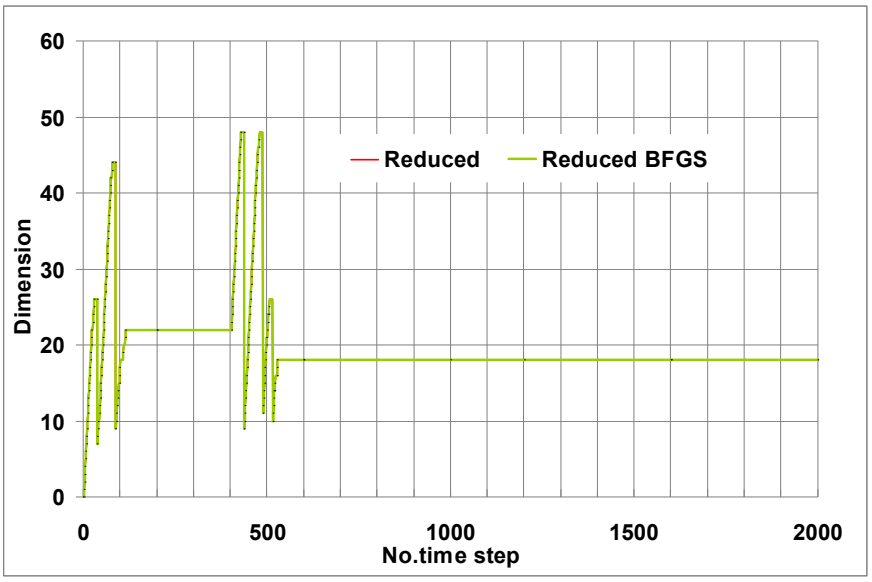


(a)

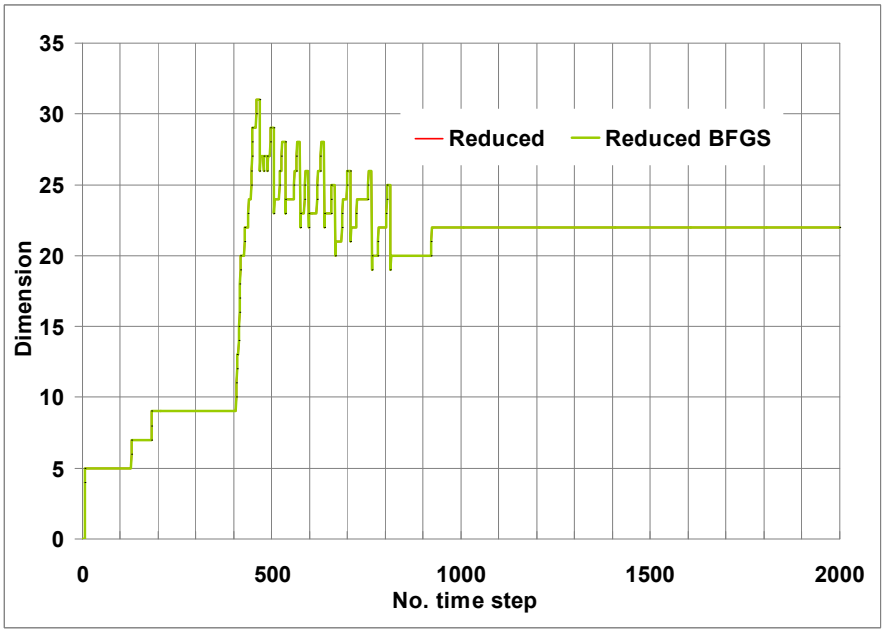


(b)

Figure 4.13: Time per step vs. number of time step (a): column model. (b): plate model.



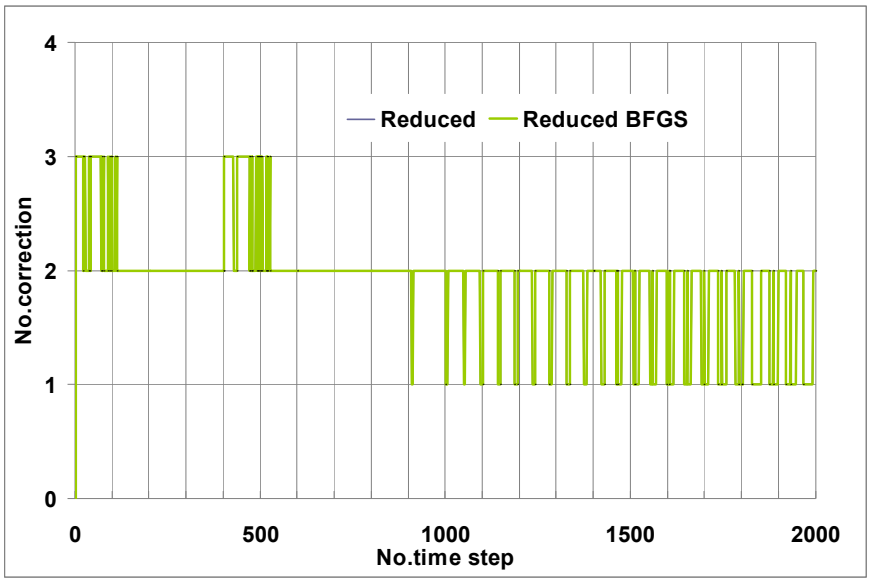
(a)



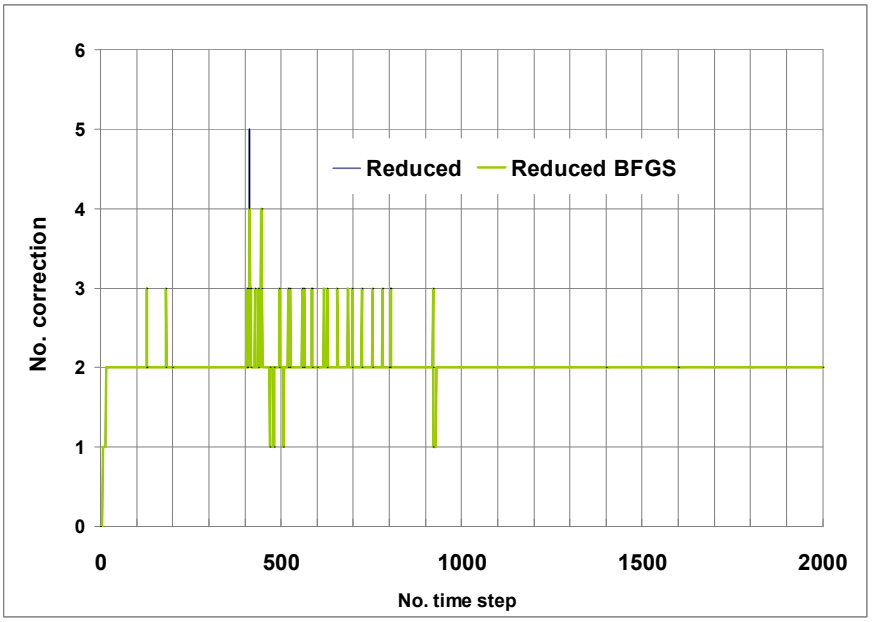
(b)

Figure 4.14: Dimension of subspace vs. number of time step
(a): column model. (b): plate model.

The difference of the convergence between the full and the reduced system can be noticed in Figure 4.16, in which the full system has quadratic convergence in the most of time (except when loads are applied). The reduced system has super-linear convergence due to projection.



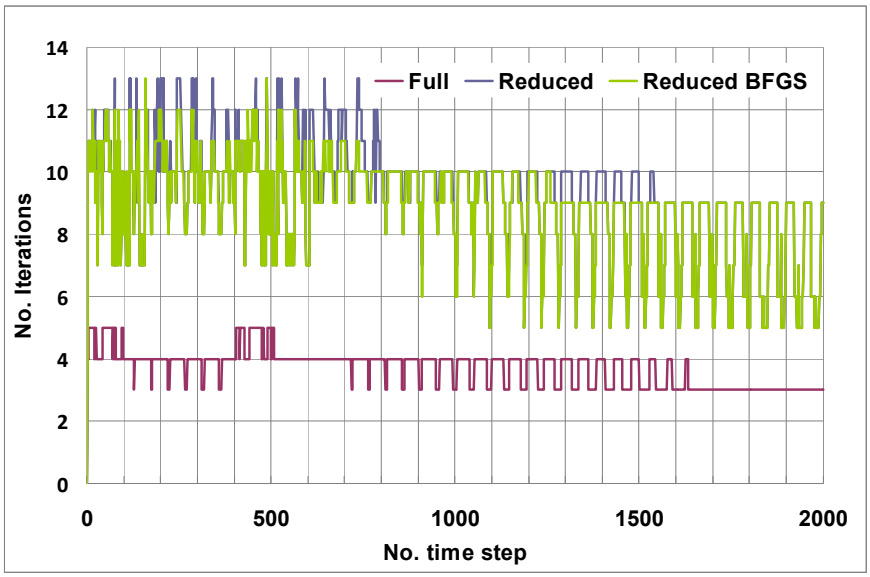
(a)



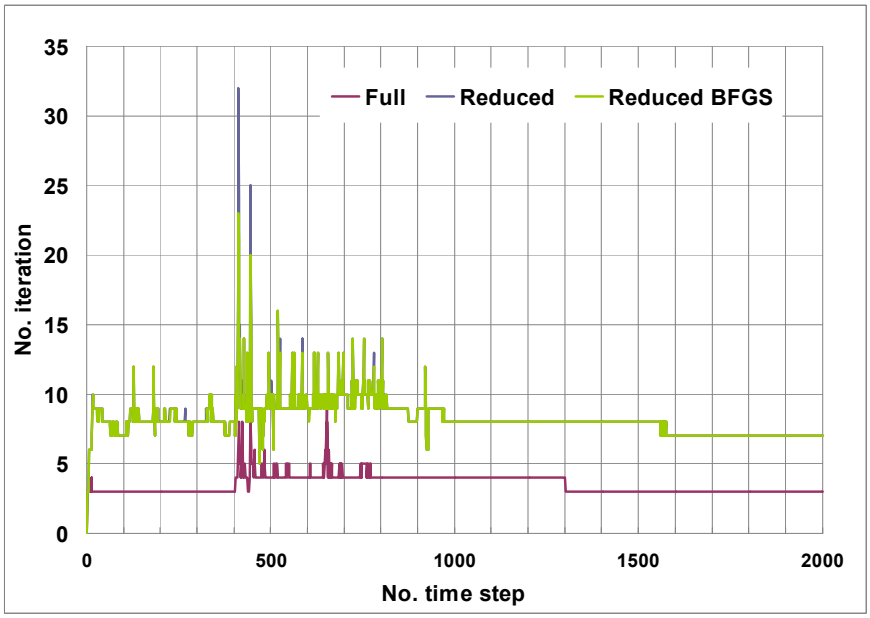
(b)

Figure 4.15: Number of corrections vs. number of time step
(a): column model. (b): plate model.

The reduced BFGS only has some meaningful effect in high displacements, when the nonlinearity is extremely strong tending to linear convergence. In general appearance, the reduced system based on projection leads to increase the number of iterations but on the other hand it is still very fast compared with the full system (Figure 4.13 and Figure 4.16).



(a)



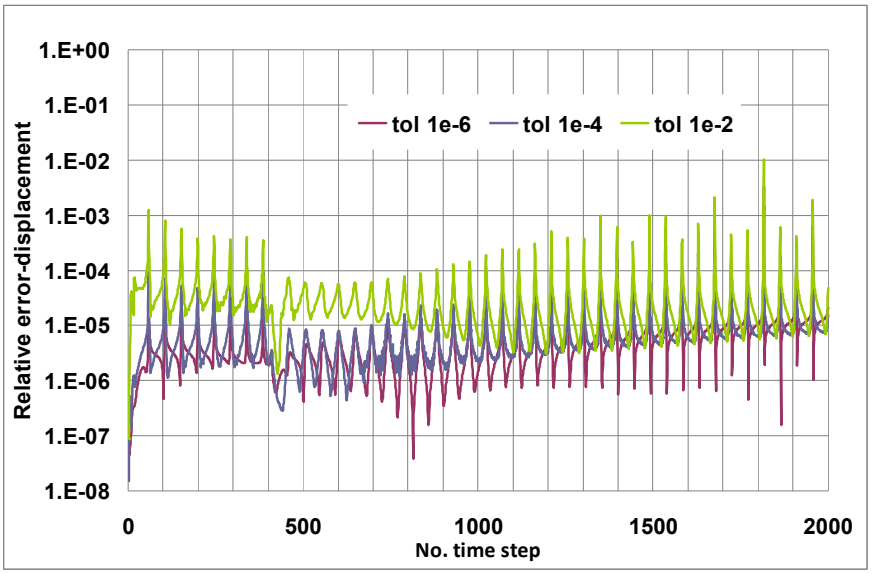
(b)

Figure 4.16: Iteration vs. number of time step
(a): column model. (b): plate model.

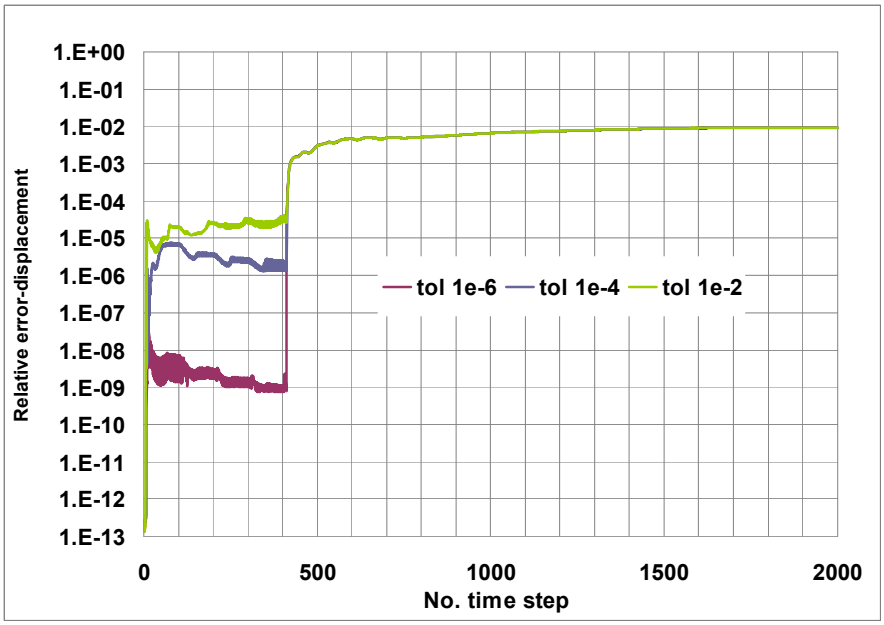
To measure the error, this work defines the relative error of the displacements for each time step according to the equation(4.11):

$$Relative\ error = \frac{\left\| \boldsymbol{u}_{\mathrm{Re}duced} - \boldsymbol{u}_{Full} \right\|}{\left\| \boldsymbol{u}_{Full} \right\|} \qquad where \quad \boldsymbol{u}_{\mathrm{Re}duced}, \boldsymbol{u}_{Full} \in \mathbb{R}^n \qquad (4.11)$$

where $\boldsymbol{u}_{\mathrm{Re}duced}$ represents the displacement vector obtained from the reduced model and $\boldsymbol{u}_{Full}$ the displacement vector obtained from the full model.



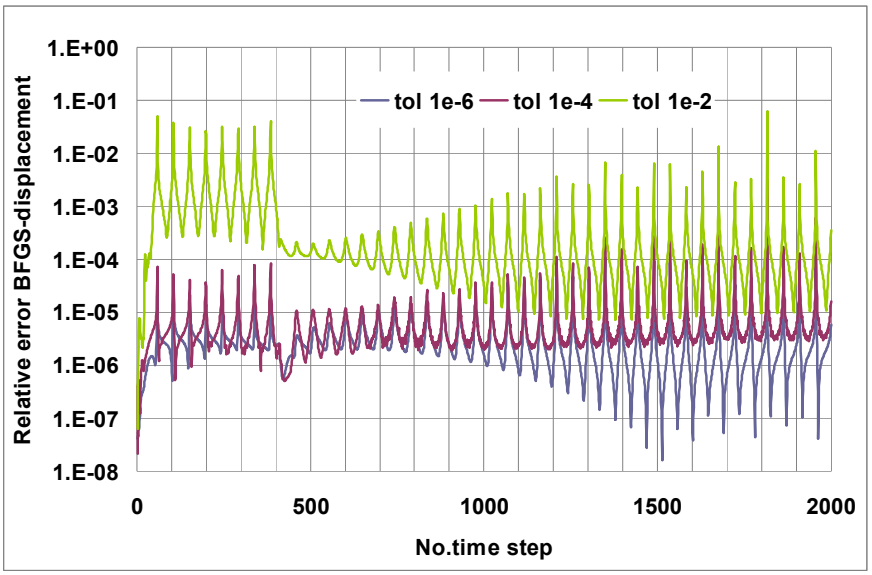
(a)



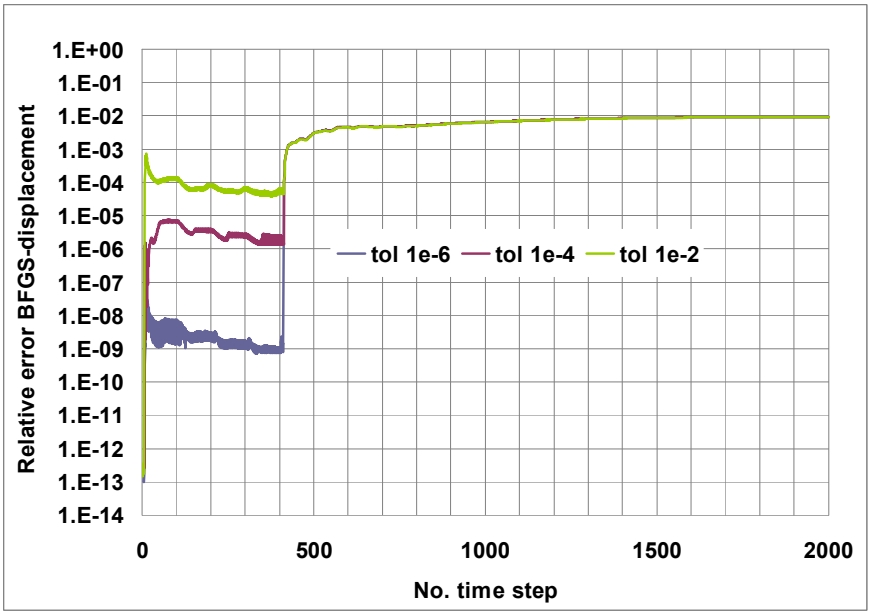
(b)

Figure 4.17: Relative error vs. number of time step
(a): column model. (b): plate model.

The relative error of displacement increases proportionally with increasing the tolerance of corrections for both reduction procedures (POD with or without BFGS) as seen in Figure 4.17a and Figure 4.18a.



(a)

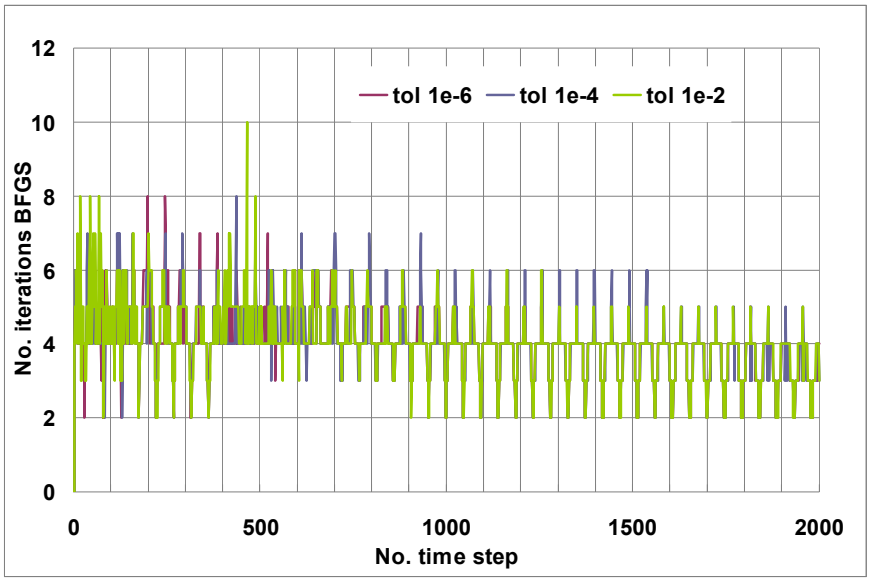


(b)

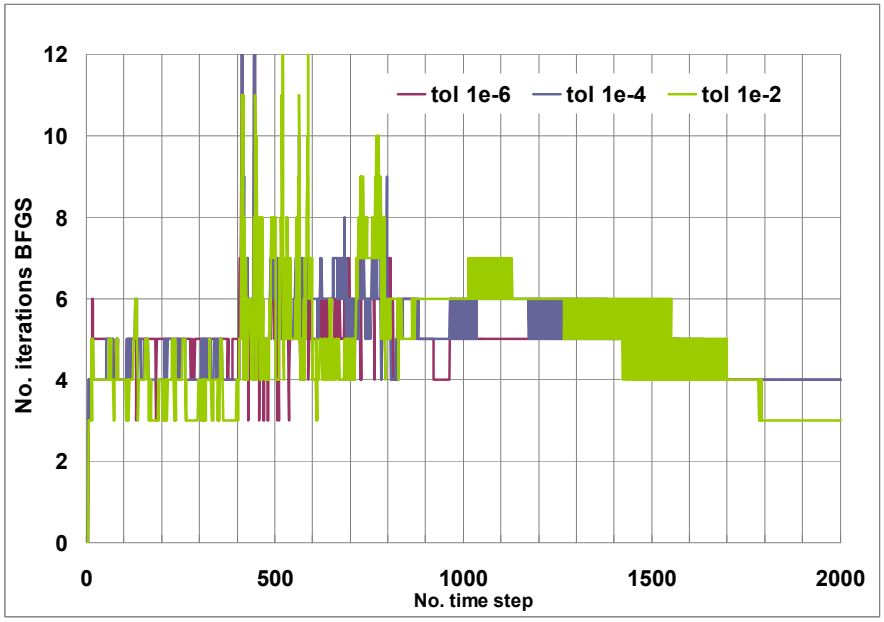
Figure 4.18 Relative error vs. number of time step
(a): column model. (b): plate model.

In Figure 4.17b and Figure 4.18b, the relative error does not change with the changing of tolerance from 2 seconds of simulation in the plate model. This happens due to that, the number of implicit modes is no significant to represent a strong local nonlinearity. It is possible to increase the maximum dimension of the reduced basis or reduced POD tolerance to minimize this effect. However,

the relative error (around one percent) is still a small error. Furthermore, Figure 4.19 presents the number of iterations at each time step for different tolerances



(a)



(b)

Figure 4.19: Iteration by using BFGS vs. number of time step
(a): column model. (b): plate model.

4.3.3 An adaptive reduced order mode by Proper Snapshots Selection (PSS) Method

The PSS method is a new approach to create and to enhance approximation subspaces. The main idea is to build a sequence of subspaces with a minimum dimension based on Ritz vectors from selected snapshots.

In the PSS method, Ritz vectors are generated through a selection of most significant snapshots. The selection procedure is similar to methods that use singular values to measure and compare energy among snapshots (eg. POD). However, this similarity does not mean that the PSS method compute singular vectors which leads PSS to save substantially a processing time.

The PSS is developed in this research for two functions. The first one is to reconstruct a subspace and the second one is to enrich the subspace. The concrete idea is to avail from stored snapshots (during the processing) for generating Ritz vectors. This will avoid expensive computations of singular vectors, usually performed in POD analysis. However, the procedure of selecting most meaningful vectors (snapshots) is based on the analogy of POD method as it's shown below. To start the PSS approach, it is necessary to define the snapshots matrix (U) as:

$$\boldsymbol{U} = \left[\boldsymbol{u}_1,...,\boldsymbol{u}_{n_\Phi}\right] \in \mathbb{R}^{n \times n_\Phi} \tag{4.12}$$

Snapshots are the previous $\boldsymbol{n_\Phi}$ displacement vectors which are collected from the previous $\boldsymbol{n_\Phi}$ time steps. where $\boldsymbol{u}_1$ is the oldest vector and $\boldsymbol{u}_{n_\Phi}$ is the newest vector. To become simpler for an understanding, a mask operator ($\boldsymbol{Z}$) is introduced as:

$$\boldsymbol{Z} = diag\left(\boldsymbol{z}_1,...,\boldsymbol{z}_{n_\Phi}\right) \in \mathbb{N}^{n_\Phi \times n_\Phi} \quad where \quad \boldsymbol{z}_i = 1 \quad or \quad \boldsymbol{z}_i = 0 \tag{4.13}$$

The diagonal matrix (4.13) is look like a filter. So, $\boldsymbol{z}_i = 1$ when the displacement vector is relevant to represent the subspace, otherwise $\mathrm{z_i} = 0$. A mask operator ($\boldsymbol{Z}$)has the following properties:

$$\boldsymbol{Z} = \boldsymbol{Z}^T, \quad \boldsymbol{Z} = \boldsymbol{Z}^T\boldsymbol{Z}, \quad \boldsymbol{Z}\boldsymbol{Z}^T = \boldsymbol{Z}^T\boldsymbol{Z} \quad and \quad \boldsymbol{Z} = \boldsymbol{Z}^p \quad where \ \boldsymbol{p} \in \mathbb{N}_* \tag{4.14}$$

Multiplying (4.13) by (4.12), we obtain:

$$\bar{\boldsymbol{U}} = \boldsymbol{U}\boldsymbol{Z} \in \mathbb{R}^{n \times n_\Phi} \tag{4.15}$$

Next, the matrices

$$\boldsymbol{W} = \boldsymbol{U}^T\boldsymbol{U} \in \mathbb{R}^{n_\Phi \times n_\Phi} \tag{4.16}$$

$$\bar{\boldsymbol{W}} = \bar{\boldsymbol{U}}^T\bar{\boldsymbol{U}} \in \mathbb{R}^{n_\Phi \times n_\Phi} \tag{4.17}$$

are defined.

A direct relation between ($\boldsymbol{W}$) and ($\bar{\boldsymbol{W}}$) can be found easily. These matrices are similar to a correlation matrix from statistical point of view. The difference between them is the key to evaluate snapshots. To reduce the operations, it is convenient to follow these considerations:

$$\bar{\boldsymbol{W}} = \left(\boldsymbol{U}\boldsymbol{Z}\right)^T \boldsymbol{U}\boldsymbol{Z} \tag{4.18}$$

$$\bar{\boldsymbol{W}} = \boldsymbol{Z}^T\left(\boldsymbol{U}^T\boldsymbol{U}\right)\boldsymbol{Z} \tag{4.19}$$

$$\bar{\boldsymbol{W}} = \boldsymbol{Z}^T\boldsymbol{W}\boldsymbol{Z} = \boldsymbol{Z}\boldsymbol{W}\boldsymbol{Z} \tag{4.20}$$

Initially, the analysis is necessary to find snapshots that have significant ranks. This can be done by the comparison between matrices $\boldsymbol{W}$ and $\bar{\boldsymbol{W}}$ as:

1- Calculating the eigenvalues of both matrices as $\boldsymbol{\sigma}, \overline{\boldsymbol{\sigma}}$

2- Calculating the ratios $\boldsymbol{M}$ and $\overline{\boldsymbol{M}}$ with respect to tolerance (tol_{PSS}) as:

$$\boldsymbol{M} = \frac{\sum_{i=n_U+1}^{n_\Phi} \boldsymbol{\sigma}_i}{\sum_{i=1}^{n_\Phi} \boldsymbol{\sigma}_i} \leq tol_{PSS} \qquad where \qquad \boldsymbol{\sigma}_i \in \mathbb{R}_+ \tag{4.21}$$

$n_{\mathbf{U}}$ is the number of eigenvectors. This number does not mean the subspace dimension in the PSS, but it is considered as a reference of evaluation. $\overline{M}$ is calculated with respect to each individual snapshot by making (z_i=0) for the particular snapshot and the rest (z_i=1).

$$\overline{\boldsymbol{M}} = \frac{\sum_{i=n_{\bar{U}}+1}^{n_\Phi} \bar{\boldsymbol{\sigma}}_i}{\sum_{i=1}^{n_\Phi} \bar{\boldsymbol{\sigma}}_i} \leq tol_{PSS} \qquad where \qquad \bar{\sigma}_i \in \mathbb{R}_+ \tag{4.22}$$

$n_{\bar{U}}$ represents the number of eigenvectors. The comparison will be evaluated for each snapshot based on two conditions (4.23) and(4.24):

$$n_U = n_{\bar{U}} \tag{4.23}$$

$$n_U > n_{\bar{U}} \tag{4.24}$$

If the condition (4.24) is satisfied, then a certain snapshot will be considered in the subspace. Otherwise, a snapshot will not take a part in the subspace (Ritz Vectors which computed based on selected snapshots). As mentioned before in the section of an incremental adaptivity, the PSS is used for enrichment or to reduce a subspace as in the next two sections:

4.3.3.1 PSS to enrich the projection matrix (PSSe)

The PSS is evaluated at the end of each step which means the analysis is just made by comparing a snapshot from the previous step with the current step. If the condition (4.23) is satisfied, the current vector replaces the previous vector. Otherwise, the current vector is added to the projection matrix. The main idea is to check whether the vector of the current step is meaningful to the subspace or not. Besides, this analysis is simpler and spends less time processing consequently.

4.3.3.2 PSS to reduce the projection matrix (PSSr)

The PSSr is required to construct a new subspace when the subspace reaches a maximum dimension ($n_{max} = 50$) or becomes stable ($n_{stable} = 25$) as mentioned before in the section of an incremental adaptivity. The idea is to start analyzing all snapshots from the oldest to the newest snapshots. This analysis is done to force the newest snapshots to be more representative (causing less instability).

4.3.3.3 PSS's tolerance and weighted snapshots

Due to different functions of the PSS (PSSe & PSSr), it is necessary to analyze the snapshots energy with respect to the following inequality:

$$tol_{PSSe} << tol_{PSSr} \tag{4.25}$$

This difference is necessary because an enrichment analysis (PSSe) is much more sensitive than the reduced analysis (PSSr). The PSS solves the problem of

increasing a subspace based on most significant snapshots (vectors of displacement field) which arises during a configuration change. Therefore, a selection of the most significant snapshots is based on the variation between them (The newest snapshot has high weight compare to the oldest one). The function of the weight was selected to be a redial basis function which is more convenient to the most of variation problems as:

$$\boldsymbol{\gamma}_i = e^{-\frac{\left(\boldsymbol{\beta}_i - \boldsymbol{\beta}_{n_\Phi}\right)^2}{2\boldsymbol{\tau}^2}} \in \mathbb{R}_+ \qquad where \qquad 0 < \boldsymbol{\gamma}_i \leq 1 \tag{4.26}$$

$$\boldsymbol{\beta}_i = \left\| \boldsymbol{u}_i \right\| \quad and \quad \boldsymbol{\beta}_{n_\Phi} = \left\| \boldsymbol{u}_{n_\Phi} \right\| \qquad where \qquad \boldsymbol{\beta}_i, \boldsymbol{\beta}_{n_\Phi} \in \mathbb{R}_+ \tag{4.27}$$

$$\boldsymbol{\tau} = \sqrt{\frac{\sum_{i=1}^{n_\Phi - 1} \left(\boldsymbol{\beta}_i - \boldsymbol{\beta}_{n_\Phi} \right)^2}{\left(n_\Phi - 1 \right)}} \qquad where \qquad \boldsymbol{\tau} \in \mathbb{R}_+ \tag{4.28}$$

$$\boldsymbol{\gamma}^T = \left[\boldsymbol{\gamma}_1, ..., \boldsymbol{\gamma}_{n_\Phi} \right] \in \mathbb{R}^{n_\Phi} \tag{4.29}$$

where $\boldsymbol{\beta}_{n_\Phi}$ is the newest snapshot norm, $\boldsymbol{\beta}_i$ are previous snapshots norms, $\boldsymbol{\tau}$ is the standard deviation related to the newest snapshot norm. The weight of each snapshot multiplies by the snapshot which belongs on it:

$$\boldsymbol{U}_\gamma = \left[\boldsymbol{\gamma}_1 \boldsymbol{u}_1, ..., \boldsymbol{\gamma}_{n_\Phi} \boldsymbol{u}_{n_\Phi} \right] \in \mathbb{R}^{n \times n_\Phi} \tag{4.30}$$

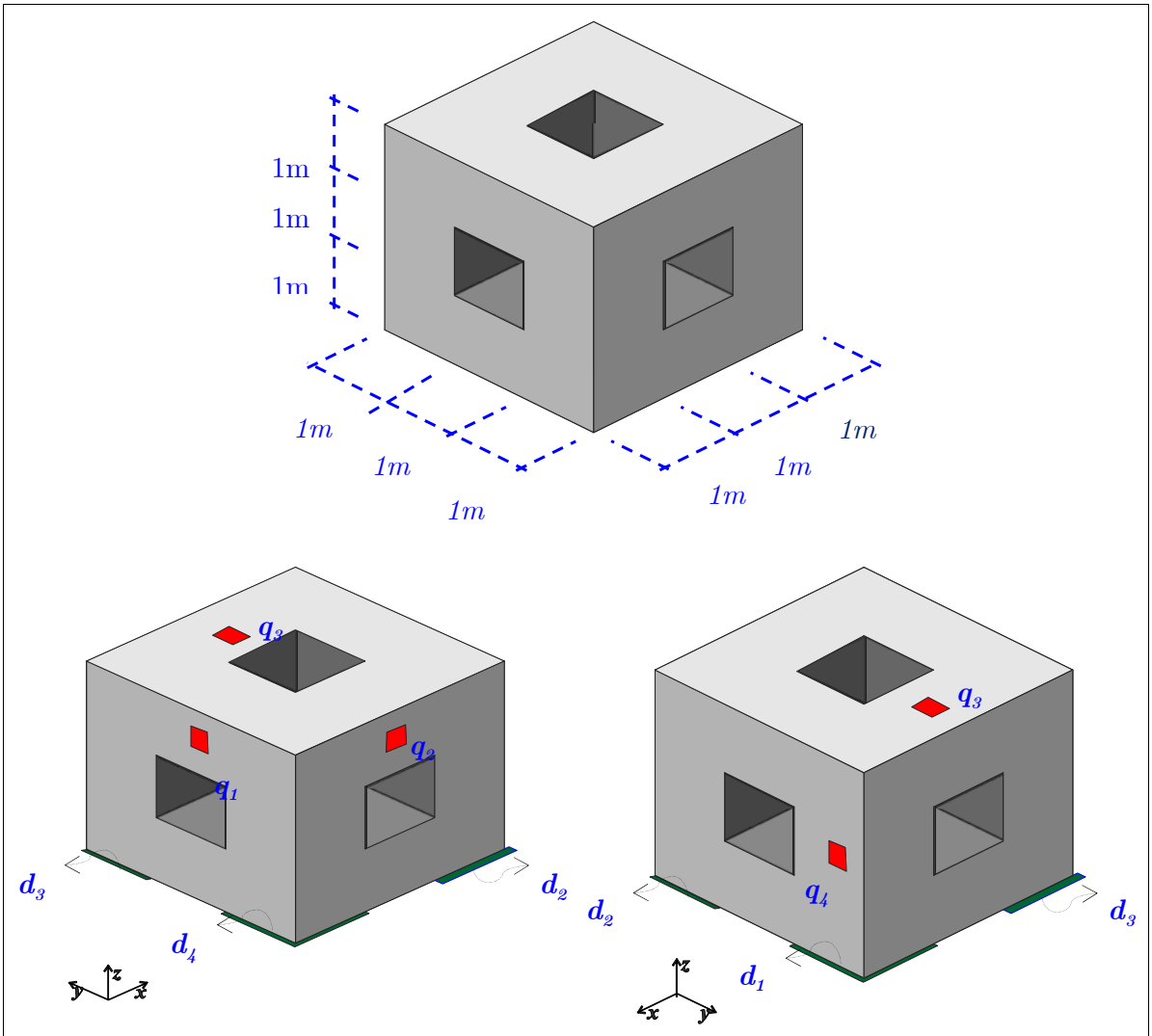
$\boldsymbol{U}_\gamma$ is ready to make the PSS analysis (equations (4.12) to (4.24)). To reduce the processing, it is recommended to change the equation (4.16) to the weight form (4.33) as:

$$\boldsymbol{W}_\gamma = \boldsymbol{U}_\gamma{}^T \boldsymbol{U}_\gamma \in \mathbb{R}^{n_\Phi \times n_\Phi} \tag{4.31}$$

$$\boldsymbol{H} = \boldsymbol{\gamma}\boldsymbol{\gamma}^T \in \mathbb{R}^{n_\Phi \times n_\Phi} \tag{4.32}$$

$$\boldsymbol{W}_\gamma = \boldsymbol{H} \circ \boldsymbol{W} \in \mathbb{R}^{n_\Phi \times n_\Phi} \tag{4.33}$$

Through the Hadamard (or Schur) product, the eigenvalues can be found. To study the convergence and accuracy of this approach for reducing the computational effort and time of the system, two models are simulated. The first model is a column of Neo-Hookean material as seen in Figure 4.7. It is discretized by the mesh with 12302 nodes and 36906 degrees of freedom.



The properties of cube system:

E(Young's modulus)=0.5e9Pa	q_1 (Load) = 9.0e11N (X-direction)
v(Poisson's ratio)=0.45	q_2 (Load) = 9.0e11N (Y-direction)
ρ(Density)=1600 kg/m^3	q_3 (Load) = -9.0e11N (Z-direction)
ζ (Damping ratio) = 0.01,	q_4 (Load) = -9.0e11N (X-direction)
Δt=0.005s	d_1(Fixed support)=restriction for all directions
T (Simulation time) =5s	d_2(Fixed support)=restriction for directions(Y&Z).
β (Newmark parameter)=0.25	d_3(Fixed support)=restriction for direction(Z).
γ(Newmark parameter)=0.5	d_4(Fixed support)=restriction for direction(Z).

Figure 4.20: Cube model

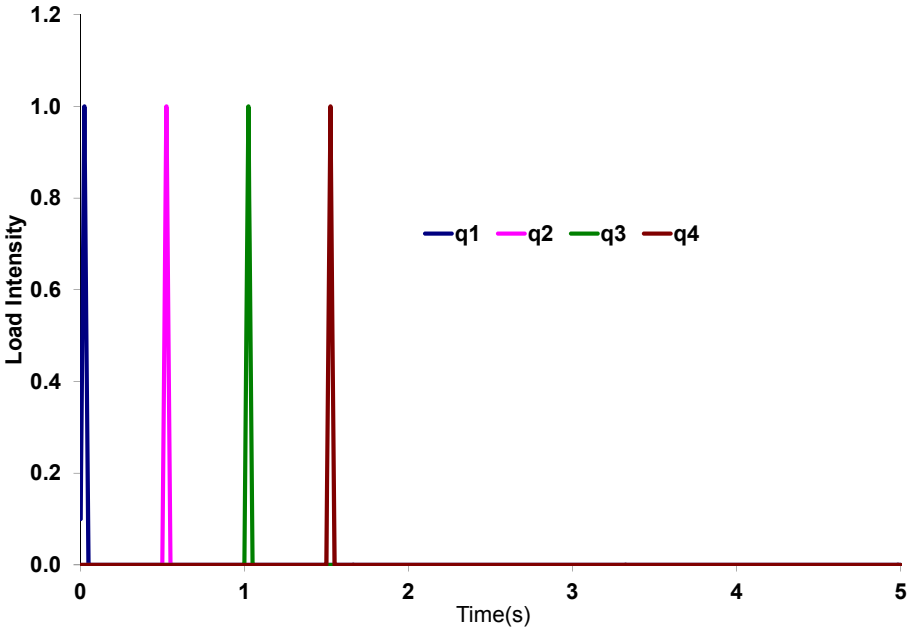


Figure 4.21: Load intensity vs. Time

a)

(b)

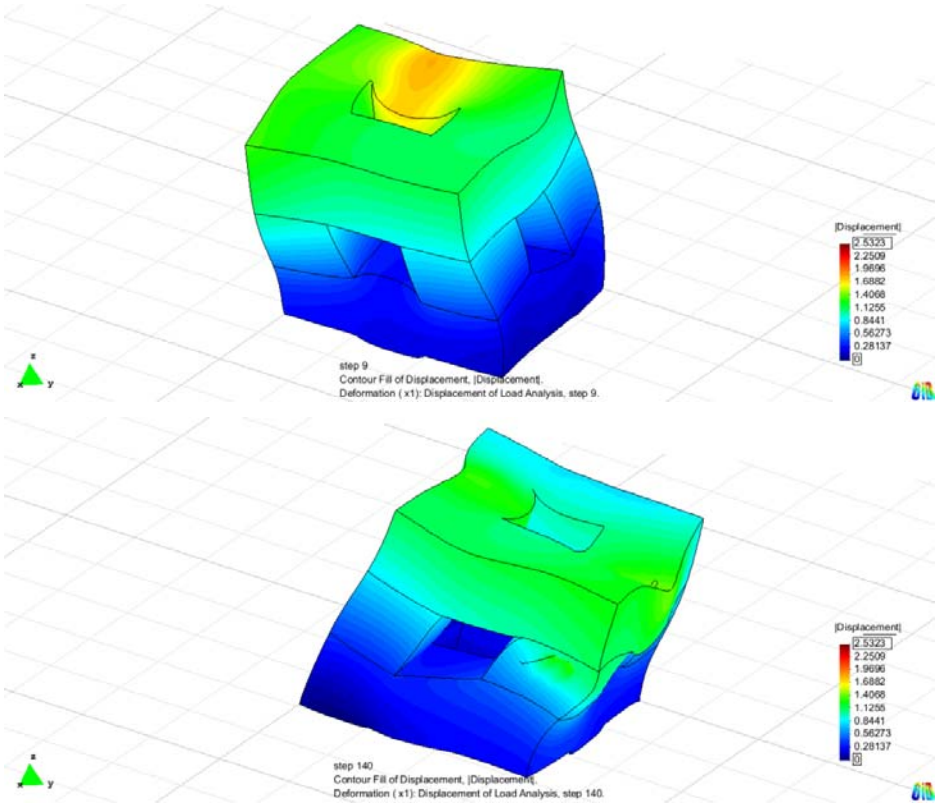


Figure 4.22: Cube model (a): Displacements (m) at t=0.1350s. (b): Displacements (m) at t=2.1025s.

The second one is a cube of Neo-Hookean material as seen in Figure 4.20 which is discretized by the mesh with 25536 nodes and 76608 degrees of freedom. The different loads intensity with the time is presented in Figure 4.21.

The comparison between the reduced system and the full system for both models are presented in Table 4.5 and Table 4.6. This comparison includes different reduced order model parameters such as processing time, iterations, corrections, and number of used PSSe & PSSr.

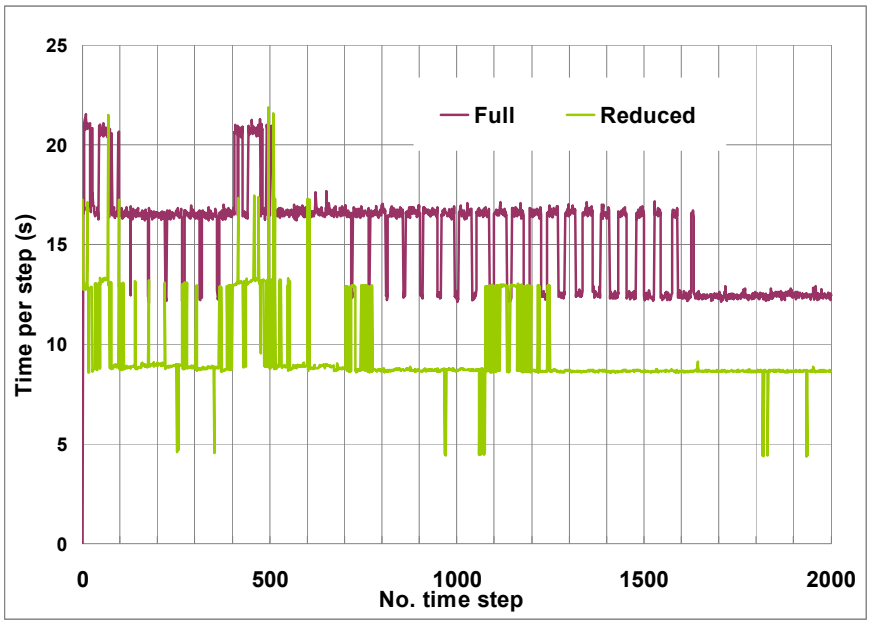
Table 4.5: An overview of the column model results

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance – tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	7412	18185	15421	14069
No. Corrections	-	4310	2964	2352
No. PSSr	-	19	16	30
No. PSSe	-	1984	1994	1997
Total time (s)	30730.84	18802.04	13039.77	10513.42
Time per step (s)	15.36	9.39	6.517224	5.253907
Total time (%)	100.00	61.18	42.43	34.21

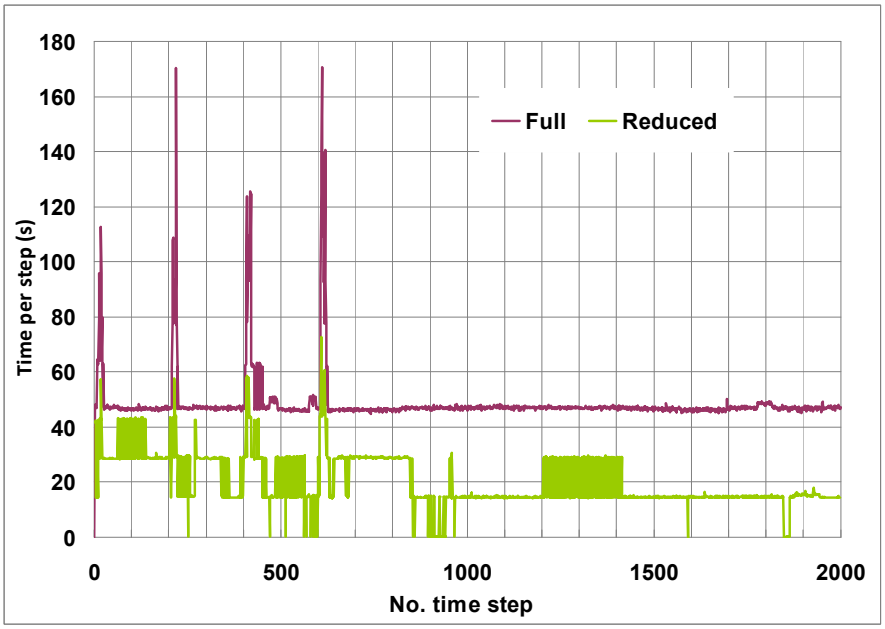
Table 4.6: An overview of the cube model results

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance – tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	6208	12896	10822	10208
No. Corrections	-	2777	1870	853
No. PSSr	-	4	5	4
No. PSSe	-	1987	1990	1983
Total time (s)	97019.75	40558.884	27408.84	12982.73
Time per step (s)	48.51	20.27	13.71	6.49
Total time (%)	100.00	41.80	28.25	13.38

For both models (column and cube), the processing time of each time step is high in case of a full system compared to the reduced system as seen in Figure 4.23.



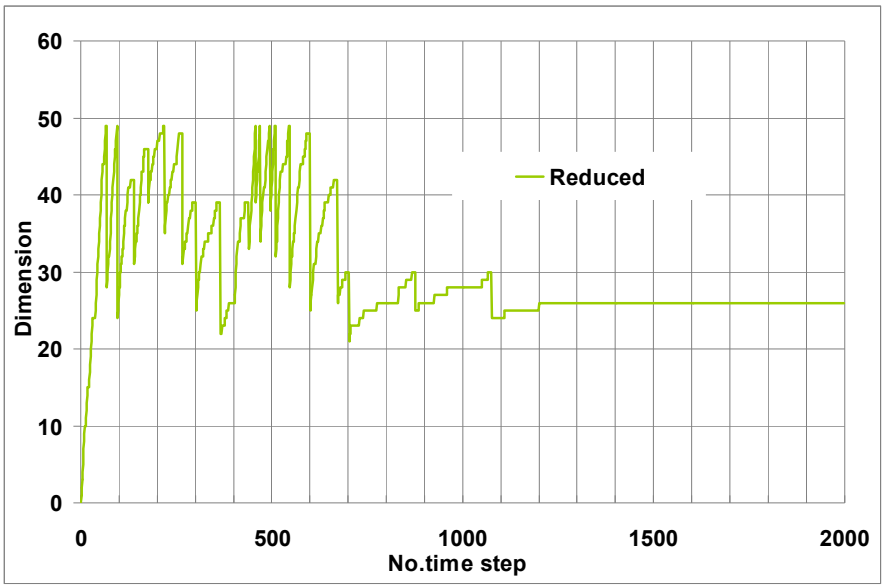
(a)



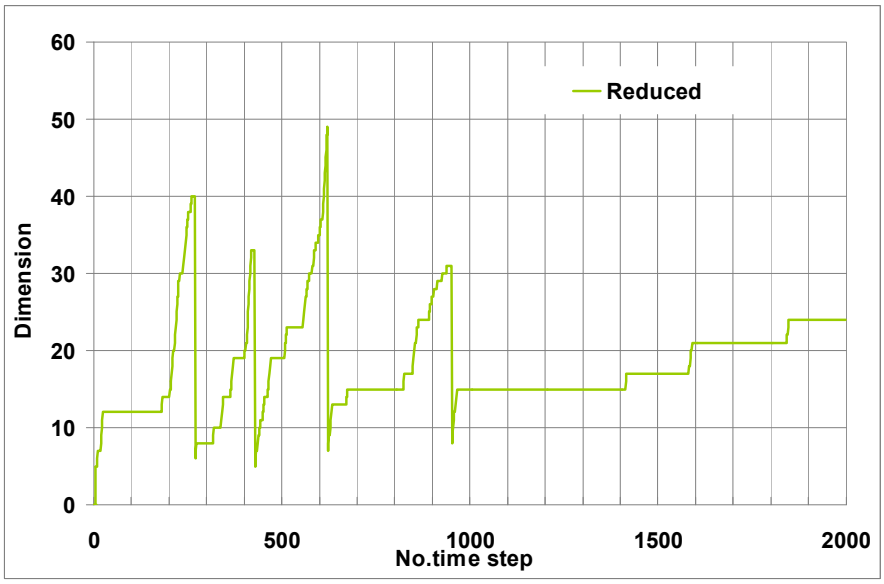
(b)

Figure 4.23: Time per step vs. number of time step (a): column model. (b): cube model.

The dimension of the subspace changes during the computational process (Figure 4.24). In fact, this is related to change of implicit modes in snapshots.



(a)

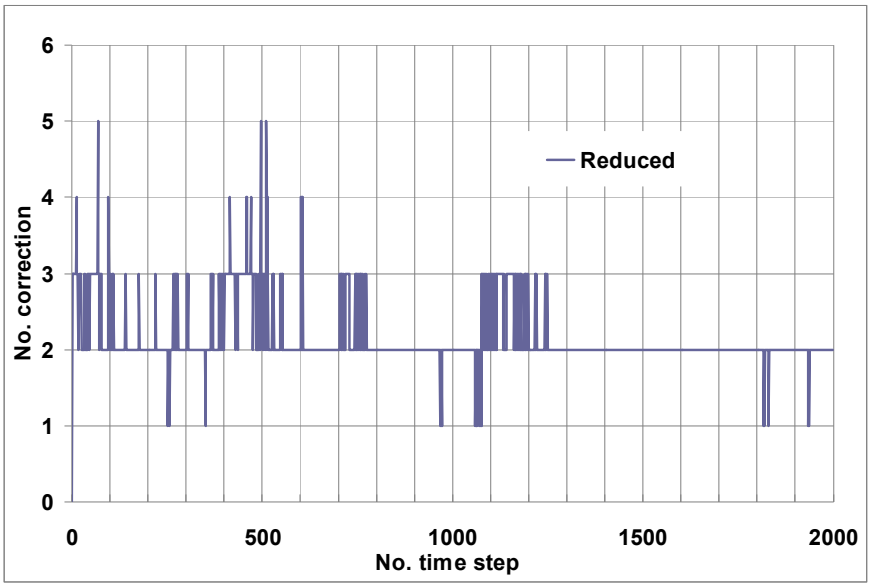


(b)

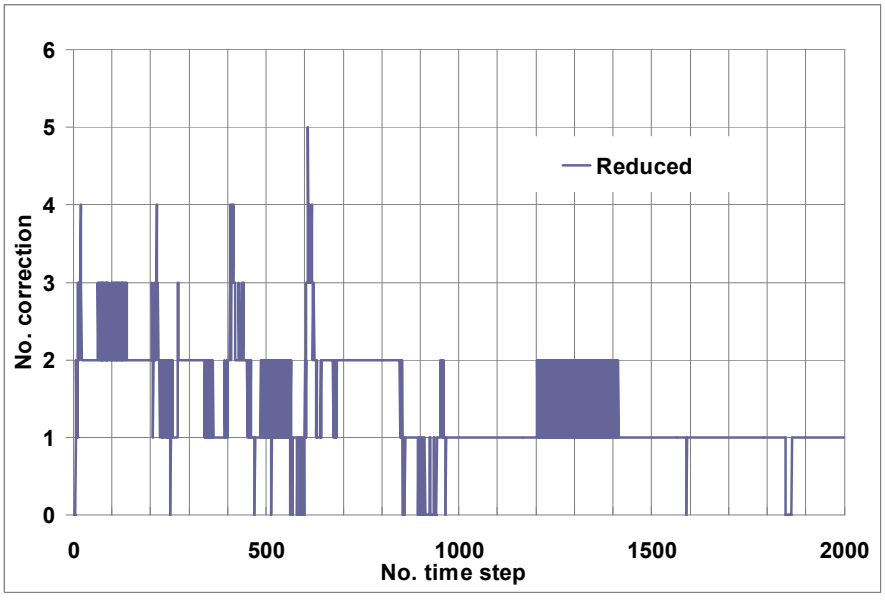
Figure 4.24: Dimension of subspace vs. number of time step (a): column model. (b): cube model.

In the same way, the number of corrections is related to changing of configurations (modes) as seen in Figure 4.25. It means that the correction increases when the configuration changes significantly in a short period due to the large displacements.

The difference between the convergence of the full and reduced system can be noticed in Figure 4.26. The full system has quadratic convergence in the most of time except when the loads are applied, while the reduced systems have superlinear convergence.



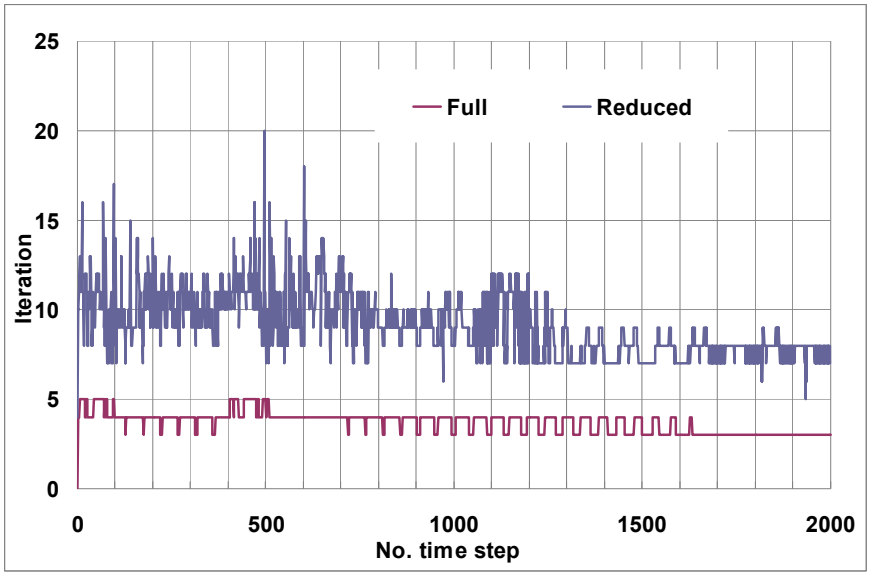
(a)



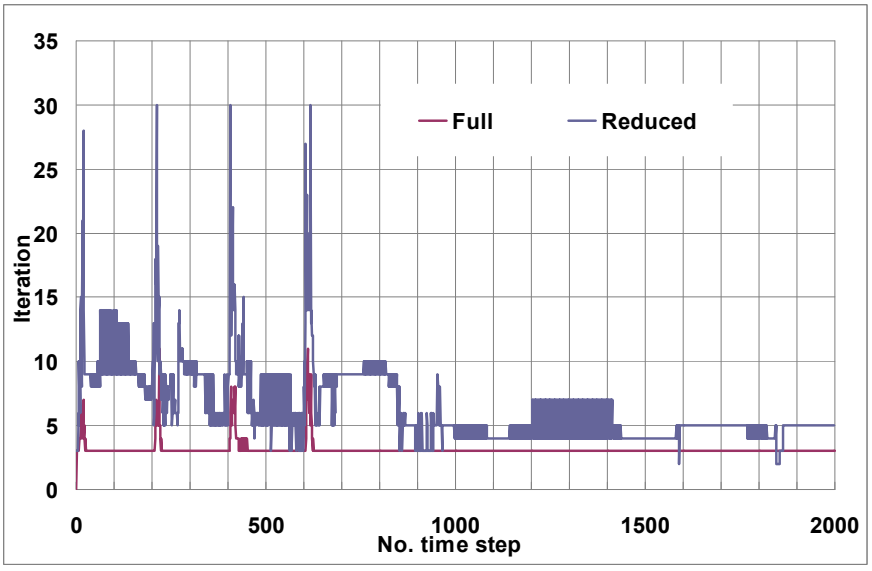
(b)

Figure 4.25: Number of corrections vs. number of time step
(a): column model. (b): cube model.

However, iterations are not significant when the corrections are few. In general, the reduced system is faster than the full system. Indeed, the number of iterations in the reduced system is greater than the full system (compare Figure 4.23 with Figure 4.26). To measure the error, this work defines the relative error of displacements for each time step. The relative error of the displacement increases proportionally with increasing the tolerance of reduced system problem as seen in Figure 4.27 a.



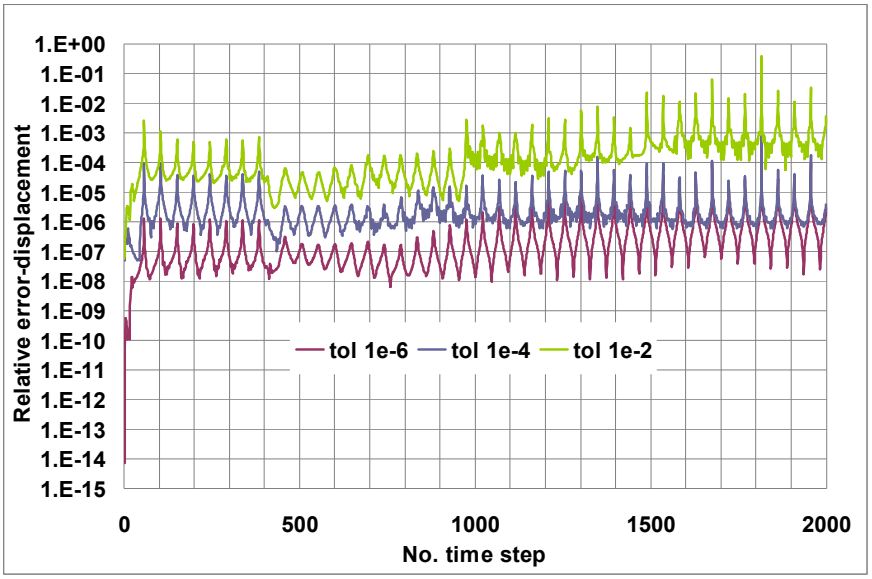
(a)



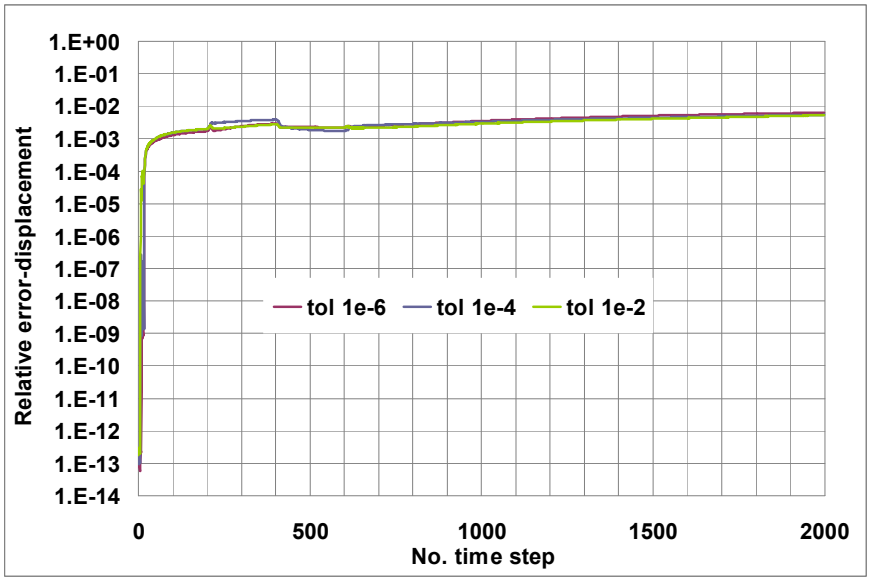
(b)

Figure 4.26: Iteration vs. number of time step (a): column. (b): cube.

In Figure 4.27b, the relative error does not change with the different tolerances since the beginning of the simulation in the cube model (due to initial impact load). This happens due to the number of modes implicit in the snapshots are not significant to represent a strong local nonlinearity that happens in 0, 0.5, 1 and 1.5 second of simulation. Increasing the maximum dimension of the reduced space can minimize this effect. However, a relative error is less than one percent.



(a)



(b)

Figure 4.27: Relative error vs. number of time step (a): column model. (b): cube model.

4.4 An adaptive Hyper-reduced order model

Unfortunately, a classical projection approach is not efficient in the nonlinear model reduction because it is not necessary to capture a nonlinearity of the problem. There are two main reasons to adaptive a subspace; changing of mode (configuration) and strong local nonlinearity. The overview of the adaptivity is illustrated in Figure 4.28. Actually, the author combined adaptivity based on Dynamic-POD with hyper-reduction based on missing point Estimation (MPE).

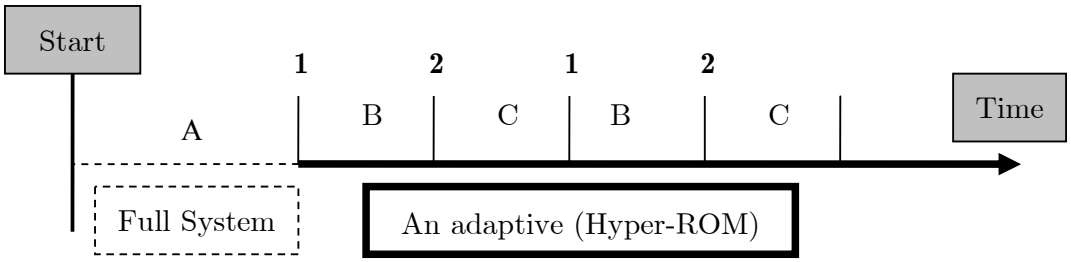


Figure 4.28: An adaptive Hyper-ROM strategy

- A: a full system is solved and a collecting of snapshots is done to construct a subspace.

- 1: a POD's subspace is constructed based on weighted snapshots. Then, a selection of Boolean matrix based on hyper reduction (missing point estimation) is done.

- B: a hyper reduction system (Petrov-Glarkein projection).

- 2: an error of model reduction is high which means the mode change or effecting of strong non-linearity.

- C: a subspace which obtained in 1 with Dynamic-POD without hyper reduction is implemented (Glarkein projection). Then, a collection of snapshots (solution of the system) and weighted of it.

4.4.1 The Dynamic-POD

It is a new approach for improving a projection based on POD subspace (Φ). A projection of the nonlinear system on subspace by using a classical POD is good when a structural mode doesn't change or when strong local non-linearity doesn't occur.

The main idea of Dynamic-POD is based on adding one vector to subspace (Φ) when the error is high during iteration and at the end of each step. But, a size of subspace (Φ) is always stay (k^{th} *+1*) which means that the adding of new vector replace the previous vector. The vector which is added should be orthogonal with all k^{th} vectors in subspace (Φ). The orthogonality of adding vector is done by using Gram-Schmidt concept. There are some ideas were arisen to increase subspace dimension by adding some vectors such as (Kerfriden et al (2011)) are not good, because the system may be shifted which leads the system to become unstable after some steps. There are two position of adding a vector to a subspace. First position is that, adding an error vector when there is no convergence of the problem after certain number of iteration. The second one is that, adding a solution to a subspace at the end of the time step.

4.4.2 Weighted Snapshots

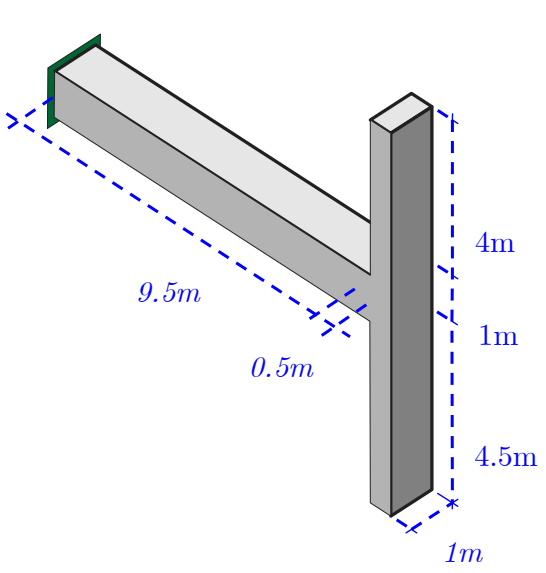
It is an approach to construct a subspace based on the suitable snapshots. This method is more suitable with correction of subspace (Adaptive subspace). A constructing of a subspace based on most significant snapshots related to mean vector of displacement field ($\mathbf{U}$) is suggested to get snapshots which are more close to mean value. This will help POD's optimization function (eq.(3.14)) which is based on least square optimization to be near optimum situation. The selection of most significant snapshots is based on the variation between each snapshot and mean vector ($\mathbf{U_m}$). The closest snapshot to the mean vector ($\mathbf{U_m}$) has a high weight (γ_i) and consequently depending on the variation. The function of the weight was selected to become similar to normal distribution (Gaussian distribution) function which is more convenient for most of variation problems.

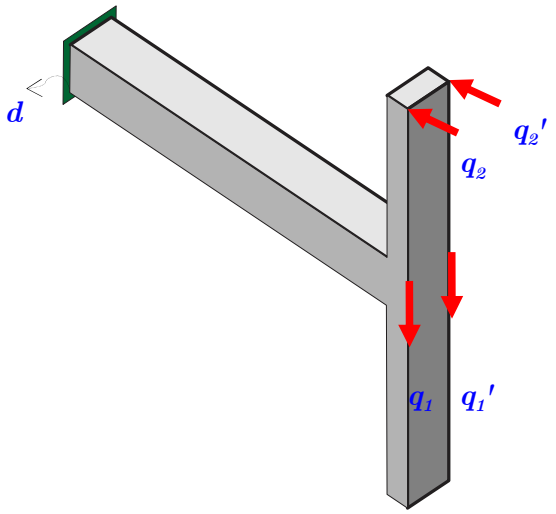
$$\gamma_i = e^{\frac{-(\beta_i - \beta_{n_\Phi})^2}{2\tau^2}} \tag{4.34}$$

where, (β_i): norm of arbitrary vector (snapshot), (β_{n_Φ}): norm of mean vector (U_m), (τ): standard deviation of the snapshots norm., (e): Natural exponential.

$$\tau = \frac{\sqrt{\sum_{i=1}^{n_\Phi} (\beta_i - \beta_{n\Phi})^2}}{n_\Phi - 1} \tag{4.35}$$

The weight of each snapshot is multiplied by snapshot which belongs to it. This procedure will change the energy of snapshots depending on a significant. Finally, the new snapshots are ready to construct subspace based on POD.





The properties of T-Beam system :

E(Young's modulus)=0.5e9 Pa	β (Newmark parameter)=0.25
v(Poisson's ratio)=0.45	γ(Newmark parameter)=0.5
ρ(Density)=1600 kg/m^3	q_1 (Load) = 5e6N
ζ (Damping ratio) = 0.01	q_2 (Load) = 5e6N
Δt=0.005s	d(Fixed support)=restriction for all directions
T (Simulation time) =2s	

Figure 4.29: T-Beam model

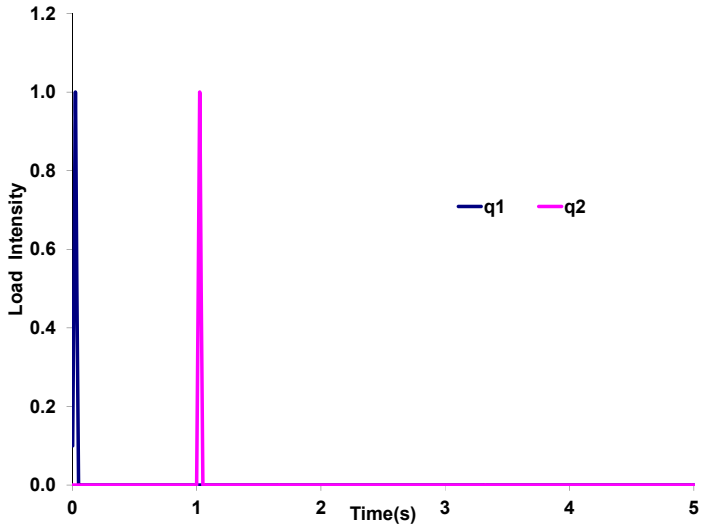


Figure 4.30: load intensity vs. time for T-Beam model.

a)

(b)

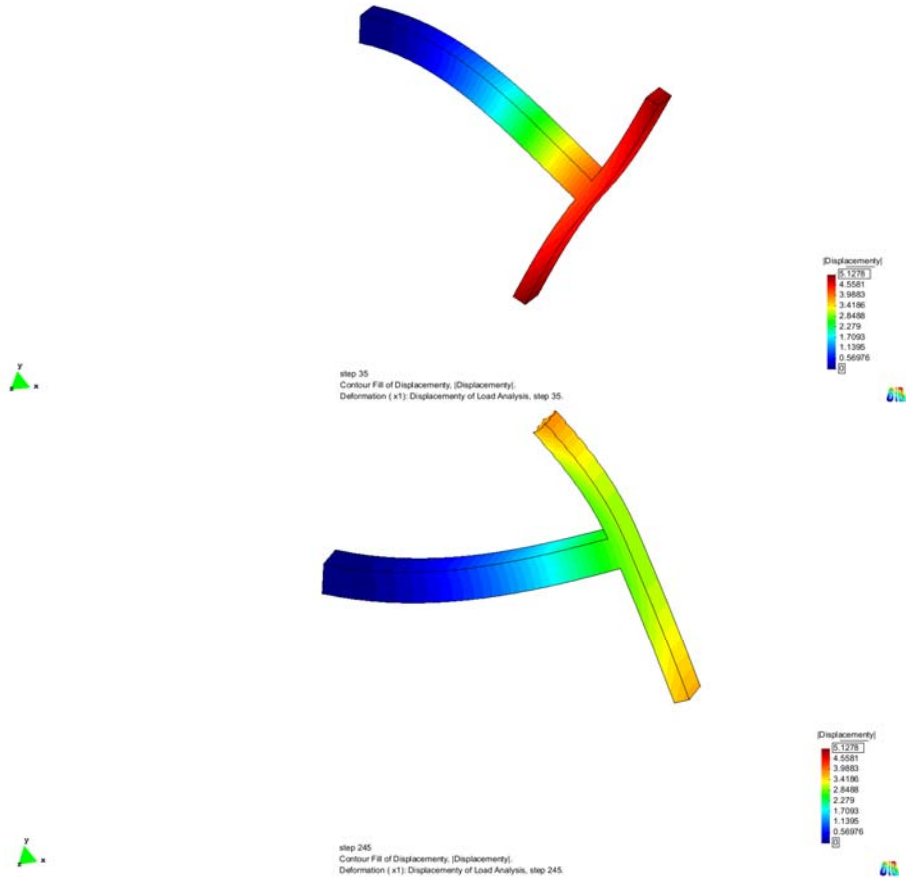


Figure 4.31: T-Beam model (a): Displacements (m) at time step=35. (b): Displacements (m) at time step=245

To study the hyper-reduction approach two models were simulated. The first one is the column model of Neo-Hookean material in Figure 4.7 and T-beam model of Neo-Hookean material in Figure 4.29.

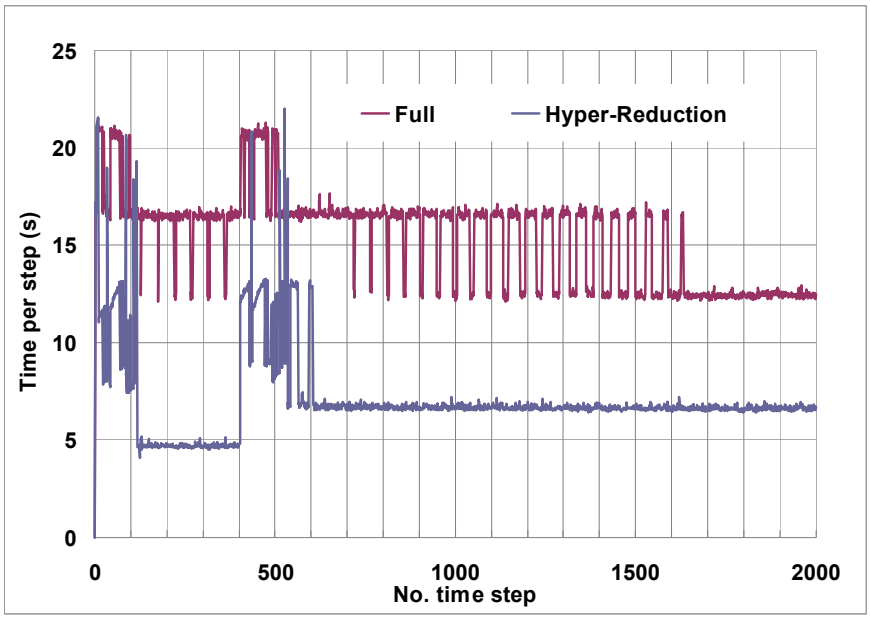
Table 4.7: An overview of the column model results

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance – tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	7412	18185	15987	13289
No. Corrections	-	1084	842.6948	697
No. MPE	-	9	5	3
Total time (s)	30730.84	14109.69	12505.61	9297.07
Time per step (s)	15.37	7.05	6.25	4.645431
Total time (%)	100.00	45.91	40.68	30.2503

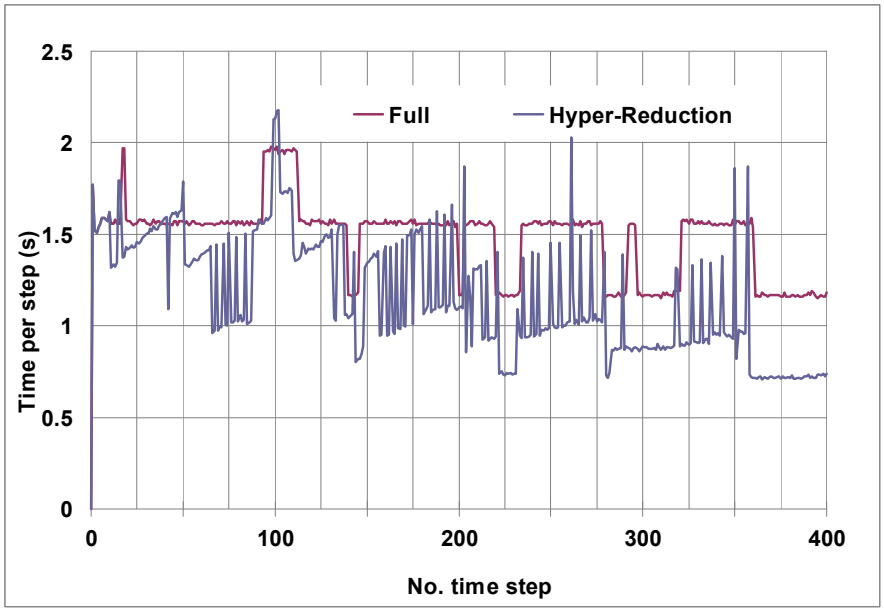
Table 4.8: An overview of the T-Beam model results

	Full	**Reduced**	**Reduced**	**Reduced**
Tolerance – tol	**1.00E-06**	**1.00E-06**	**1.00E-04**	**1.00E-02**
No. Iterations	1521	4128	3629.148	3266.233
No. Corrections	-	353	306.9846	276.2861
No. MPE	-	12	8	7
Total time (s)	594.07	464.77	411.93	370.73
Time per step (s)	1.49	1.16	1.03	0.94
Total time (%)	100	78.23	69.33	62.39

The adaptive hyper reduction approach seems to be a promising technique to reduce the computation time. In fact, this can be seen in the column model in Figure 4.32a. But, it is an inverse in Figure 4.32b for the T-beam model due to that, the number of constructing a hyper reduction system (matrix (D)) is computationally expensive. This is clear from the peaks in the Figure 4.32 when the computational time is high.



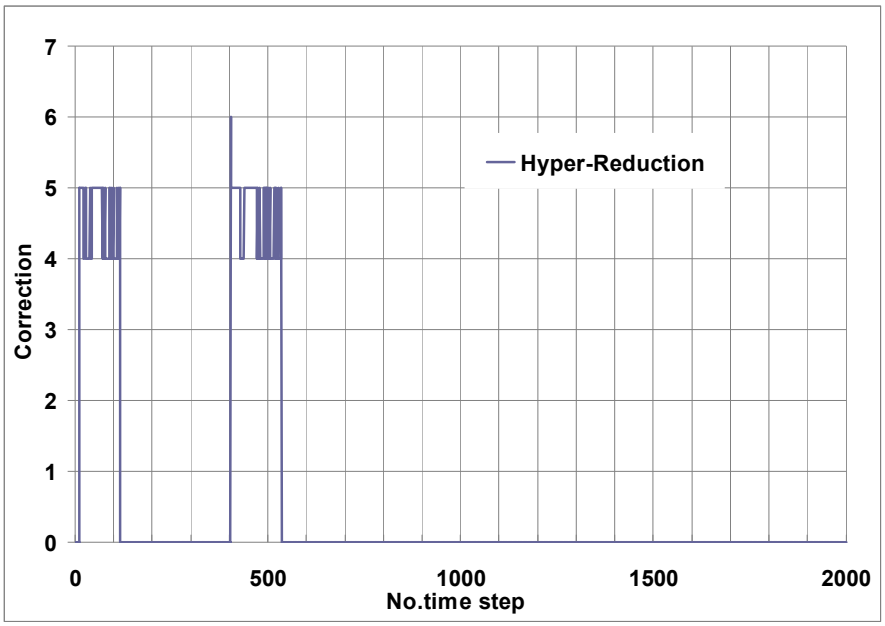
(a)



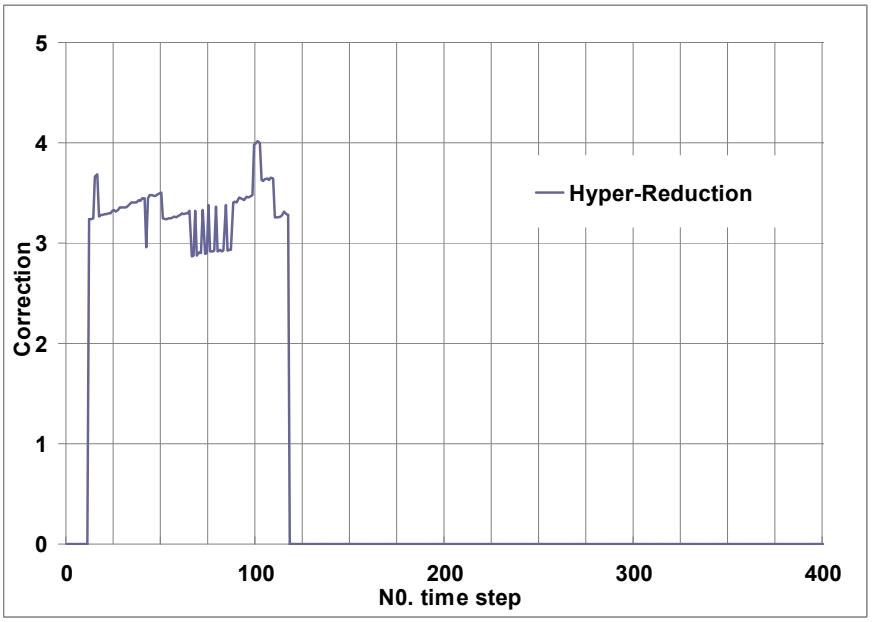
(b)

Figure 4.32: Time per step vs. number of time step (a): column model. (b): T-Beam model.

The number of corrections based on adaptivity (Dynamic-POD) is high due to the fact that the subspace dimension does not increase with the adaptivity. This can be seen clearly in Figure 4.33 for both systems.



(a)

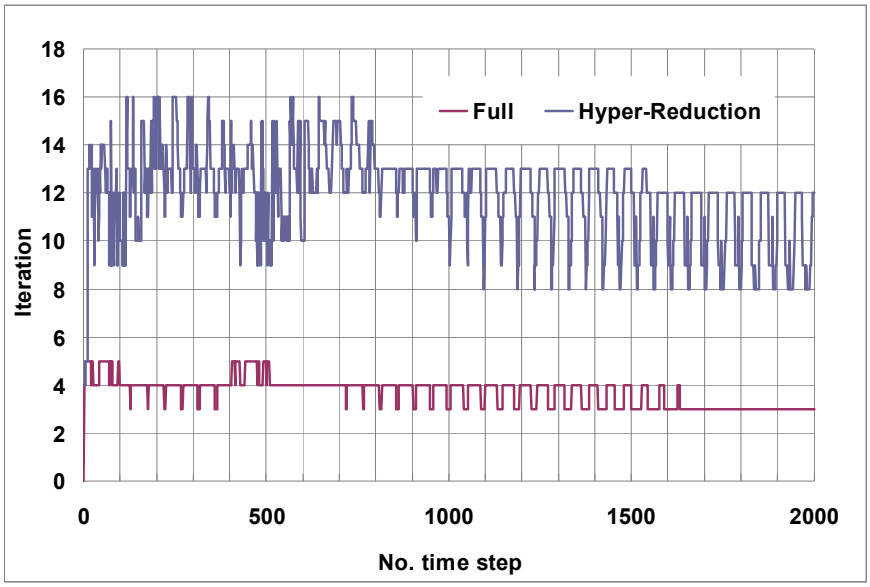


(b)

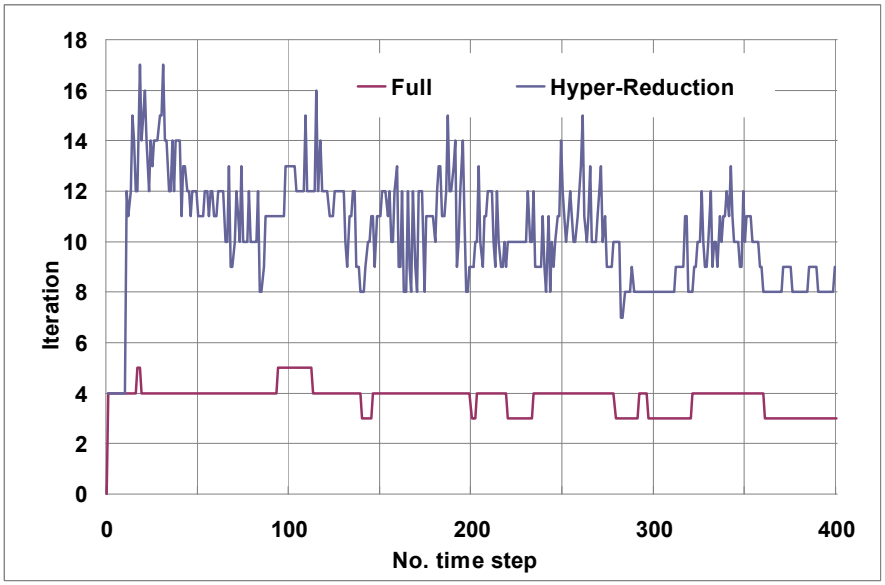
Figure 4.33: Number of corrections vs. number of time step (a): column model. (b): T-Beam model.

Compared to the previous approaches, the hyper reduction contains more corrections in some iteration and no corrections in other iterations. This is in fact due to reconstruct a reduced subspace simultaneously which will give a good approximate of the system. The hyper reduction approach has more

iteration than the full system. This is related to the projection which will never give a quadratic convergence. The results can be seen in Figure 4.34, Figure 4.35, and Figure 4.36.

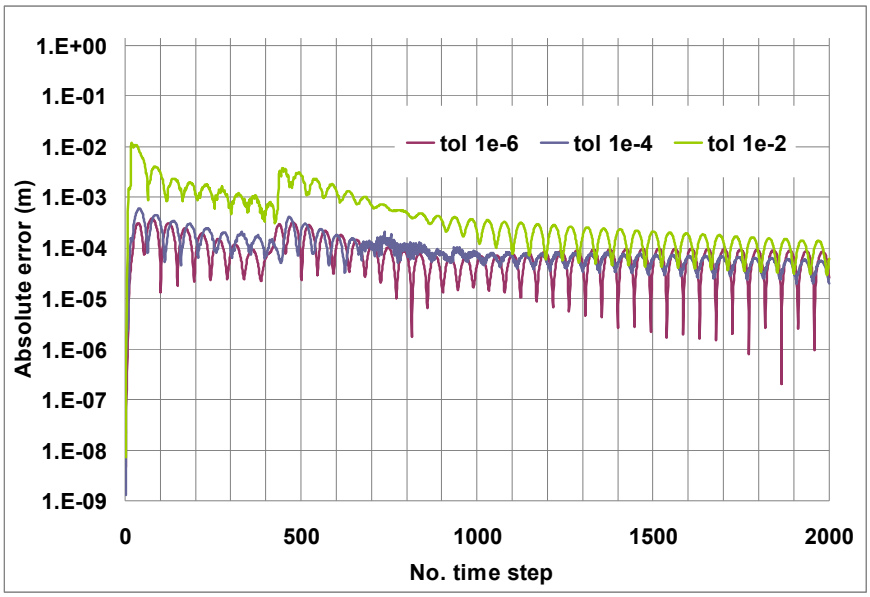


(a)

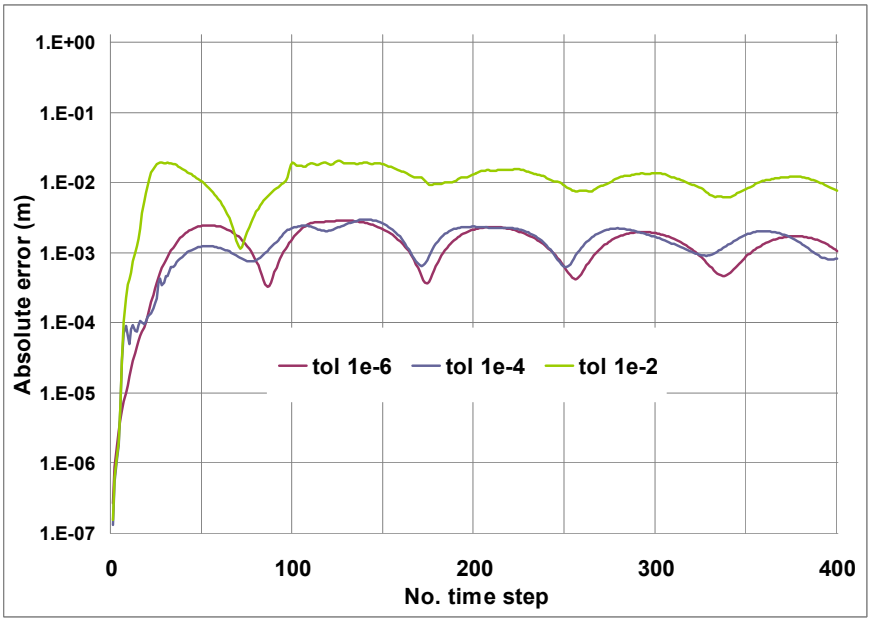


(b)

Figure 4.34: Iteration vs. number of time step (a): column model. (b): T-Beam model.

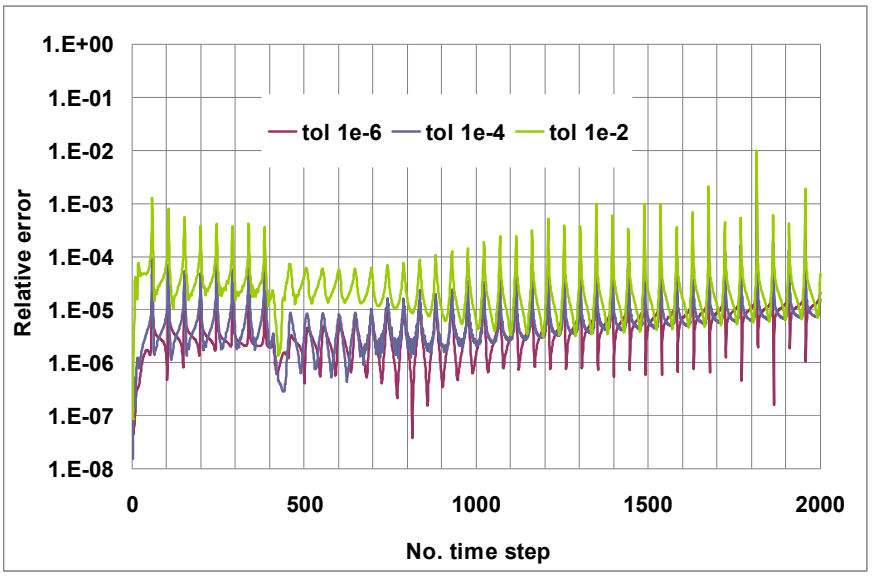


(a)

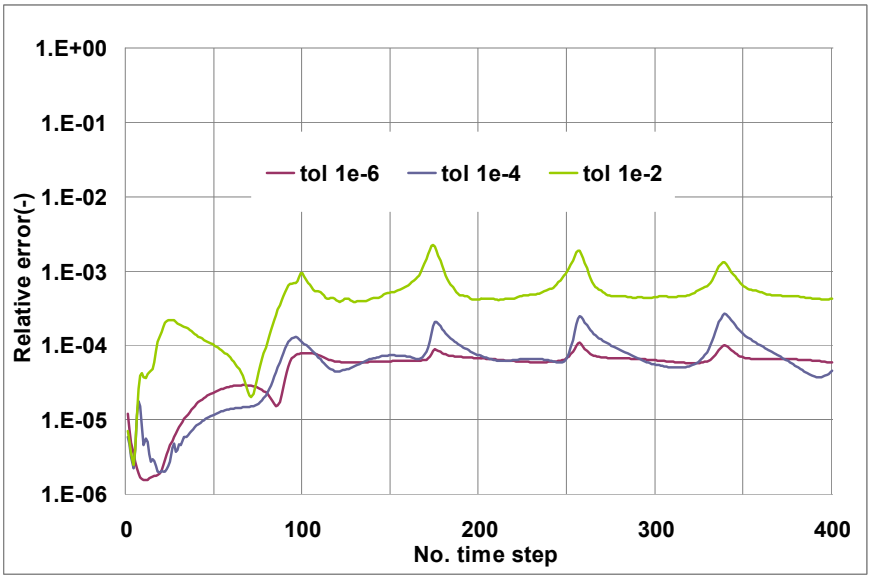


(b)

Figure 4.35: Absolute error vs. number of time step (a): column model. (b): T-Beam model.



(a)



(b)

Figure 4.36: Relative error vs. number of time step (a): column model. (b): T-Beam model.

Chapter 5 Machine Learning Approach

5.1 Support vector machine (SVM)

It is probably the most widely used algorithm in the field of machine learning. It is based on the kernel learning approach, the pattern recognition and the optimization theory (Vapnik (1995)). The performance of learning is limited by three sources of error (an approximation error, an estimation error, and an optimization error) (Bottou & Lin (2007)). Furthermore, the scale of the problem (learning problem) has also a significant effect on the limitation of SVM. The SVM can relate to two features, the support vector for the classification purpose, and the support vector for the regression purpose.

5.1.1 Support vector for classification

The earlier work of pattern recognition was linear classification (Nilsson (1965)). The classification of pattern (x) is performed by giving a class ($y = \pm 1$). The linear classification is the dot product of (inner product or scalar product) as:

$$f\left(x\right) = w^T \boldsymbol{x} + b \tag{5.1}$$

where w is known as the weight vector, and b is the bias.

$$\begin{aligned} &w^T \boldsymbol{x} + b \geq 1, \quad if \quad y = 1 \\ &w^T \boldsymbol{x} + b \leq -1, \; if \quad y = -1 \end{aligned} \tag{5.2}$$

The previous two constraints can be combined as:

$$y\left(w^T \boldsymbol{x} + b\right) \geq 1 \tag{5.3}$$

To construct a hyper plane which is proposed by (Vapnik &Lerner (1963)), the bias $b = 0$ is assumed. Then, the hyper plane becomes a plane which contains all points perpendicular to w and satisfied $w^T \boldsymbol{x} = 0$ as it is shown in Figure 5.1.

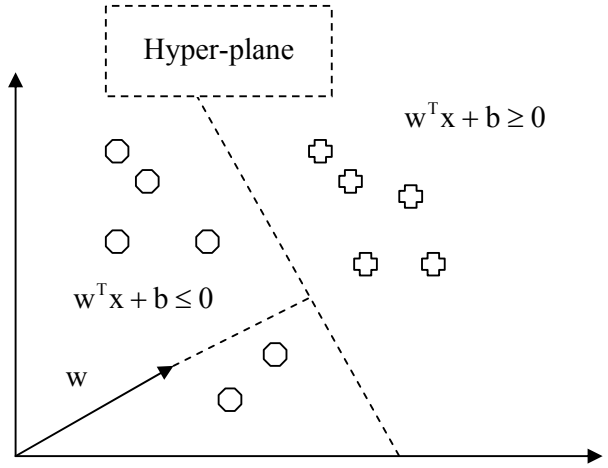


Figure 5.1: a linear classification (SVM)

The transfer of a hyper plane away from the origin along a vector (h) leads to that the equation of hyper-plane will be$(\boldsymbol{x}$ - $h)^T w = 0$, $b = h^T w$, $\|h\| = \frac{|b|}{\|w\|}$.

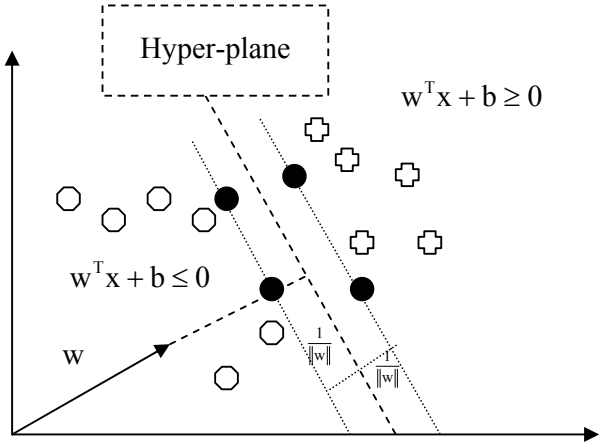


Figure 5.2: the margin of linear separable case with circled data points are the support vectors (SVM)

Thus, it is possible to construct more hyper-planes than a separate hyper-plane as in Figure 5.2. The distance between each hyper-plane and separate hyper plane is $\frac{1}{\|w\|}$. Therefore, the total distance will be$\frac{2}{\|w\|}$.

To find the hyper-planes, the optimization problem in (5.4) should be solved. In fact, the name of support vector comes from the data (points) which lie on the hyper plane and are called support vectors (Figure 5.2).

$$\begin{aligned} &\underset{w,b}{minimize} && \frac{1}{2}\|w\|^2 \\ &subject\ to: && y_i(w^T \boldsymbol{x}_i + b) \geq 1, \quad i = 1,\ 2,\ 3,... \end{aligned} \tag{5.4}$$

The previous optimization problem can be solved with quadratic optimization. Therefore, the transformation of the problem to dual problem can be done by reformulating the problem using dual form (a saddle point problem $\min i_w \max_\alpha$) as:

$$\ell(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \alpha_i \Big[y_i(w^T \boldsymbol{x}_i - b) - 1 \Big], \quad i = 1,\ 2,\ 3,... \tag{5.5}$$

Once the objective function is convex, the solution can be found by derivative with respect to (w)

$$\frac{\partial \ell}{\partial w} = w - \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i = 0 \ \Rightarrow w = \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i \tag{5.6}$$

and with respect to (b) as well

$$\frac{\partial \ell}{\partial b} = \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{5.7}$$

Inserting equations (5.6) and (5.7) into equation (5.5) leads to:

$$\begin{aligned} &maximize\ \ell(D) && = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j}^{N} \alpha_i \alpha_j\ y_i y_j\ \boldsymbol{x}_i^T \boldsymbol{x}_j \\ &subject\ to: && \sum_{i=1}^{N} \alpha_i y_i = 0 \\ & && \alpha_i \geq 0 \end{aligned} \tag{5.8}$$

The dual problem depends only on the inner product $\boldsymbol{x}_i^T \boldsymbol{x}_j$. According to the theory of duality, the dual problem will be concave when the original objective function is convex problems. Therefore, the unique solution of the original problem is the unique solution of the dual problem (duality gap is zero).

The KKT (Karush-Kuhn-Tucker) conditions are required to be satisfied to solve the saddle point problem. They can be found by the previous derivatives with respect to w & b to be zero. Furthermore, the constraints are part of these conditions and also the Lagrange multipliers are non-negative. Finally, the important constraint is called "complementary slackness" which should be satisfied too.

$$\frac{\partial \ell}{\partial w} = w - \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i = 0 \tag{5.9}$$

$$\frac{\partial \ell}{\partial b} = \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{5.10}$$

$$y\left(w^T \boldsymbol{x} + b\right) - 1 \geq 0 \tag{5.11}$$

$$\alpha_i \geq 0 \ (Lagrangemultiplier\ condition) \tag{5.12}$$

$$\alpha_i \Big[\ y_i(w^T \boldsymbol{x}_i - b) - 1 \Big] = 0 \quad (complementary\ slackness) \tag{5.13}$$

In case of non-separable (nonlinear SVM) situation:

To separate the data easily, the mapping of the data from the initial space to the feature space is required ($\boldsymbol{x} \Rightarrow \varphi(x)$). This can be done by finding the inner product through the kernel function as:

$$\begin{aligned}
&\boldsymbol{x}^T.\boldsymbol{x}' \Rightarrow \varphi(\boldsymbol{x})^T.\varphi(\boldsymbol{x}') \\
&\varphi(\boldsymbol{x})^T.\varphi(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}') \qquad where\ (k)\ is\ a\ \text{kernel function} \\
&k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}, \boldsymbol{x}' + 1)^d \qquad Polynomial\ Kernel \\
&k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(\gamma \left\| \boldsymbol{x} - \boldsymbol{x}' \right\|^2\right) \quad RBF\ Kernel
\end{aligned} \tag{5.14}$$

The equation (5.8) will become as the equation (5.15) in case of nonlinear SVM:

$$\begin{aligned}
&\max imize\ \ell(D) \quad = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j}^{N} \alpha_i \alpha_j\ y_i y_j\ k\left(\boldsymbol{x}_i^T \boldsymbol{x}_j\right) \\
&subject\ to: \sum_{i=1}^{N} \alpha_i y_i = 0 \\
&\qquad\qquad \alpha_i \geq 0
\end{aligned} \tag{5.15}$$

In case of soft margin:

The slack variable (ζ) should be added to the constraint which allows for violations of the constraint. Thus, The penalty function ($C\sum_i \zeta_i$) also should by add to the objective function as:

$$\begin{aligned} \min_{w,b} mize \quad & \frac{1}{2}\|w\|^2 + C\sum_i \xi_i \\ subject\ to: \quad & y_i(w^T \boldsymbol{x}_i - b) - 1 + \xi_i \geq 0 \\ & \xi_i \geq 0 \end{aligned} \tag{5.16}$$

The optimization problem can be reformulated to be as:

$$\ell(w,b,\alpha,\zeta,\mu) = \frac{1}{2}\|w\|^2 + C\sum_i \zeta_i - \sum_{i=1}^{N} \alpha_i \Big[y_i(w^T \boldsymbol{x}_i - b) - 1 + \zeta_i \Big] - \mu_i \zeta_i \tag{5.17}$$

Then, the KKT (Karush-Kuhn-Tucker)

$$w - \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i = 0 \tag{5.18}$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{5.19}$$

$$C - \alpha_i - \mu_i = 0$$

$$y(w^T \boldsymbol{x} + b) \text{ - } 1 + \zeta_i \geq 0 \tag{5.20}$$

$$\alpha_i \geq 0 \ (Lagrangemultiplier\ condition) \tag{5.21}$$

$$\zeta_i \geq 0 \tag{5.22}$$

$$\mu_i \geq 0 \tag{5.23}$$

$$\alpha_i \Big[y_i(w^T \boldsymbol{x}_i - b) - 1 + \zeta_i \Big] = 0 \quad (complementary\ slackness) \tag{5.24}$$

$$\mu_i \zeta_i = 0 \quad (complementary\ slackness) \tag{5.25}$$

5.1.2 Support vector for regression.

The main purpose of support vector regression (SVR) is to find a function ($\mathrm{f(x)}$) from training data $\{(\mathrm{x}_1, \mathrm{y}_1), \ldots\ldots (\mathrm{x}_L, \mathrm{y}_L)\}$ that has at most deviation

(ε) from the actual target (y_i) as shown in Figure 5.3. The convex optimization problem is:

$$\begin{aligned} &\underset{w,b}{minimize} && \frac{1}{2}\|w\|^2 \\ &subject\ to: && y_i - (w^T \boldsymbol{x}_i) - b \le \varepsilon\ , \quad i = 1,\ 2,\ 3,... \\ & && (w^T \boldsymbol{x}_i) + b - y_i \le \varepsilon \end{aligned} \tag{5.26}$$

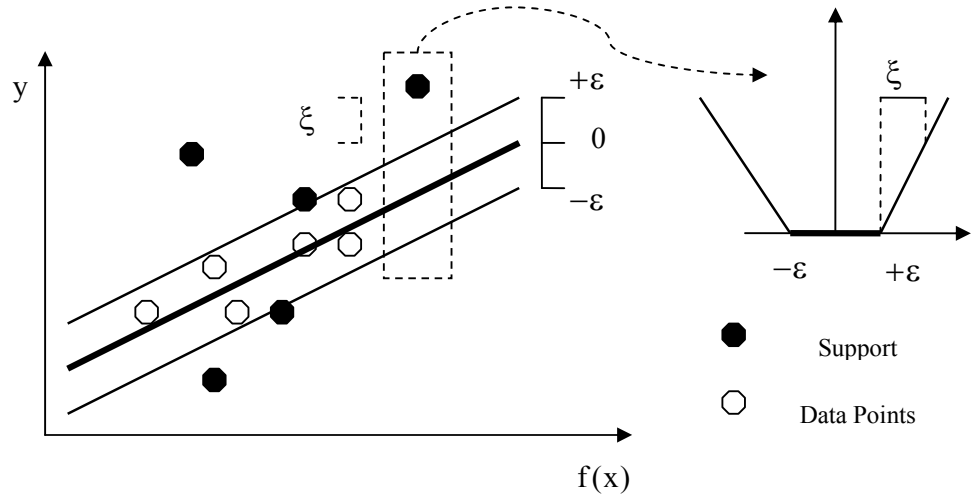


Figure 5.3: Support vectors regression (SVR).

The slack variables (ξ_i, ξ_i^*) are introduced in the case of the soft margin by (Cortes & Vapnik (1995)). Thus, the optimization problem will become as :

$$\begin{aligned} &\underset{w,b}{minimize} && \frac{1}{2}\|w\|^2 + C\sum_{i}^{N}(\xi_i, \xi_i^*) \\ &subject\ to: && y_i - (w^T \boldsymbol{x}_i) - b \le \varepsilon + \xi_i, \quad i = 1,\ 2,\ 3,... \\ & && (w^T \boldsymbol{x}_i) + b - y_i \le \varepsilon + \xi_i^* \\ & && \xi_i, \xi_i^* \qquad \ge 0 \end{aligned} \tag{5.27}$$

The constant $C > 0$ can be found by a trade-off between flatness of function $f(x)$ and deviation larger than (ε − insenstive loss function $\left|\xi\right|_\varepsilon$)as:

$$\left|\xi\right|_\varepsilon = \begin{cases} 0 & if\ \left|\xi\right| \le \varepsilon \\ \left|\xi\right| - \varepsilon & otherwise \end{cases} \tag{5.28}$$

The dual problem formulation of the previous optimization problem (5.27) is written as:

$$\begin{aligned}\ell(w,b,\alpha,\zeta,\mu) \quad &= \quad \frac{1}{2}\|w\|^2 + C\sum_{i}^{N}\ (\xi_i + \xi_i^*) - \sum_{i}^{N}\ (\eta_i\xi_i + \eta_i^*\xi_i^*) \\ &\quad -\sum_{i=1}^{N}\ \alpha_i\Big[\varepsilon + \xi_i - y_i + (w^T\boldsymbol{x}_i) + b\Big] \\ &\quad -\sum_{i=1}^{N}\ \alpha_i^*\Big[\varepsilon + \xi_i^* + y_i - (w^T\boldsymbol{x}_i) - b\Big]\end{aligned} \tag{5.29}$$

where (ℓ) is a lagrangian, and ($\eta_i, \eta_i^*, \alpha_i, \alpha_i^*$) are Lagrange multipliers.

$$\eta_i, \eta_i^*, \alpha_i, \alpha_i^* \geq 0 \tag{5.30}$$

Following the saddle point condition, the derivative of lagrangian with respect to (w, b, ξ_i, ξ_i^*) should vanish as:

$$\frac{\partial \ell}{\partial w} = w - \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\boldsymbol{x}_i = 0 \tag{5.31}$$

$$\frac{\partial \ell}{\partial b} = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \tag{5.32}$$

$$\frac{\partial \ell}{\partial(\xi\ ,\xi^*)} = C - \alpha_i - \alpha_i^* - \eta_i - \eta_i^* = 0 \tag{5.33}$$

Substituting equations (5.31), (5.32) and (5.33) into (5.29) leads to dual problem (5.34).

$$\begin{aligned} &\textit{maximize} \qquad \begin{cases} \dfrac{-1}{2}\displaystyle\sum_{i,j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\big\langle \boldsymbol{x}_i, \boldsymbol{x}_j \big\rangle \\ -\varepsilon\displaystyle\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \end{cases} \\ &\textit{subject to:} \qquad \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \ \text{ and } \alpha_i, \alpha_i^* \ \in \big[0, C\big] \end{aligned} \tag{5.34}$$

Thus:

$$w = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\ \boldsymbol{x}_i \ , \quad \text{f(x)} = \big\langle w\ .\boldsymbol{x} \big\rangle + b \ = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\ \big\langle \boldsymbol{x}_i.\boldsymbol{x} \big\rangle + b \tag{5.35}$$

5.1.3 Multi-output Support Vector Regression (M-SVR).

It is an expansion of the single-output regression in the previous section. There are different models in the literatures such as neural networks which are used to

estimate parameters (regression). Neural networks have poor performance with few data points (Smola & Schölkopf (2004)). So, The M-SVR can be an alternative. Its formulation can be derived similar to the support vector regression and based on (Sanchez-Fernandez et al. (2004) and, Tuia et al. (2011)). Hence, the lagrange formulation is:

$$\begin{aligned} \ell(w,b) &= \frac{1}{2}\sum_{j=1}^{Q} \left\|w^j\right\|^2 + C \sum_{i=1}^{N} L(u_i) \\ & u_i = \left\|e_i\right\| = \sqrt{e_i^T.e_i}\,, \qquad e_i^T = y_i \text{ - } \varphi\left(\boldsymbol{x}_i\right)^T w \text{ - } b^T \end{aligned} \tag{5.36}$$

The insensitive loss function ($L(u_i)$) for multidimensional case:

$$L(u_i) = \begin{cases} 0, & u < \varepsilon \\ u^2 - 2u\varepsilon + \varepsilon^2, & u \geq \varepsilon \end{cases} \tag{5.37}$$

To solve the optimization problem(5.36), the iterative reweighted least-squares (IRWLS) method is used for each iteration (k). First, the approximation of equation (5.36) is done by using first order Taylor expansion of ($L(u)$) such as:

$$l'(w,b) = \frac{1}{2}\sum_{j=1}^{Q} \left\|w^j\right\|^2 + C\left(\sum_{i=1}^{N} L(u_i^k) + \left.\frac{dL(u)}{du}\right|_{u_i^k} \frac{\left(e_i^k\right)^T}{u_i^k}\left[e_i \text{ - } e_i^k\right]\right) \tag{5.38}$$

Thus, the quadratic approximation can be constructed from the equation (5.38).

$$\begin{aligned} \ell''(w,b) &= \frac{1}{2}\sum_{j=1}^{Q} \left\|w^j\right\|^2 + C\left(\sum_{i=1}^{N} L(u_i^k) + \left.\frac{dL(u)}{du}\right|_{u_i^k} \frac{u_i^2 - \left(u_i^k\right)^2}{2u_i^k}\right) \\ &= \frac{1}{2}\sum_{j=1}^{Q} \left\|w^j\right\|^2 + \frac{1}{2}\sum_{i=1}^{N} a_i u_i^2 + \tau \end{aligned} \tag{5.39}$$

where

$$a_i = \frac{C}{u_i^k}\left.\frac{dL(u)}{du}\right|_{u_i^k} = \begin{cases} 0, & u_i^k < \varepsilon \\ \dfrac{2C\left(u_i^k - \varepsilon\right)}{u_i^k} & u_i^k \geq \varepsilon \end{cases} \tag{5.40}$$

(τ) is a sum of constant terms which independent from (w) and (b). To obtain (w) and (b), the solving of a weighted least square problem (5.39) is required. This can be done by derivative with respect to w

$$\frac{\partial \ell''}{\partial w} = w^j - \sum_{i=1}^{N} \varphi(\boldsymbol{x}_i)\, a_i \left(y_{ij} - \varphi(\boldsymbol{x}_i)^T\ w^j - b^j \right) = 0 \tag{5.41}$$

and with respect to b

$$\frac{\partial \ell''}{\partial b} = -\sum_{i=1}^{N} \quad a_i \left(y_{ij} - \varphi(\boldsymbol{x}_i)^T\ w^j - b^j \right) = 0 \tag{5.42}$$

This will lead to a linear system of equation (5.43).

$$\begin{bmatrix} \varphi^T D_a \varphi + I & \varphi^T a \\ a^T \varphi & 1^T a \end{bmatrix} \begin{bmatrix} w^j \\ b^j \end{bmatrix} = \begin{bmatrix} \varphi^T D_a y^j \\ a^T y^j \end{bmatrix} \tag{5.43}$$

where $(j = 1,....,\mathrm{Q})$, $(\varphi = \left[\varphi(\boldsymbol{x}_1),.....,\varphi(\boldsymbol{x}_n)\right]^T)$, $(a = \left[a_1,.....,a_n\right]^T)$, $\left(D_a\right)_{ij} = a_i \delta\left(i-j\right)$, and (1) is an all-one column vector. The linear system (5.43) requires a nonlinear mapping $(k(\boldsymbol{x},\boldsymbol{x}') = \varphi(\boldsymbol{x})^T.\varphi(\boldsymbol{x}'))$. Thus, the learning problem will be a linear combination of the training data in feature space (representer theorem $(w^j = \sum_i \varphi(\boldsymbol{x}_i)\,.\,\beta^j = \varphi^T.\,\beta^j)$). Then, the equation (5.43) will be:

$$\begin{bmatrix} K + D_a^{-1} & 1 \\ a^T K & 1^T a \end{bmatrix} \begin{bmatrix} \beta^j \\ b^j \end{bmatrix} = \begin{bmatrix} y^j \\ a^T y^j \end{bmatrix} \tag{5.44}$$

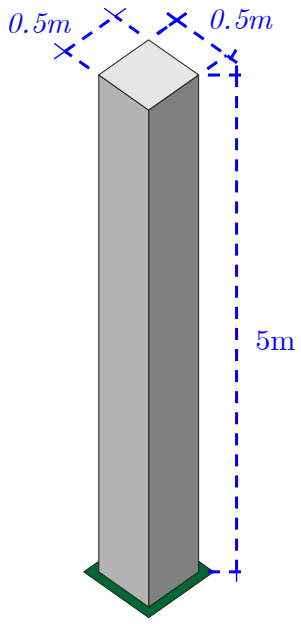
where $((K)_{i,j} = k(\boldsymbol{x}_i,\boldsymbol{x}_j))$ is a kernel matrix. The solution of the problem is found, iteratively. In the beginning, the values of $(\beta\ \&\ b)$ are equal to zero and the variables $(u\ \&\ a)$ are computed. Then, the linear system (5.44) is solved for $(\beta\ \&\ b)$ with fixed value of the variable (a). This operation is repeated iteratively until the convergence is satisfied.

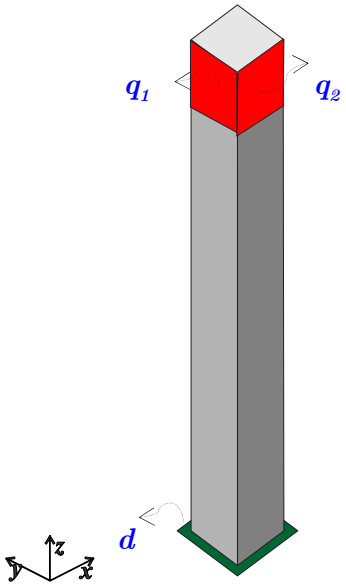
5.1.4 Numerical example of M-SVR.

The column in Figure 5.4 is used as a numerical model. The system was divided into 183 parts to make the regression easily. The input of each part are the degrees of freedom (DOF=231) which contain the applied load (external loads). Thus, the applied loads are taken into account for the regression as:

$$\underbrace{U_t \times (1 + P_{t+1})}_{\textit{Input}} \quad \Rightarrow \quad \underbrace{U^n_{t+1}}_{\textit{Output}} \tag{5.45}$$

where (U_t) is a displacement vector of degrees of freedom which contains applied loads (DOF=231) at the end of time step (t) (converged solution), (1) is an all-one column vector, (P_{t+1}) is a vector of applied force of freedom which contains applied loads at the next time step (t+1), $(U_{t+1})$is a displacement vector (converged solutions) for particular part (n), and (n) is an indication of the parts (n=1,.... 183((200DOF for 182)+143DOF)).





The properties of column system :

E(Young's modulus)=0.5e9Pa	β (Newmark parameter)=0.25
v(Poisson's ratio)=0.45	γ(Newmark parameter)=0.5
ρ(Density)=1600 kg/m^3	q_1 (Load) = 3.63e6N (X-direction)
ζ (Damping ratio) = 0.01,	q_2 (Load) = 3.63e6N (Y-direction)
Δt=0.005s	d(Fixed support)=restriction for all directions
T (Simulation time) =5s	

Figure 5.4: column beam model

The different parts of the system were trained with the regression parameters (Epsilon (ε) = 0.1, C=10, Kernel is a redial basis function (RBF) with

parameter (γ) = (10-100). The M-SVR still an approximation system. Therefore, it is combined with a Newton method. Thus, the new model can give the accurate solution. The Figure 5.5 shows the schematic overview of the algorithm.

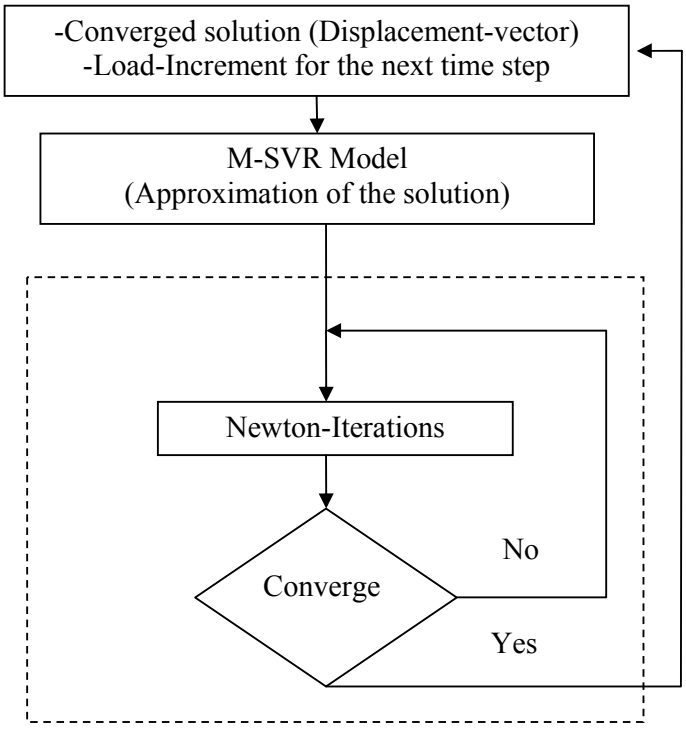
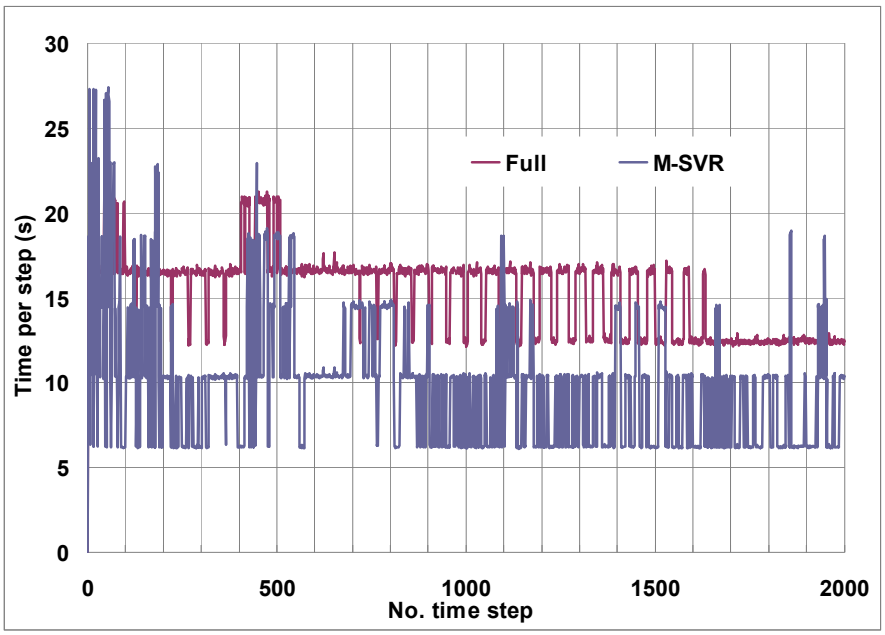


Figure 5.5: Illustration of M-SVR model.

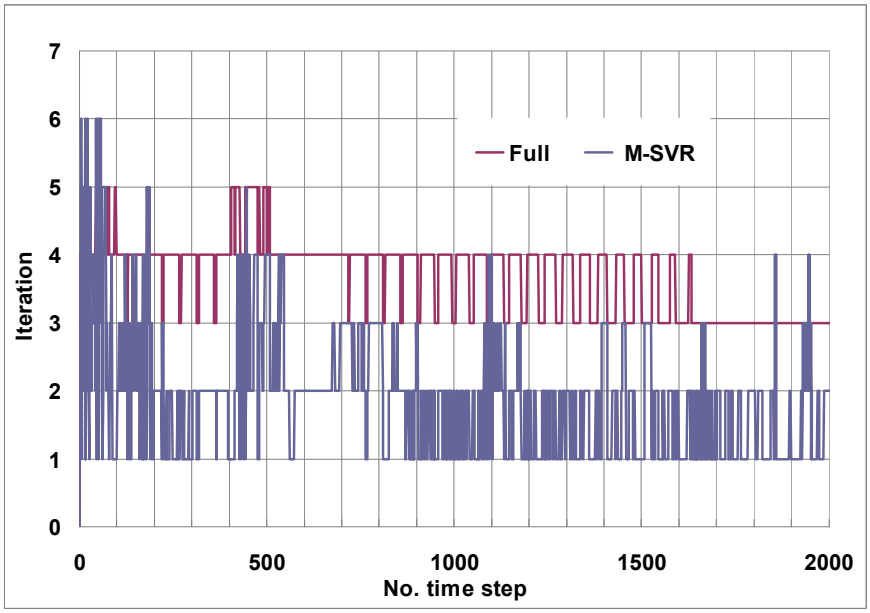
The advantage of surrogate model (M-SVR) can be seen from the number of iteration of the system and the total time of computation for each time step as it is observed in Figure 5.6.

Table 5.1:An overview of column model results (M-SVR)

	Full	**M-SVR**
Tolerance - tol	**1.00E-06**	**1.00E-06**
No. Iterations	7412	3748
Total Time(s)	30730.84	19684.08
Time per step (s)	15.36	9.84
Total Time (%)	100.00	64.06



(a)



(b)

Figure 5.6: Comparison between Full model and M-SVR model.

Chapter 6 Comparison, Conclusions and Recommendations

6.1.1 Comparison of different approaches

The comparison of different reduced model approaches is done for the column model as shown in Figure 6.1. It can be clearly seen that the hyper reduction approach based on missing point estimation seems to be the best. In fact, the hyper reduction approach is good when the system configuration does not change a lot otherwise it can be the worse approach. This is related to that the computing of the masked operator (matrix ($\boldsymbol{D}$)) is computationally expensive, especially when the subspace change a lot. The POD approach with or without BFGS is almost close to each other.

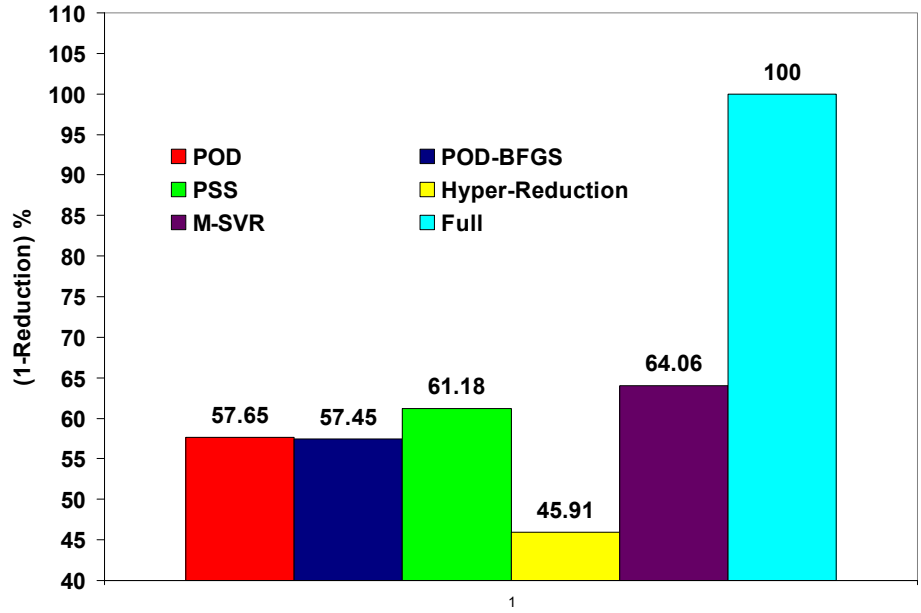


Figure 6.1: comparison of different reduced model approaches based on column model.

The proper snapshots selection approach is almost close to POD approach from the computation time but it is required less adaptation of the subspace. This is in fact related to that the subspace which constructed by PSS is well

representative to the system of interest when it is compared with POD's subspace.

Support vector machine approach is considered as one of the alternative approach of reduced order model. Its disadvantage is that it requires training of the system and its sensitivity is high with small change of the loading parameters. Furthermore, its training is time consuming and it is still a black box.

6.1.2 Conclusions

In this research, one of the challenging problems in the field of computational mechanics called reduced order model is studied for nonlinear structural mechanics systems. Therefore, different proposed approaches were introduced to ensure the performance and accuracy. These approaches were suggested to avoid problems of the classical approaches such as configuration changes with the time. The proposed approaches are compared in the previous section. All approaches except the machine learning approach can be classified as a projection based reduced order model.

The first approach is an adaptive reduced order model based on Proper orthogonal decomposition (POD) while it considered the best technique to construct a projection subspace. The adaptivity is necessary for the POD's subspace while the POD is based on least square optimization. Furthermore, the computing of POD's subspace required the computing of the singular value decomposition which is computational expensive. Therefore, this approach was combined with BFGS method to increase the performance while BFGS compute the tangent matrix based on secant formulation. In fact, The combination of this approach with BFGS leads to almost the same computational time because BFGS has a super convergence and also the projection will leads to further approximation. This will lead to more corrections (adaptivity).

The second approach is adaptive reduced order model base on Proper Snapshots Selection (PSS). This approach is following the same adaptivity with the previous approach. But, the projection subspace was constructed by the suitable snapshots as RITZ based method. This will lead to avoid the computing of

singular value decomposition such as POD. The construction and the adaptation of the projection subspace were done based on the sensitivity analysis. The dimension of projection subspace based on PSS is bigger than the projection subspace based on POD but the PSS's subspace is better approximation for the system (less corrections) when it is compared with POD's subspace.

The third approach is adaptive hyper- reduced order model. This approach is based on the hyper reduction technique called missing point estimation (MPE), which is based on POD. It is elimination of some rows from both residual vector and tangent matrix before projection. This will lead to reduce the computational time while the algebraic system becomes more sparse. Furthermore, the new adaptive strategy called Dynamic-POD is suggested to avoid the stability which may appear from the adaptivity in the case of strong local nonlinearity. This approach has a disadvantage when the configuration of the system changes a lot with the time. This is due to that the finding of elimination rows is based on subspace which will change due to changing the configuration (mode).

The forth approach is a machine learning approach based on multi support vector regression (M-SVR). It is good for some problems in mechanics when the simulations are repeated. Their disadvantages are highly sensitive to the changing of parameters, high training time, and black box.

6.1.3 Limitations and future research

The limitation of the proposed approaches is for the structural mechanics system with strong local nonlinearity such as contact problem, plasticity with finite strain. This kind of nonlinearity needs more studies to find a robust reduced order model due to the fact that the Jacobi matrix of such systems has bad conditional number. Therefore, it will be difficult to get a good approximation by projection based technique even with use the partition techniques. Furthermore, it can not approximate the Jacobi matrix by using the secant formulation such as BFGS method.

The solution of the previous kind of problem is a domain decomposition technique such as Finite Element tearing and Interconnecting (FETI). So, the entire structure can be divided into sub domains. Then, a reduced order model will implement in the regions when linear and weak nonlinear problems are appearing.

In fact, this approach (FETI) needs computational time to combine the sub-domains together. Finally, the further research is required for such high strong local nonlinearity problem in structural mechanics

Appendices

A. Programming Details

The algorithm is implemented in MATLAB platform and software compiled with C++ code linked by MEX function. The software is responsible for generate and assemble the stiffness matrix and internal force vector. All the rest, it is developing in MATLAB including matrix operations and linear system solve. The parallel processing with shared memory is applied in the MATLAB platform through its parallel computing toolbox applied in the linear system solver. The pre-processing and post-processing is developed in the GiD software. All examples are processed in a personal computer with processor of 8 cores. Both reduced and full models keep a parallel processing in the same parts of the code for a fair comparison.

B. Parameters related to adaptive model order reduction (POD&BFGS).

n_k	2	k_Φ	0.001
$n_{\Phi_{stable}}$	10	k_r	0.95
n_{stable}	25	tol_{POD}	1e-3
n_{max}	50	tol_{BFGS}	1e-8
tol_R	1e-6	tol_u	1e-6

C. Parameters related to adaptive model order reduction (PSS).

n_k	2	k_Φ	0.001
$n_{\Phi_{stable}}$	10	k_r	0.95
n_{stable}	25	tol_{PSSr}	1e-3
n_{max}	50	tol_{PSSe}	1e-6
tol_R	1e-6	tol_u	1e-6

Bibliography

- Almroth B.O., Stern P. & Brogan F. A. "Automatic choice of global shape functions in structural analysis". AIAA Journal, **16(5)**, 525–528, 1978.
- Amabili M., Sarkar A. & Paidoussis MP. "Reduced-order models for nonlinear vibrations of cylindrical shells via the proper orthogonal decomposition method". Journal of Fluids and Structures, **18**, 227–250, 2003.
- Amsallem D., Carlberg K., Cortial J.& Farhat C. "A method for interpolating on manifolds structural dynamics reduced-order models" International Journal for Numerical Methods in Engineering , 78, 275–295, 2009.
- Astrid P., Weiland S., Willcox K., & Backx T. "Missing point estimation in models described by proper orthogonal decomposition". IEEE Transactions on Automatic Control, 53, 2237–2251, 2008.
- Avery P., Farhat C., Reese G. "Fast frequency sweep computations using a multi-point Padé-based reconstruction method and an efficient iterative solver". International Journal for Numerical Methods in Engineering; 69(13):2848–287., 2007.
- Bergmann M., Cordier L.& Brancher J-P "Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model" Physics of Fluids, 80–101, 2005.
- Bottou L. & Line C. -J. "Support vector machine solvers" Large Scale Kernel Machines. L. Bouttou, O.chappelle, D.DeCoste and J.Weston, eds., MIT Press, 2007.
- Cabos C. "Error bounds for dynamic responses in forced vibration problems". SIAM J Sci Comput 15 (1):1-15, 1994.
- Carlberg K. & Farhat C. "An adaptive POD-Krylov Reduced order Model for Structural Optimization" 8th World Congress on Structural and Multidisciplinary Optimization, Lisbon. Portugal, 2009.
- Carlberg K., Bou-Mosleh C & Farhat C. "Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and

compressive tensor approximations" Int. J. Numer. Meth. Engng, 86,155–181, 2011.

- Carlberg K. & Farhat C. "A low-cost, goal-oriented (compact proper orthognol decomposition) basis for model reduction of static systems" International Journal for Numerical Methods in Engineering, 86, 381–402, 2011.
- Chan A. S. L. & Hsiao K. M. "Nonlinear analysis using Areduced number of variables". Comp.Methods Appl.Mech.Eng, 52, 899-913, 1985.
- Chaturantabut S.,& Sorensen D. C. "Non-linear model reduction via Discrete empirical Interpolation". SIAM J Sci Comput 32 (5):2737-2764, 2010.
- Cortes C. & Vapnik V "Support vector networks" machine learning, 20: 273-297, 1995.
- Du J., Fang F., Pain C. C., Navon I. M., Zhu J., & Ham D. A. " POD reduced-order unconstructed mesh modeling applied to 2D and 3D fluid flow" computers and Mathematics with applications, 65, 362-379, 2013.
- Dulong J., Druesne F., & Villon P. "A model reduction approach for real-time part deformation with nonlinear mechanical behavior" Int J Interact Des Manuf , 1, 229–238, 2007.
- Epureanu BI. "A parametric analysis of reduced order models of viscous flows in turbo machinery". Journal of Fluids and Structures, 17, 971–982, 2003.
- Eriksson K. Estep D., Hansbo P. & Johnson C. "Computational Differential Equations". Cambridge University Press, 1996.
- Everson R. & Sirovich L. "The Karhunen-Loeve procedure for gappy data."Journal Opt. Soc. Am., 12:1657-1664, 1995.
- Farhat C., Avery P.,Chapman T.,&Cortial J. "Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency". Int. J. Numer. Meth. Engng, , 2014.
- Grepl MA., Maday Y., Nguyen NC., AT. "Patera Efficient reduced-basis treatment of nonaffine and nonlinear partial differential

equations". Mathematical Modeling and Numerical Analysis, **41**, 575–605, 2007.

- Hetmaniuk U., Tezaur R., Farhat C. "An adaptive scheme for a class of interpolatory model reduction methods for frequency response problems". International Journal for Numerical Methods in Engineering; **93**, 1109–1124, 2013.
- Ibrahimbegovic A. & Wilson E. L. "Automated truncation of Ritz vector basis in modal transformation". Journal of Engineering Mechanics ASCE;116(11), 2506–2520, 1990.
- Idelson S. R. & Cardona A. "A load-dependent basis for reduced nonlinear structural dynamic". *Computer & structure*; 20, 203–210, 1985.
- Joo K.L., Wilson E.L., Legrer P. "Ritz vector and generation criteria for mode superposition analysis" Earthquake Engineering and Structural Dynamics, 18, 149–167, 1989.
- Kapania R. K. & Byun C. "Reduction methods based on eigenvectors and Ritz vectors for nonlinear transient analysis" computational mechanics, 11, 65-82, 1993.
- Kerfriden P., Gosselet P., Adhikari S., & Bordas S. "Bridging Proper Orthogonal Decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems" Comp.Methods Appl.Mech.Eng, 200, 850-866, 2011.
- Kerschen G., Golinval J.C., Vakakis A., & Bergman L. "The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: an overview. Nonlinear Dynamic. **41**, 147–169, 2005.
- Kline K .A. "Dynamic analysis using a reduced basis of exact modes and Ritz vectors" AIAA Journal, 24, 2022–2029, 1986.
- Kreuzer E., Kust O. "Analysis of long torsional springs by proper orthogonal decomposition" Archive of Applied Mechanics, 67, 68–80, 1996.

- Krysl P., Lall S., &Marsden J.E. "Dimensional model reduction in nonlinear finite element dynamics of solids and structures" Int. J. Numer. Methods Eng, 51, 479–504, 2001.
- Kunisch K. & Volkwein S. "Glarcian Proper orthogonal Decomposition Methods for a General Equation in Fluid Dynamics" SIAM Jornal on Numerical Analysis, 40, 492-515, 2002.
- Lenaerts V., Kerschen G., Golinval J., &Chevreuils C. D. " Proper orthogonal decomposition for model updating of nonlinear mechanical systems" GOLINVAL Mechanical Systems and Signal Processing, 31–43 , 2001.
- Matthies H., Strang G. "The solution of nonlinear finite element equations" International Journal for Numerical Methods in Engineering, Vol.14, No.11, 31–43 , 1979.
- Meyer M. "Computational Model Order Reduction of Linear and Nonlinear Dynamical systems" Institute of scientific computing, TU-Braunschweig, 2006.
- Nickell R. "Nonlinear dynamics by mode superposition" Comput. Methods Appl. Mech. Eng, 107–129, 1976.
- Nilsson, N.J. "Learning machines" McGraw Hill. New Yourk, 1965.
- Noor A. K. & Peters J. M. "Reduced basis techniques for nonlinear analysis of structures" AIAA Journal,18, 455–462, 1980.
- Noor A. K. & Peters J. M. "Instability analysis of space trusses" Comp.Methods Appl.Mech.Eng, 40, 199-218, 1983.
- Noor A. K. "Recent advances and applications of reduction methods" Appl. Mech. Review, 125–146, 1994.
- Nour-Omid B. & Clough R.W. "Dynamic analysis of structures using Lanczos co-ordinates" "Dynamic Analysis by direct super position of Ritz vector" Earthquake Engineering and structural dynamics, 12, 565-577, 1984.
- Price N. D., Reed J. L., Papin J. A., Famili I., Palsson B. O. "Analysis Metabolic Capabilities Using Singular Value Decomposition of Extreme Pathway Matrices" Biophysical Journal. 84,794-804, 2003.
- Radermacher A. & Reese S. "A comparison of projection-based model reduction concepts in the context of nonlinear Biomechanics" Journal

of Archive of applied mechanics, Spring-Verlag Berlin Heidelberg, 2013.

- Reese S. & Radermacher A. "A new strategy for model reduction in nonlinear biomechanics" proceedings for the 23rd ICTM, Beiijing (China), august 2012.
- Ryckelynck D. "A poriori Hyperreduction method: an adaptive approach" Journal of Computational Physics,202, 346-366, 2005.
- Sanchez-Fernandez M., De-Prado-Cumplido M., Arenas-Garcia J., & Perez-Cruz F. "SVM multiregression for nonlinear channel estimation in multi-input multi-Output systems" IEEE Trans.Signal Proc., 52(8),2298-2307,2004
- Shao A., Xi S., Qiu C. " A variational method for correcting Nonsystematic Errors in Numerical Weather Prediction" Science in China Series B, Chemistry, Life science and Earth science, 52, 1650-1660, 2009.
- Sirovich L. "Turbulence and the dynamics of coherent structures" Part I: Coherent structures. Quarterly of Applied Mathematics, 45, 561–571, 1987.
- Slaats P.M.A, de Jonge J., Sauren "Model reduction tools for nonlinear structural dynamics" Computaer & structures, 54, 1155–1171, 1995.
- Smola A. J., Schölkopf B. " A tutorial on support vector regression" Stat.Comput, 14(3), 199–222, 2004.
- Spiess H. & Wriggers, P. "Reduction methods for FE analysis in nonlinear structural dynamics" PAMM. Proc. Appl. Math. Mech, 5, 135-136, 2005.
- Spiess H. "Reduction Methods in Finite Element Analysis of Nonlinear Structural Dynamics" Dissertation, IBNM.Leibniz University of Hannover, February, 2006.
- Thomas J. P., Dowell E. H., Hall K. C. "Three-dimensional transonic aeroelasticity using proper orthogonal decomposition-based reduced-order models" *Journal of Aircraft;* **40**(3):544–551.2003.
- Tuia D., Verrelst J., Alonso L., Perez-Cruz F., & Camps-Valls G. "Multioutput support vector regression for remote sensing biophysical

parameter estimation" IEEE Geoscience and Remote sensing Letters, 2011.

- Vapnik V. and Lerner A. "Patter recognition using generalized portrait method" Automation and remote control, 24: 774-780,1963
- Vapnik V. "the nature of statistic learning theory" Springer, 1995.
- Vendl, A. & Faβbender, H. "Proper Orthogonal Decomposition for steady aerodynamic applications" PAMM. Proc. Appl. Math. Mech, 10, 635-636, 2010.
- Vendl, A. & Faβbender, H. "Missing Point Estimation for Steady Aerodynamic Applications" PAMM. Proc. Appl. Math. Mech,11,839-840, 2011.
- Wilson E. L., Yuan M.,& Dickens J.M "Dynamic Analysis by direct super position of Ritz vector" Earthquake Engineering and structural dynamics, 10, 813-821, 1982.
- Wriggers,P"Nonlinear Finite Element Methods" Spring-Verlag Berlin Heidelberg, 2008.
- Yuan G. & Yao S. "A BFGS algorithm for solving symmetric nonlinear equations." Journal of Mathematical Programming and operations research, 2011.
- Zienkiewicz. O.C & Taylor "The finite element method", 2000.

Curriculum vitae

Name:	Mohamed Ahmed Annour Anndif
Date of birth	29.03.1984
Place of birth	Alghrifa, Libya
Nationality:	Libyan
Marital Status	Single

Education

2002-2007	BSc. Civil engineering *Sebha university*, Sebha, Libya
2008-2010	MSc. Construction engineering and management *Arab academy for Science and technology*, Alexandria, Egypt
2012	MSc. Thesis Defense (interruption of North-Africa situation 2011) *Arab academy for Science and technology*, Alexandria, Egypt
2010-2012	MSc. Advanced computational & Structural Engineering *TU-Dresden*, Dresden, Germany.

Occupation

2012-2015	Research assistance at the institute of continuum mechanics *Leibniz University of Hannover*, Hannover, Germany.