

Doctoral Thesis

New model order reduction strategies in earthquake engineering and structural dynamics

submitted in satisfaction of the requirements for degree of
Doctor of Science in Civil Engineering
of the Vienna University of Technology, Faculty of Civil Engineering

Dissertation

Neue Modellreduktionsstrategien im Erdbebeningenieurwesen und in der Strukturdynamik

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften
eingereicht an der Technischen Universität Wien, Fakultät für Bauingenieurwesen
von

Dipl.-Ing. **Franz Bamer**
Matrikelnummer 0227201
3143 Pyhra, Hauptstraße 26, Österreich

1. Gutachter: Univ.Prof. Dipl.-Ing. Dr.techn. **Christian Bucher**
Institut für Hochbau und Technologie
Technische Universität Wien
Karlsplatz 13 / E2063, 1040, Wien, Österreich
2. Gutachter: Univ.Prof. Dipl.-Ing. Dr.techn. **Christoph Adam**
Institut für Grundlagen der Technischen Wissenschaften
Universität Innsbruck
Technikerstraße 13, 6020 Innsbruck, Österreich
3. Gutachter: Professor Dr. Sc. **Alexander Belyaev**
Institute of Applied Mathematics and Mechanics
State Polytechnical University of St. Petersburg
29, Polytechnicheskaya st., 195251, St.Petersburg, Russia

Wien, Juli, 2014

Vorwort

Diese Dissertation ist das Ergebnis der Forschungsarbeit, die ich auf der Technischen Universität Wien ausübte. Ich durfte hier am Institut für Hochbau und Technologie in dem Forschungsbereich für Allgemeine Mechanik und Baudynamik für vier Jahre als Universitätsassistent meine Dienste verrichten. So war es mir möglich nicht nur durch meine Tätigkeit in der Forschung sondern auch durch die Ausübung der Lehre eine hervorragende und breit gefächerte Ausbildung im Bereich der technischen Mechanik zu erhalten.

Mein Betreuer, Herr Professor Christian Bucher, stand mir mit Rat und Tat stets zur Seite und betreute mich die ganze Zeit über in meiner Forschungstätigkeit. Er setzte immer die richtigen Akzente um mein Vorankommen bei der Dissertation optimal zu steuern. Wir diskutierten regelmäßig Vorgangsweisen und fachliche Probleme, die durch sein breitgefächertes Wissen rasch beseitigt werden konnten.

Ich möchte Herrn Professor Christoph Adam für die genaue Begutachtung und die konstruktiven Anregungen bedanken. Er hat mit sehr großem Arbeitsaufwand dazu beigetragen die Dissertation zu verbessern.

Auch dem Drittgutachter, Herrn Professor Belyaev, gilt großer Dank, der, obwohl er unter beruflichem Zeitdruck stand, sich für mich die Zeit nahm und die Arbeit zügig bewertete.

Durch die tollen Mitarbeiter am Institut kam ein hervorragendes Arbeitsklima zustande. Nicht selten führten Diskussionen bei der Kaffeemaschine oder beim Mittagessen dazu Probleme schneller zu lösen oder sogar dafür zu sorgen, dass viele Probleme im Kern erstickt wurden und daher gar nicht erst auftraten. Es war mir eine Freude in diesem Zentrum des Wissens arbeiten zu dürfen.

Sehr wichtig war für mich in dieser Zeit auch der Beistand der Familie. Ich wurde stets unterstützt und, obwohl ich vor zehn Jahren nicht einmal im Traum daran gedacht hatte, dass ich jetzt soweit gekommen bin, zweifelten sie niemals an mir und gaben mir somit den nötigen Ansporn.

Diese Dissertation wird im Rahmen des internationalen Doktoratskollege, "Vienna Doctoral Programme on Water Resource Systems" (DK-Plus W1219), abgeschlossen, der ich als "Associate Student" zugeteilt war. So war es mir auch möglich mich mit Dissertanten anderer Gebiete regelmäßig auszutauschen und mein Blickfeld somit zu erweitern und Parallelen zu erkennen.

Abstract

The evaluation of the response of a structure in the time domain is without any doubt one of the main tasks in structural dynamics and earthquake engineering. In order to create proper models, which are able to describe the main behavior of the system, a high dimensional spatial discretization by e.g. finite elements is often inevitable.

High dimensionality demands computational effort. In the literature a large number of numerical and analytical methods are discussed to solve linear and nonlinear ordinary differential equations, but they all involve high dimensional matrix operations that either lead to a long calculation time and/or demand a huge amount of computational storage.

This dissertation deals with model order reduction (MOR) of linear and nonlinear systems in order to minimize the computational effort while preserving the main properties of the dynamical system. The proper orthogonal decomposition (POD) method provides a low dimensional description of a high dimensional process and is presented in this work as useful model order reduction method in earthquake engineering and structural dynamics. Several new strategies based on the POD method are discussed.

The POD method is applied to earthquake excited linear and nonlinear frame structures. Nonlinearities are a result of plastic material behavior and of friction based seismic isolation devices located at the basements of the structure. In the linear case the POD method is compared to the method of modal truncation. It provides a useful alternative to the classical method. In the nonlinear case an accurate approximation of the response is achieved. The POD method is implemented in that way that snapshots (observations) are taken by integrating over a small time period of the beginning of the earthquake excitation and with this information the transformation matrix into the POD subspace is generated. Subsequently, the system is integrated numerically over the whole time period in the reduced space. In the nonlinear case it is important to capture the relevant nonlinear behavior in the snapshot time period in order to be able to describe the nonlinear response.

The MOR reduction procedure utilizing the POD is expanded to more than one earthquake excitation. With the information of the response of the structure to a small time segment of one earthquake a transformation matrix is generated, which is not only used to reduce the system with this excitation but also for the reduction process of this structure excited by other earthquakes. The new strategy, which is called "universal" POD MOR strategy in this work, is tested on far-fault and near-fault events.

Subsequently, this "universal" MOR method is verified on a three-dimensional nonlinear building construction. The nonlinear effect is realized by a bilinear stress-strain curve in axial direction in order to model elastic-plastic material behavior. The method is tested on six representative earthquake events. Not only for the two but also for the three dimensional test object the "universal" MOR strategy provides accurate approximations of the full system if the nonlinear motion forms can be mapped in the snapshot matrix.

The "universal" MOR strategy is applied to realize high dimensional nonlinear Monte Carlo Simulations using explicit numeric integration. The procedure is tested on a two-dimensional nonlinear frame structure with a moderate number of degrees of freedom in order to compare the results of the reduced with those of the full simulation. The introduced Monte Carlo Simulation strategy is presented on a high dimensional more complex model, where the advantages can be pointed out entirely. The combination of POD reduction and explicit integrators does not only provide a fast calculation procedure but also the possibility

to implement a multithreaded algorithm, which allows the calculation of a large number of sample records.

The full eigenvalue solution becomes expensive if the number of degrees of freedom is large. Therefore, even the analytic calculation algorithms are time-consuming. A modal truncation with only a small number of lower modes associated with the solution of only a small part of the eigenvalue problem often leads to an effective and accurate approximation of the full problem. This method fails for impact problems, where high frequency motions are responsible for stresses and forces near the impact point of the structure. For this class of problems a new hybrid MOR strategy is presented. The high frequency motions are captured in the snapshots, which are taken directly after the impact time instant. The low frequency motion is captured by the truncation of a few lower modes, which are added into the snapshot matrix. An accurate approximation can be achieved by the application of this method, whereas the method of modal truncation fails completely.

Kurzfassung

Die Berechnung der Antwort einer Struktur im Zeitbereich ist ohne Zweifel einer der wichtigsten Herausforderungen im Erdbebeningenieurwesen und in der Strukturmechanik. Um geeignete Modelle zu erzeugen, mit denen das grundsätzliche dynamische Verhalten beschrieben werden kann, ist eine hoch-dimensionale räumliche Diskretisierung durch z.B. finite Elemente oft unerlässlich.

Hoch-dimensionale Berechnungen sind kostspielig. Es existiert eine Vielzahl an analytischen und numerischen Methoden zur Lösung von linearen und nicht linearen gewöhnlichen Differenzialgleichungen, jedoch sind diese alle mit kostspieligen Matrixoperationen verbunden, die zu langen Rechenzeiten führen oder große Speicherkapazitäten benötigen.

Diese Dissertation befasst sich mit Modellreduktionsmethoden linearer und nichtlinearer Systeme um den Rechenaufwand zu minimieren, jedoch dabei die wichtigen Eigenschaften des dynamischen Systems beizubehalten. Die "Proper Orthogonal Dekomposition" (POD) Methode ist in der Lage eine niedrig-dimensionale Darstellung eines hoch-dimensionalen Prozesses zu erzeugen. In dieser Arbeit wird sie als sinnvolle Modellreduktionsmethode im Erdbebeningenieurwesen und in der Strukturmechanik vorgestellt. Einige neue Strategien, die auf das Prinzip dieser Methode aufbauen, werden diskutiert.

Die POD Methode wird bei einer erdbebenangeregten linearen und nichtlinearen Struktur angewendet. Nichtlinearitäten werden durch Reibungselemente hervorgerufen, die unter den Fundamenten der Struktur angebracht sind. Im linearen Fall wird die POD Methode mit der Methode der Modalen Reduktion verglichen. Hier stellt sie eine sinnvolle Alternative zu dieser klassischen Methode dar. Im nichtlinearen Fall stellt sich heraus, dass die POD Lösung eine gute Approximierung zur Lösung des gesamten Systems ist. Die POD Methode ist so realisiert, dass "Schnappschüsse" (Beobachtungen) gemacht werden, indem zu Beginn der Erdbebenanregung über eine kurze Zeitspanne numerisch integriert wird, und mit diesem Informationsgehalt die Transformationsmatrix in den reduzierten Raum erstellt wird. Anschließend wird das System im reduzierten Raum über die gesamte Zeitspanne des Erdbebens numerisch integriert. Im nichtlinearen Fall ist es essentiell nichtlineares Verhalten im Beobachtungszeitraum zu erfassen um in der Lage zu sein nichtlineare Moden zu definieren.

Die neue Modellreduktionsstrategie wird auf ein größeres Anwendungsgebiet als nur für eine einzige Erdbebenanregung erweitert. Mit der Information eines kleinen Ausschnittes des Antwortverlaufes zu einer Erdbebenanregung wird eine Transformationsmatrix generiert, die nicht nur für die Transformation des Systems mit dieser Anregung in den reduzierten Raum herangezogen wird, sondern auch zur Reduktion derselben Struktur, angeregt durch andere Erdbebenschriebe. Diese neue Strategie, in dieser Arbeit "universelle" Modellreduktion, wird für Erdbeben, die nahe dem Epizentrum und weit weg vom Epizentrum gemessen wurden, getestet.

Anschließend wird die "universelle" Modellreduktionsstrategie für hoch-dimensionale nichtlineare Monte Carlo Simulationen angewendet, die mithilfe expliziter numerischer Integration durchgeführt werden. Die Prozedur wird an einer nichtlinearen zweidimensionalen Rahmenstruktur mit einer verhältnismäßig geringen Anzahl an Freiheitsgraden getestet um einen Vergleich mit der Lösung des nichtreduzierten Modells zu ermöglichen. Die vorgestellte Monte Carlo Simulation wird nun an einer vergleichsweise hochdimensionalen nichtlinearen Gebäudestruktur getestet, wo die Vorteile der neuen Strategie voll zur Wirkung

kommen. Die Kombination von POD Reduktion und expliziter Integration zeichnet sich nicht nur durch eine beträchtliche Verringerung der Berechnungszeit aus, sondern auch durch die Möglichkeit Multithread Algorithmen zu implementieren, wodurch die Anzahl der Stich-Proben nochmal beträchtlich erhöht werden kann.

Die Berechnung des vollen Eigenwertproblems ist für hoch dimensionale Systeme sehr rechenintensiv. Folglich werden auch analytische Lösungsalgorithmen zeitintensiv. Durch die Abspaltung einer geringen Anzahl von niedrigfrequenten Strukturmoden muss nur mehr ein kleiner Teil der Eigenwertanalyse durchgeführt werden. In der Strukturmechanik führt diese Modale Reduktion oft zu einer genauen Annäherung des vollen Problems. Diese Methode versagt z.B bei der Berechnung von Stoßproblemen, wo Kräfte und Spannungen in der Nähe des Kontaktes von der Antwort der hochfrequenten Moden abhängen. Aus diesem Grund wird eine neue Hybrid-Modellreduktionsstrategie vorgeschlagen. Die hochfrequenten Bewegungsformen werden in den Schnappschüssen durch Zeitintegration direkt nach dem Kontakt berücksichtigt. Die niedrigfrequenten globalen Moden werden durch wenige niedrigfrequente Strukturmoden realisiert und zur Snapshot-Matrix hinzugefügt. Die Approximation der Hybridmethode ist vielversprechend, wo die klassische Methode der Modalen Reduktion versagt.

Contents

1	Motivation	1
2	Introduction	3
3	Theoretical fundamentals	8
3.1	Equations of motion	8
3.2	The linear system	9
3.2.1	Natural vibration	9
3.2.2	Modal analysis	10
3.2.3	Duhamel Integral	11
3.2.4	Piecewise analytic response	12
3.2.5	Transformation into the state space and modal analysis	14
3.2.6	Piecewise analytic response in the state space	15
3.2.7	Direct time integration	15
3.3	Nonlinear systems	21
3.3.1	Direct time integration	22
3.4	Model order reduction	25
3.4.1	Modal truncation	26
3.4.2	Proper Orthogonal Decomposition	26
3.5	Summary	29
4	The proper orthogonal decomposition as model order reduction method in earthquake engineering	29
4.1	Application of the proper orthogonal decomposition for structures under transient excitation	29
4.2	The procedure for linear structures	31
4.2.1	Linear dynamic model	31
4.3	The procedure for nonlinear structures	36
4.3.1	Nonlinear dynamic model	36
4.3.2	POD modes	39
4.3.3	Numerical efficiency	42
4.3.4	Conclusion	43
4.4	A universal model order reduction method	44
4.4.1	Dynamic structure - test object	46
4.4.2	Earthquake excitation	47
4.4.3	Numerical demonstration	50
4.4.4	Numerical efficiency	55
4.4.5	Conclusion	57
4.5	Application of the universal model order reduction strategy to structures with elastoplastic material properties	58
4.5.1	Structural model	58
4.5.2	Numerical demonstration	61
4.5.3	Conclusion	66
4.6	An optimized Monte Carlo simulation strategy by the proper orthogonal decomposition	66

4.6.1	Earthquake sampling	66
4.6.2	"Universal" POD reduction - statistical evaluation	67
4.6.3	"Universal" POD reduction - Monte Carlo Simulation numerical example	68
4.6.4	Conclusion	72
5	A new hybrid model order reduction strategy for impact problems	73
5.1	Numerical demonstration	75
5.2	Numerical efficiency and limits	78
5.3	Conclusion	79
6	General conclusion	79
A	Implementation Codes	87
A.1	Implementation of piecwise analytic solution methods	87
A.1.1	Implementation of the response calculation of a structure to a piecewise constant force function	87
A.2	Implementation of numeric time integration methods for linear systems . . .	89
A.2.1	Implementation of the central difference algorithm for linear systems	89
A.2.2	Implementation of the Newmark algorithm for linear systems	90
A.2.3	Numerical demonstration and test of the presented numeric time integration methods for linear systems	92
A.3	Implementation of numeric time integration methods for nonlinear systems .	95
A.3.1	Implementation of the central difference algorithm for nonlinear systems	95
A.3.2	Implementation of the Newmark algorithm for nonlinear systems . . .	96
A.3.3	Numerical demonstration and test of the presented numeric time integration methods for nonlinear systems	97
A.4	Implementation of mechanical models	100
A.4.1	Two dimensional frame structure	100
A.4.2	Three dimensional frame structure	102
A.4.3	Three dimensional building construction - Las Vegas building	106
A.5	Earthquake generation	109
A.6	Damping	111
B	Test run demonstration of the Monte Carlo Simulation for a three-dimensional building structure	112

1 Motivation

An important task in structural dynamics is the evaluation of the solution of a dynamical system. Models in structural dynamics consist of a system of partial and/or ordinary dif-

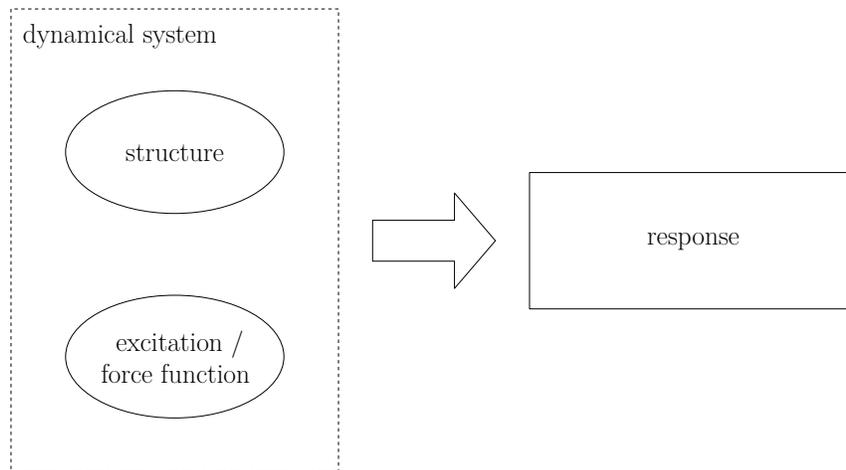


Figure 1: Overview basic goal

ferential equations, supplemented with boundary conditions. It is often defined that the right hand side describes an external force function, the excitation, and the left hand side describes the structural resistance dependent on the state variables, the displacement vector and its derivatives. The main goal of the engineer is to calculate the reaction of a structure subjected to external influences, i.e. to evaluate the maximum displacements and the maximum internal forces to be able to predict structural resistance. Fig. 1 gives a sketch to focus on the main idea of this work, which is to obtain a useful approximation of the solution of the differential equation as fast as possible.

In engineering it is common practice to determine three-dimensional structures and create two-dimensional surrogate models that are able to describe the main dynamical behavior of the system. Often this process is not trivial and demands a lot of experience.

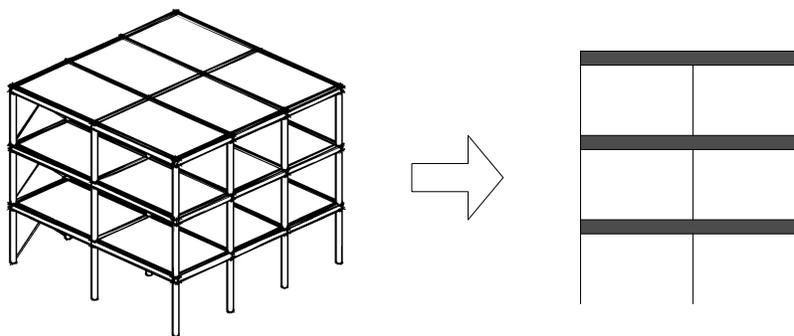


Figure 2: Simple modeling

Fig. 2 presents a rather easy symmetric and consistent supporting structure of a building and its linear three degrees of freedom surrogate model. Here the aim must be to capture

as much dynamic information into the surrogate model as possible, but even here essential motion patterns, such as torsional modes, are neglected. The next step is to create a more complicated surrogate model in order to describe also torsional and other essential motion forms, or to switch to other modeling strategies. As even for rather simple structures modeling as presented in Fig. 2 can be a difficult challenge, it is risky or even sometimes impossible to capture all important motion patterns of a complicated asymmetric structure.

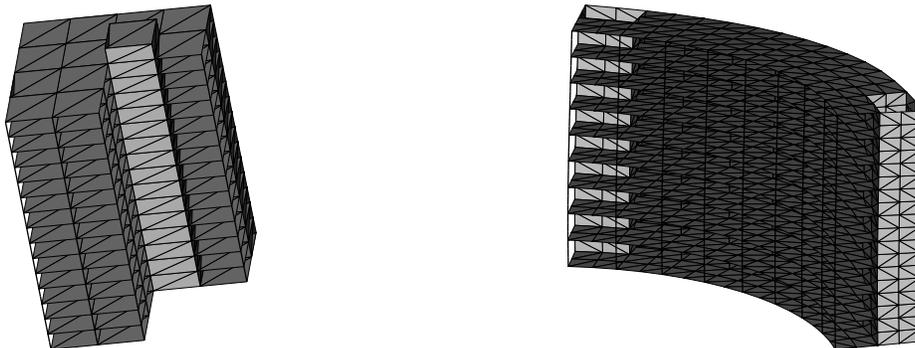


Figure 3: Examples for complicated non symmetric structures

Fig. 3 presents structures, that require a spacial discretization, e.g. by finite elements, in order to create proper models. Finely meshed models lead to high dimensional equations of motion that demand a lot of computational storage and/or time. The implication is to switch to low dimensional descriptions that approximate the full process as accurately as possible.

Therefore, this work presents model order reduction strategies that have potential concerning the application in earthquake engineering as well as other problems in structural dynamics for linear as well as nonlinear systems.

In the following an academic example of model order reduction is presented, which demonstrates the basic idea of this dissertation. It should be mentioned that theoretical details are not explained in this motivation example, however, the reader is referred to the theoretical parts of this dissertation. Imagine a structure, e.g. the FE discretized structure on the right side of Fig. 3, subjected to a harmonic ground induced motion $\ddot{\mathbf{x}}_{\mathbf{g}} = \mathbf{x}_0 \cos \nu t$. Now the goal is to find the harmonic oscillation of the structure in the stationary state, i.e. after the transient effect, which is equivalent to the particular solution of the differential equation to the force function induced by the ground motion. The high dimensional linear equation of motion is then

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = -\mathbf{M}\ddot{\mathbf{x}}_{\mathbf{g}} \quad .$$

The particular solution to the differential equation is then the summation of particular response functions in the uncoupled modal space

$$\mathbf{x} = \sum_{k=1}^n \Phi_k a_{0,k} \cos(\nu t - \beta) = \Phi \mathbf{a}_0 \cos(\nu t - \beta) = \mathbf{a}_p \cos(\nu t - \beta) \quad ,$$

where the vector \mathbf{a}_0 and the delay angle β can be evaluated by insertion of the solution ansatz function into the equation of motion. Generally all modes must be considered to obtain the

exact solution vector although different components of \mathbf{x} oscillate with the same oscillation frequency however with different amplitude. As a consequence it should be possible to describe the oscillation by only one displacement pattern (which is a linear combination of all modes) and therefore by one equation of motion.

The observation of the solution vector at arbitrary different time instants leads to a matrix that is assembled of a set of s records of the deformation status of the structure at these time instants

$$\mathbf{X}_S = [\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_s)] \quad ,$$

which leads in this special case of the particular solution response function to the assumed harmonic excitation to

$$\mathbf{X}_S = [\mathbf{a}_p \cos(\nu t_1 - \beta), \mathbf{a}_p \cos(\nu t_2 - \beta), \dots, \mathbf{a}_p \cos(\nu t_s - \beta)] \quad .$$

Obviously, all column vectors in this observation matrix are linearly dependent, i.e. they form a singular matrix with the rank 1. Consequently the singular value decomposition of this rectangular $n \times s$ matrix leads to only one singular value, which differs from zero

$$s = \left(\sum_{i=1}^s \frac{\cos^2 \nu t_i}{n} \right)^{\frac{1}{2}} \|\mathbf{a}_p\| \quad ,$$

which is equivalent to the non-zero eigenvalue of the matrix $\mathbf{X}_S^T \mathbf{X}_S$. In this context to this one singular value, which is non zero, there exists one corresponding left singular vector φ that is equivalent to the first eigenvector of the matrix $\mathbf{X}_S^T \mathbf{X}_S$.

As a consequence it is possible to describe the particular response function of the n dimensional equation of motion to a harmonic excitation by a single degree of freedom system applying only one deterministic displacement pattern without error, while for modal analysis (i.e. the full solution procedure) all modes of vibration have to be considered in order to obtain the exact response function.

It has to be mentioned that this is an academic example, as the reduction process requires already the solution function, but based on this idea this work creates model order reduction strategies, which have huge potential of future applications in earthquake engineering and structural dynamics.

2 Introduction

Generating of low dimensional and simplified models is an important and often challenging skill. But often a system can have a more complex dynamic behavior and the utilization of basic models would lead to wrong approximations or to an oversimplification. Complex systems demand complex models in order to provide a sufficiently accurate approximation of the dynamic behavior. In this context the spacial discretization of dynamic systems by the finite element (FE) method has become common practice in engineering and structural dynamics.

The calculation of the response of a linear or nonlinear structure to an arbitrary excitation, i.e. the solution of the second order differential equations of motion in the time domain,

is mostly the basis for investigations of the behavior of a system concerning structural failure. Numeric integration of high dimensional systems is, depending on the considered method, time consuming or requires a lot of computational storage. Even the solution of linear ordinary differential equations (ODE) requires excessive computational effort as the calculation of the eigenvalue problem includes large matrix operations. Consequently, it can be determined that high dimensionality demands computational effort. Although the capacity of computers is constantly increasing, the models being used are constantly being refined, and therefore finite element calculations with thousands (or even millions) degrees of freedom (DOF) are still very time-consuming. Especially in dynamical systems the amount of calculation time may increase dramatically when using a finely meshed model. As mentioned in Qu (2004) although the computer capacity doubles every 18 months (Moore's law) the demand of storage and speed will grow similarly, which has been demonstrated in the last 50 years of finite element analysis. Additionally one of the engineer's main tasks is the design of structures, which is an iterative process that requires lots of recalculations in order to optimize the properties of the system. In this context the Monte Carlo method has become very popular in the last decades as it is a useful strategy for analyzing nonlinear problems. For calculation procedures that demand days or even weeks the application of the Monte Carlo simulation would exceed any limit of acceptance, i.e. a practical realization would be impossible. In such a setting the question for reduced order models (ROM) naturally arises.

Model order reduction (MOR) has become a huge field of research with many application areas. The aim is a simplification of a dynamical model by capturing its essential features. The first studies in this context came from Fourier in the year 1807, where he published the idea of approximating a function by the sum of a few trigonometric terms. Cornelius Lanczos (1893-1974) provided the basis for the Lanczos and Arnoldi methods, or more generally known as Krylov methods. He searched for a way to reduce a matrix in a tridiagonalized form and Arnoldi realized that a smaller matrix could be a good approximation of the full matrix (Antoulas (2005), Schilders et al. (2008)).

The research field of model order reduction boomed in the 1980's, when the utilization of computers and with it the demand of the calculation of high dimensional models became omnipresent. Asymptotic Waveform Evaluation (AWE) was introduced by Pillage and Rohrer (1990) as one of the first and basic MOR algorithms. The transfer function is described by a Padé approximation, which is the realization of the ratio of two polynomials. The further development was done by Feldmann and Freund (1995), who introduced the Padé via Lanczos (PVL) method, where the Padé approximation is calculated by the application of a two sided Lanczos algorithm. The computational effort is comparable to the AWE method but PVL is more robust. Obadasioglu and Celik (1998) presented a new algorithm named passive reduced-order interconnect macro modeling algorithm (PRIMA) based on the Arnoldi process. In Knockaert and Zutter (1999) and Knockaert and Zutter (2000) a reduced order multiport modeling algorithm based on the decomposition of orthogonal scaled Laguerre functions is proposed. They claim that the link with Padé approximation, the block Arnoldi process and singular value decomposition (SVD) leads to a simple and stable implementation of the algorithm. It should be mentioned that the presented methods (often they are called Krylov based methods as well) are applicable to algebraic equations. Laplace transformation has to be applied in order to transform the system described by the differential equation to a purely algebraic equation. Additionally a collective disadvantage of these methods is the limited application to only linear systems.

Another group of reduction methods that are more closely related to this work are truncation methods. The basic idea is to truncate the dynamic system at some point. Modal truncation is a basic and very effective method for linear systems and is applied several times in this dissertation. Important papers are Davidson (1966), Guyan (1965), Bampton and Craig (1968). It is probably the oldest method in structural dynamics but often one of the most effective MOR strategy as well. The method of modal truncation is presented in section 3.4.1. An important method of this group is truncated balanced realization (TBR) as well. Here the reader is referred to Antoulas (2005).

The review papers of Rega and Troger (2005) and Koutsovasilis and Beitelshmidt (2008) give profound information about model order reduction for dynamical systems and present relevant literature in this field of research. The book of Qu (2004) provides an overview to model order reduction in FE analysis by the application of truncation methods.

Qu (2004) sub-classifies MOR methods for FE applications into three basic categories: physical coordinate reduction, generalized coordinate reduction and hybrid coordinate reduction. The physical coordinate reduction is the most straightforward method. The reduced order model is obtained by removing a part of the physical coordinates. As a consequence, the coordinates of the reduced model belong to a subset of the full model. The generalized coordinate reduction represents the main category for the methods used in this work. Two examples for generalized coordinates are modal and Ritz coordinates. The hybrid coordinate reduction is a combination of the first two mentioned reduction methods.

This work concentrates on MOR strategies using the *proper orthogonal decomposition* (POD) and its advantages in comparison to the method of modal truncation. The aim is to find a set of deterministic modes (POD modes). The POD provides an optimal low dimensional description of a high dimensional process in the least square sense and has found application in various fields of research including turbulent fluid systems and coherent structures, wind engineering, image processing, dynamics of structures, and others. The method is also known as Karhunen-Loeve decomposition and principal component analysis. Additionally it should be added up front that the singular value decomposition (SVD) is closely related with the POD as it realizes the calculation of the POD modes and the POD values in a stable algorithm.

First ideas have arisen from the paper of Pearson (1901), who dealt with a representation of a system of points by the best fitting lines and planes, which is a mathematical basis for the now omnipresent SVD algorithm. Another early work was realized by Hotelling (1935), who presented an iterative scheme of the calculation of principal components. The theory of the POD was first mentioned by Kosambi (1943), the Karhunen-Loeve decomposition by Karhunen (1947) and Loeve (1946) as statistical formulations. In the field of turbulence and coherent structures basic works have been done by Holmes et al. (1996) and Sirovich (1987), who discusses the orthogonal decomposition of the covariance and introduces the method of snapshots, a powerful strategy based on the POD method. A research field, where the POD is also a frequently used term, is the area of wind engineering. Examples are Bienkiewicz et al. (1995), Wang et al. (2010) and Kikitsu et al. (2008).

In the field of structural dynamics POD was first used in the early 1990's by Cusumano et al. (1993), who presented an experimental study of dimensionality in an elastic impact oscillator. Others followed such as Feeny and Kappagantu (1998) and Kappagantu and Feeny (1999), where the POD modes for dynamical systems are discussed together with their relationship to normal modes. Further work was done in Kappagantu and Feeny

(2000a) and Kappagantu and Feeny (2000b), where the POD modes from an experimental set up in their former papers are used for numeric simulations and validation. A profound overview work of MOR techniques in connection with POD give Kerschen et al. (2005) and a physical interpretation of the POD modes are provided by Kerschen and Golivani (2002). Liang et al. (2002) give a summary of POD methods and in Kerschen and Golivani (2002) a physical interpretation of the POD modes is done and its relation to singular value decomposition is facilitated.

Tubino et al. (2003) used the POD method for investigating the seismic ground motion of the support points of a structure and denominates POD as a very efficient tool to simulate those multi-variate processes. Kerschen and Golivani (2003) investigated the POD based on auto-associative neural networks and illustrated the proposed procedure using data from a three-dimensional portal frame. Ritto et al. (2010) measured the efficiency of the POD method and compared it to the method of Modal Analysis using the example of a bar with an axial displacement field.

This work investigates MOR strategies using the POD in the field of earthquake engineering and structural dynamics. In this context a new MOR strategy for linear and nonlinear structures under transient excitations is presented. The innovative focal point of the section 4.1 is the application of the POD as a useful model reduction technique in earthquake engineering and as a consequence for structures under transient excitations for linear and nonlinear structures. This section is based on the paper of Bamer and Bucher (2012). The main goal of POD is its applicability to the analysis of nonlinear structures. However, before complex nonlinear structures under earthquake excitations can be treated, MOR by the POD method for linear systems under earthquake excitation should be analyzed.

The POD method is used in such a way that *snapshots* are taken by integrating the equations of motion over a short initial time period of the earthquake from which the reduced order model (ROM) is assembled. This ROM is time integrated over the whole time period (earthquake duration) and finally transformed back to the dimension of the original model. The main idea for both the linear and nonlinear system is to demonstrate that the information of the response within a small part of the random excitation is sufficient to draw conclusions about the response to the entire random process. For an application example see e.g. Bucher (2009a). Firstly, it has to be tested if it makes sense to use the POD method for model reduction of an earthquake excited linear system. Here the output of *modal truncation* provides a useful reference solution. In section 4.2 the application of the procedure to linear structures is presented. Subsequently, in section 4.3 the strategy is applied to a nonlinear structure. The nonlinearities are caused by friction-based seismic isolation devices. The nonlinear example shows the applicability of the POD method for nonlinear structures under earthquake excitations. Error estimations are performed in the linear and the nonlinear case and the property of optimality of the POD method is shown by proving that only a small number of POD modes is sufficient to provide very accurate solutions of the reduced order model in comparison to the full model. In addition the error of the POD - reduced system, which is not only related to the number of used POD modes but also to the number of *snapshots*, is quantified and analyzed using a time-independent quantity. The main aim for using this method is on the one hand to reduce substantially the calculation time and on the other hand not to lose required accuracy due to the approximate nature of the reduced order system. All numeric integration procedures are realized by the central difference method here. Sections 4.3.3 and 4.3.4 include the discussion of the

numerical efficiency, where accuracy is set in relation to calculation time, and the conclusion, where all results are assessed and an outlook for future works is given.

In section 4.4 the POD modes are calculated with the information of one nonlinear earthquake excited structure, and subsequently are used to reduce this structure excited by another transient excitation with similar properties (i.e. frequency content). The new universal MOR approach is presented by using two main kind of earthquake excitations: far-fault and near-fault earthquake records. These two types of earthquake motions are well known as qualitatively quite different, e.g. Chopra and Chintanapakdee (2001).

A test structure (nonlinear frame system) is excited by a record data of a transient excitation (i.e. Fukushima foreshock). The structure is integrated over a small time period within the main period of the earthquake ("snapshot time period"), in contrast to section 4.1, where the snapshot matrix is calculated by integration over the initial time period of the earthquake excitation. With this information the universal transformation matrix for model order reduction is assembled. The structure, which is now excited by the main acceleration data, is transformed into the reduced space and integrated over the whole time period.

It is demonstrated that earthquakes with similar properties (e.g. frequency range, spacial properties) are suitable for this new reduction method. On the one hand with the information of a foreshock event or an event with similar properties it is possible to draw conclusions about the resistance of structural failure to the main earthquake excitation. This excitation can be a generated earthquake based on the properties of the first excitation (i.e. foreshock), but in this work acceleration record data are used to demonstrate this MOR method. The advantage of this approach is that the POD modes are calculated according to real data and not artificially generated transient excitations and therefore contain information of real measured earthquake records.

The new POD strategies are discussed on planar, considerably simple linear frame structures resting on nonlinear friction elements. In this section the "universal" POD strategy is presented on a three dimensional nonlinear building construction. The nonlinearities are realized by a bilinear elastic-plastic stress-strain curve in axial direction. The method is tested on six earthquake records.

Section 4.6 discusses the application of the new approach, which is presented in section 4.4, for nonlinear high dimensional Monte Carlo simulations. As the calculation of one response is very time consuming the procedure of a Monte Carlo simulation, i.e. the calculation of a whole sample set of excitations, would exceed any limit of acceptance. It is suggested to obtain the observation matrix by a small part of the response of the nonlinear structure to one excitation sample and use this response information to assemble a POD surrogate model for every other excitation sample. In order to present the applicability of this method a statistical evaluation of the responses to 1000 excitation samples on a simple planar frame structure with a considerably small number of DOF is conducted. Additionally the new Monte Carlo Simulation procedure together with the POD reduction is applied to a three-dimensional nonlinear building structure with a large numbers of DOF by developing a multi-thread algorithm.

The exact calculation of linear high dimensional structures becomes disproportionately expensive because the eigenvalue problem includes the matrix factorization of the stiffness matrix and the evaluation of all eigenvectors. Modal truncation is often very effective to approximate the system by a small number of modes accurately. For calculations such as impact problems, where a broad frequency band is excited, the classical method of modal

truncation fails, since high frequency motions that are responsible for local peak stresses and forces are not considered in the reduced model. The idea is to capture local high frequency motions by taking snapshots directly after the impact time instant using a direct integration method and additionally feed the snapshot matrix with a small number of the global low frequency truncated normal modes. These here so-called hybrid POD modes are able to approximate the solution much more accurately than the method of modal truncation does.

3 Theoretical fundamentals

Two main topics are discussed in this dissertation, i.e. numerical integration and model order reduction.

Dynamic equilibrium is described by a system of differential equations. Especially if a structure is discretized in space (semi discretization process) it is described by a system of second order ordinary differential equations. This section deals with a variety of methods to solve these equations in the linear and the nonlinear case. Advantages and disadvantages of the different algorithms are elaborated.

Two methods to reduce a high dimensional set of equations of motion to a representative low dimensional one are introduced: the classical modal truncation, which is practically limited to the application to linear systems and the nonclassical proper orthogonal decomposition method, which can be applied to nonlinear systems as well.

A mathematical toolbox is provided that allows to combine algorithms of the two different groups in order to present new methods and strategies in earthquake engineering and structural dynamics and to reduce computational effort by preserving a required level of accuracy.

3.1 Equations of motion

The dynamic equilibrium is assembled by the sum of all on the system acting forces and terms of inertia dependent on time (cf. Bathe (1995))

$$\mathbf{F}_i(t) + \mathbf{F}_d(t) + \mathbf{F}_k(t) = \mathbf{F}_{ex}(t) . \quad (1)$$

The length of the vectors that appear in this equation define the dimension n of the system, i.e. the number of degrees of freedom (DOF). \mathbf{F}_i , \mathbf{F}_d and \mathbf{F}_k are the term of inertia, the dissipative damping force vector and the inertial restoring force vector. They are acting opposite to the direction of motion and are therefore located positive on the left side of the force equilibrium (Bathe (1995)).

The term of inertia, $\mathbf{F}_i = \mathbf{M}\ddot{\mathbf{x}}$, depends on the mass of the system and the acceleration, to which this mass is subjected to. The distribution of the mass is defined by the mass matrix \mathbf{M} , which is in structural dynamics mostly realized by a spacial discretization using the finite element (FE) method (semi-discretization). $\ddot{\mathbf{x}} = \frac{d^2\mathbf{x}}{dt^2}$ denotes the second derivative of the coordinate \mathbf{x} , i.e. the acceleration. This work examines only systems with non variable mass, therefore \mathbf{M} is constant and symmetric and positive definite.

The restoring force vector \mathbf{F}_k depends on the displacement vector \mathbf{x} and (in the case of plastic or visco-plastic material behavior) on the velocity vector $\dot{\mathbf{x}}$. In the special case of

a linear elastic material behavior the restoring force is $\mathbf{F}_k = \mathbf{K}\mathbf{x}$, where \mathbf{K} is the stiffness matrix, which is symmetric and real. \mathbf{K} is obtained by the FE discretization process. If the system is nonlinear \mathbf{F}_k is calculated directly from the stresses obtained by the deformation \mathbf{x} . Depending on the solution algorithm, i.e. explicit or implicit, the force \mathbf{F}_k can be directly used in the solution process or a tangent stiffness matrix is calculated based on the internal restoring force vector $\mathbf{F}_k(\mathbf{x})$.

The damping force is mostly governed by damping values, which are chosen empirically or experimentally. This work applies velocity proportional damping, $\mathbf{F}_d = \mathbf{C}\dot{\mathbf{x}}$, where \mathbf{C} is the damping matrix and $\dot{\mathbf{x}} = \frac{d\mathbf{x}}{dt}$ is the first derivative of the vector \mathbf{x} with respect to time, i.e. the velocity. In this work the classical Rayleigh damping strategy, where $\mathbf{C} = a_0\mathbf{M} + a_1\mathbf{K}$ is defined as a linear combination of mass- and stiffness matrix, is applied, which has advantages concerning the diagonalization process within the linear solving algorithms. For detailed information about the evaluation of a Rayleigh damping matrix the reader is referred to e.g. Chopra (2001) or Craig and Kurdila (2006).

The vector \mathbf{F}_{ex} denotes the sum of all on the system acting external forces consisting mainly of the excitation force $\mathbf{F}(t)$ and the gravitational force. Practically the external force vector is often known from measurements at equidistant time intervals. In the special case of an earthquake excitation, the excitation vector is evaluated based on measured ground acceleration data or by a sampling procedure using a filtered white noise excitation. However, every force function can be defined by sample points, dependent on the sampling rate, with adjustable accuracy.

3.2 The linear system

The n dimensional set of equations of motion of a linear damped system is a set of ordinary second order differential equation (ODE) (cf. Chopra (2001))

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{F}(t) . \quad (2)$$

The goal is to find the solution $\mathbf{x}(t)$, i.e. the response of the structure defined by the constant matrices \mathbf{M} , \mathbf{C} and \mathbf{K} excited by an arbitrary force function $\mathbf{F}(t)$.

3.2.1 Natural vibration

The homogenous undamped problem is governed by (cf. Chopra (2001))

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{0} . \quad (3)$$

and the solution ansatz is (cf. Chopra (2001))

$$\mathbf{x}(t) = \phi_{i,m} e^{-i\omega_i t} , \quad (4)$$

where the vector ϕ_i and the number ω_i are invariant with respect to time. Insertion of the solution ansatz and its second derivative into Eq. (3) leads to the n -dimensional eigenvalue problem (cf. Chopra (2001)):

$$[\mathbf{K} - \omega_i^2 \mathbf{M}] \phi_{i,m} = \mathbf{0} . \quad (5)$$

Based on the property that \mathbf{K} and \mathbf{M} are both positive definite and symmetric, which is assured for all structures prevented from rigid body motion, the eigenvalues ω_n are all real and positive. To each eigenvalue ω_i , called the i_{th} natural frequency or natural vibration frequency, there exists an independent vector $\phi_{i,\mathbf{m}}$, called the i_{th} mode shape of vibration or modal mode (Chopra (2001)).

The natural modes satisfy the orthogonality condition (cf. Chopra (2001))

$$\phi_{i,\mathbf{m}}^{\mathbf{T}} \mathbf{K} \phi_{j,\mathbf{m}} = 0 \quad \text{and} \quad \phi_{i,\mathbf{m}}^{\mathbf{T}} \mathbf{M} \phi_{j,\mathbf{m}} = 0 \quad \text{for} \quad i \neq j \quad (6)$$

Because in this dissertation Rayleigh damping is implemented, and therefore the damping matrix \mathbf{C} is a linear combination of mass and stiffness matrix, the orthogonality condition is valid for the damping matrix as well (cf. Chopra (2001)):

$$\phi_{i,\mathbf{m}}^{\mathbf{T}} \mathbf{C} \phi_{j,\mathbf{m}} = 0 \quad \text{for} \quad i \neq j \quad . \quad (7)$$

It makes sense to define a spectral matrix, where the squared eigenfrequencies are ordered in a diagonal matrix in ascending order according to (cf. Chopra (2001))

$$\mathbf{\Omega} = \begin{pmatrix} \omega_1^2 & & & \\ & \omega_2^2 & & \\ & & \ddots & \\ & & & \omega_n^2 \end{pmatrix} \quad (8)$$

and a modal matrix, in which the modes are arranged according to the spectral matrix (cf. Chopra (2001))

$$\mathbf{\Phi}_{\mathbf{m}} = [\phi_{1,\mathbf{m}}, \phi_{2,\mathbf{m}}, \dots, \phi_{n,\mathbf{m}}] \quad (9)$$

For the numerical solution of the eigenvalue problem the reader is referred to Bathe (1995). Dependent on the magnitude of high dimensionality of the system, i.e. if n is a large number, the numerical evaluation of the eigenfrequencies and mode shapes may require a substantial computational effort and storage.

3.2.2 Modal analysis

A new vector of coordinates $\mathbf{q}_{\mathbf{m}}$ is defined through the sum of n independent expressions $\phi_{i,\mathbf{m}} q_{i,m}$ (cf. Chopra (2001))

$$\mathbf{x} = \sum_{i=1}^n \phi_{i,\mathbf{m}} q_{i,m} = \mathbf{\Phi}_{\mathbf{m}} \mathbf{q}_{\mathbf{m}} \quad . \quad (10)$$

Insertion of this transformation property into Eq. (2) and left multiplication by $\mathbf{\Phi}_{\mathbf{m}}^{\mathbf{T}}$ leads to the linear equation of motion in the modal space (Chopra (2001))

$$\mathbf{m}_{\mathbf{m}} \ddot{\mathbf{q}}_{\mathbf{m}} + \mathbf{c}_{\mathbf{m}} \dot{\mathbf{q}}_{\mathbf{m}} + \mathbf{k}_{\mathbf{m}} \mathbf{q}_{\mathbf{m}} = \mathbf{f}_{\mathbf{m}}(t) \quad , \quad (11)$$

where $\mathbf{m}_{\mathbf{m}} = \mathbf{\Phi}_{\mathbf{m}}^{\mathbf{T}} \mathbf{M} \mathbf{\Phi}_{\mathbf{m}} = \text{diag}[m_i^*]$, $\mathbf{c}_{\mathbf{m}} = \mathbf{\Phi}_{\mathbf{m}}^{\mathbf{T}} \mathbf{C} \mathbf{\Phi}_{\mathbf{m}} = \text{diag}[c_i^*]$, $\mathbf{k}_{\mathbf{m}} = \mathbf{\Phi}_{\mathbf{m}}^{\mathbf{T}} \mathbf{K} \mathbf{\Phi}_{\mathbf{m}} = \text{diag}[k_i^*]$ due to the orthogonality condition, Eq. (6), are diagonal matrices, and $\mathbf{f}_{\mathbf{m}}(t)$ is the modal

force vector. Thus the coupled n -dimensional Eq. (2) is transformed into n independent single degree of freedom (SDOF) systems in the modal space (Chopra (2001))

$$m_i \ddot{q}_i + c_i \dot{q}_i + k_i q_i = f_i(t) , \quad (12)$$

which can be solved separately applying a closed form or numeric solution ansatz for this linear ODE with an arbitrary force function. In this work most of the closed form solution methods for SDOF systems are presented for the decoupled coordinate \mathbf{q}_m , which is transformed back into the full coordinate \mathbf{x} by the transformation condition in Eq. (10). For more detailed information about modal analysis and its properties the reader is referred to Chopra (2001).

3.2.3 Duhamel Integral

The Duhamel integral represents a closed form solution of the second order ODE in Eq. (12) excited by an arbitrary force function $f(t)$. Considering a function value at a time instant τ the small duration impulse at the time point τ is $f(\tau)\Delta\tau$ and the response to this impulse is (Clough and Penzien (1995))

$$dq(t) = \frac{1}{m\omega_d} e^{-\zeta\omega t} \sin[\omega_d(t - \tau)] d\tau , \quad (13)$$

where $\omega_d = \omega\sqrt{1 - \zeta^2}$ is the damped natural frequency. A sketch is shown in Fig. 4. A

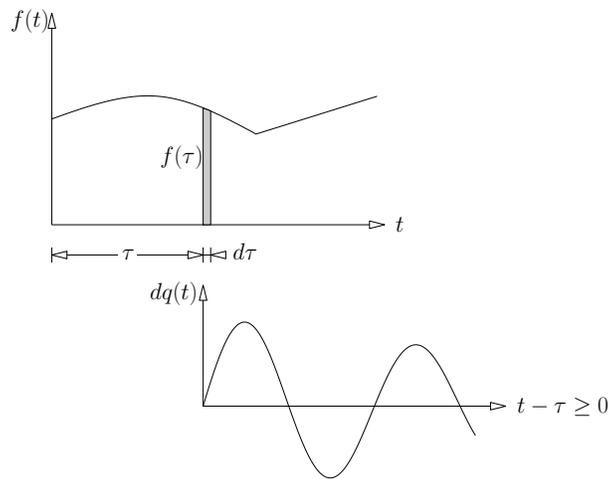


Figure 4: Response to the small duration impulse (Clough and Penzien (1995))

summation of all impulse responses leads to the convolution integral (Duhamel integral) (Clough and Penzien (1995)):

$$q(t) = \frac{1}{m\omega_d} \int_0^t f(\tau) e^{-\zeta\omega t} \sin[\omega_d(t - \tau)] d\tau . \quad (14)$$

Here the initial conditions $q(t = 0) = 0$ and $\dot{q}(t = 0) = 0$ are assumed. The calculation of the general response requires the initial conditions $q(t = 0) = q_0$ and $\dot{q}(t = 0) = \dot{q}_0$, which

leads to the general response function (c.f. Clough and Penzien (1995))

$$q(t) = e^{-\zeta\omega t} \left[q_0 \cos \omega_d t + \frac{\dot{q}_0 + q_0 \zeta \omega}{\omega_d} \sin \omega_d t \right] + \frac{1}{m\omega_d} \int_0^t f(\tau) e^{-\zeta\omega(t-\tau)} \sin[\omega_d(t-\tau)] d\tau, \quad (15)$$

where the first part of the response function is the homogenous solution of the ODE and the second part is the convolution integral (Duhamel integral).

If $f(\tau)$ is a simple function the Duhamel integral is an alternative method to the classical solution of the equation of motion. But for an arbitrary force function that is described numerically an analytic solution of the Duhamel integral does not exist. In this context the integration must be realized numerically. From the numerical point of view the application of the Duhamel integral in the time domain is often inefficient as for every time step the integral has to be evaluated again, which depends on the implementation code.

3.2.4 Piecewise analytic response

The numerical description of the excitation is provided by data points which are measured or generated. The space between these data points, which are mostly defined equidistant over the whole time definition period, needs to be defined by polynomial functions. The goal is to find an analytic solution in the defined continuous time interval between the data points and consequently move forward to the next continuous time interval.

The simplest approximation attempt is the application of a piecewise constant excitation (i.e. a polynomial of order zero), which can deliver useful results if the time interval Δt between the data points is small, see Fig 5.

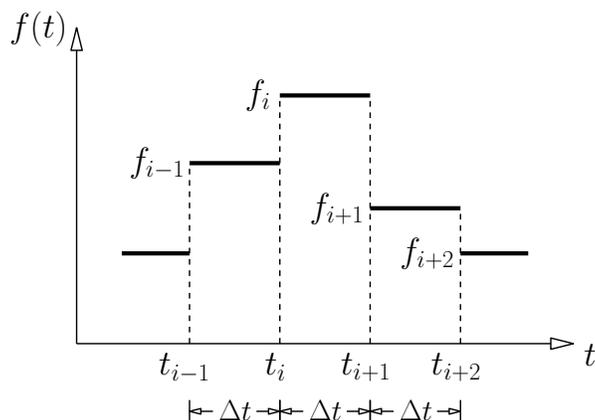


Figure 5: Approximation of an excitation by a piecewise constant excitation function

The exact step by step solution procedure to a piecewise constant excitation function, considering the initial displacement $q(t = t_i) = q_i$ and the initial velocity $\dot{q}(t = t_i) = \dot{q}_i$ at

time instant t_i is (cf. Clough and Penzien (1995))

$$\begin{aligned}
q_{i+1} &= e^{-\zeta\omega\Delta t} [A \cos(\omega_d\Delta t) + B \sin(\omega_d\Delta t)] + \frac{f_i}{k} \\
\dot{q}_{i+1} &= -\zeta\omega e^{-\zeta\omega\Delta t} [A \cos(\omega_d\Delta t) + B \sin(\omega_d\Delta t)] \\
&\quad + e^{-\zeta\omega\Delta t} [-A\omega_d \sin(\omega_d\Delta t) + B\omega_d \cos(\omega_d\Delta t)] \\
A &= q_i + \frac{f_i}{k}, \quad B = \frac{-\zeta\omega A + \dot{q}_i}{\omega_d} .
\end{aligned} \tag{16}$$

The assumption of higher order polynomials within the time interval Δt can lead to an increasing accurateness of the solution, especially when Δt is large. Fig 6 presents a linear varying loading during the time step. Within the time interval $t_i \leq t \leq t_{i+1}$ the force

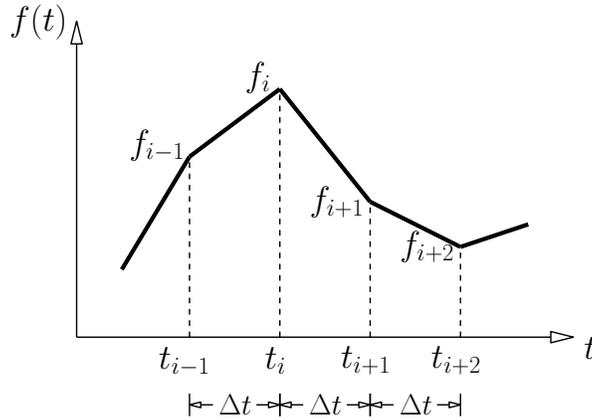


Figure 6: Realization of an excitation force by a piecewise linear force function

function is described by (cf. Clough and Penzien (1995))

$$f(t) = f_i + \frac{\Delta f_i}{\Delta t_i} t, \quad \Delta f_i = f_{i+1} - f_i \tag{17}$$

and the step by step response procedure by consideration of the initial conditions $q(t = t_i) = q_i$ and $\dot{q}(t = t_i) = \dot{q}_i$ is (cf. Clough and Penzien (1995))

$$\begin{aligned}
q_{i+1} &= e^{-\zeta\omega\Delta t} [A \cos(\omega_d\Delta t) + B \sin(\omega_d\Delta t)] + a_{0p} + a_{1p}t \\
\dot{q}_{i+1} &= -\zeta\omega e^{-\zeta\omega\Delta t} [A \cos(\omega_d\Delta t) + B \sin(\omega_d\Delta t)] \\
&\quad + e^{-\zeta\omega\Delta t} [-A\omega_d \sin(\omega_d\Delta t) + B\omega_d \cos(\omega_d\Delta t)] + a_{1p} \\
a_{0p} &= \frac{f_i}{k} + \frac{\Delta f_i c}{\Delta t k^2}, \quad a_{1p} = \frac{\Delta f_i}{\Delta t k}, \quad A = q_i - a_{0p}, \quad B = \frac{\dot{q}_i + \zeta\omega A - a_{1p}}{\omega_d} .
\end{aligned} \tag{18}$$

For linear structures under transient excitation the application of this approximation is the most efficient way to solve problems with a moderate number of degrees of freedom (DOF). Naturally any analytic force function can be defined by discrete points as well and the error

of the response can be reduced to any acceptable value by scaling down the length of the time step. In the case of given measured data a refinement is not possible. However, a better interpolation function can be used to improve the representation of reality, but it is up to the engineer to decide if this is necessary. The numerical implementation of the piecewise analytic solution is presented in Appendix A.1.1.

3.2.5 Transformation into the state space and modal analysis

The orthogonality condition of the natural modes in Eq. (6) with respect to mass and stiffness matrix is the reason for the diagonalization of the equation of motion. The classical Rayleigh damping formulation is a linear combination of mass and stiffness matrix, and therefore, it is diagonalized by the application of modal modes as well. If nonclassical damping is utilized, the equation of motion is not diagonalizable by modal modes any more. In this dissertation only Rayleigh damping is implemented, and thus non-diagonalization of the damping matrix by modal modes is not considered. Much more attention is drawn to the fact that the transformation by a set of deterministic orthonormal modes (e.g. POD modes) does not necessarily diagonalize the mass-, stiffness- and damping matrix in the equations of motion. I.e., another strategy must be applied in order to obtain the analytic solution of the linear ODE.

Subsequently the equations of motion (2) are transformed into the state space by the introducing the variables $\mathbf{y}_1 = \mathbf{x}$ and $\mathbf{y}_2 = \dot{\mathbf{x}}$. The first order linear ODE is then (cf. Clough and Penzien (1995) and Veletsos and Ventura (1986))

$$\begin{pmatrix} \dot{\mathbf{y}}_1 \\ \dot{\mathbf{y}}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{F}(t) \end{pmatrix} \rightarrow \dot{\mathbf{y}} = \mathbf{G}\mathbf{y} + \mathbf{g}(t), \quad (19)$$

where \mathbf{G} is the $2n \times 2n$ system matrix and $\mathbf{g}(t)$ is the excitation vector in state space. The second order differential equation with n DOF is now transformed to a first order differential equation with $2n$ DOF.

The homogenous problem is (cf. Veletsos and Ventura (1986) and Guckenheimer and Holmes (1983))

$$\dot{\mathbf{y}} = \mathbf{G}\mathbf{y}, \quad (20)$$

with the ansatz function $\mathbf{y}(t) = \Phi e^{\lambda t}$ leads to the eigenvalue problem (cf. Guckenheimer and Holmes (1983))

$$[\mathbf{G} - \lambda_i \mathbf{I}] \Phi_i = \mathbf{0}. \quad (21)$$

The eigenvectors are assembled in the $2n \times 2n$ eigenmatrix $\Phi = [\Phi_1, \Phi_2, \dots, \Phi_{2n}]$ and the $2n$ eigenvalues appear in conjugated complex pairs. Eq. (19) can now be diagonalized by the transformation $\mathbf{z}(t) = \Phi \mathbf{y}(t)$ (cf. Veletsos and Ventura (1986))

$$\dot{\mathbf{z}} = \Phi^{-1} \mathbf{G} \Phi \mathbf{z} + \Phi^{-1} \mathbf{g} \quad \longrightarrow \quad \dot{\mathbf{z}} = \text{diag}[\lambda_i] \mathbf{z} + \mathbf{g}_m, \quad i = 1 \dots 2n \quad (22)$$

into the decoupled modal state space, and Eq. (22) can be written as a set of $2n$ independent single degree of freedom linear first order ODE (Guckenheimer and Holmes (1983))

$$\dot{z}_i = \lambda_i z_i + g_i. \quad (23)$$

3.2.6 Piecewise analytic response in the state space

The step by step solution of a linear first order SDOF ODE to a piecewise constant force function as shown in Fig. 5 under consideration of the initial condition $z(t=0) = z_0$ is (cf. Chopra (2001))

$$z_{i+1} = \left(z_i + \frac{f_i}{\lambda_i} \right) e^{\lambda_i \Delta t} - \frac{f_i}{\lambda_i} . \quad (24)$$

Correspondingly, the step by step response to a linear excitation function as presented in Fig. 6 is (cf. Chopra (2001))

$$z_{i+1} = \left(z_i - \frac{-f_i - \frac{a_1}{\lambda}}{\lambda} \right) e^{\lambda \Delta t} - \frac{a_1}{\lambda} \Delta t + \frac{-f_i - \frac{a_1}{\lambda}}{\lambda} , \quad (25)$$

where $a_1 = \frac{f_{i+1} - f_i}{\Delta t}$ is the slope of the excitation function within the time interval $t_i \leq t \leq t_{i+1}$.

For the sake of completeness the piecewise analytic solution of the SDOF state space equation to a piecewise constant and piecewise linear force function is presented but the implementation is based on the following procedure: The solution of the first order ODE in Eq. (19) is (Guckenheimer and Holmes (1983))

$$\mathbf{y} = \mathbf{y}_h + \mathbf{y}_p = e^{\mathbf{G}t} \mathbf{y}_0 + e^{\mathbf{G}t} \mathbf{c}(t) , \quad (26)$$

where $\mathbf{c}(t)$ is evaluated by insertion of the particular solution ansatz into the equation of motion (19), and $\mathbf{y}_0 = [\mathbf{x}_0, \dot{\mathbf{x}}_0]^T$ is composed of the initial conditions. The full solution in the state space is then (cf. Guckenheimer and Holmes (1983))

$$\mathbf{y}(t) = -\mathbf{G} [\mathbf{I} - e^{\mathbf{G}t}] \mathbf{g} . \quad (27)$$

The calculation of the term $e^{\mathbf{G}t}$ is then realized by the eigenvalue factorization of \mathbf{G} into $\mathbf{\Phi}^{-1} \mathbf{\Lambda} \mathbf{\Phi}$, where $\mathbf{\Lambda}$ is a diagonal matrix with complex conjugated eigenvalues. The exponential term is therefore (cf. Bellman (1970))

$$e^{\mathbf{G}t} = \mathbf{\Phi}^{-1} e^{\mathbf{\Lambda}t} \mathbf{\Phi} = \mathbf{\Phi}^{-1} \begin{pmatrix} e^{\lambda_1} & & & \\ & e^{\lambda_2} & & \\ & & \ddots & \\ & & & e^{\lambda_{2n}} \end{pmatrix} \mathbf{\Phi} . \quad (28)$$

The implementation code of the analytic solution in the state space is presented in Appendix A.1.1.

3.2.7 Direct time integration

The expression *direct* means in this context that integration can be applied in the coordinate \mathbf{x} and no transformation of the equations is needed. There exist more reasons why someone would choose a purely numerical method to calculate the response of a system. Definitely the most relevant one is that for the treatment of nonlinear systems analytic solutions often do not exist, and therefore, the application of direct time integration is necessary to solve

the problem. The treatment of linear systems by analytic or exact methods involves the analyzation of the eigenvalue problem (Eq. (5)), which demands plenty of computational effort for large system matrices because a matrix inversion and an iterative solution procedure has to be conducted to calculate all eigenvalues and eigenvectors. In this case the direct methods can be an effective alternative in transient analysis. Additionally one important reason can be the engineer's convenience, as if a numeric algorithm is once implemented there is a broad area of application (i.e. linear and nonlinear response analysis).

Direct numeric integration is basically defined by two main ideas. Firstly, the dynamic equilibrium is assumed to be satisfied at discrete time points, which are mostly equidistant and defined by the prior time discretization. Secondly, an assumption for the displacement $\mathbf{x}(t)$, the velocity $\dot{\mathbf{x}}(t)$ and the acceleration $\ddot{\mathbf{x}}(t)$ is done within each equidistant time interval Δt . The integration schemes differ merely because of these assumptions and thus each method has different features concerning accuracy, stability and computational effort.

All these methods can basically be classified in two groups, i.e. explicit and implicit time integration schemes, each of them with advantages and disadvantages. Generally for explicit methods the computational effort for each time step (calculation step) is relatively small but, since they are not unconditionally stable, the time step must be considerably small. Many implicit methods on the contrary benefit from the property of unconditional stability. Consequently, the definition of the magnitude of the time step is only dependent on the required level of accuracy.

A variety of direct time integration methods for differential equations describing dynamical systems does exist in the literature. However, in this dissertation only a few representative methods are applied. In Chopra (2001) and Bathe (1995) the most important direct integration methods for structural dynamic problems are presented and the practical application of several algorithms is discussed.

Explicit methods In the step by step procedure the solution vector \mathbf{x}_{i+1} at time t_{i+1} is directly calculated by considering the equilibrium condition at time t_i , i.e. the solution vector \mathbf{x}_{i+1} is only a function of either the excitation or the structural stiffness at time t_i (and not t_{i+1}). The consequence is that no iteration is required in the algorithm.

The basis for explicit numeric integration is provided by Leonhard Euler in his work *Institutionum calculi interalis* (1768), simply called *Euler's method* here. Displacement and velocity are approximated by a first order polynomial

$$\begin{aligned}\mathbf{x}_{i+1} &= \mathbf{x}_i + \dot{\mathbf{x}}_i \Delta t \\ \dot{\mathbf{x}}_{i+1} &= \dot{\mathbf{x}}_i + \ddot{\mathbf{x}}_i \Delta t ,\end{aligned}\tag{29}$$

where $\ddot{\mathbf{x}}_i$ is obtained considering the linear equation of motion (2) at time instant t_i

$$\ddot{\mathbf{x}}_i = \mathbf{M}^{-1} [-\mathbf{C}\dot{\mathbf{x}}_i - \mathbf{K}\mathbf{x}_i + \mathbf{F}(t_i)] .\tag{30}$$

Since this is a first order approximation the error of the Euler's method is proportional to the square of the time step. The approximation of a simple harmonic oscillator shows that the numerical solution by Euler's method grows to infinity with increasing time. Therefore, for many problems the level of accuracy of this method is insufficient and thus often higher order methods should be applied.

A second order approximation, the *central difference approximation*, is the most widely used explicit technique in large scale dynamical systems because the possible time step is maximum among all possible order two approximations while preserving a high level of accuracy.

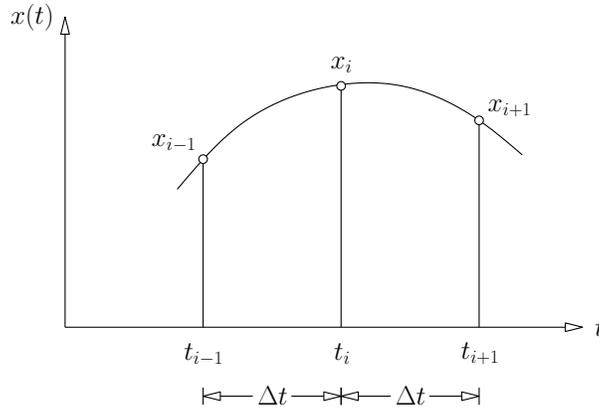


Figure 7: Central difference approximation, geometric assumptions for a SDOF system

The forward and backward Taylor expansions of the solution vector $\mathbf{x}(t)$ with the expansion vector $\mathbf{x}_i = \mathbf{x}(t_i)$ at time instant t_i are (cf. Greenberg (1998))

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta t \dot{\mathbf{x}}_i + \Delta t^2 \frac{\ddot{\mathbf{x}}_i}{2} + \Delta t^3 \frac{\dddot{\mathbf{x}}_i}{6} + \mathcal{O}(\Delta t^4) \quad (31)$$

$$\mathbf{x}_{i-1} = \mathbf{x}_i - \Delta t \dot{\mathbf{x}}_i + \Delta t^2 \frac{\ddot{\mathbf{x}}_i}{2} - \Delta t^3 \frac{\dddot{\mathbf{x}}_i}{6} + \mathcal{O}(\Delta t^4) \quad . \quad (32)$$

They provide the basis for the geometrical assumption of the central difference approximation shown in Fig. 7. Summation of \mathbf{x}_{i+1} and \mathbf{x}_{i-1} leads to the central difference approximation of the acceleration at time instant t_i (cf. Dokainish and Subbaray (1989a))

$$\ddot{\mathbf{x}}_i = \frac{\mathbf{x}_{i+1} - 2\mathbf{x}_i + \mathbf{x}_{i-1}}{\Delta t^2} + \mathcal{O}(\Delta t^2) \quad , \quad (33)$$

subtraction of \mathbf{x}_{i+1} from \mathbf{x}_{i-1} leads to the central difference approximation of the velocity at the time instant t_i (Dokainish and Subbaray (1989a))

$$\dot{\mathbf{x}}_i = \frac{\mathbf{x}_{i+1} - \mathbf{x}_{i-1}}{2\Delta t} + \mathcal{O}(\Delta t^1) \quad . \quad (34)$$

The result is a second order approximation of the acceleration with an error term of order two. Therefore, the error is divided into four if the time step is divided into two. The velocity term is a first order expression with an error of order one. Eq. (33) leads directly to the basic Verlet integration scheme (cf. Verlet (1967)):

$$\mathbf{x}_{i+1} = 2\mathbf{x}_i - \mathbf{x}_{i-1} + \ddot{\mathbf{x}}_i \Delta t^2 + \mathcal{O}(\Delta t^4) \quad . \quad (35)$$

Verlet (1967) used it in the 1960's for problems of molecular dynamic. This integration method is very efficient for conservative systems or systems without velocity dependent

terms as it has an error term of the order of four. It has symplectic properties and is therefore appropriate for calculations concerning gravitational astronomy where a long time stability is necessary. For more information about symplectic integrators, especially the symplectic Verlet method, the reader is referred to e.g. Hairer et al. (2003) and Verlet (1967). Another form of the Verlet integration scheme is the velocity Verlet algorithm where the result of the displacement and the velocity is calculated in one integration loop.

Damping plays an important role in problems concerning dynamics of structures and can therefore not be neglected. Another strategy needs to be chosen to implement the central difference approximation. In this context the acceleration term in Eq. (33) and the velocity term in Eq. (34) is inserted into the equations of motion (2), which leads to an explicit expression of the displacement in the next time step (c.f. Chopra (2001) and Dokainish and Subbaray (1989a))

$$\mathbf{x}_{i+1} = \mathbf{D}^{-1} \left\{ [2\mathbf{M} - \Delta t^2 \mathbf{K}] \mathbf{x}_i + \left[\frac{\Delta t}{2} \mathbf{C} - \mathbf{M} \right] \mathbf{x}_{i-1} + \Delta t \mathbf{F}_i \right\} - \mathbf{D}^{-1} [\mathcal{O}(\Delta t^4) + \mathcal{O}(\Delta t^2)] \quad (36)$$

where $\mathbf{D} = \mathbf{M} + \frac{\Delta t}{2} \mathbf{C}$. The big advantage of this method is that no factorization of the stiffness matrix is necessary. However, the the matrix \mathbf{D} needs to be inverted once. Therefore, this algorithm has the best efficiency concerning computational effort if a diagonal mass matrix and a diagonal damping matrix is assumed. The error

$$\mathcal{R} = \left[\mathbf{M} + \frac{\Delta t}{2} \mathbf{C} \right]^{-1} [\mathcal{O}(\Delta t^4) + \mathcal{O}(\Delta t^3)] = \mathcal{O}(\Delta t^2) \quad (37)$$

is of order Δt^2 , where \mathbf{D}^{-1} is of order Δt^{-1} and the residual composed of the acceleration and the velocity residual is of order Δt^3 .

Every calculation loop involves two already known displacement vectors, \mathbf{x}_i and \mathbf{x}_{i-1} . Therefore a special start procedure must be employed, which contains not only the initial values \mathbf{x}_0 and $\dot{\mathbf{x}}_0$ but also the artificial initial value \mathbf{x}_{-1} . The second order backward Taylor expansion according to Eq. (31) leads to

$$\mathbf{x}_{-1} = \mathbf{x}_0 - \Delta t \dot{\mathbf{x}}_0 + \Delta t^2 \frac{\ddot{\mathbf{x}}_0}{2} \quad , \quad (38)$$

where $\ddot{\mathbf{x}}_0$ is obtained by insertion of the initial values, \mathbf{x}_0 and $\dot{\mathbf{x}}_0$, into Eq. (2) according to Eq. (30).

The big drawback of explicit methods is the lack of stability. The investigation of stability is here presented based on the paper of Leech et al. (1965), who were one of the first scientists, who dealt with the stability problem of the finite difference approximation used to solve the dynamic equation of motion (2). Eq. (39) - (52) can be compared with the paper of Leech et al. (1965). The undamped free vibration is described by the homogenous problem

$$\mathbf{M} \ddot{\mathbf{x}} + \mathbf{K} \mathbf{x} = \mathbf{0} \quad . \quad (39)$$

If Δt is invariant with respect to time, the time at the i_{th} time step is $t = i \Delta t$ and the finite difference equivalent of Eq. (39) is

$$\mathbf{x}_{i+1} - 2\mathbf{x}_i + \mathbf{x}_{i-1} + \Delta t^2 \mathbf{M}^{-1} \mathbf{K} \mathbf{x}_i = \mathbf{0} \quad . \quad (40)$$

In this context the error is

$$\epsilon_{i+1} - 2\epsilon_i + \epsilon_{i-1} + \Delta t^2 \mathbf{M}^{-1} \mathbf{K} \epsilon_i = \mathbf{0} \quad . \quad (41)$$

The solution $\epsilon = \delta e^{\alpha t}$ is assumed and insertion of this ansatz function at the time instants t_{i-1} , t_i and t_{i+1} into Eq. (41) leads to:

$$\left[\left(\zeta - 2 + \frac{1}{\zeta} \right) \mathbf{I} + \Delta t^2 \mathbf{M}^{-1} \mathbf{K} \right] \delta = \mathbf{0}, \quad \zeta = e^{-i\alpha \Delta t} \quad . \quad (42)$$

The eigenvalue problem can then be written as

$$[\mathbf{P} - \beta_n \mathbf{I}] \delta_n = \mathbf{0} \quad , \quad (43)$$

where $\mathbf{P} = \mathbf{M}^{-1} \mathbf{K}$ and $\beta_n = -\frac{1}{\Delta^2} \left(\zeta - 2 + \frac{1}{\zeta} \right)$ with the eigenvalues β_n , which are equivalent to the eigenfrequencies of the system defined by \mathbf{P} :

$$\beta_n = \omega_n^2 \quad . \quad (44)$$

The number n denotes the number of the corresponding natural frequency, and if \mathbf{M} and \mathbf{K} are symmetric, real and positive definite, every ω_n^2 is real and positive. It follows

$$\omega_n^2 = -\frac{1}{\Delta^2} \left(\zeta - 2 + \frac{1}{\zeta} \right) \quad (45)$$

or

$$\zeta^2 + 2A\zeta + 1 = 0 \quad , \quad (46)$$

where $A = \frac{1}{2} \Delta t^2 \omega_n^2 - 1$ and the solution for $\zeta_{1,2}$

$$\zeta_1 = -A + (A^2 - 1)^{\frac{1}{2}} \quad \text{and} \quad \zeta_2 = -A - (A^2 - 1)^{\frac{1}{2}} \quad . \quad (47)$$

The solution of Eq. (41) is then

$$\epsilon_i = [c_1 \zeta_1^i + c_2 \zeta_2^i] \delta \quad . \quad (48)$$

c_1 and c_2 are constant numbers and δ is a constant vector. Consequently ζ_1 and ζ_2 are

$$|\zeta_{1,2}| \leq 1 \quad , \quad (49)$$

which is equivalent to

$$\left| -A + (A^2 - 1)^{\frac{1}{2}} \right| \leq 1 \quad \text{and} \quad \left| -A - (A^2 - 1)^{\frac{1}{2}} \right| \leq 1 \quad , \quad (50)$$

leading to the boundary $-1 \leq A \leq 1$ and

$$0 \leq \frac{1}{2} \Delta t^2 \omega_n^2 \leq 2 \quad . \quad (51)$$

The left hand side is fulfilled for every time step Δt but the right hand side reads as

$$\Delta t \leq \frac{2}{\omega_n} \quad , \quad (52)$$

where n denotes here the maximum possible value, consequently ω_n defines the highest natural frequency. The conclusion is that the application of the finite difference approximation only implies conditional stability. If the condition in Eq. (52) is not fulfilled, i.e. if the chosen time step exceeds a critical value, the solution grows exponentially to infinity. For very finely meshed models the highest natural frequency of the system is very large and as a consequence the critical time step is very small. In this context a large number of calculation loops has to be processed during the whole calculation time history, which involves considerably large amount of computational effort and hence time consuming procedures.

Implicit methods This work deals only with the Newmark family of methods (Newmark (1959)) although there exist many other schemes such as the Wilson Θ and the Houbolt method. An overview of the most important implicit integration schemes the reader is provided in Dokainish and Subbaray (1989b). The presentation of the Newmark method in this work is mainly based on Chopra (2001), consequently Eq. (53) - (63) can be compared to those presented in Chopra (2001) or Dokainish and Subbaray (1989b).

The integration scheme is based on the following equations:

$$\dot{\mathbf{x}}_{i+1} = \mathbf{x}_i + [(1 - \gamma) \Delta t] \ddot{\mathbf{x}}_i + \gamma \Delta t \ddot{\mathbf{x}}_{i+1} \quad (53)$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta t \dot{\mathbf{x}}_i + \left[\left(\frac{1}{2} - \beta \right) \Delta t^2 \right] \ddot{\mathbf{x}}_i + \beta \Delta t^2 \ddot{\mathbf{x}}_{i+1} \quad , \quad (54)$$

where the parameters β and γ define the variation of the acceleration (typical: $\beta = \frac{1}{4}$ and $\gamma = \frac{1}{2}$). As the acceleration vector $\ddot{\mathbf{x}}_{i+1}$ appears on the right hand side of both equations an implicit step is necessary to solve the equations. However, a noniterative formulation exists for linear systems.

Noniterative formulation The displacement, velocity and acceleration vectors and the excitation vector is now transformed into incremental quantities:

$$\Delta \mathbf{x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i \quad , \quad \Delta \dot{\mathbf{x}}_i = \dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i \quad , \quad \Delta \ddot{\mathbf{x}}_i = \ddot{\mathbf{x}}_{i+1} - \ddot{\mathbf{x}}_i \quad , \quad \Delta \mathbf{F}_i = \mathbf{F}_{i+1} - \mathbf{F}_i \quad . \quad (55)$$

By consideration of these incremental representations the Newmark equations can be rewritten as

$$\Delta \dot{\mathbf{x}}_i = \Delta t \ddot{\mathbf{x}}_i + \gamma \Delta t \Delta \ddot{\mathbf{x}}_i \quad \text{and} \quad \Delta \mathbf{x}_i = \Delta t \dot{\mathbf{x}}_i + \frac{\Delta t^2}{2} \ddot{\mathbf{x}}_i + \beta \Delta t^2 \Delta \ddot{\mathbf{x}}_i \quad , \quad (56)$$

where the second equation leads to an explicit formulation of the incremental acceleration

$$\Delta \ddot{\mathbf{x}}_i = \frac{1}{\beta \Delta t^2} \Delta \mathbf{x}_i - \frac{1}{\beta \Delta t} \dot{\mathbf{x}}_i - \frac{1}{2\beta} \ddot{\mathbf{x}}_i \quad (57)$$

and insertion of Eq. (57) into the first incremental equation of Newmark's formulation gives

$$\Delta \dot{\mathbf{x}}_i = \frac{\gamma}{\beta \Delta t} \Delta \mathbf{x}_i - \frac{\gamma}{\beta} \dot{\mathbf{x}}_i + \Delta t \left(1 - \frac{\gamma}{2\beta} \right) \ddot{\mathbf{x}}_i \quad (58)$$

The incremental formulation of the equation of motion at time instant $t = t_{i+1}$ is

$$\mathbf{M}\Delta\ddot{\mathbf{x}}_i + \mathbf{C}\Delta\dot{\mathbf{x}}_i + \mathbf{K}\Delta\mathbf{x}_i = \Delta\mathbf{F}_i \quad (59)$$

and insertion of Eq. (57) and Eq. (58) into Eq. (59) leads to the incremental displacement

$$\Delta\mathbf{x}_i = \hat{\mathbf{K}}^{-1}\Delta\hat{\mathbf{F}}_i \quad (60)$$

at time instant $t = t_{i+1}$, where

$$\hat{\mathbf{K}} = \mathbf{K} + \frac{\gamma}{\beta\Delta t}\mathbf{C} + \frac{1}{\beta\Delta t^2}\mathbf{M} \quad (61)$$

and

$$\Delta\hat{\mathbf{F}}_i = \Delta\hat{\mathbf{F}}_i + \left(\frac{1}{\beta\Delta t}\mathbf{M} + \frac{\gamma}{\beta}\mathbf{C} \right) \dot{\mathbf{x}}_i + \left[\frac{1}{2\beta}\mathbf{M} + \Delta t \left(\frac{\gamma}{2\beta} - 1 \right) \right] \ddot{\mathbf{x}}_i \quad (62)$$

If $\Delta\mathbf{x}_i$ is known the incremental velocity $\Delta\dot{\mathbf{u}}_i$, and acceleration $\Delta\ddot{\mathbf{u}}_i$ can be calculated, and finally the relations (55) lead to the displacement \mathbf{x}_{i+1} , the velocity $\dot{\mathbf{x}}_{i+1}$ and the acceleration $\ddot{\mathbf{x}}_{i+1}$ at the time point t_{i+1} .

The disadvantage of this method is that the matrix $\hat{\mathbf{K}}$ must be inverted, which can be expensive if the dimension is high. The advantage is the property of stability of the Newmark method. The algorithm is stable if the following condition is fulfilled:

$$\frac{\Delta t\omega_n}{2} \leq \frac{1}{\sqrt{2}} \frac{1}{\sqrt{\gamma - 2\beta}} \quad (63)$$

For the parameters $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ the critical time step grows to infinity, which means that the Newmark method is unconditionally stable if a constant acceleration within the time step is assumed.

3.3 Nonlinear systems

The n dimensional set of equations of motion of the nonlinear damped system is the ODE (cf. Chopra (2001))

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{R}(\mathbf{x}) = \mathbf{F}(t) \quad (64)$$

where the vector $\mathbf{R}(\mathbf{x})$ is the vector of the nonlinear restoring forces. In this dissertation the mass- and damping matrix is constant, although systems with variable mass matrices and nonlinear damping matrices could be included. In this work nonlinearities depend only on $\mathbf{R}(\mathbf{x})$.

Piecewise analytic or closed form solutions are presented in the sections 3.2.3 and 3.2.4 but for nonlinear problems according to Eq. (64), they are very rare and mostly approximations or simply do not exist. Direct numerical time integration is essential to obtain the response. The limitation of solution algorithms to numerical methods leads automatically to the discussion of computational efficiency.

3.3.1 Direct time integration

The big advantage of numerical integration methods is their broad area of application. In this context they can be utilized to solve nonlinear problems, where the application of analytic or exact procedures would not make any sense. As the approximation assumptions of displacement, velocity and acceleration do not depend on the linearity or nonlinearity of the structure the equations are based on those presented in section 3.2.7.

Explicit time integration - central difference approximation for nonlinear systems Substitution of the central difference approximation, Eq. (33) and (34), in the nonlinear set of equations of motion (64) leads to the explicit formulation of the displacement at time instant $t = t_{i+1}$

$$\mathbf{x}_{i+1} = \mathbf{D}^{-1} \left\{ 2\mathbf{M}\mathbf{x}_i - \left[\mathbf{M} - \frac{1}{2}\Delta t\mathbf{C} \right] \mathbf{x}_{i-1} + \Delta t^2\mathbf{F}_i - \Delta t^2\mathbf{R}(\mathbf{x}_i) \right\} + \mathbf{R}(\mathcal{O}^2) \quad , \quad (65)$$

where $\mathbf{D} = \mathbf{M} - \frac{1}{2}\Delta t\mathbf{C}$, which is analogous to the response of linear systems. The vector of the restoring forces depends on the displacement at time t_i , in case of plasticity it depends additionally on the sign of the velocity and in case of visco-plasticity on the magnitude of the velocity as well. Therefore, the explicit formulation of \mathbf{x}_{i-1} can be calculated an iteration. Thus, the central difference scheme is one of the simplest methods for nonlinear systems. The residual term is of second order, which is the same as for the linear system. Hence , the level of accuracy here can be compared to the level of accuracy of linear systems.

Solutions cannot be compared to exact solutions as they mostly do not exist. Thus, the correctness of the solution should be checked applying other numerical integration schemes.

Implicit time integration - Newmark method for nonlinear systems The nonlinear incremental equilibrium equation reads as (cf. Chopra (2001) or Bathe (1995))

$$\mathbf{M}\Delta\ddot{\mathbf{x}}_i + \mathbf{C}\Delta\dot{\mathbf{x}}_i + \Delta\mathbf{R}_i = \Delta\mathbf{F}_i \quad (66)$$

and consideration of Eq. (60) leads to the relation (cf. Chopra (2001) or Bathe (1995))

$$\hat{\mathbf{K}}_{sec,i}\Delta\mathbf{x}_i = \Delta\hat{\mathbf{F}}_i \quad , \quad (67)$$

where (cf. Chopra (2001) or Bathe (1995))

$$\hat{\mathbf{K}}_{sec,i} = \mathbf{K}_{sec,i} + \frac{\gamma}{\beta\Delta t}\mathbf{C} + \frac{1}{\beta\Delta t^2}\mathbf{M} \quad (68)$$

and \mathbf{K}_i is the secant stiffness matrix from time instant t_i to t_{i+1} , as demonstrated in Fig. 8.

The secant stiffness is not known since it depends on the displacement at time instant t_{i+1} . Thus, another algorithm has to be applied in order to obtain \mathbf{x}_{i+1} , which is discussed.

The tangent stiffness matrix $\mathbf{K}_{T,i}^1$ is calculated using the restoring force that depends on the displacement \mathbf{x}_i^0 , $\mathbf{R}(\mathbf{x})_i$ (for further information the reader is referred to publications dealing with nonlinear finite element problems, e.g. Bathe (1995)). Considering Eq. (68) and substituting $\hat{\mathbf{K}}_{sec,i}$ with $\hat{\mathbf{K}}_{T,i}^0$ and in this manner $\mathbf{K}_{sec,i}$ with $\mathbf{K}_{T,i}^0$ leads to the tangential incremental equilibrium equation of the initial iteration step (cf. Chopra (2001))

$$\hat{\mathbf{K}}_{T,i}^1\Delta\mathbf{x}_i^1 = \Delta\hat{\mathbf{F}}_i \quad . \quad (69)$$

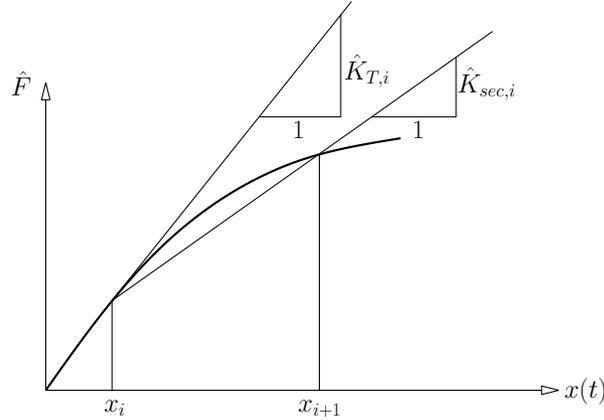


Figure 8: Secant stiffness $\hat{\mathbf{K}}_{sec,i}$ and tangent stiffness $\mathbf{K}_{T,i}$

The complete Newton Raphson iteration procedure is given in Table 1 and illustrated in Fig. 9. For the sake of clarity, i denotes the number of the time step and j the number of the iteration loop within the time step.

1	Initial conditions	
	$\mathbf{x}_{i+1}^0 = \mathbf{x}_i$	$\mathbf{R}^0 = \mathbf{R}(\mathbf{x}_i)$
		$\Delta \mathbf{r}_1 = \Delta \hat{\mathbf{F}}_i$
2	Iterative calculation	
2.1	calculate \mathbf{K}_T^j from \mathbf{R}^{j-1} and then $\hat{\mathbf{K}}_T^j$ considering Eq. (68)	
2.2	$\hat{\mathbf{K}}_T^j \Delta \mathbf{x}^j = \Delta \mathbf{R}^j \implies \Delta \mathbf{x}^j$	
2.3	$\mathbf{x}_{i+1}^j = \mathbf{x}_{i+1}^{j-1} + \Delta \mathbf{x}^j$	
2.4	calculate $\mathbf{R}^j = \mathbf{R}(\mathbf{x}_{i+1}^j)$	
2.5	$\Delta \mathbf{f}^j = \mathbf{R}^j - \mathbf{R}^{j-1} + (\mathbf{K}_T^j - \mathbf{K}_T) \Delta \mathbf{x}^j$	
2.6	$\mathbf{r}^{j+1} = \mathbf{r}^j - \Delta \mathbf{f}^j$	

Table 1: Newton Raphson iteration within the time step t_i to t_{i+1} (cf. Chopra (2001))

This algorithm converges very fast, but the recalculation of $\hat{\mathbf{K}}_T^j$ and the calculation of the inversion in order to solve the linear systems of equations to obtain \mathbf{x}_{i+1}^j becomes time-consuming if high dimensional models are used, i.e. if the tangent stiffness matrix becomes large. Therefore often a more efficient algorithm is used: the modified Newton Raphson algorithm.

In this procedure the tangent stiffness matrix is only calculated once per time step. As a consequence $\hat{\mathbf{K}}_T$ does not depend on the number of iterations. Table 2 presents the iteration procedure of the modified Newton Raphson algorithm illustrated Fig. 10.

The number of iterations is larger than for the standard Newton Raphson iteration but the computational cost per iteration step is much smaller since no assembling and factorization of $\hat{\mathbf{K}}_T$ is necessary. It is possible to operate over more than one time step with the same tangent stiffness matrix as well if the number of iterations does not become too large.

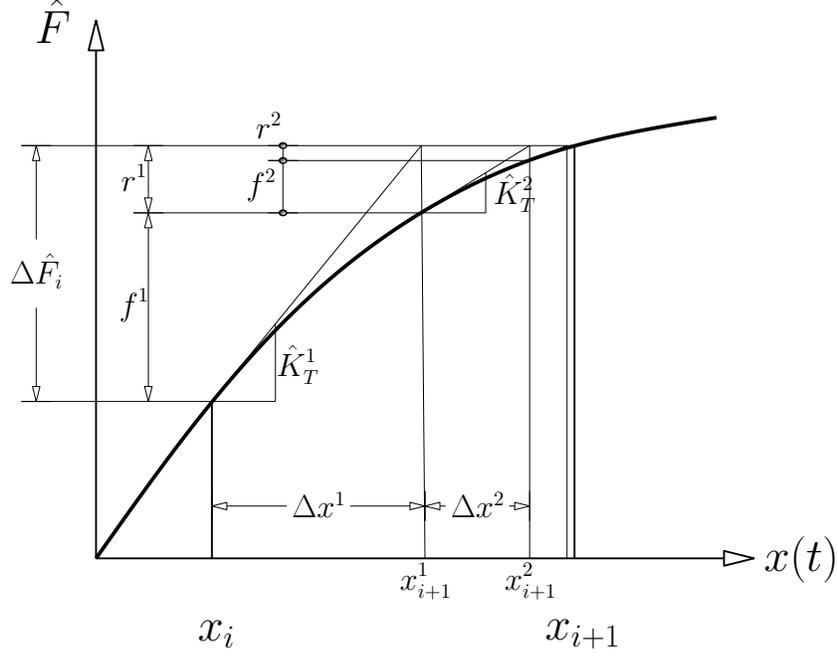


Figure 9: Newton Raphson iteration (cf. Chopra (2001))

1	Initial conditions	
	$\mathbf{x}_{i+1}^0 = \mathbf{x}_i$	$\mathbf{R}^0 = \mathbf{R}(\mathbf{x}_i)$
		$\Delta \mathbf{r}_1 = \Delta \hat{\mathbf{F}}_i$
	(calculate \mathbf{K}_T from \mathbf{R}^0 and then $\hat{\mathbf{K}}_T$ considering Eq. (68) $\implies \hat{\mathbf{K}}_T^{-1}$)	
2	Iterative calculation	
2.1	$\Delta \mathbf{x}^j = \hat{\mathbf{K}}_T^{-1} \Delta \mathbf{R}^j$	
2.2	$\mathbf{x}_{i+1}^j = \mathbf{x}_{i+1}^{j-1} + \Delta \mathbf{x}^j$	
2.3	calculate $\mathbf{R}^j = \mathbf{R}(\mathbf{x}_{i+1}^j)$	
2.4	$\Delta \mathbf{f}^j = \mathbf{R}^j - \mathbf{R}^{j-1} + (\mathbf{K}_T^j - \mathbf{K}_T) \Delta \mathbf{x}^j$	
2.5	$\mathbf{r}^{j+1} = \mathbf{r}^j - \Delta \mathbf{f}^j$	

Table 2: Modified Newton Raphson iteration within the time step t_i to t_{i+1} (cf. Chopra (2001))

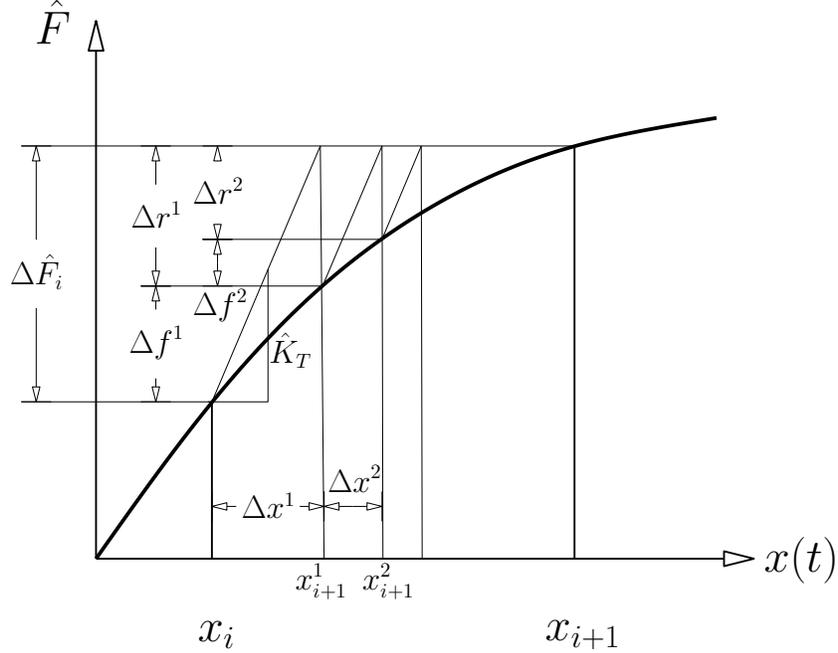


Figure 10: Modified Newton Raphson iteration

3.4 Model order reduction

The main goal of the most model order reduction (MOR) techniques is primarily to define a transformation matrix $\mathbf{T} \in \mathbb{R}^{n \times m}$, $m \ll n$ to approximate the coordinate vector $\mathbf{x} \in \mathbb{R}^n$ through a reduced coordinate vector $\mathbf{q}_r \in \mathbb{R}^m$ by the relation (cf. Koutsovasilis and Beiteltschmidt (2008))

$$\mathbf{x} = \mathbf{T}\mathbf{q}_r$$

so that the dynamic properties of the system are preserved and the error is small. The notation of the variables $n, m \in \mathbb{N}$, the number of DOF of the system and the dimension in the reduced subspace, are used consistently in this work.

The projection of the linear system defined by Eq. (2) into that subspace leads to the second order ODE (cf. Koutsovasilis and Beiteltschmidt (2008))

$$\mathbf{m}_r \ddot{\mathbf{q}}_r + \mathbf{c}_r \dot{\mathbf{q}}_r + \mathbf{k}_r \mathbf{q}_r = \mathbf{f}_r, \quad (70)$$

where $\mathbf{m}_r = \mathbf{T}^T \mathbf{M} \mathbf{T}$, $\mathbf{c}_r = \mathbf{T}^T \mathbf{C} \mathbf{T}$, $\mathbf{k}_r = \mathbf{T}^T \mathbf{K} \mathbf{T} \in \mathbb{R}^{m \times m}$ are mass-, damping- and stiffness matrix and $\mathbf{f}_r = \mathbf{T}^T \mathbf{F}(t) \in \mathbb{R}^{m \times 1}$ is the force vector in the reduced subspace. It should be noted that the reduced system matrices \mathbf{m}_r , \mathbf{c}_r and \mathbf{k}_r are not necessarily diagonal. In this work this procedure is called transformation by \mathbf{T} .

The projection of the nonlinear system defined by Eq. (64) leads to the equation of motion (cf. Koutsovasilis and Beiteltschmidt (2008))

$$\mathbf{m}_r \ddot{\mathbf{q}}_r + \mathbf{c}_r \dot{\mathbf{q}}_r + \mathbf{r} = \mathbf{f}_r, \quad (71)$$

in the reduced subspace, where \mathbf{m}_r , \mathbf{c}_r and \mathbf{f}_r are equivalent to those in Eq. 70. The vector of the restoring forces in the reduced subspace is

$$\mathbf{r} = \mathbf{T}^T \mathbf{R}(\mathbf{x}) = \mathbf{T}^T \mathbf{R}(\mathbf{T}\mathbf{q}_r) \quad . \quad (72)$$

Consequently the vector of the restoring forces must be calculated on the full system at every time step.

3.4.1 Modal truncation

Modal truncation is a very old and simple but also very effective MOR method. First papers appeared in the 1960's dealing with the problem of modal dimension reduction (e.g. Davidson (1966), Gyan (1965), Bampton and Craig (1968)). Following the method of modal analysis and respectively the method of modal truncation is excessively prepared in e.g. Chopra (2001), Qu (2004), Silva (2007). This method is very popular due to its simplicity and accuracy but it has a limited area of application for nonlinear systems. Additionally the frequency band of the excitation force has to be known and comparatively narrow, otherwise a truncation of only a few modes can not consider the complete main dynamic behavior of the system.

In engineering applications dynamical systems with thousands and more DOF are often used but depending on the excitation force vector only a few lower L ($L \ll n$) modes are of interest for the calculation:

$$\mathbf{x}(t) = \sum_{i=1}^L \varphi_{\mathbf{m},i} q_{m,i} = \mathbf{\Phi}_{\mathbf{m},r} \mathbf{q}_{\mathbf{m},r} \quad . \quad (73)$$

This method is called modal truncation (Qu (2004)). If the excitation forces are in the middle frequency range, the number of considered modes L would have to be very large. As a consequence one can use low-high modal truncation scheme where only the modes between the L_1 th and the L_2 th mode are considered:

$$\mathbf{x}(t) = \sum_{i=L_1}^{L_2} \varphi_{\mathbf{m},i} q_{m,i} = \mathbf{\Phi}_{\mathbf{m},r} \mathbf{q}_{\mathbf{m},r} \quad . \quad (74)$$

Substitution of the relation in Eq. (73) into the linear equation of motion [2] and left multiplication with $\mathbf{\Phi}_{\mathbf{m},r}$ leads to the L dimensional coordinate in the reduced modal space

$$\mathbf{m}_{\mathbf{m},r} \ddot{\mathbf{q}}_{\mathbf{m},r} + \mathbf{c}_{\mathbf{m},r} \dot{\mathbf{q}}_{\mathbf{m},r} + \mathbf{k}_{\mathbf{m},r} \mathbf{q}_{\mathbf{m},r} = \mathbf{f}_{\mathbf{m},r} \quad , \quad (75)$$

where the reduced modal mass -, damping and stiffness matrices, which are transformed by $\mathbf{\Phi}_{\mathbf{m},r}$, are the diagonal square matrices of the dimension L and $\mathbf{f}_{\mathbf{m},r}$ is the L dimensional modal force vector transformed by $\mathbf{\Phi}_{\mathbf{m},r}$.

3.4.2 Proper Orthogonal Decomposition

The Proper Orthogonal Decomposition (POD) (Chatterjee (2000), Volkwein (2008a), Volkwein (2008b), Holmes et al. (1996), Qu (2004), Liang et al. (2002)) is a straight forward method for obtaining a low dimensional uncorrelated process of a correlated high dimensional or even infinite dimensional process. Holmes et al. (1996) examine the theoretical background of the POD and its properties profoundly. The aim of the POD calculation is to find a set of ordered orthonormal basis vectors in a subspace so that samples in a sample space are expanded in terms of l basis vectors in an optimal form. This means that

the POD is able to find an orthonormal basis, which describes a observation vector in a subspace better than any other orthonormal basis can do. A measure for this problem is the mean square error (Qu (2004))

$$E \{ \|\mathbf{x} - \mathbf{x}(l)\|^2 \} \leq E \{ \|\mathbf{x} - \hat{\mathbf{x}}(l)\|^2 \} , \quad (76)$$

where $\mathbf{x} \in \mathbb{R}^{n \times 1}$ is the random vector, $\mathbf{x}(l)$ is the approximation of this random vector in an l dimensional POD subspace and $\hat{\mathbf{x}}(l)$ is the approximation of the random vector by any other possible orthonormal basis. Therefore the random vector can be expressed as (cf. Qu (2004))

$$\mathbf{x} = \Phi_{\mathbf{p}} \mathbf{q}_{\mathbf{p}} , \quad \Phi_{\mathbf{p}} = [\varphi_{\mathbf{p},1}, \varphi_{\mathbf{p},2}, \dots, \varphi_{\mathbf{p},s}] \quad \text{and} \quad \mathbf{q}_{\mathbf{p}} = [q_{p,1}, q_{p,2}, \dots, q_{p,s}] , \quad (77)$$

where $\varphi_{\mathbf{p},i}$ are the POD modes and $q_{p,i}$ denote the coordinates in the POD subspace and s is the number of realizations of the random vector. This leads to the objective function of the optimization problem (cf. Qu (2004))

$$\epsilon^2(l, t) = E \{ \|\mathbf{x} - \mathbf{x}(l)\|^2 \} \rightarrow \min \quad (78)$$

subject to the orthonormality condition (cf. Qu (2004))

$$\varphi_{\mathbf{p},i}^T \varphi_{\mathbf{p},j} = \delta_{ij} \quad (i, j = 1, 2, \dots, s) . \quad (79)$$

The transformation into the l dimensional POD subspace is a truncation of the first l lower POD modes (cf. Qu (2004))

$$\mathbf{x}(l) \approx \Phi_{\mathbf{p},r} \mathbf{q}_{\mathbf{p},r} , \quad \Phi_{\mathbf{p},r} = [\varphi_{\mathbf{p},1}, \varphi_{\mathbf{p},2}, \dots, \varphi_{\mathbf{p},l}] , \quad l < s \ll n . \quad (80)$$

In structural dynamics systems are discretized in space and time and the random vector is realized by s observations at different time instants (cf. Han and Feeny (2003))

$$\mathbf{X}_{\mathbf{s}} = [\mathbf{x}_{t_1}, \mathbf{x}_{t_2}, \dots, \mathbf{x}_{t_s}] = \begin{pmatrix} x_1(t_1) & \cdots & x_1(t_s) \\ \cdots & \cdots & \cdots \\ x_n(t_1) & \cdots & x_n(t_s) \end{pmatrix} . \quad (81)$$

These observations \mathbf{x}_{t_i} are called snapshots and therefore in the literature often the observation matrix $\mathbf{X}_{\mathbf{s}}$ is called snapshot matrix. $\mathbf{x}_{t,i}$ can be measurements or they are solution vectors of a dynamical system at different time instants (Chatterjee (2000)). If $\boldsymbol{\mu}$ is the expectation of all observations then the sample covariance matrix $\boldsymbol{\Sigma}_{\mathbf{s}}$ of the random vector, which is realized by the observation matrix, is defined by (cf. Kerschen et al. (2005))

$$\boldsymbol{\Sigma}_{\mathbf{s}} = E \{ (\mathbf{x} - \boldsymbol{\mu})^T (\mathbf{x} - \boldsymbol{\mu}) \} . \quad (82)$$

The POD modes and the POD values are defined by the eigensolution of the sample covariance matrix. If the data have zero mean the covariance matrix is (cf. Kerschen et al. (2005))

$$\boldsymbol{\Sigma}_{\mathbf{s}} = \mathbf{X}_{\mathbf{s}}^T \mathbf{X}_{\mathbf{s}} \quad (83)$$

and the POD is realized by the singular value decomposition of the observation matrix $\mathbf{X}_{\mathbf{s}}$. The POD modes $\varphi_{\mathbf{p},i}$ are equal to the left singular vectors and the POD values $\lambda_{p,i}$ to

the singular values of \mathbf{X}_s , which are all real and positive and arranged in an rectangular diagonal matrix in descending order. The energy, which is contained by the snapshot matrix is defined by the summation of the POD values, i.e. $V = \sum_{i=0}^n \lambda_{p,i}$. As a consequence the energy ratio of the i_{th} POD mode is (cf. Kerschen et al. (2005))

$$V_i = \frac{\lambda_{p,i}}{\sum_{i=0}^n \lambda_{p,i}} . \quad (84)$$

In dynamic problems often the sum of only a few POD values often captures 99.99 percent of the total energy included in the observation matrix, which reflects the big advantage of the POD, i.e. the property of optimality with respect to the energy in a least square sense.

According to Liang et al. (2002) there are three schemes that can realize the POD. These are the *Karhunen-Loeve Decomposition* (KLD), the *Principal Component Analysis* (PCA) and the *Singular Value Decomposition* (SVD). It should be noted that in a finite-dimensional space the schemes can be converted into one another, so that formally they are equivalent.

Singular value decomposition The Singular Value Decomposition (Chatterjee (2000), Volkwein (2008a), Volkwein (2008b), Holmes et al. (1996), Bronstein et al. (2001)) can be viewed as the generalisation of the eigenvalue problem and is the decomposition of a rectangular $n \times m$ matrix \mathbf{A} into three matrices.

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad (85)$$

$\mathbf{U} = [\mathbf{U}_1, \dots, \mathbf{U}_n]$ is a $n \times n$ orthonormal matrix in which the columns are the eigenvectors of the matrix $\mathbf{A}^T\mathbf{A}$, $\mathbf{V} = [\mathbf{V}_1, \dots, \mathbf{V}_m]$ is a $m \times m$ orthonormal matrix in which the columns are the eigenvectors of the matrix $\mathbf{A}\mathbf{A}^T$ and $\mathbf{\Sigma}$ is a $n \times m$ rectangular diagonal matrix containing the singular values, which are equal to the square roots of the eigenvalues of $\mathbf{A}\mathbf{A}^T$ (or, equivalently of $\mathbf{A}^T\mathbf{A}$) in descending order. The columns of \mathbf{U} are called the left singular vectors and the columns of \mathbf{V} are called the right singular vectors. The SVD realizes the POD, the mathematical proof is given in Volkwein (2008a).

For detailed information about KLD and PCA the reader is referred to Liang et al. (2002).

Practical computation of a POD basis The practical computation of the i_{th} POD basis vector is as follows (cf. Volkwein (2008a))

$$\mathbf{u}_i = \frac{1}{\sqrt{\lambda_i}} \mathbf{X}_s \mathbf{v}_i \quad i = 1 \dots l , \quad (86)$$

where \mathbf{v}_i are the first l left singular vectors and λ_i are the first l singular values of $\Phi_{\mathbf{p},\mathbf{r}}$ (Volkwein (2008a)). The POD basis $\Phi_{\mathbf{p},\mathbf{r}}$ with l considered basis vectors is (cf. Volkwein (2008a))

$$\Phi_{\mathbf{p},\mathbf{r}} = [\mathbf{u}_1, \dots, \mathbf{u}_l] \text{ with } (l \ll n) . \quad (87)$$

3.5 Summary

This section provides tools, which can be utilized to solve in homogenous linear problems with arbitrary force functions. The force functions are defined piecewise continuous by interpolation polynomials of order 0 and 1. The formally closed solution by the Duhamel integral and piecewise analytic solution algorithms are presented under consideration of the diagonalization of the equations using the mode superposition. Additionally the piecewise analytic solution in the state space, for nondiagonalizable matrices is shown. Direct time integration is introduced. Explicit time integration is presented with emphasis on the second order central difference method as well as implicit methods by the example of the Newmark algorithm. On the one side explicit integrators have the property of low cost matrix operations per time step but on the other side the problem of stability forces the time step of the algorithm to be very small. Thus the number of time steps that must be worked off is very large. The Newmark method is unconditionally stable if a constant acceleration within one time step is assumed. Therefore the number of time steps depends only on the level of required accuracy. Although the Newmark method belongs to the family of implicit time integrators, it is possible to define an explicit formulation for linear systems. However, the Newmark method requires the factorization of the stiffness matrix, which can be very time consuming if the dimensions are high.

Nonlinear problems in structural dynamics require numeric time integration algorithms in order to obtain the response function. Therefore an extension to the direct integration methods for linear systems needs to be applied. In this context the central difference approximation as example for explicit integration schemes and the Newmark method as example for implicit integration schemes for nonlinear systems is discussed. Based on the information of section 3.2.7 the central difference approximation can easily be adapted for nonlinear systems. Only the linear restoring force $\mathbf{K}\mathbf{x}_i$ is substituted by the vector of the nonlinear restoring force $\mathbf{R}(\mathbf{x}_i)$. From this point of view the direct numeric integration schemes for nonlinear systems are not a closed operation since the calculation of the vector of the restoring forces depends on the nonlinearity. The problem of stability is comparable to that of linear problems. The explicit formulation of the Newmark method would lead to inaccurate response functions, and therefore an iterative formulation is presented. The Newton Raphson iteration is discussed. This algorithm is inefficient as in every iteration step the tangent stiffness matrix must be assembled and inverted, which leads to unacceptable calculation times. As a consequence the modified Newton Raphson iteration is presented. Though this algorithm does not converge so fast, a lot of computational effort can be saved since the cost per iteration step is considerably smaller.

4 The proper orthogonal decomposition as model order reduction method in earthquake engineering

4.1 Application of the proper orthogonal decomposition for structures under transient excitation

Time integration of the full system over the snapshot time period provides the snapshot matrix \mathbf{X}_s . According to section 3.4.2 the POD - basis $\Phi_{p,r}$ is calculated, which should

be capable of representing the main behavior of the system. The POD basis is the transformation matrix into the l - dimensional POD subspace, where time integration over the whole time period happens. The approximated solution in the POD subspace can now be transformed back into the full space. A sketch of the realization of the POD strategy in this context is given in Fig. 11.

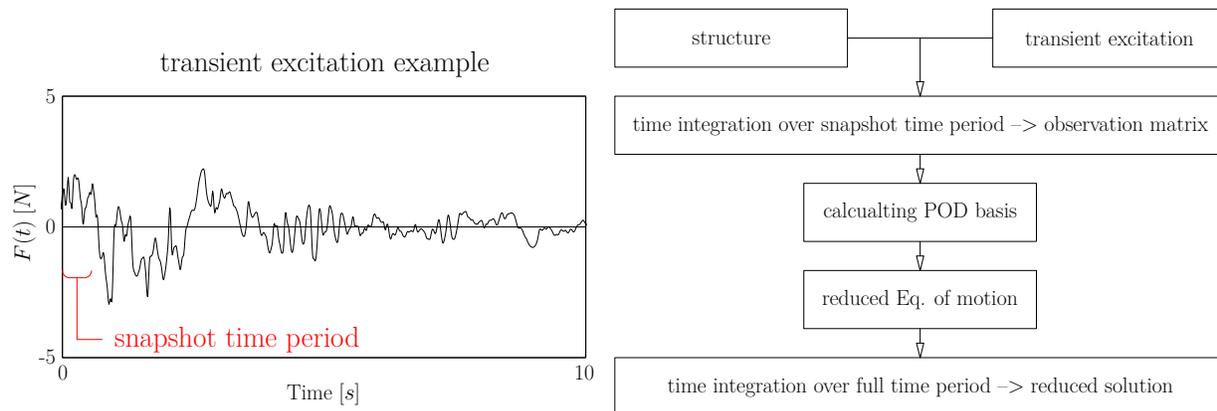


Figure 11: Left subplot: transient excitation (earthquake), right subplot: calculation scheme

An earthquake record is presented as example for a transient excitation, i.e. the north-south component of the *El Centro* Earthquake. (http://www.vibrationdata.com/elcentro_NS.dat). The acceleration time history with a sampling rate of $1/50$ [s] is shown in Fig. 12. In the following the response of a linear high dimensional structure is calculated apply-

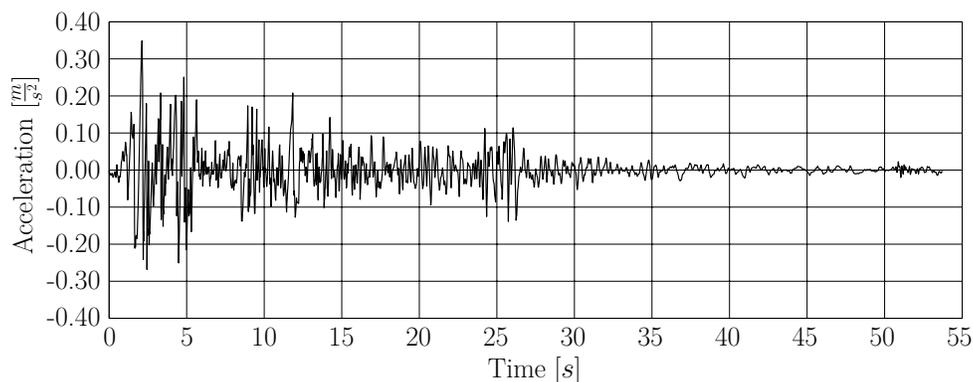


Figure 12: NS acceleration of the El Centro earthquake

ing the proposed MOR procedure using the POD. In order to evaluate the results the POD solution is compared with outcomes using the classical method of modal truncation and the full solution. After proving the adaptability of this procedure it is applied to a nonlinear structure.

4.2 The procedure for linear structures

4.2.1 Linear dynamic model

The dynamical system is chosen as a two dimensional $Y \times Z$ frame system representative for civil engineering applications. Y denotes the number of stories and Z the number of bays of the frame system. The system is discretized by finite beam elements. For detailed information about finite beam elements the reader is invited to study Bathe (1995).

The following parameters are chosen:

- *geometrical parameters*: $Y = 3$, $Z = 5$, elements per column... 5, elements per beam... 5, total height = $20 \times Y$, total width = $20 \times Z$
- *structural parameters*: $E = 2.0 \times 10^{11} [N/m^2]$, $\rho = 7850 [kg/m^3]$, rectangular cross section of the beam elements... $0.3 \times 0.3 [m]$, damping coefficient $\zeta = 0.01$ (implemented for both Rayleigh damping coefficients)

A script has been written, which allows to adapt Y , Z , the total height and width and the element size of the beam elements not only for the beams but also for the columns (Appendix A.4.1). Hence it is possible to increase or decrease the DOF in order to test the solution of the ROM in comparison to the full model. The structure is simply supported at the base. Fig. 13 shows a sketch of the structure.

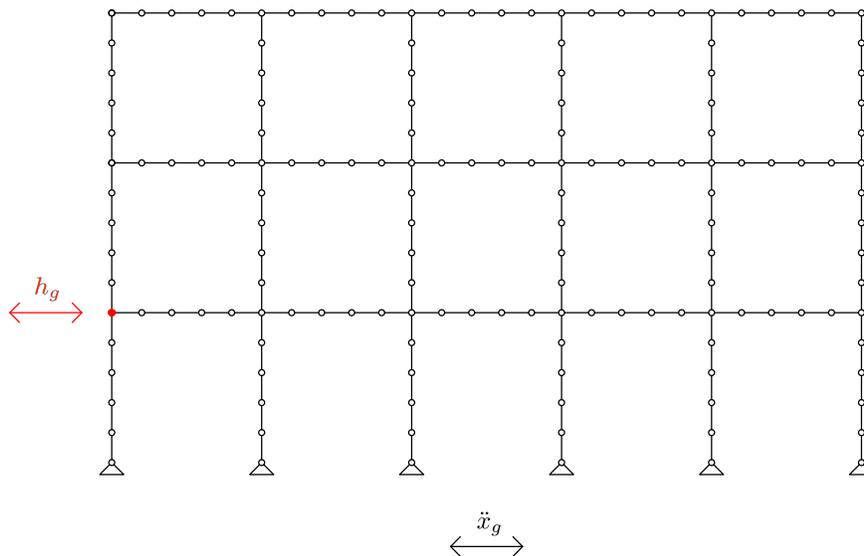


Figure 13: Discretized three-story structure with five bays, $h_g(t)$ denotes the horizontal displacement of the first floor at the left corner (Bamer and Bucher (2012))

Full system The set of equations of motion for a dynamic system under single-point earthquake excitation reads as (Chopra (2001))

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = -\mathbf{M}\ddot{x}_g(t)\mathbf{f} \quad . \quad (88)$$

The right hand side describes the excitation of the structure due to an earthquake excitation. $\ddot{x}_g(t)$ is the ground acceleration depicted in Fig. 12 and \mathbf{f} is the quasi-static influence vector. In case of only horizontal ground accelerations the entries of \mathbf{f} , which are related to the horizontal degrees of freedom, are *one*, all others are *zero*.

In order to compare the results of the full and the reduced solutions the full response \mathbf{x} is obtained by time integration using the method of *central differences*, which is applicable to nonlinear systems as well. The correctness of the solution is tested by applying two other algorithms. According to the transformation of Eq. (10) the decoupled system is solved by the by a piecewise analytic solution algorithm presented in Eq. (16) and transformed back to the coordinate \mathbf{x} . Additionally the Newmark algorithm presented in Eq. (53) is implemented. The full response functions of the central difference algorithm, the piecewise analytic algorithm and the Newmark algorithm coincide sufficiently accurately. the appropriate algorithm depends on the level of high dimensionality. While the Newmark method and piecewise analytic methods are limited considering high dimensionality because of the matrix inversion of the stiffness matrix and iteration in each time step, the method of central differences is not so limited considering storage and much larger systems can be calculated. Therefore for both calculations, the full and the reduced system, the computational time of the central difference method is compared.

The critical time step of the central difference method is according to Eq. (52)

$$\Delta t_{crit} = \frac{T_n}{\pi} \quad (89)$$

where $T_n = \frac{2\pi}{\omega_n}$ is the smallest period of vibration. The finer the structure is discretized, the smaller is the smallest oscillation period and as a consequence the smaller is the critical time step. If the chosen time step is larger then the critical time step explicit integration methods become unstable and the solution grows exponentially to infinity (see section 3.2.7 as well).

The chosen time step Δt for the calculation is 10^{-5} seconds. Although the number of DOF is *only* 456, the calculation is already fairly time-consuming (computational time 350 [s]). The smallest period of vibration is 6.56×10^{-4} seconds and the critical time step is therefore 2.09×10^{-4} seconds.

POD - reduced order model In order to derive the snapshot matrix \mathbf{X}_s , composed of the *snapshot vectors*, the full system is integrated numerically over a short time period of the beginning of the earthquake record. According to Eq. (86) and (87) the first l POD basis vectors (POD modes) are calculated and arranged into the POD basis matrix $\Phi_{p,r}$. Insertion of the POD transformation relation in Eq. (80) into Eq. (88) and multiplying with $\Phi_{p,r}$ leads to the reduced linear equation of motion in the POD - subspace,

$$\mathbf{M}_{pod}\ddot{\mathbf{q}} + \mathbf{C}_{pod}\dot{\mathbf{q}} + \mathbf{K}_{pod}\mathbf{q} = \mathbf{F}_{pod}(t) \quad (90)$$

where $\mathbf{M}_{pod} = \Phi_{p,r}^T \mathbf{M} \Phi_{p,r}$, $\mathbf{C}_{pod} = \Phi_{p,r}^T \mathbf{C} \Phi_{p,r}$, $\mathbf{K}_{pod} = \Phi_{p,r}^T \mathbf{K} \Phi_{p,r}$ and $\mathbf{F}_{pod}(t) = \Phi_{p,r}^T \mathbf{F}(t)$, considering that \mathbf{M}_{pod} , \mathbf{C}_{pod} and \mathbf{K}_{pod} are not necessarily diagonal matrices. This equation is a particular realization of Eq. (70). In contrast to modal analysis and modal truncation the system is not uncoupled and it can be solved by numeric integration using the central difference approximation. Application of Eq. (80) leads subsequently to the approximation of the full displacement vector \mathbf{x} .

In this example 100 snapshots are taken in the first 2 seconds, which means the POD basis for this system is assembled with the information of only the first 2 seconds of the earthquake. The number of the considered POD basis vectors l is chosen as 2 for this example. The reduced system is now integrated numerically by the method of central differences for which, similar to the modal reduction, the critical time step can be dramatically enlarged compared to the full calculation.

The reduced model leads to a very accurate approximation of the full model with a short snapshot period and only 2 POD modes ($\Delta t = 0.02$ [s], calculation time 4 [s]). Using the POD method it is not necessary to analyze the excitation about frequency content, because the POD basis vectors describe the excited system in this time period, in which the snapshots are taken, optimally. Since these POD modes are used for the remaining excitation period, the main error depends only on the changing of the behavior of the excitation force in the remaining time period compared to the snapshot time period.

Modal truncation reduced order model According to Eq. (75) the first two modes are considered in the reduced modal matrix $\Phi_{m,r}$, which means that 456 DOF are approximated by only two uncoupled eigenmodes. Applying the method of modal truncation one should investigate the excitation. Otherwise it could be possible that the frequency range of the excitation is not close to the eigenfrequencies of the used modes and the approximation through modal truncation could fail, which is actually a disadvantage in comparison to the POD method.

Due to the reduction of the DOF the cost of calculation in every time step decreases and as T_n increases to 1.56 seconds, the critical time step gets larger and the number of necessary time steps decreases dramatically, which is the main responsible factor for the saving of computational time. While the full calculation takes a few minutes, the ROM through modal truncation takes only a few seconds ($\Delta t = 0.02$ [s], calculation time 3 [s]).

Comparison of results - error estimation Fig. 14 shows the horizontal displacements of the first floor (left corner), $h_g(t)$ (cf. Fig. 13). The solution of the full model is approximated accurately not only by the modally reduced system but also by the POD reduced system. Although modal truncation is an effective method for earthquake excited linear structures, the POD method provides a powerful alternative.

In order to obtain more information of the error through model order reduction, the mean horizontal error is calculated:

$$E_{mean}(t) = \frac{\sum_{i=1}^{n_h} x_i(t) - x_{i,red}(t)}{n_h} . \quad (91)$$

$E_{mean}(t)$ is the mean error of the horizontal displacements of every node, x_i is the i_{th} horizontal component of the solution vector of the full system $\mathbf{X}(t)$, $x_{i,red}$ is the i_{th} horizontal component of the solution vector $\mathbf{X}_{red}(t)$ and n_h denotes all horizontal DOF. Fig. 15 presents the mean error of the modally reduced system $E_{mean,modal}(t)$ and Fig. 16 the mean error of the POD reduced system $E_{mean,POD}(t)$. The error of the modally reduced linear system is the sum of those modes, which are not included in the reduced calculation. In this case it is the mean of all horizontal displacements of the third to the n_{th} mode. The POD error shown in Fig. 16 is obviously larger than the modally reduced error presented in Fig. 15. A small phase shift of the POD solution can be observed caused by the fact that not the complete

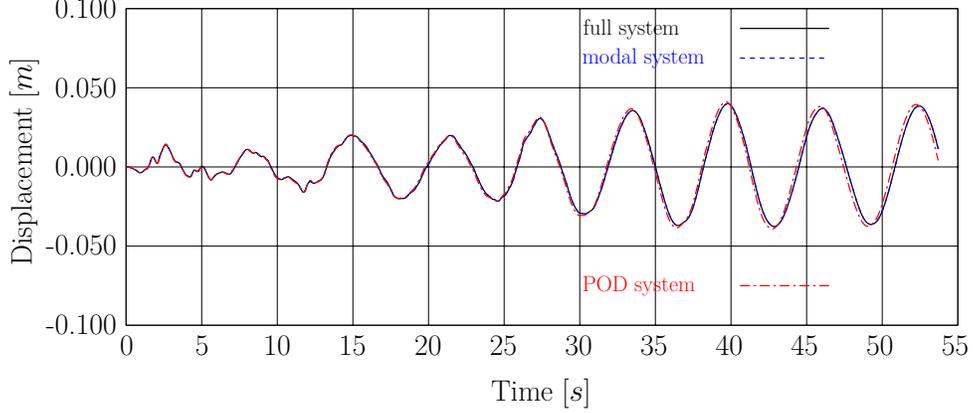


Figure 14: Horizontal displacements based on different solution procedures (Bamer and Bucher (2012))

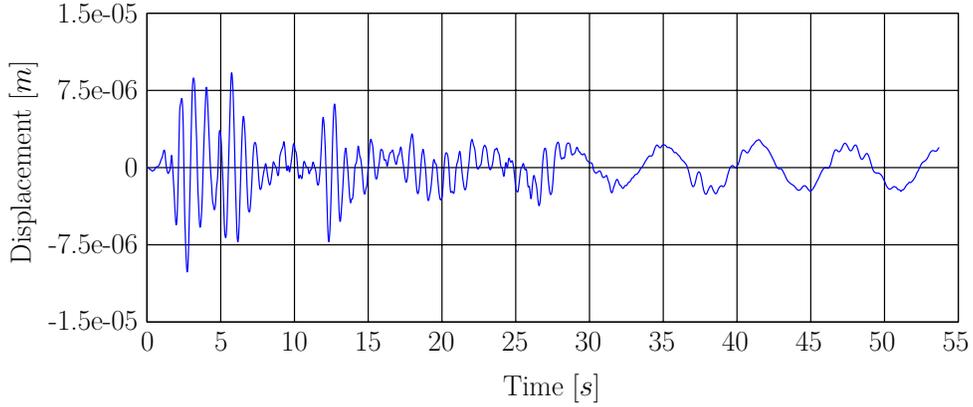


Figure 15: Mean modal truncation error, $E_{mean,modal}(t)$ (Bamer and Bucher (2012))

frequency information of the earthquake excitation is contained in the snapshot time period at the beginning of the excitation. From the engineering point of view the approximation of both methods is sufficiently accurate. Additionally the errors are both within a range that can be observed approximating analytic full solutions by numerical approximations applied to the full system as well, therefore the error is neglect able concerning engineering problems.

POD modes Fig. 17 depicts the first six POD modes, which depend not only on the dynamical system but also on the excitation.

The *importance* of these modes can be visualized by using the error, which is caused by the approximation of the system through the modes,

$$\mathbf{E} = \mathbf{X} - \mathbf{X}_{red} \quad , \quad (92)$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_s]$ describes the exact solution, $\mathbf{X}_{red} = [\mathbf{x}_{1,red}, \mathbf{x}_{2,red}, \dots, \mathbf{x}_{s,red}]$ is the reduced solution and s is the number of time steps. To get information about the error dependent on the number of considered modes, the variances (i.e. the diagonal elements

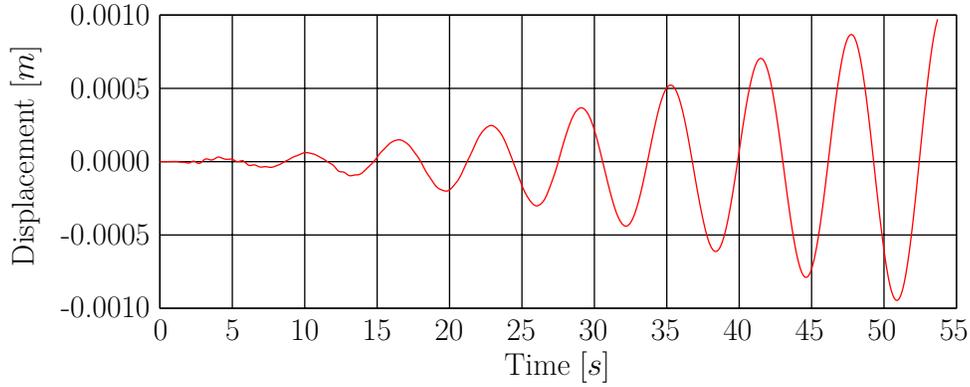


Figure 16: Mean POD reduction error, $E_{mean,POD}(t)$ (Bamer and Bucher (2012))

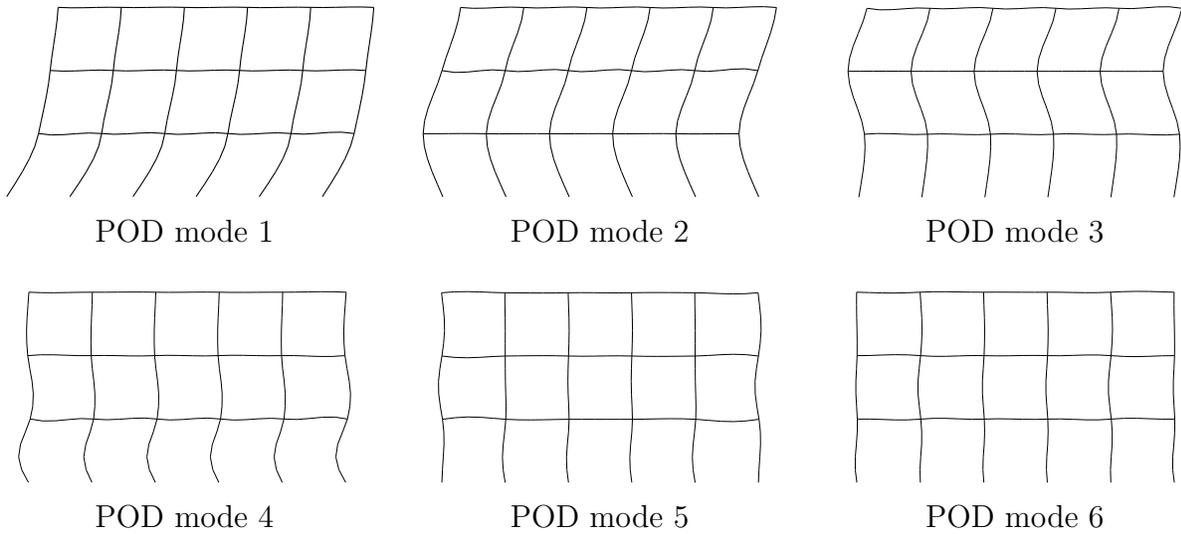


Figure 17: POD modes of the linear system (Bamer and Bucher (2012))

of the covariance matrix $\mathbf{D} = \frac{1}{s-1}\mathbf{E}^T\mathbf{E}$) are added. Fig. 18 shows the first invariant of the covariance matrix \mathbf{D} within the snapshot time period as a function of the considered modes. Since the POD satisfies the condition that the quadratic error is smaller than the quadratic error caused by any other orthonormal set of basis vectors of the same dimension, the error caused by the POD modes must be smaller than the error based on *normal* modes. Fig. 19 represents the first invariant of the covariance matrix \mathbf{D} dependent on the number of considered modes within the whole time period of the excitation. It demonstrates that there is hardly any difference between POD and modal reduction when using *only* 3 or more modes.

For this specific example, the conclusion can be drawn that the approximation of the solution is not essentially improving when using more than three POD modes.

The invariant of the covariance matrix \mathbf{D} is here presented as possibility to measure the error of the approximated solution vector as a function of time by only one scalar. Therefore, within the snapshot time period the optimality condition of the POD method, i.e. no other orthonormal basis does exist that is able to approximate the solution within the snapshot time period more accurately, is confirmed. This is shown in Fig. 18. Due to

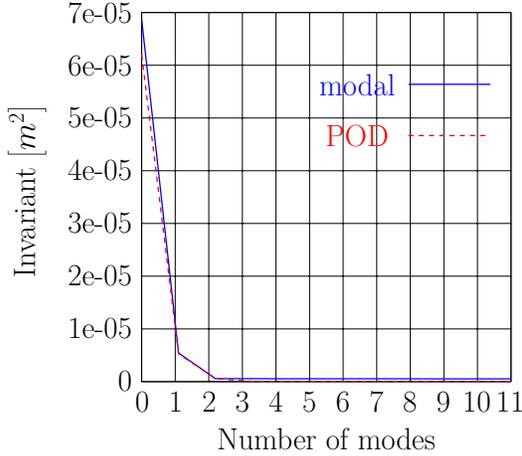


Figure 18: Invariant of the covariance matrix \mathbf{D} in the snapshot time period (Bamer and Bucher (2012))

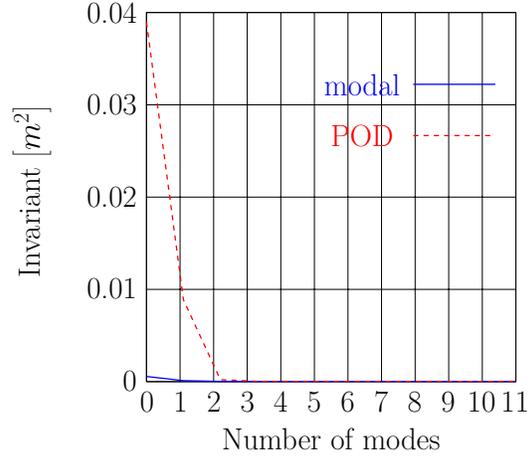


Figure 19: Invariant of the covariance matrix \mathbf{D} in the whole observation period (Bamer and Bucher (2012))

frequencies of the earthquake record that do not appear within the snapshot time period (or only related with a small energy content) there are motions patterns that cannot be described by the POD modes in the whole time period. Fig. 19 demonstrates this property clearly. Where the method of modal truncation is able to approximate the solution with only one mode accurately, the POD method requires three modes for an accurate solution. Numerically there is hardly any difference concerning computational effort by integration of a one-dimensional or three-dimensional system, as for both systems the critical time step is much smaller than the resolution time step of the earthquake record. I.e. the POD method can be qualitatively compared to the method of modal truncation. A downside is that the snapshot matrix has to be calculated but on the other hand if dimensions become so high that the factorization of the stiffness matrix, when using the method of modal truncation, would exceed the limits of storage, the presented POD procedure could be executed as the central difference algorithm is economical concerning storage as well as the SVD algorithm if the number of columns of the snapshot matrix is smaller than the number of rows (i.e. DOF).

4.3 The procedure for nonlinear structures

4.3.1 Nonlinear dynamic model

The consistent $Y \times Z$ frame system exhibits the same geometrical and structural parameters, however, $Z + 1$ nonlinear *friction elements* are added to represent friction bearings. The dynamic model of these elements is shown in Fig. 20. $k_0 + k_1$ describes the main stiffness of the element. When the critical friction force r is exceeded, only spring k_0 is active. The friction elements can only be loaded in longitudinal direction by force $H(t)$. The whole frame system is shown in Fig. 21. The properties of the friction bearing elements are chosen as $k_1 = 30000 \text{ N/m}$, $k_0 = 10000 \text{ N/m}$, $r = 50 \text{ N}$.

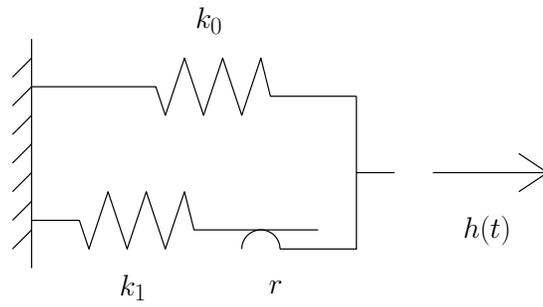


Figure 20: Finite friction element (Bamer and Bucher (2012))

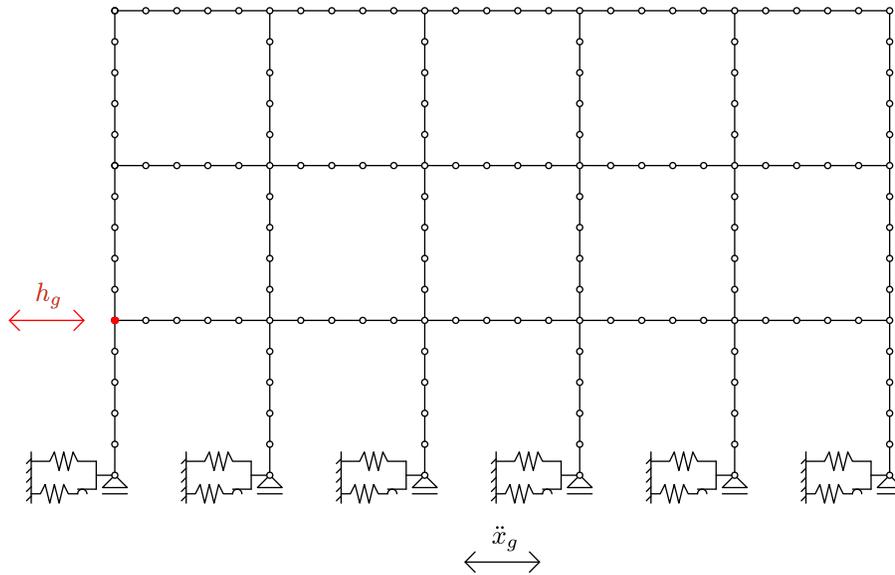


Figure 21: Planar frame structure with friction elements at the supports (Bamer and Bucher (2012))

Full system The coupled set of equations of motion for the nonlinear system subjected to earthquake excitation reads as

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{R}(\mathbf{x}) = -\mathbf{M}\ddot{x}_g(t)\mathbf{f} \quad . \quad (93)$$

Numeric integration is the only useful possibility to derive the solution and as well as for the linear system it is realized by the central difference approximation.

Due to the small Δt_{crit} (2.09×10^{-4} [s]), the time step is chosen as 10^{-5} [s], the total number of DOF is 462 and the computational time is 3150[s].

A part of the dissipation of energy comes from the viscous damping of the structure but the contributions comes from the friction work supplied form the friction elements. This means, that only a part of the whole energy supplied by the earthquake records arrives the actual linear part of the frame structure. Therefore the relative displacements of the frame structure and the friction elements are of great interest and responsible for the magnitude of internal forces.

POD - reduced order model Firstly, the \mathbf{X}_s -matrix is calculated and then the $\Phi_{p,r}$ -matrix (see section 3.4.2), which is the transformation matrix containing the l considered POD basis vectors. 200 Snapshots are taken in equidistant time instants at the beginning of the earthquake record. Thus a snapshot time period of 4 [s] is chosen. In order to compare the quality of result a second POD computation is performed as well, where 20 snapshots every 0.02 [s] are taken, i.e. a snapshot time period of 0.4 [s] at the beginning of the earthquake record is applied. Inserting Eq. (80) into Eq. (93) and left multiplication by $\Phi_{p,r}$ leads to the l - dimensional nonlinear differential equation

$$\mathbf{M}_{pod}\ddot{\mathbf{q}} + \mathbf{C}_{pod}\dot{\mathbf{q}} + \Phi_{p,r}^T \mathbf{R}(\Phi_{p,r}\mathbf{q}) = \mathbf{F}_{pod}(t) \quad . \quad (94)$$

This reduced nonlinear differential equation is solved by the method of central differences considering that $\mathbf{R}(\Phi_{p,r}\mathbf{q}) = \mathbf{R}(\mathbf{x})$ still has to be calculated for the full system at every time step. Nevertheless, the total amount of calculation is much smaller compared to the full system, because if only l (i.e. four) POD modes are considered, the critical time step for the reduced systems is much larger than for the full system. Again the transformation presented in Eq. (80) leads the response in the physical coordinates \mathbf{x} .

Comparison of results - error estimation The horizontal displacements of the left corner in the first floor, $h_g(t)$ (cf. Fig. 21), of the full system and the POD-reduced systems are shown in Fig. 22. The horizontal displacements of the bearing and the relative displacements are presented in Fig. 23 and 24. The relative displacement is the difference of the horizontal displacement of the left support to the displacement in the the first floor (left corner) $h_g(t)$. It is responsible for stresses and forces in the linear part of the structure. The results in Fig. 22 - 27 show the results of the computations. Not only the displacements, i.e. $h_g(t)$, the horizontal displacement of the left support and the relative horizontal displacement of the structure but also shear forces and moments of the full solution are approximated accurately by the solution obtained by the POD method, if 200 snapshots are applied. However, the POD calculation procedure does not provide satisfactory approximations using 20 snapshots, i.e. only 1/10 of the length of the snapshot time period applied before. At about the first 20 seconds of the response history is usefully approximated but

the error increases with time and not only a phase shifting appears but also an overestimation of the maximum displacement of about 50 % can be observed (cf. Fig. 22 - 24). Obviously not only the main frequency content can be captured within this time period of 0.4 seconds but also the activation of sliding by the nonlinear friction elements. Thus the choice of an appropriate length of the snapshot time period is essential for the quality of the approximation applying the POD MOR strategy. However there is no mathematical criteria to determine a priori the length of the snapshot time period to ensure that an accurate approximation can be obtained. Much more this is an "engineering decision", where understanding of dynamics as well as a certain level of experience is required.

The mean error of all horizontal displacements with respect to time, $E_{mean,POD}(t)$, which is calculated according to Eq. (91), is shown in Fig. 25. While the error of the 200 snapshots approximation is negligible the error of the 20 snapshots approximation increases dramatically with time. Consequently the choice of the length of the snapshot time period is the most important decision when using the POD method. Also the bending moments and the shear forces as a function of time of the full system and the POD - reduced systems are calculated and presented in Fig. 26 and 27. The results of the shear forces and the moments point out the conclusions presented in Fig. 22 - 24.

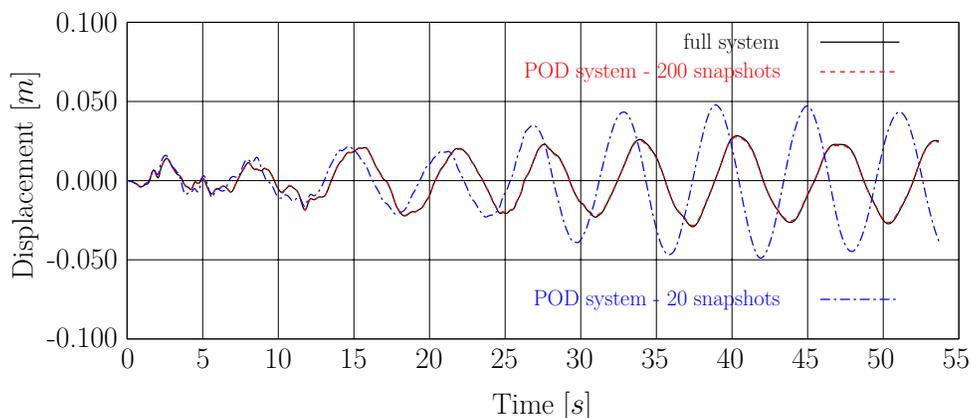


Figure 22: Horizontal displacement left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))

4.3.2 POD modes

Fig. 28 presents the first 6 POD modes of the nonlinear system equivalent to section 4.2.1. They look similar to linear structural modes but they carry nonlinear information of the friction elements.

In Fig. 29 the *importance* of the POD modes is shown, which are described by their corresponding singular values (i.e. POD values). The POD values carry information about the energy content of the corresponding POD mode. The POD is defined by the eigenvalue solution of the sample covariance matrix, i.e. if zero mean is assumed, of the matrix $\mathbf{X}_s^T \mathbf{X}_s$ (cf. Eq. (83)). Since the eigenvalues of the sample covariance matrix are equivalent to the singular values of the snapshot matrix, Fig. 29 is comparable to Fig. 18, where the trace of the covariance matrix \mathbf{D} , i.e. the first invariant of the covariance matrix of the error, is

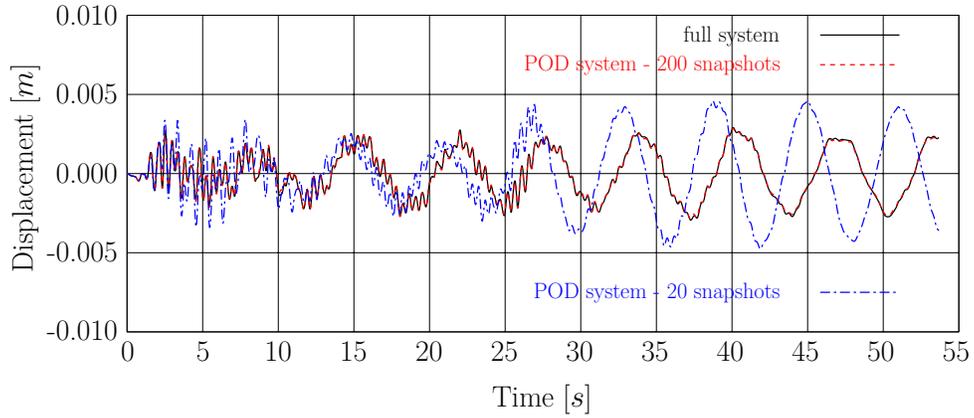


Figure 23: Horizontal displacement - left bearing, full system compared to POD reduced system (Bamer and Bucher (2012))

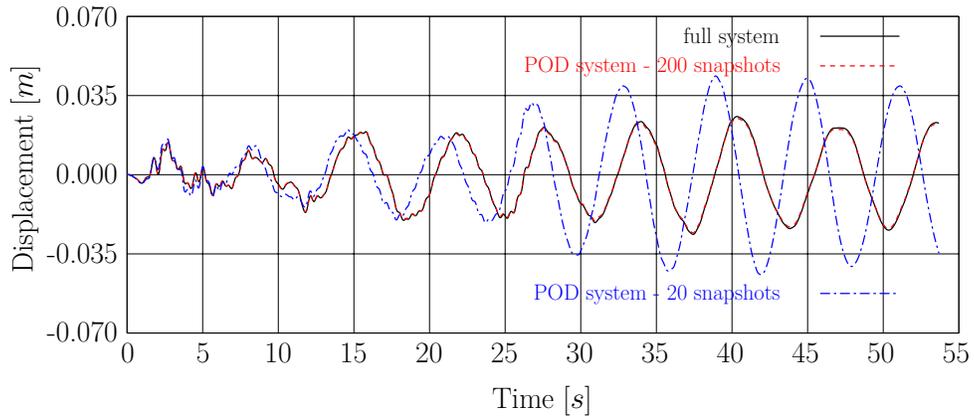


Figure 24: Horizontal displacement - relative, full system compared to POD reduced system (Bamer and Bucher (2012))

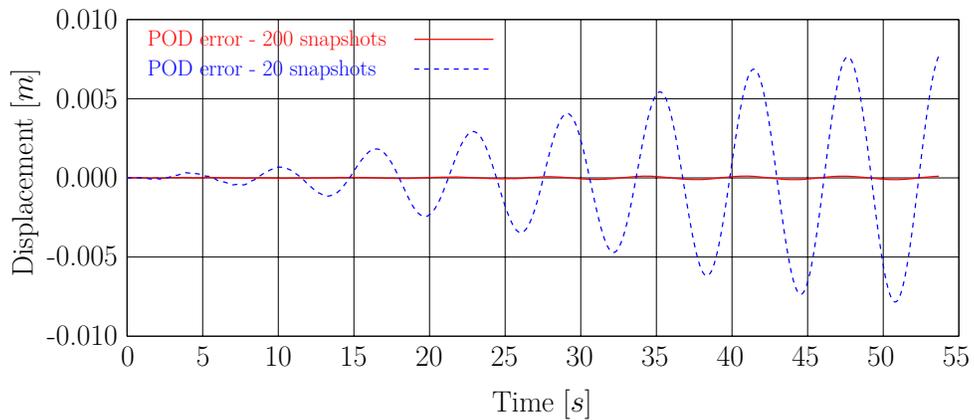


Figure 25: Mean POD error, $E_{mean,POD}(t)$ (Bamer and Bucher (2012))

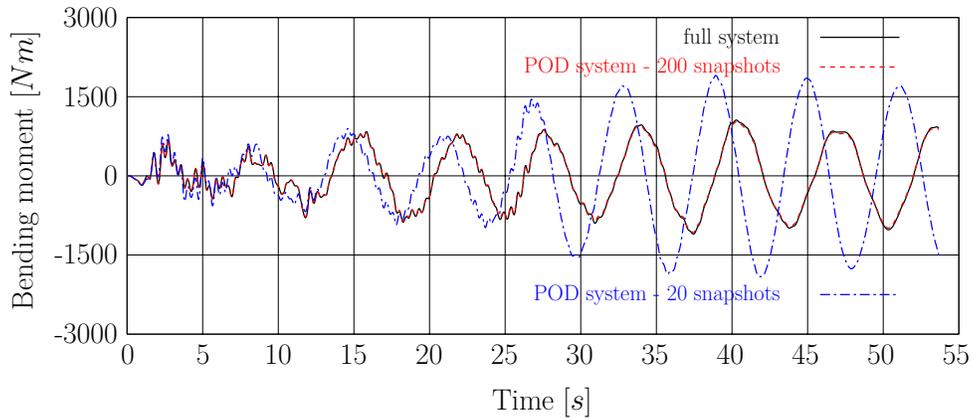


Figure 26: Bending moment left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))

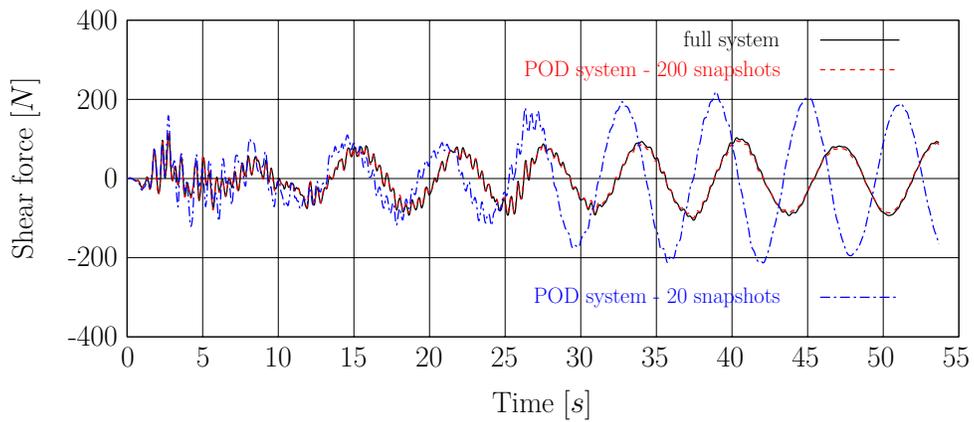


Figure 27: Shear force left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))

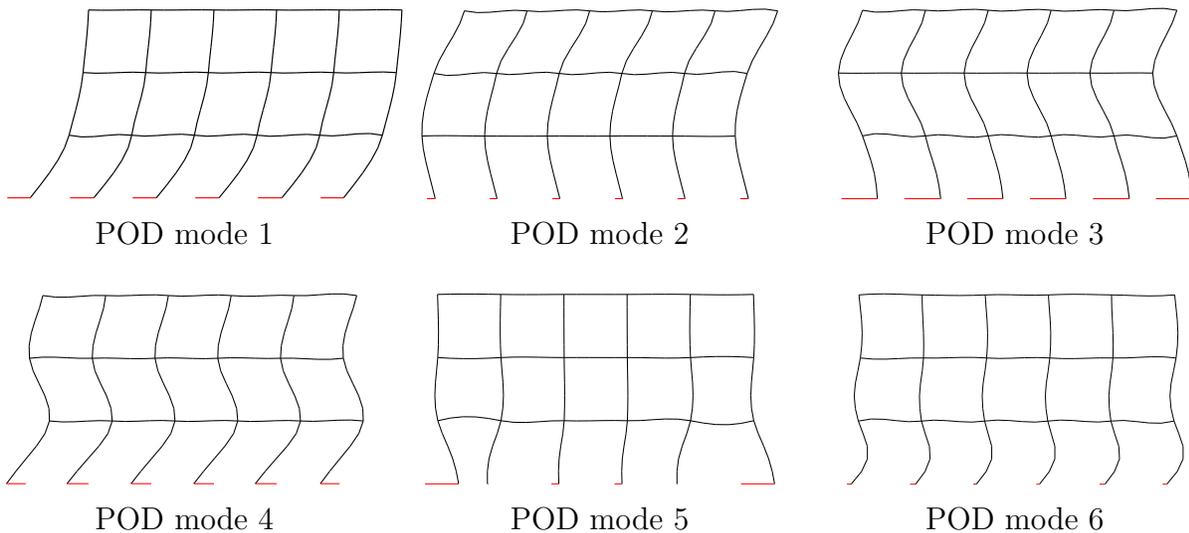


Figure 28: POD modes of the nonlinear system (Bamer and Bucher (2012))

presented. The advantage using the singular values as benchmark for the importance of the POD modes is that only one calculation procedure is conducted instead of more calculations applying different numbers of POD modes.

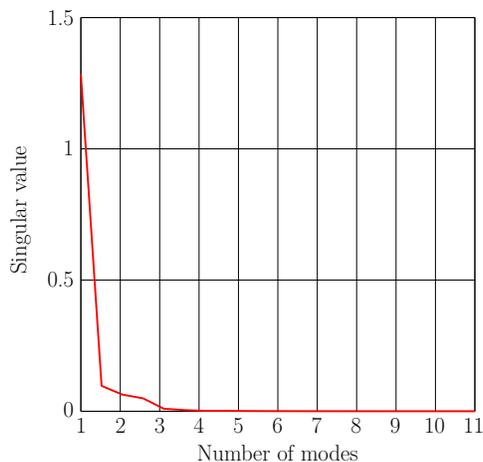


Figure 29: Singular values of the corresponding POD modes (Bamer and Bucher (2012))

4.3.3 Numerical efficiency

For the practical applicability of a numerical approximation method such as the POD it is essential to investigate the trade-off between accuracy and computational effort. Obviously, the results of an analysis based on reduction method can be made more accurate by including more basis vectors. This increases on the other hand the computational effort in a disproportional way, because of the property of optimality, which characterizes the POD.

In this context, it is important to have a qualitative assertion of the error through a scalar, which can be represented by the first invariant of the covariance matrix \mathbf{D} (see section 4.2.1). This scalar is related to the computational time. The first invariant of the covariance matrix depends on the number of snapshots but also on the number of used POD modes. As only a few POD modes describe the main behavior of the system (which is the property of optimality) the approximation of the system is not noticeably improved when using more than 5 modes (see Fig. 19 and 29). The critical time step in this configuration (5 used POD modes) is larger than 0.02 [s], which is the resolution of the earthquake record.

Hence, the main interest concerning computational time and accuracy should be drawn to the number of used snapshots, by using a fixed number of POD modes, which allows to identify the largest possible time step (which is typically given by the time resolution of the earthquake record). The computational time, which increases linearly with the number of snapshots, is influenced only by assembling the snapshot matrix and the POD basis. Since the number of POD modes is fixed, there is no dependency to the computational time. Simultaneously, the first invariant of \mathbf{D} drops rapidly with an increasing number of snapshots (see Fig. 75). This figure clearly shows that the accuracy cannot be improved substantially when using more than 40 snapshots. Beyond that point only the computation time increases but no additional accuracy is gained.

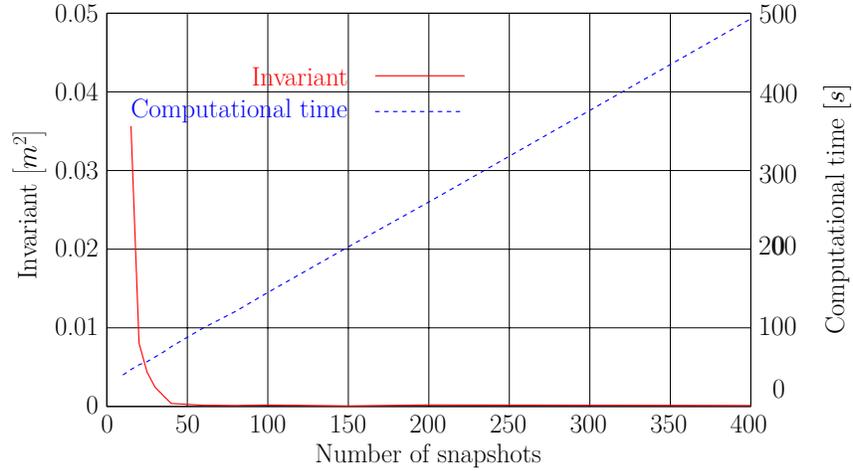


Figure 30: Invariant of \mathbf{D} and calculation time as a function of the used snapshots (Bamer and Bucher (2012))

4.3.4 Conclusion

The analysis of structural systems under severe time-variant (transient) excitation such as impact loads or earthquakes frequently requires the severe consideration of nonlinear structural effects. Consequently, the dynamic analysis must be carried out by utilizing suitable time integration schemes. Due to their simplicity, explicit methods are very popular. However, they typically require the time step to be smaller than a critical value. In many cases, this critical time step is so small that the computation time exceeds acceptable limits.

One possible approach to overcome this difficulty is the application of reduction methods, which attempt to eliminate the high-frequency components from the response and in this way increase the critical time step. In this dissertation the Proper Orthogonal Decomposition technique (POD) is applied as model order reduction method for structures under earthquake excitation. Although earthquake excitations are random processes, the approximation of a system through the POD method by considering snapshots only in a small time period is very accurate. The advantage of the POD method over the method of modal truncation is the property of optimality and its applicability for the analysis of nonlinear systems.

The numerical results of the application to both linear and nonlinear structural models indicate that the POD method is indeed a very powerful method for reducing the degrees of freedom. By varying the number of POD modes retained in the analysis and by varying the number of snapshots used to compute the POD modes, the method provides much flexibility to adjust the balance between computational effort and desired accuracy.

It can be expected that the POD method also works for more general structures although in this paper only the two test objects, a linear and a nonlinear frame system, have been analyzed. In addition, the focus of further analysis should not only be the testing for more general types of structures but also the investigation concerning structural safety and probabilistic design where the saving of time and computational effort plays an essential role.

4.4 A universal model order reduction method

The main idea is to use one POD basis for the dimension reduction of the structure subjected to different earthquake excitations in order to save time and computational effort. As this consideration represents an expansion to the strategy presented in section 4.1 firstly a similar structure with the same nonlinear properties (i.e. friction elements) is chosen for the demonstration. Afterwards this strategy will be further demonstrated by the application of high dimensional building constructions with nonlinear material parameters.

The snapshot matrix \mathbf{X}_s is taken from the response of the structure to one transient excitation (i.e. Fukushima foreshock). The structure is time integrated over a short snapshot time period of the foreshock acceleration. The starting point does not have to be at the very beginning of the earthquake but somewhere in the middle (e.g. 50 seconds in the Fukushima foreshock record), where the main period of excitation is happening. The wavelet transformation of the excitation provides the frequency content dependent on time and supports to find the optimal starting time point of the snapshot time period. From the snapshot matrix \mathbf{X}_s the POD modes φ_i are calculated and the first l POD modes (l defines the chosen dimension in the POD subspace) assemble the transformation matrix $\Phi_{p,r} = [\mathbf{u}_1, \dots, \mathbf{u}_l]$, which transforms the full coordinate \mathbf{x} into the reduced coordinate $\mathbf{q}_{p,r}$ as presented in Eq. 80. The approach to obtain this universal transformation matrix is shown in Fig. 31.

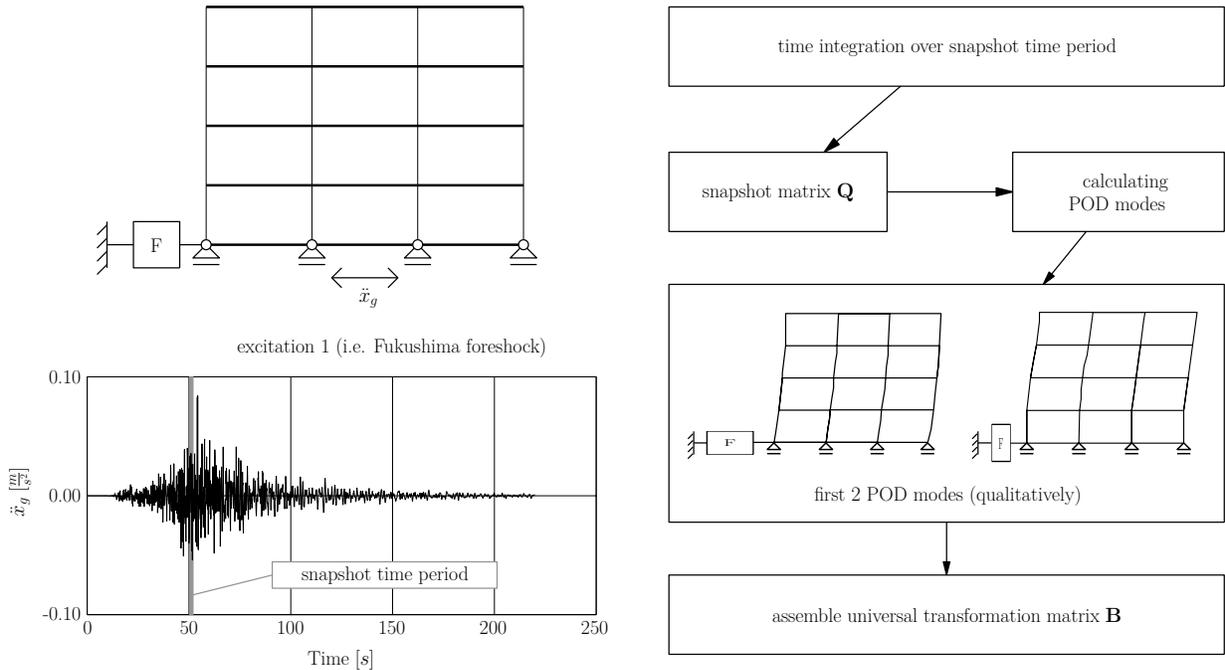


Figure 31: Approach for universal POD modes (transformation matrix)

The same structure is now excited by a second transient ground acceleration (i.e. Fukushima earthquake record). The set of equations of motion for the structure (Eq. (93)) is now transformed into the POD subspace using the universal transformation matrix $\Phi_{p,r}$ as shown in Figure 31 (no system identification through a snapshot matrix gained by integrating over a small time period of this earthquake is calculated). The ROM is now time integrated

over this transient excitation. Only a small number, l , of POD modes is often sufficient to approximate the full solution accurately. Taking more POD modes into account would not change the quality of the approximated solution noticeable but the critical time step would be scaled down and the calculation time would be enlarged. The reason is the property of optimality, which characterizes the POD method (4.1). Because of the smaller number of DOF, the critical time step increases and the calculation time decreases considerably.

Considering t_0 being the first and t_{end} the last time step, the solution matrix $[\mathbf{q}_{p,r}(t_0), \mathbf{q}_{p,r}(t_1), \dots, \mathbf{q}_{p,r}(t_{end})]$ in the POD subspace is obtained and the transformation (Eq. 80) leads to the response in the physical coordinate $[\mathbf{x}(t_0), \mathbf{x}(t_1), \dots, \mathbf{x}(t_{end})]$. Fig. 32 presents the calculation approach for this universal POD reduced system subjected to the Fukushima earthquake.

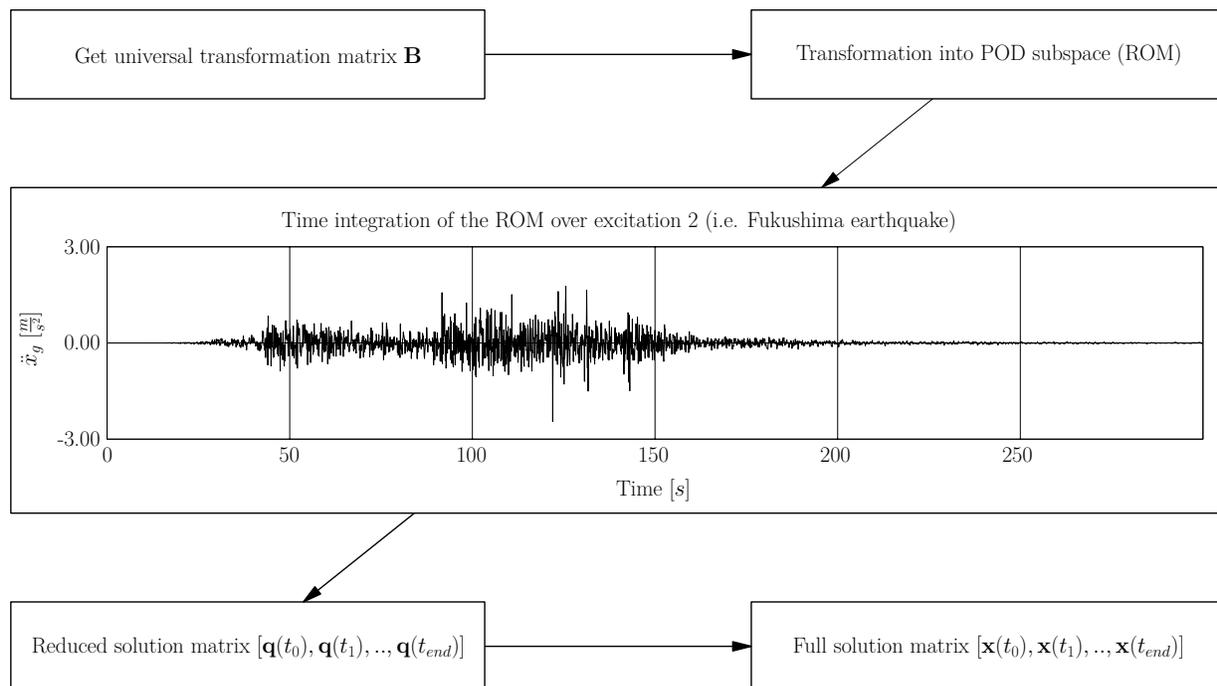


Figure 32: Approach for MOR with universal POD modes

In summary, the most time consuming factor is the calculation of the snapshot matrix \mathbf{X}_s because the full system must be integrated over the snapshot time period using a very small time step. But as mentioned, it is not necessary to calculate a snapshot matrix for each earthquake record. Once a snapshot matrix is determined the POD basis can be used as universal POD basis for the structure excited by earthquakes with "similar" properties (i.e. similar frequency range).

This procedure can be used for an estimation of structural damage. The foreshock of an earthquake is analyzed to obtain the POD modes, which can then be used to analyze this structure subjected to a stronger generated earthquake with similar properties and prevent possible damage in an acceptable expenditure of time. The advantage is that these modes carry information of real measured data in contrary to the POD investigation using only one generated transient excitation, where POD modes are calculated purely according to artificial values.

In alternative statistical methods (i.e. Monte Carlo method) this universal reduction method can be seen as an advantage as well, where the numerical effort plays an essential role. Using the Monte Carlo method, thousands transient excitations and their responses are analyzed, dependent on the desired probability of failure. The demand of calculation time can grow unacceptably by investigating high dimensional models. Using a "universal" ROM the whole calculation time can be decreased dramatically.

4.4.1 Dynamic structure - test object

The structural realization is similar to that presented in section 4.3.1. A two dimensional frame system is used as a test object for building constructions. The height of each storey is 4 meters. The width of the structure is 4×8 meters. The modulus of elasticity of the material is $3.57 \times 10^{10} \frac{N}{m^2}$, which should present a linearized approximation of the material parameter of reinforced concrete. The cross section of the columns is quadratic ($0.2 m \times 0.2 m$) and is identical for all columns in the structure. The cross section of the bars is also rectangular ($0.2 m \times 1.4 m$) and is also identical for all bars in the structure. The base of the structure is equipped with friction-based seismic isolation devices (friction isolators), which separate the structure from the ground and should avoid structural failure in case of an earthquake (analog to the nonlinear friction elements in section 4.1). A simplified structural model is presented in Figure 33.

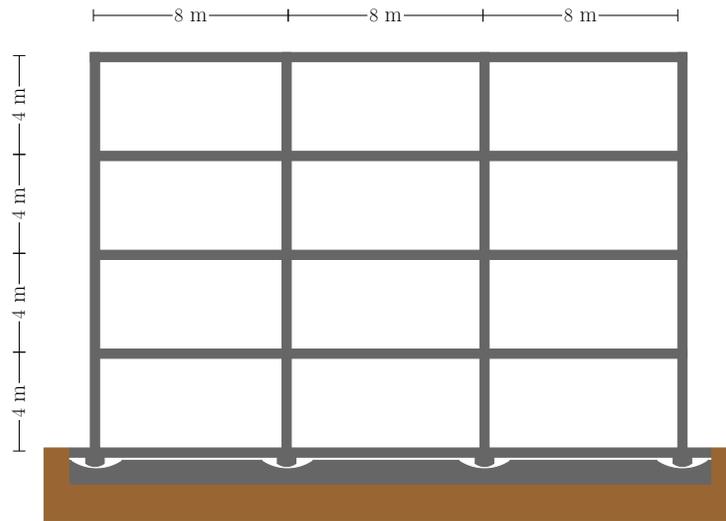


Figure 33: Building construction

The system is discretized by linear finite beam elements (column elements and bar elements). In the structural model all friction-based isolation devices are represented by one friction isolator on the left side of the structure. This friction-based seismic isolation device (friction isolator) discussed in section 4.3.1, is responsible for the nonlinear response of the structure. Subsequently, the horizontal displacement of the first floor, $h(t)$, and the horizontal displacement of the slide bearing, $h_g(t)$, is visualized. Important magnitude of the results is the relative horizontal displacement, h_{rel} , i.e. is the difference of $h(t)$ and $h_g(t)$ that is actually responsible for the internal forces and stresses.

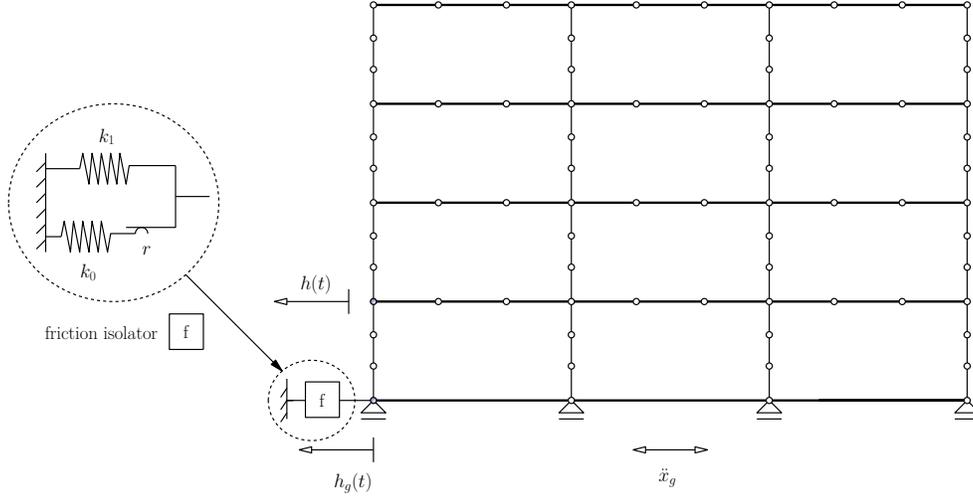


Figure 34: Left hand side: Finite element discretized structure, right hand side: friction isolator

4.4.2 Earthquake excitation

Two types of earthquake records are discussed: one type represents far-fault earthquake excitations, the other near-fault earthquake excitations.

Far-fault earthquake excitations Two ground acceleration data of the Fukushima earthquake in March 2011 are used. One is the record of the Fukushima foreshock (2011) and the second the record of the Fukushima main earthquake (2011). Both data sets have been recorded at the station *FKSH16* in Fukushima in Japan, which is located about 200 km away from the epicenter. Figure 35 gives information about the geographical situation of the Fukushima foreshock and the Fukushima main earthquake and the record station *FKSH16* (image data have been taken from Google Earth and subsequently adapted). Detailed information (date and time, lateral coordinate of the epicenter Φ_e , longitudinal coordinate of the epicenter λ_e , lateral coordinate of the station Φ_s , longitudinal coordinate of the station λ_s , distance to the epicenter d , magnitude M) is given in Table 3. Figure 36 presents the east-west acceleration data of the ground, \ddot{x}_g , of the Fukushima foreshock and the corresponding Morlet wavelet transformation. Fig. 37 presents the east-west acceleration data of the ground of the Fukushima main earthquake and the corresponding wavelet transformation. Record data are taken from Kyoshin-Network (2012).

The result of the wavelet transformation of the foreshock record presented in Fig. 36 shows that the maximum values of wavelet power spectrum appear between second 40 and 70. For the determination of the snapshot time period not only the intense and the frequency of an earthquake is important, which could be evaluated by application of the classic fourier transformation, but also the intensity and frequency content as a function of time. Thus, it is recommendable to chose the snapshot time period within this time interval, i.e. $40 \leq t_s \leq 80$ [s]. In the main earthquake period, i.e. second 40 to 80 a maximum value of the wavelet power spectrum of about $0.2 [m^2/s^3]$ is observed. The frequency content of the Fukushima main record (cf. Fig. 37) is similar to that of the Fukushima foreshock record, however the maximum value of the wavelet power spectrum is about $6 [m^2/s^3]$, i.e.

earthquake	date/time	Φ_e	λ_e	Φ_s	λ_s	d	M
foreshock	2011.03.09 14:45:00	38.328°	143.278°	37.764°	140.377°	262 km	7.3
main event	2011.03.11 11:45:00	38.103°	142.860°	37.764°	140.377°	219 km	9.0

Table 3: Information data of the Fukushima foreshock and the Fukushima main earthquake

about 30 times larger.

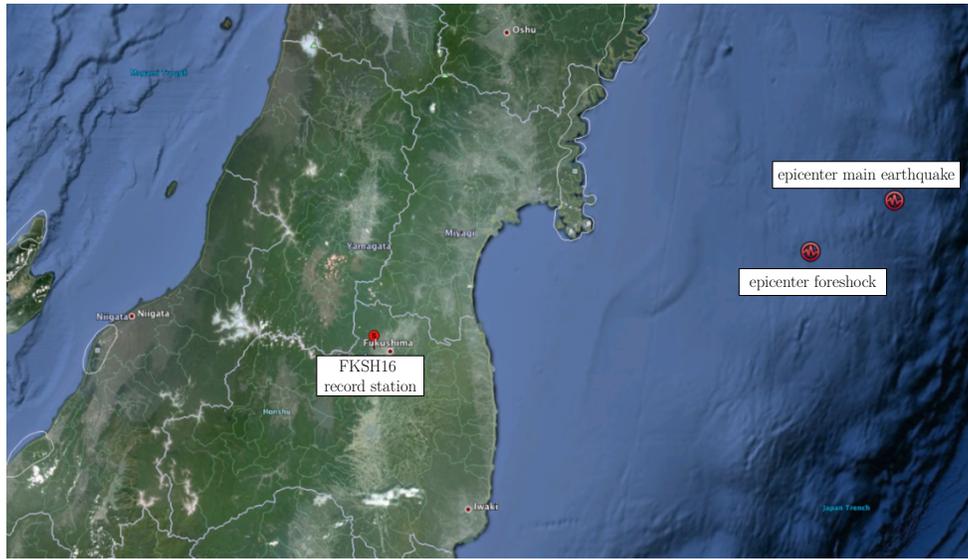


Figure 35: Geographical visualization of the epicenters and the record station FKSH16; note that this is a photo using *GoogleEarth* (<https://earth.google.com>) to illustrate the geographical distance

Near-fault earthquake excitations Two near-fault ground accelerations recorded in California are used. One is the record of the Imperial Valley earthquake (1979), the other the record of the Northridge earthquake (1994). Both acceleration data are recorded less than eight kilometers from each epicenter. Detailed information (date and time, lateral coordinate of the epicenter Φ_e , longitudinal coordinate of the epicenter λ_e , record station, distance to the epicenter d , magnitude M) about the record data is given in Table 4. Figure 45 presents the normal-fault acceleration components of the ground, \ddot{x}_g , of the Imperial Valley earthquake and its corresponding Morlet wavelet transformation and Figure 46 presents the normal-fault acceleration components of the ground of the Northridge earthquake and its corresponding wavelet transformation. Earthquake data are taken from PEER (2012) from the University of California.

The results of the wavelet transformations in Fig. 45 and 46 show that both earthquakes have about the same maximum value of the wavelet power spectrum, i.e. $30 [m^2/s^3]$. Although the maximum peak of the Northridge spectrum is at a higher frequency than the peak of the Imperial Valley spectrum. However they have similar properties concerning frequency

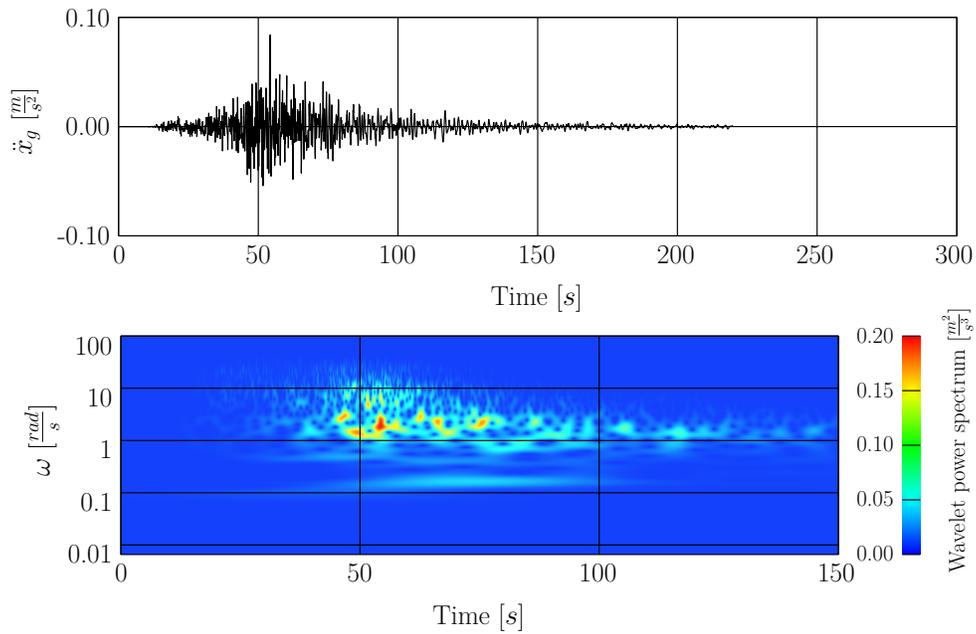


Figure 36: Top: East-west acceleration record of the Fukushima foreshock, bottom: Morlet wavelet transformation of the Fukushima foreshock accelerogram, note that time scales are different

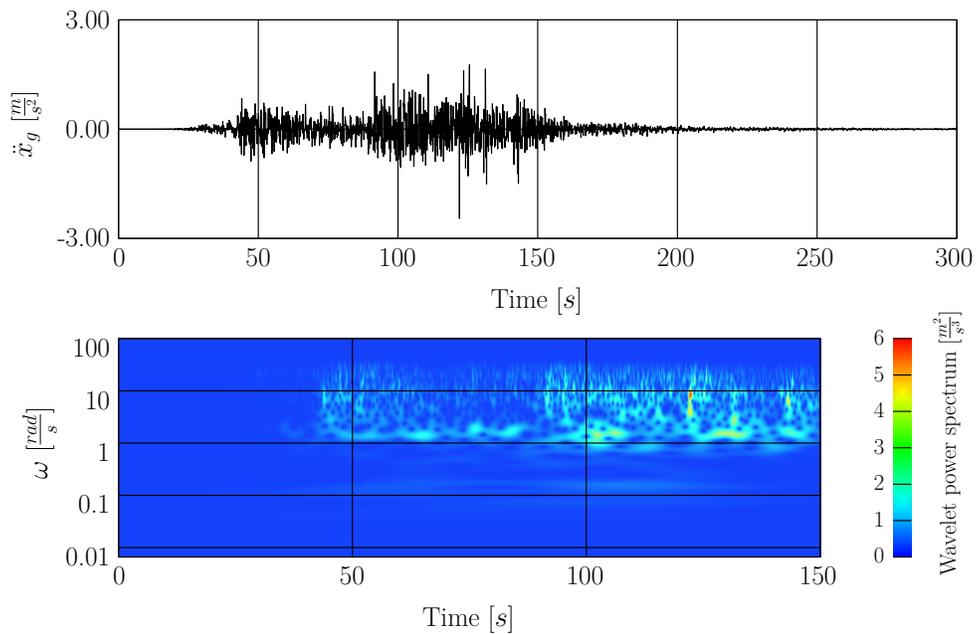


Figure 37: Top: East-west acceleration record of the Fukushima main earthquake, bottom: Morlet wavelet transformation of the Fukushima main earthquake accelerogram, note that time scales are different

earthquake	date/time	Φ_e	λ_e	record station	d	M
Imperial Valley	1979.10.15 23:16:00	32.617°	-115.317°	El Centro Array 6	1.2 km	6.5
Northridge	1994.01.17 12:30:00	34.213°	-118.537°	Rinaldi Receiving Station FF	7.5 km	6.7

Table 4: Information data of the Imperial Valley earthquake and the Northridge earthquake

content. Based on the results of the Imperial Valley spectrum it is advisable to chose the snapshot time period somewhere in the time interval $3 \leq t_s \leq 10$ [s] in order to obtain the optimal POD approximated result.

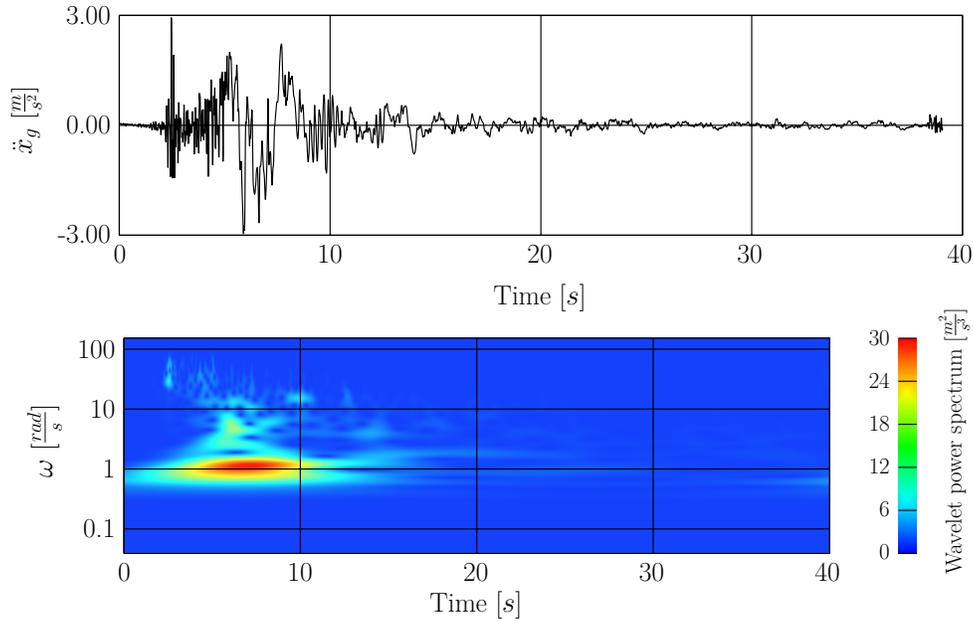


Figure 38: Top: Imperial Valley acceleration, bottom: Morlet wavelet transformation of the Imperial Valley acceleration

4.4.3 Numerical demonstration

Far-fault earthquake excitation example The nonlinear set of equations of motion, Eq. (93), obtained by the finite element structure presented in Fig. 34, is integrated by the central difference method in the snapshot time period ($50 \leq t_{snapshot} \leq 50.8$ [s]) of the Fukushima foreshock. With this information the universal POD modes are calculated. Only four modes are used for the MOR of the system.

The reduced equation of motion in the POD subspace, Eq. (90), is now integrated over the whole time period with a chosen time step of $\Delta t = 0.01$ s, which is the resolution of the earthquake record. The full system is calculated as well to assess the quality of the reduced solution. Displacements $h(t)$, $h_g(t)$ and $h_{rel}(t)$ of the full solution and the POD reduced solution are shown in Fig. 40.

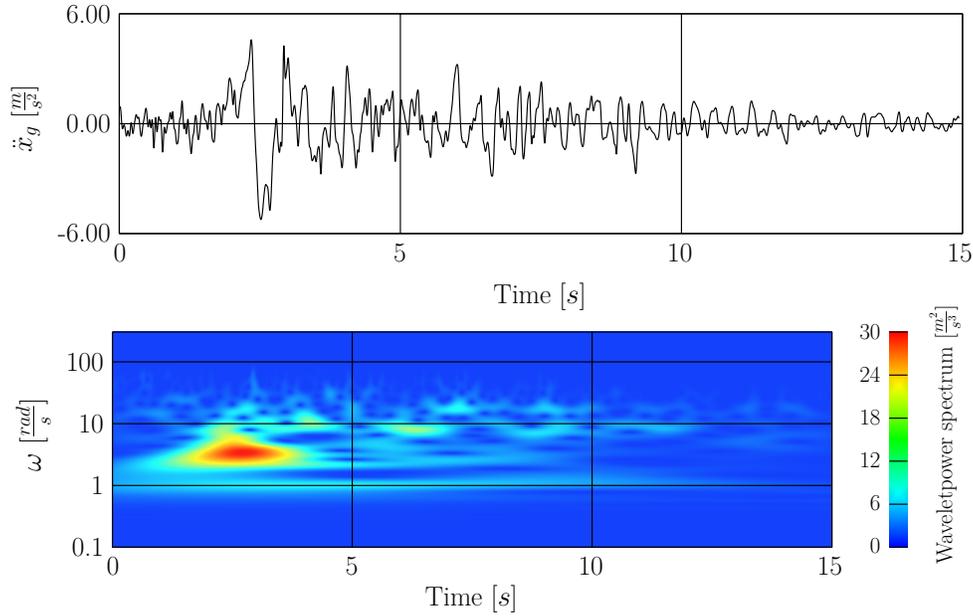


Figure 39: Top: Northridge acceleration, bottom: Morlet wavelet transformation of the Northridge acceleration

An accurate approximation of the response due to the full computation can be observed in the time histories of the horizontal displacements $h(t)$, support displacements $h_g(t)$ and relative displacements $h_{rel}(t)$. This special example follows the same MOR procedure as discussed in section 4.3, only the snapshot time period is not chosen at the very beginning of the record but the 80 snapshots are taken at a time section within the record (i.e. $(50 \leq t_{snapshot} \leq 50.8 \text{ [s]})$ in an equidistant time interval of 0.01 seconds. If the snapshots are calculated at the very beginning of the record in this special example no POD basis can be generated with sufficient information about linear and nonlinear behavior. However, the results presented in Fig. 40 confirm the results of section 4.3, where the POD method is applied to the El Centro record.

The utilized four universal POD modes are displayed in Figure 41. They look similar to normal modes, but actually they are "modules" of the nonlinear system carrying information of the structure and the excitation.

The structure excited by the Fukushima main record is transformed into the POD subspace by the universal transformation matrix, which contains the four universal POD modes. Time integration by the central difference method leads to the approximated POD solution. The full solution of the system is also calculated.

Displacements $h(t)$, $h_g(t)$ and $h_{rel}(t)$ of the response due to the full computation and the POD reduced computation are presented in Fig. 42. The nonlinear effect of the friction elements at the support is not as accurate as at the results presented in Fig. 40 or 22. Obviously the nonlinear sliding effect of the friction element cannot be captured in the snapshot time period calculated integrating over a small time section of the Fukushima foreshock record. However, the maximum relative displacements $h_{del}(t)$ are overestimated of about 10 percent, which can still be seen as a useful approximation for civil engineering problems.

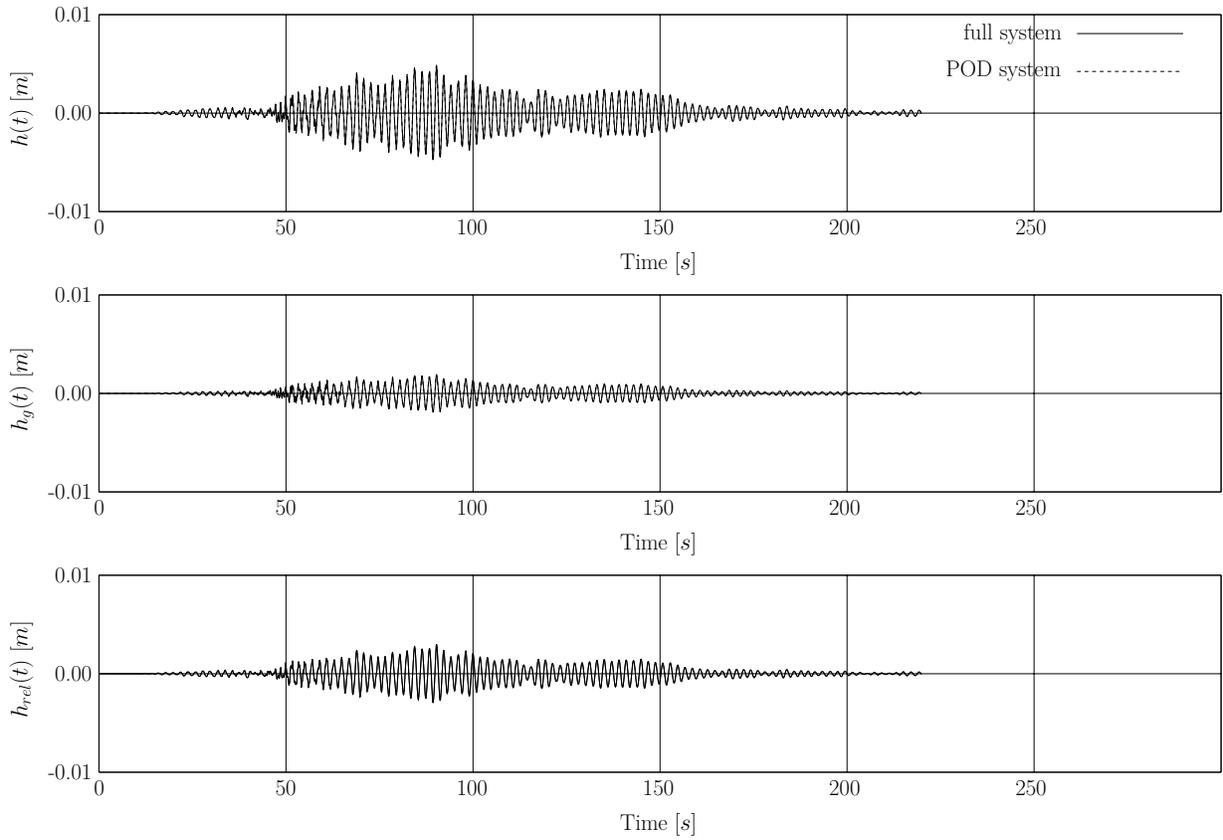


Figure 40: Displacements of the Fukushima foreshock full model and POD reduced model (equivalent to 4.3)

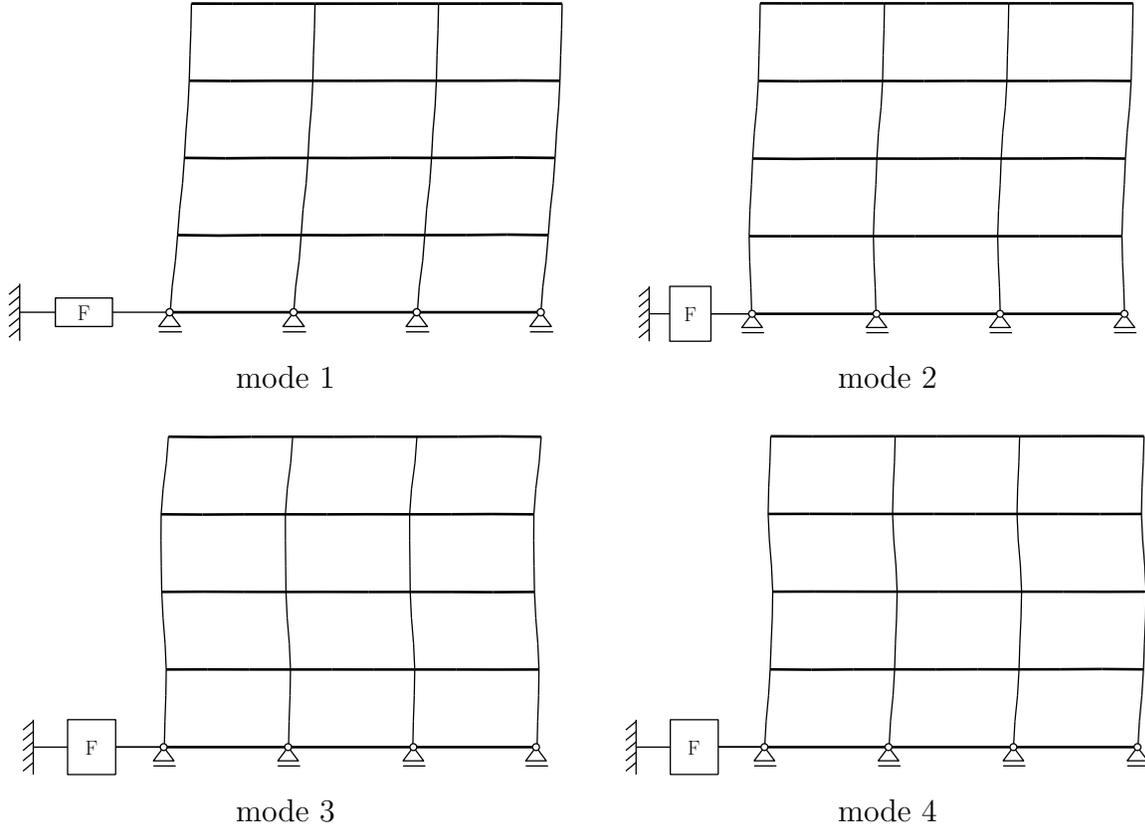


Figure 41: Universal POD modes

In order to assess the result of the full and reduced system, error estimations are done. The mean error $E_{mean}(t)$ for both earthquake records is calculated according to Eq. (91) and presented in Fig. 43 and 44.

Obviously the universal POD response of the structure excited by the Fukushima main earthquake approximates the full system very accurately. As expected the universal POD error in Fig. 43 is larger than the POD error in Fig. 44, which can be explained that the POD modes, calculated with the frequency information of the snapshot time period in the foreshock, do not exactly contain the same frequencies as in the Fukushima main earthquake. Consequently a phase shift of the universal POD solution and the full solution appears. Additionally there is a huge difference of intensity concerning the Fukushima foreshock and the Fukushima main earthquake. Therefore, the nonlinear sliding effect of the friction elements cannot be described in the snapshot time period in the Foreshock record. Consequently, in the application of universal MOR it is also advantageous that not only earthquakes with same frequency content but also with similar energy content are chosen.

Nevertheless the approximation of the Fukushima main earthquake is promising by using the universal POD method although a huge difference in magnitude and energy content of these two records can be observed. This can be seen as an example of robustness of this universal MOR procedure.

Universal POD reduction should only be implemented for similar transient excitations (earthquakes), i.e. the dominant frequencies of the excitation should be in a similar range and the main frequencies should occur in the snapshot time period. Additionally the intensity of the records should be in a similar range. Otherwise an accurately approximated

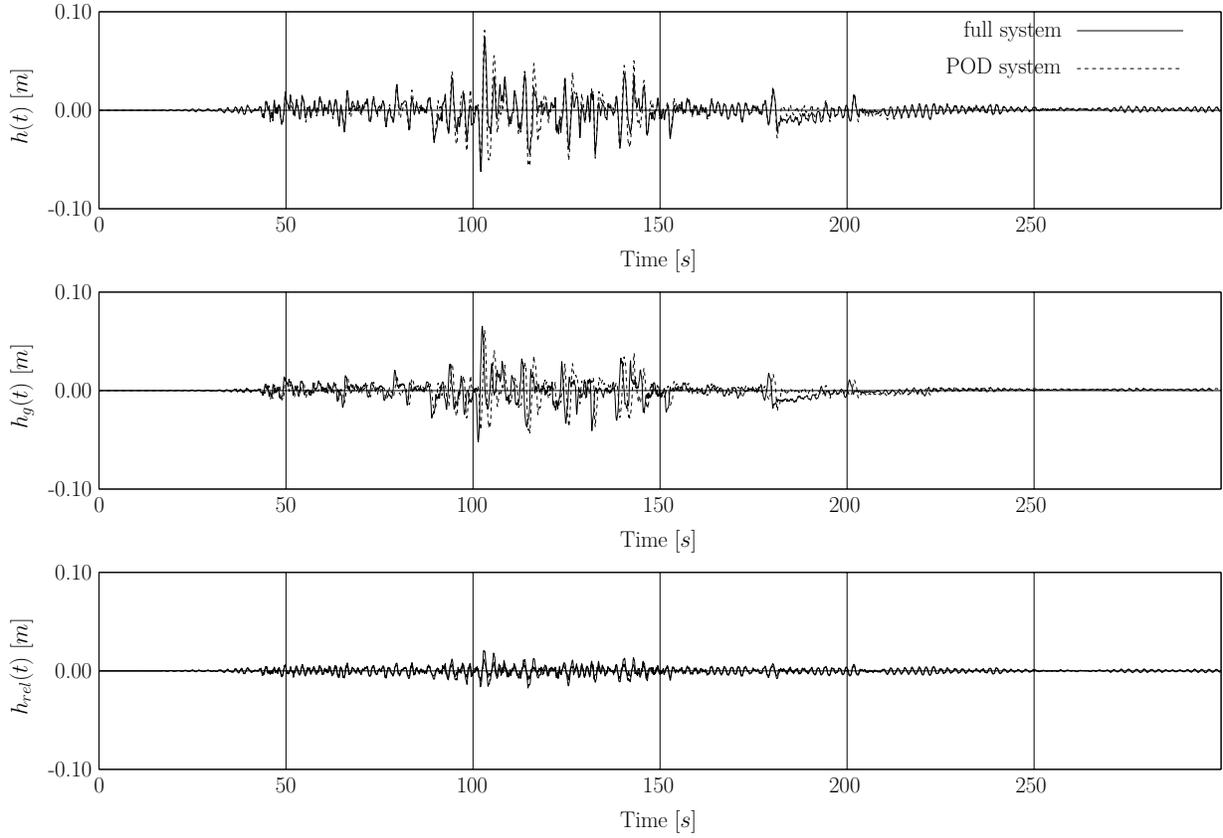


Figure 42: Displacements of the Fukushima earthquake full model and POD reduced model

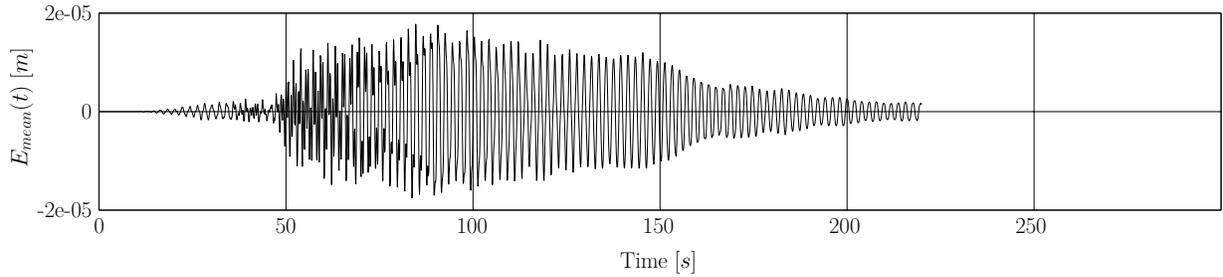


Figure 43: Mean error of horizontal displacements - Fukushima foreshock

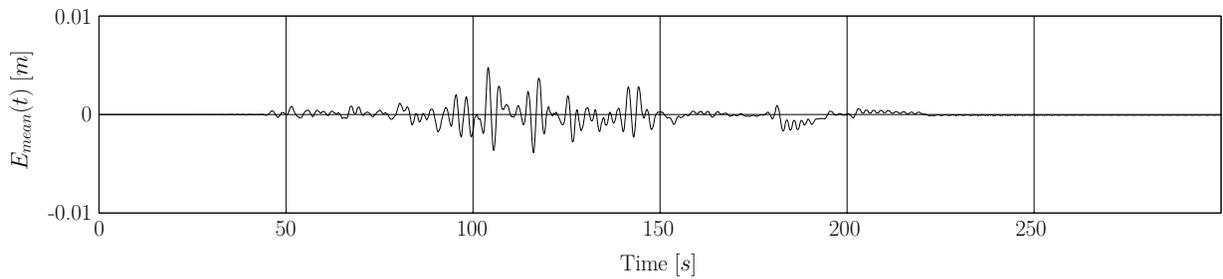


Figure 44: Mean error of horizontal displacements - Fukushima earthquake

solution of the reduced system cannot be guaranteed.

Near-fault earthquake excitation example The calculation approach is analogous to the one described in section 4.4.3. The POD modes are obtained by integration over the snapshot time period of the Imperial Valley earthquake. The starting point of the snapshot time period is second 7, 80 snapshots are calculated and four POD modes are utilized in the universal transformation matrix. Fig. 45 presents the full and the reduced solutions, $h(t)$, $h_g(t)$ and h_{rel} of the Imperial valley earthquake, see also section 4.1. Fig. 46 presents the full and the universal reduced solutions, $h(t)$, $h_g(t)$ and h_{rel} of the Northridge record. Error estimations are presented in Fig. 47 and 48.

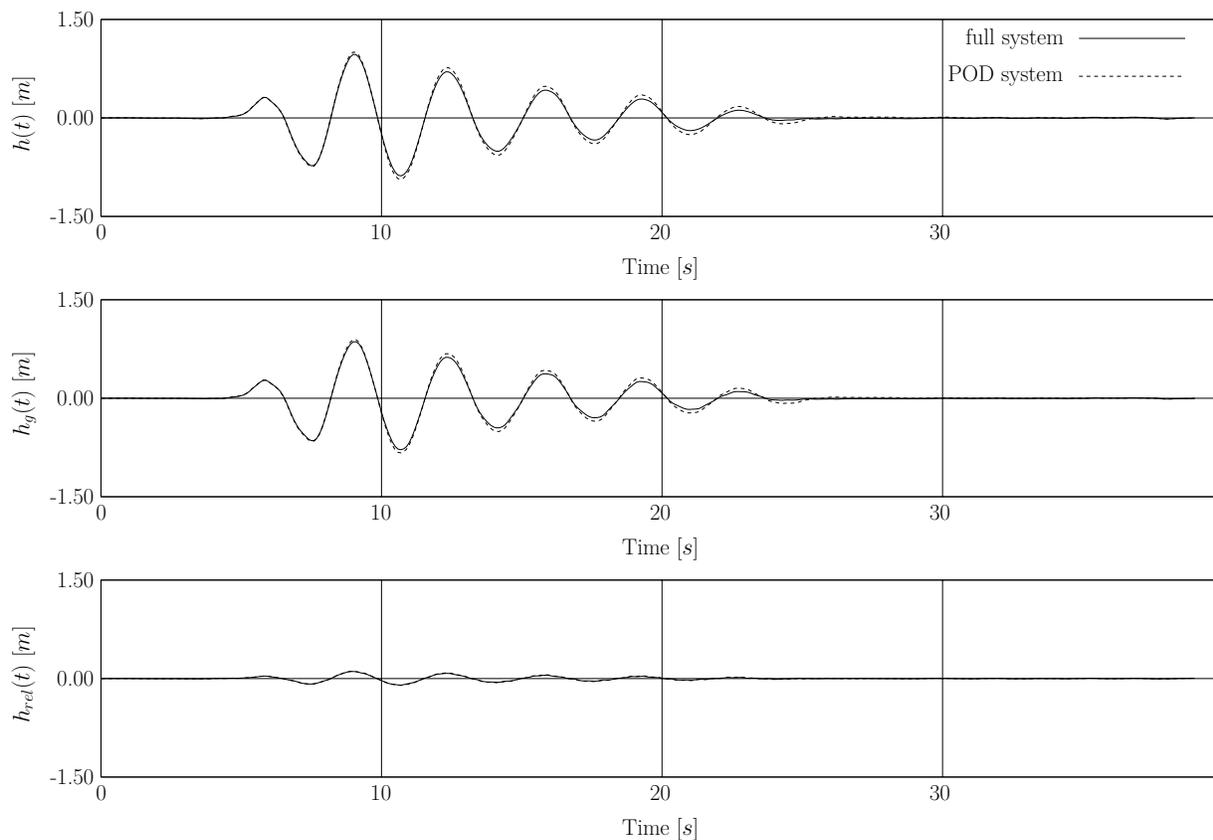


Figure 45: Displacements of the Imperial Valley earthquake full model and POD reduced model (equivalent to 4.3)

4.4.4 Numerical efficiency

The number of DOFs (224) of the structure is rather moderate, but the dynamic system is seen as test object for the universal POD method. Nevertheless the full calculations by application of the central difference algorithm are considerably time-consuming. The computational time depends, considering the rather moderate number of DOFs, on the number of iterations but hardly on the matrix operations in one loop. The chosen time step, which is actually smaller than the critical time step, is 10^{-4} seconds for each system

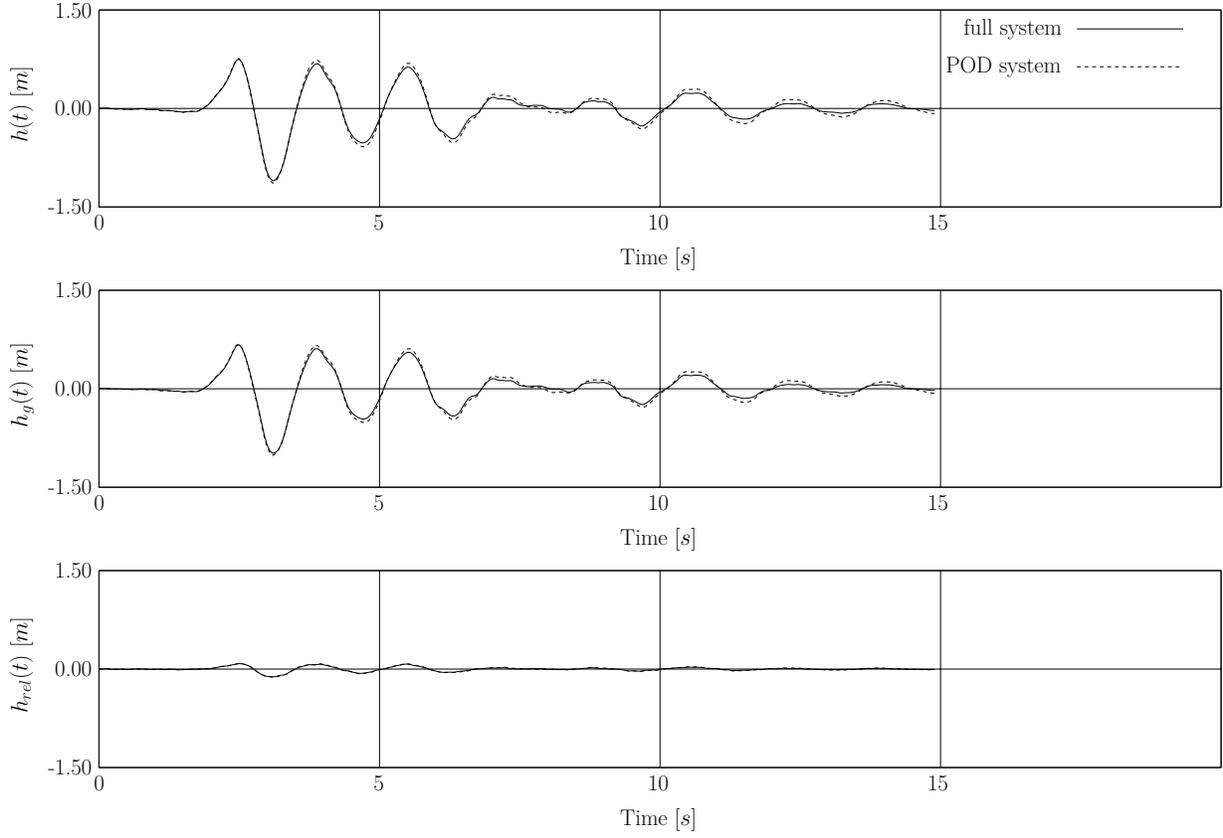


Figure 46: Displacements of the Northridge earthquake full model and POD reduced model

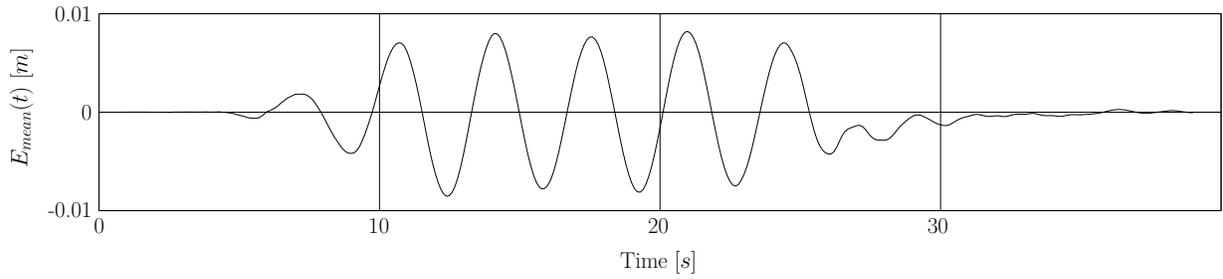


Figure 47: Mean error of horizontal displacements - Imperial Valley earthquake

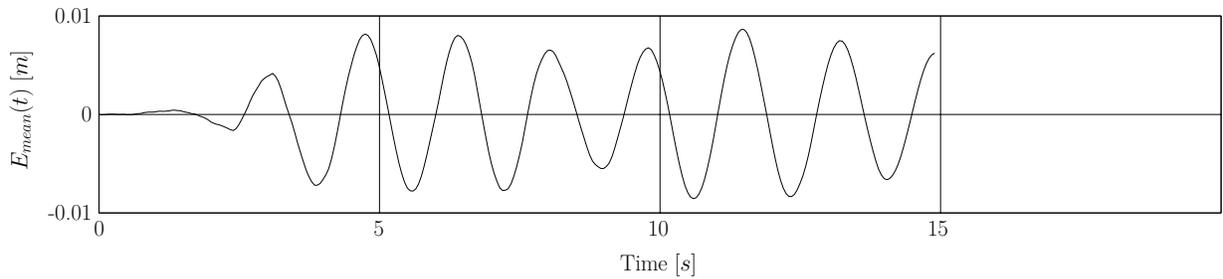


Figure 48: Mean error of horizontal displacements - Northridge earthquake

earthquake	full calculation	POD calculation	universal POD calculation
Fukushima foreshock	1661	19 + 6	-
Fukushima earthquake	2265	-	28
Imperial Valley	302	3 + 6	-
Northridge	231	-	2

Table 5: Calculation time [s]: Full calculation (central differences), POD calculation, universal POD calculation (calculation time of the snapshot matrix: 6 seconds)

in this section. The chosen time step in the POD reduced subspace is always equal to the resolution of the earthquake record, which is the maximum possible value. The condition for using the earthquake resolution as time step is that the critical time step of the POD reduced system has to be larger than the time resolution of the earthquake records, which is obviously always the case in this example. As a consequence the time of the POD calculation depends on the sampling rate of the earthquake record and in addition on the assembling process of the snapshot matrix. Table 5 presents the calculation times of the full systems, the POD reduced systems (equivalent to section 4.1) and the universal POD reduced systems. Furthermore, universal POD reduced calculations are even more optimized in the sense of computational effort because the recalculation of the snapshot matrix is not necessary.

4.4.5 Conclusion

Based on section 4.1 a new model order reduction strategy by the proper orthogonal decomposition method is presented. By integrating over a small part of the response of a nonlinear structure subjected to one earthquake the transformation matrix for model order reduction is assembled and the structure, excited by another earthquake, is transformed into the reduced space and subsequently integrated over the whole time period. Numerical examples are presented not only on far-fault but also on near-fault excitations. The far-fault excitation pair is the Fukushima foreshock earthquake record and the Fukushima main earthquake record. The numeric demonstration shows that this pair of earthquakes is not that adequate for this MOR strategy in contrary to other example pairs. Although they show similar properties concerning frequency content they are completely different concerning intensity, i.e. a scaling factor of about 30 is observed. The nonlinear effect cannot be captured in the snapshot time period integrating over a small selected time period of the foreshock record since the excitation is too weak to induce sliding of the friction elements. However, the full solution of the Fukushima main earthquake can be approximated with an error of about 10 % of the maximum displacement. This is still a useful approximation concerning civil engineering problems, which demonstrates the robustness of this method, i.e. the strategy does not completely fail if there are not optimal conditions concerning the input parameters. The near-fault excitation pair is the Imperial Valley earthquake record and the Northridge earthquake record. The wavelet transformations show similar properties concerning frequency content and maximum energy. The wavelet power spectrum shows a clearly defined time window for the calculation of the snapshots. The approximation of the imperial valley excited structure, analogous to the "standard" POD strategy discussed in section 4.1, as well as approximation of the Northridge excited structure, i.e. the example for

the new "universal" POD strategy, show accurate results. The mean error of both examples dependent on time are comparable in the magnitude and insignificantly small, which can be compared to the POD error of the El Centro example in section 4.1, i.e. "standard" POD strategy.

Furthermore it can be expected that this MOR method works not only for systems with many more DOFs but also for more complex structures. Because of the occurrence of frequencies that are not captured in the snapshots a phase shifting of the response can be observed, but the error can be neglected for problems concerning civil engineering. It is obvious that the calculation of a POD reduced system is sufficient to predict the response of a structure to an earthquake excitation from the engineering point of view. On the one hand this method should make it possible to draw conclusions very fast about the resistance of a structure to an earthquake excitation because of the information of an earthquake with similar properties. On the other hand this method provides the basis for the realization of high dimensional Monte Carlo investigations regarding structural safety.

4.5 Application of the universal model order reduction strategy to structures with elastoplastic material properties

In section 4.1 the new MOR strategy was presented and tested on linear and nonlinear structures. In section 4.4 this MOR strategy was expanded to a broader application area using one transformation matrix for different transient excitations. Strategies were applied to relatively simple and uniform linear and nonlinear planar models (test objects) so far. Nonlinearities were realized by friction elements in order to model friction based seismic isolation devices. Although nonlinear reactions can be observed in the response histories, nevertheless the structure, which rests on these friction bearings, behaves linear.

This section introduces the application of the "universal" MOR strategy on a more complex nonlinear three dimensional building structure subjected to six different earthquake records. It should be mentioned that the records seem to have rather near-field characteristics.

4.5.1 Structural model

Elastoplastic material in the axial stresses and strain curve of the beam elements is implemented as presented in Fig. 49. E_1 describes Young's modulus of the linear material and E_2 is the post-yield Young's modulus. f_{yd} defines the yield strength. For detailed information about the modeling of elastoplastic material behavior the reader is referred to Bathe (1995).

The three-dimensional building construction is realized by a finite element discretized mesh. Floor sections are modeled by shell/plate triangle elements and the columns are realized by beam elements including the nonlinear material parameters presented in Fig. 49. The shape of the building is motivated by the appearance of hotels and casinos in Las Vegas. A three-dimensional visualization of the building is given in Fig. 50. A plan of the structural design of one story of the building shows the left subplot of Fig. 51 on the left side and the corresponding FE discretized mesh of one storey is displayed on the right subplot. In the implementation the number of stories is variable. The chosen number is seven. Every column per storey is discretized by three nonlinear beam elements. The columns are designed by a quadratic cross section ($0.25 \times 0.25 [m]$) and for the floors a

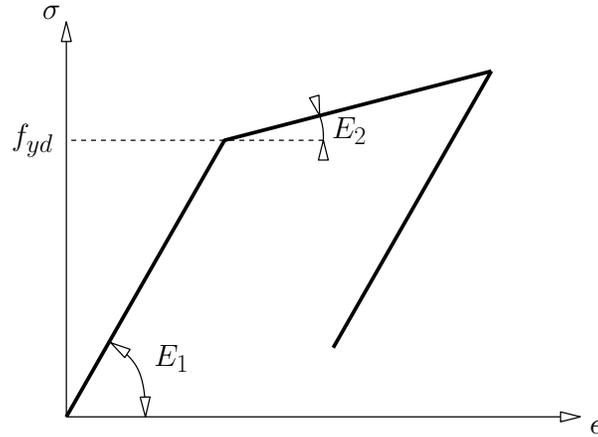


Figure 49: Stress strain behavior of a yielding material with strain hardening

thickness of 0.4 [m] is chosen. The floors are designed on purpose to ensure considerably large plastic deformations in order to test the nonlinear MOR strategy.

The structure is subjected to ground acceleration only in x -direction. The ground acceleration in y -direction is set as zero for all calculations in this section and section 4.6.

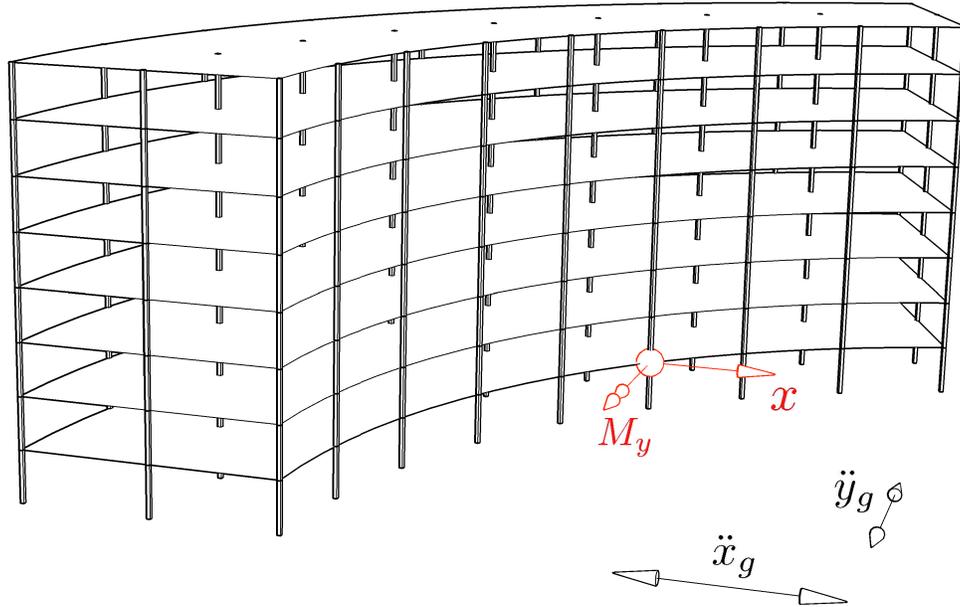
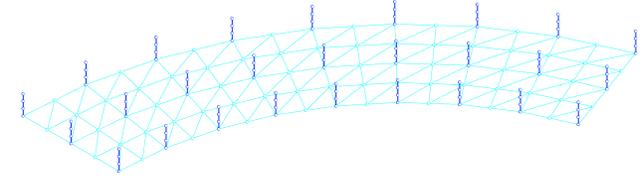
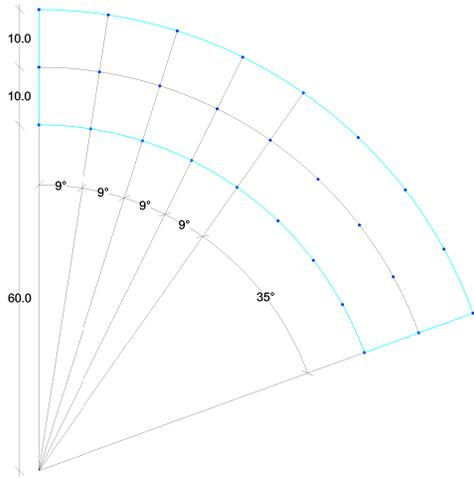


Figure 50: 3-D visualization of the building and direction of the earthquake excitation

The natural mode shapes of the linearized structure, i.e. if the axial stress component of the beam elements does not exceed f_{yd} , are presented in Fig. 52.

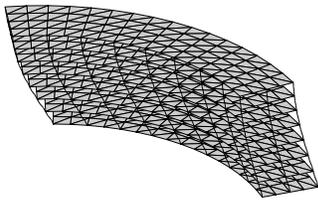
Upfront the structure is excited by the harmonic ground acceleration $\ddot{x}_g = x_0 \sin \nu t$ ($x_0 = 5$ [m/s²], $\nu = 3$ [rad/s]) in order to test the nonlinear effect of the system to a rather simple excitation, where the behavior of the response function can be pre-estimated qualitatively. The system response is derived by the application of two methods the central



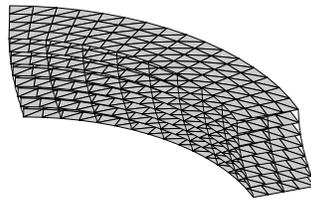
FE mesh of one storey, shell / plate elements in magenta and nonlinear column elements in blue

Construction plan of one storey $[m]$,
 columns with quadratic cross section
 $0.25 \times 0.25 [m]$, floor dimension $d = 0.4 [m]$

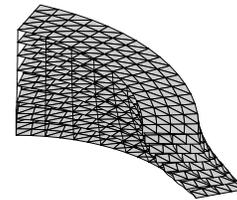
Figure 51: Left side: construction plan of one storey; right side: FE discretized mesh of one storey



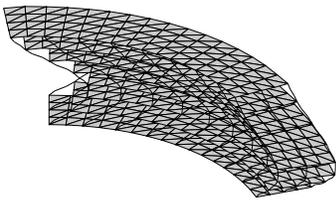
Natural mode shape 1,
 $\omega_1 = 2.98 [rad/s]$,
 $f_1 = 0.47 [1/s]$



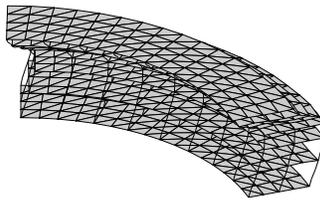
Natural mode shape 2,
 $\omega_1 = 3.02 [rad/s]$,
 $f_1 = 0.48 [1/s]$



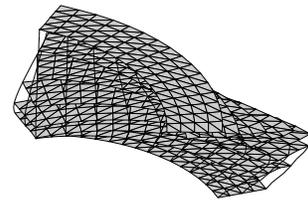
Natural mode shape 3,
 $\omega_1 = 3.27 [rad/s]$,
 $f_1 = 0.52 [1/s]$



Natural mode shape 4,
 $\omega_1 = 8.98 [rad/s]$,
 $f_1 = 1.42 [1/s]$



Natural mode shape 5,
 $\omega_1 = 9.79 [rad/s]$,
 $f_1 = 1.43 [1/s]$



Natural mode shape 6,
 $\omega_1 = 14.63 [rad/s]$,
 $f_1 = 1.56 [1/s]$

Figure 52: First 6 natural mode shapes of the linearized structure with the corresponding eigen angular frequencies and eigen frequencies

difference approximation and the Newmark method. Fig. 53 shows the harmonic excitation and Fig. 54 the corresponding response in the time domain, i.e. the horizontal displacement of the red marked node shown in Fig. 50 and the hysteresis, i.e. the moment M_y as function of the horizontal displacement x in order to investigate the level of nonlinear behavior of the structure.

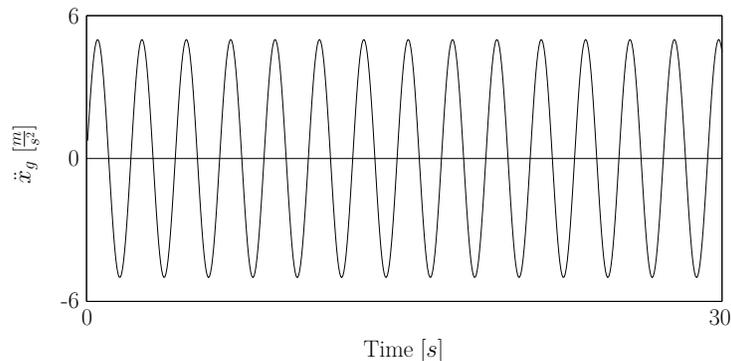


Figure 53: Harmonic ground acceleration

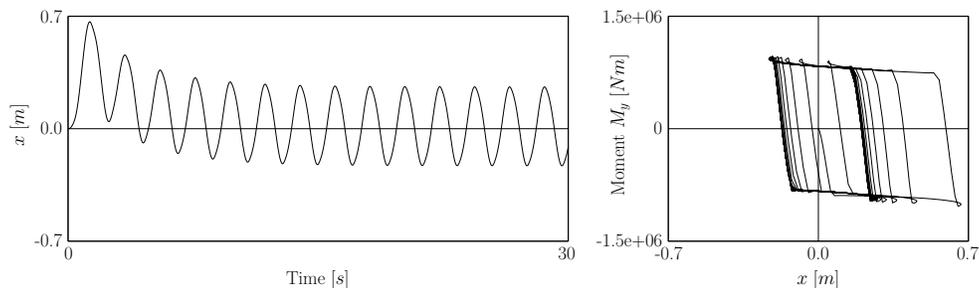


Figure 54: Left subplot: relative storey displacement x of the output node, right subplot: hysteresis - displacement x and moment M_y

The response is calculated by the Newmark method and verified by the Central difference algorithm. It can be seen that for a harmonic excitation with a magnitude of about $5 \left[\frac{m}{s^2} \right]$ considerably large nonlinear deformations occur. In this context it makes sense to move on to the next step and investigate the universal MOR strategy with transient excitations.

4.5.2 Numerical demonstration

Six different earthquake excitations are used to demonstrate the universal MOR strategy. The excitation set includes the Bam earthquake (2003) in Iran and the following five representative events in California: Northridge Rinaldi (1994), Imperial Valley (1979), Landers (1992), Loma Prieta (1989), North Palm Springs (1986). Table [6] presents a list of the events taken from PEER (2012) from the University of California.

The response to the the Northridge Rinaldi record, is used to assemble the snapshot matrix. 300 snapshots are taken in equidistant time periods spread over the whole history of the earthquake response. The truncation of the POD transformation matrix (POD modes)

Event	time	location	Δt	T	d	M	PGA
Bam	2003	Iran	0.05		-	6.6	7.16
Northridge Rinaldi	1994	California / Newhall	0.005	14.945	6.7	6.7	5.23
Imperial Valley	1979	California / Huston Road	0.01	39.38	10	6.5	4.79
Landers	1992	California / Barstow	0.02	79.98	36	7.3	4.13
Loma Prieta	1989	California / Gilroy	0.02	39.98	12	7.0	9.51
North Palm Springs	1986	California / Palm Springs	0.02	59.98	6.7	6.0	9.99

Table 6: Earthquake excitation list; Δt [s] resolution of the record data, T [s] duration of the record, d [km] distance from epicenter, M moment magnitude, PGA [m/s^2] peak ground acceleration

is chosen that 99 % of the total energy of the snapshots is captured. The number of modes is therefore 17. The first four modes are shown in Fig. 55.

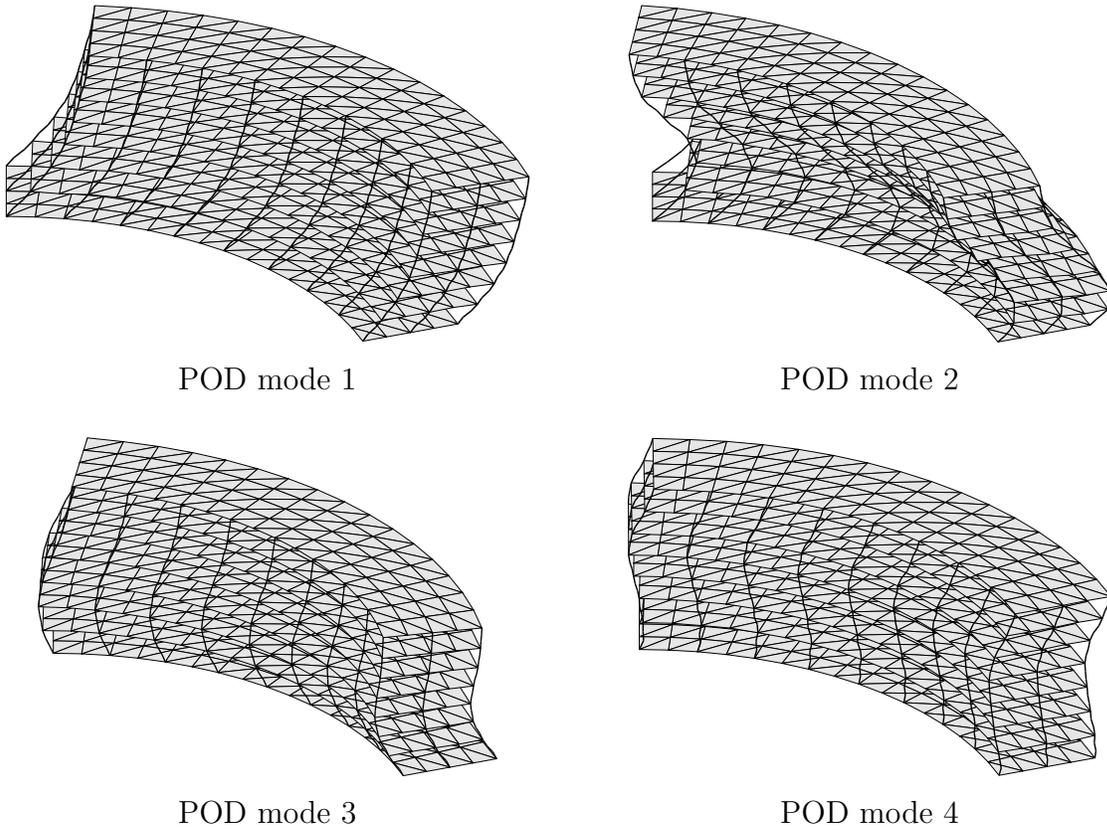


Figure 55: Universal POD modes

The modes are similar to the natural modes of the linearized system. However, they carry nonlinear reactions of the system to the earthquake excitation. Otherwise, it would not be possible to approximate plastic deformations as presented in this example.

In the following the ground excitation records are presented together with the response functions of the characteristic output node in Fig. 50, i.e. the horizontal response in x direction and the hysteresis realized by the bending moment M_y dependent on the horizontal displacement x . The responses are presented in Fig. 56, 57, 58, 59, 60 and 61.

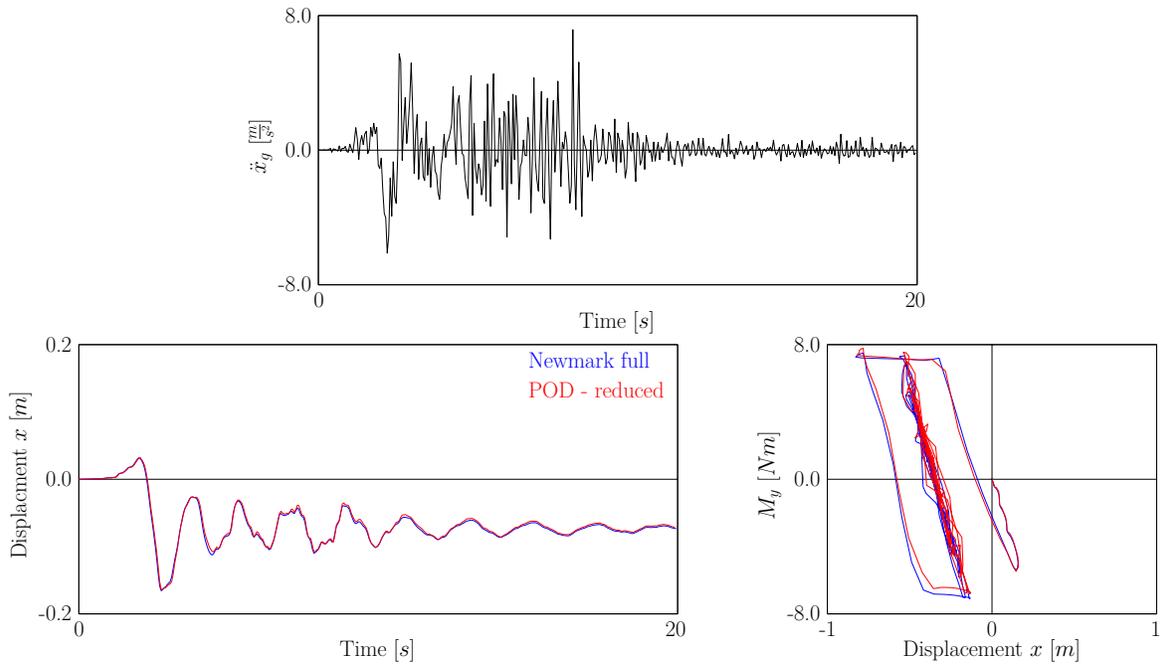


Figure 56: Response to the Bam earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - displacement x and moment M_y

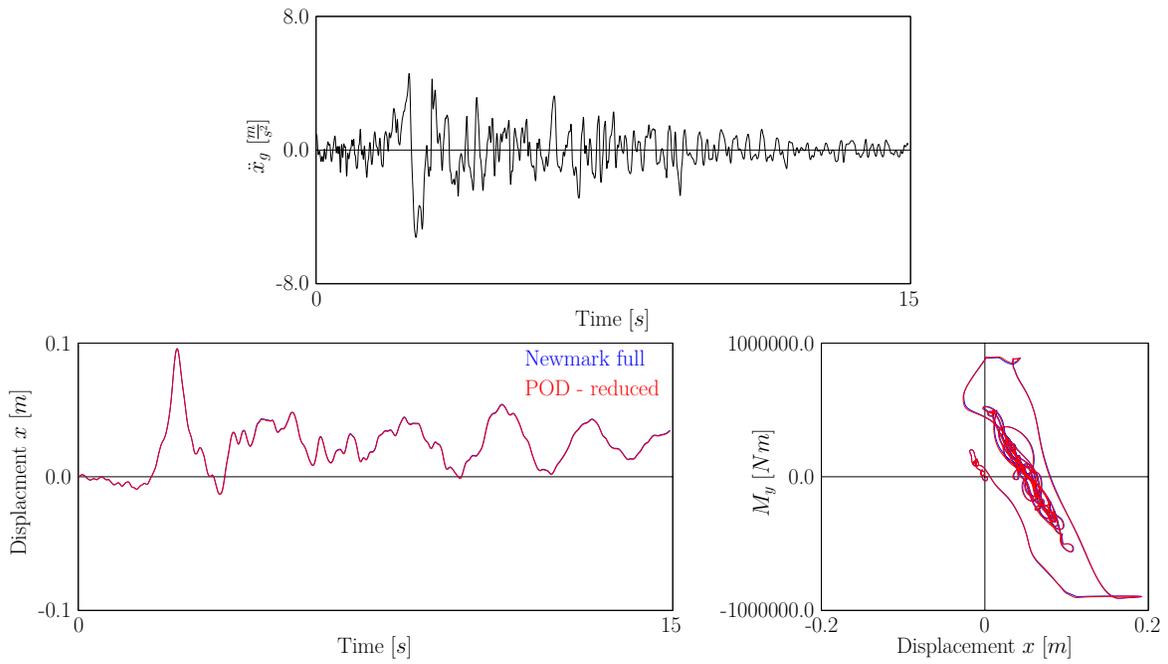


Figure 57: Response to the Northridge earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y

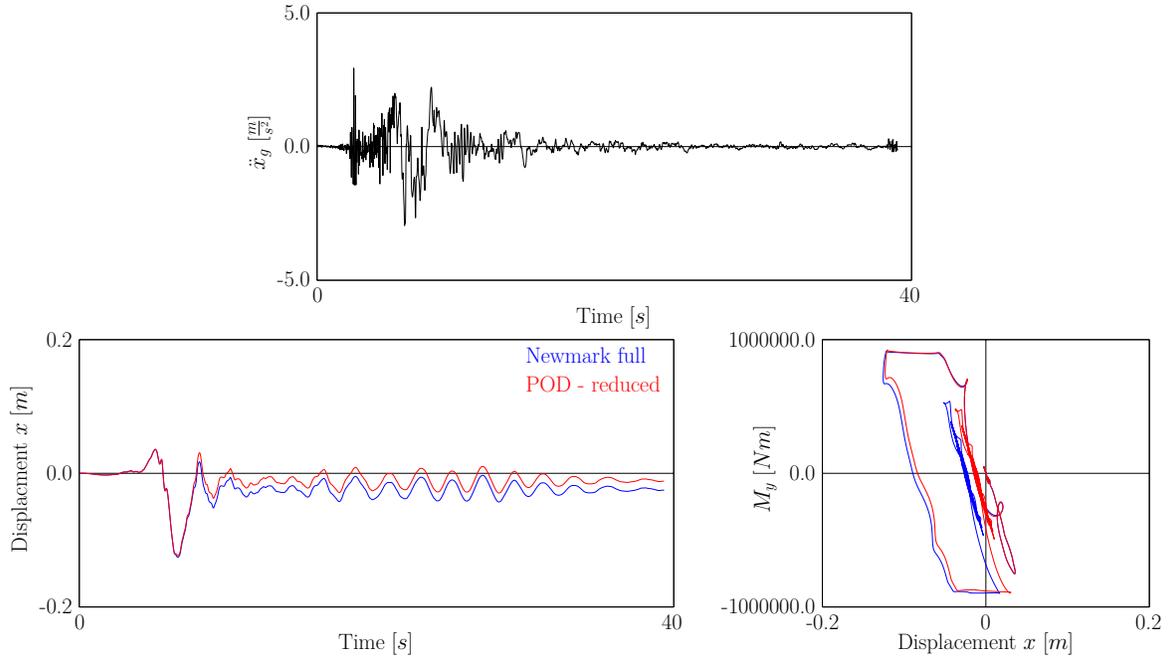


Figure 58: Response to the Imperial Valley earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y

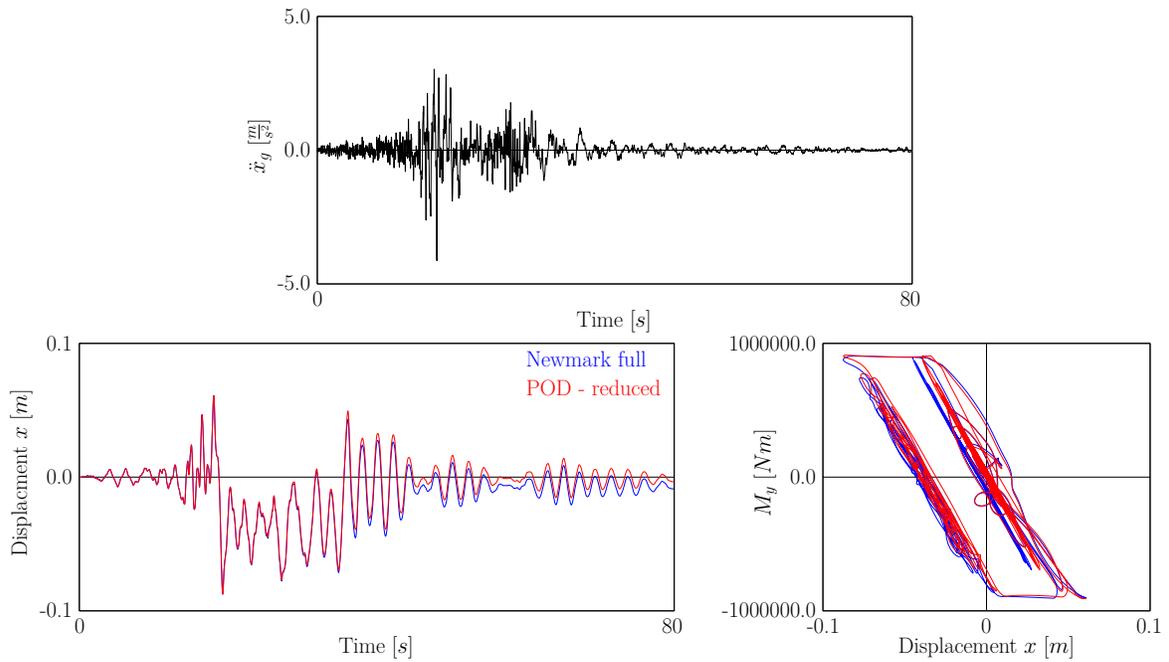


Figure 59: Response to the Landers earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y

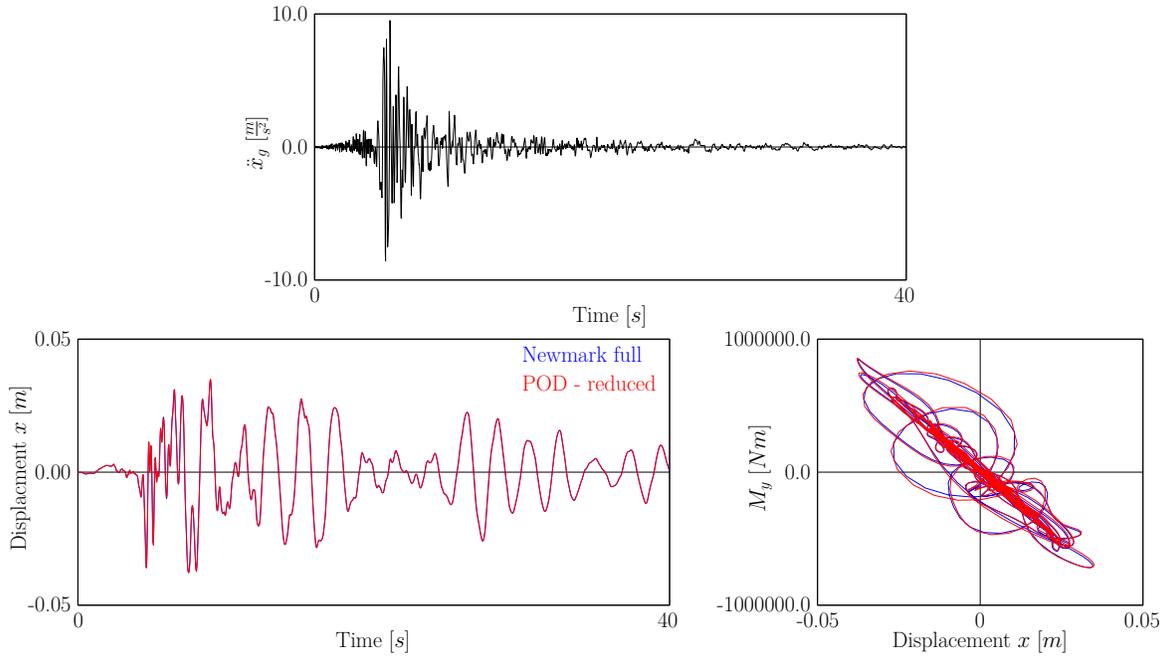


Figure 60: Response to the Loma Prieta earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y

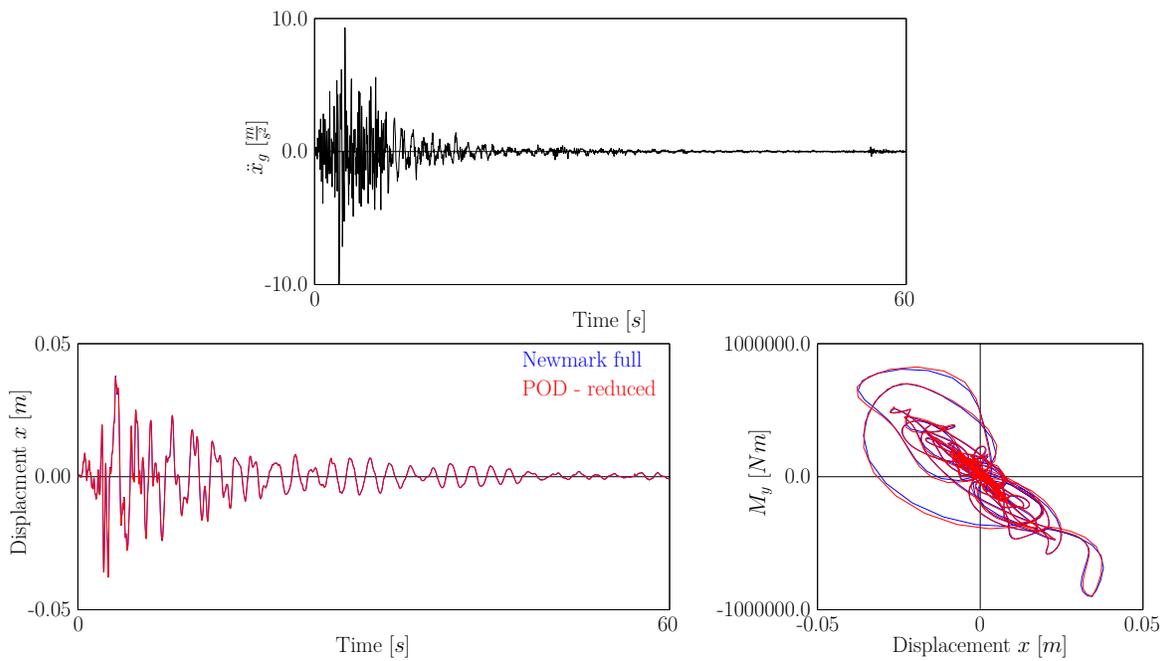


Figure 61: Response to the Palm Springs earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y

4.5.3 Conclusion

The "universal" MOR reduction presented in section 4.4 has been applied to a more complex structure with different nonlinear properties, i.e. elasto plastic material behavior. The numerical examples demonstrate that POD reduced model can describe the nonlinear behavior of the structure and that an accurate approximation of the full system is possible. Hence, the expectation that the POD MOR procedure is applicable for more complex structures with a higher degree of nonlinearity is confirmed.

4.6 An optimized Monte Carlo simulation strategy by the proper orthogonal decomposition

Estimations of structural resistance to various kinds of excitation is without any doubt one of the main problems in engineering. Especially analyzation of earthquake excited structures involve uncertainties not only in the excitation but also in the structure that necessitate expensive calculation procedures in order to compute probabilities of structural failure or to design structures from the human safety and from the monetary point of view.

The analysis requires the application of high dimensional nonlinear finite element structures in order to describe the behavior of the system as realistic as possible. The problem of nonlinear random vibrations is very difficult and often analysis is based on linearization methods, which try to describe the mean value and covariance function of the response of the structure, referred to as *equivalent linearization* (Bucher (2009b)).

An alternative method, which describes an artificial realization, is the well known Monte Carlo (MC) method. It estimates exact response statistics of randomly excited structures, which can be seen as random computation experiments (Roberts and Spanos (1990)). Shinozuka (1972) considers the Monte Carlo method as extremely useful for numerical analysis of nonlinear structures subjected to random excitation.

In case of structural dynamics, samples of a certain kind of excitation (e.g. earthquake) are generated and responses are calculated, i.e. by numeric integration. Depending on certain confidence levels a large number of samples is necessary to obtain acceptable estimation of the response statistics. Therefore even SDOF systems often demand a very large amount of computational effort. For multi degrees of freedom (MDOF) systems, especially nonlinear problems, where the computation of the response to only one sample excitation can already be very time consuming, the required calculation time using the MC method would exceed any limit of acceptance. MOR is a definitive necessity in this context.

4.6.1 Earthquake sampling

The ground acceleration $\ddot{x}_g(t)$ is composed of a filtered white noise excitation $a(t)$ and an envelope function $e(t)$ (Bucher (2009b)):

$$\ddot{x}_g(t) = a(t) e(t) \quad . \quad (95)$$

For the filter a Kanai-Tajimi model is used, where the ground acceleration $\ddot{x}_g(t)$ is defined as a linear combination of the displacement and velocity of a system with a SDOF. This SDOF system is characterized by the natural frequency ω_g and the damping ratio ζ_g . The

power spectral density of the ground acceleration is (Bucher (2009b))

$$S_{aa}(\omega) = S_0 \frac{4\zeta^2\omega_g^2 + \omega_g^4}{(\omega_g^2 - \omega^2)^2 + 4\zeta^2\omega_g^2\omega^2} . \quad (96)$$

The filtered white noise excitation $a(t)$ in Eq. (95) is generated based on the power spectral density.

The envelope function

$$e(t) = 4 \left[\exp\left(-\frac{t}{4}\right) - \exp\left(-\frac{t}{2}\right) \right] \quad (97)$$

is necessary to represent the time-dependent intensity of the ground motion. Some examples of generated earthquakes are shown in Fig. 62.

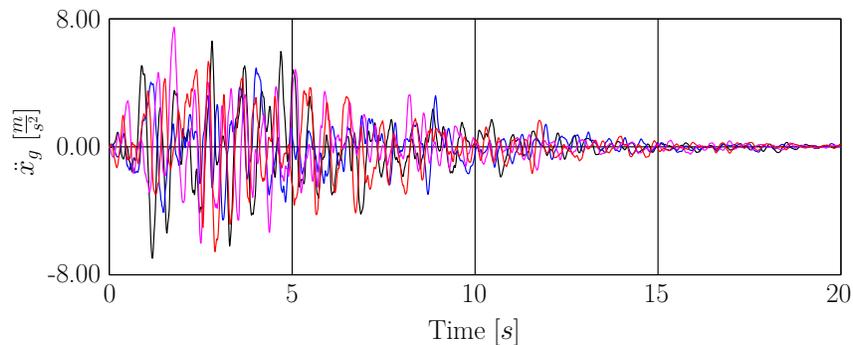


Figure 62: Samples of generated earthquakes - ground acceleration $\ddot{x}_g(t)$; $S_0 = 0.1 [m^2/s]$, $\zeta = 0.3 [-]$, $\omega_g = 15 [rad/s]$

4.6.2 ”Universal” POD reduction - statistical evaluation

In this section another application of ”universal” model order reduction by the POD method, which is much more appropriate for MC simulation, is presented. The snapshot matrix is calculated by integrating over a small time period of one generated earthquake excitation of the sample training set, and subsequently the ”universal” transformation matrix $\Phi_{p,r}$ is computed. Every system is now truncated into the ”universal” POD subspace using $\Phi_{p,r}$ and the approximated response is computed. One Monte Carlo run with 1000 samples is performed. The parameters of the power spectral density using the Kanai-Tajimi filter are $S_0 = 0.1 [m^2/s]$, $\omega_g = 15 [rad/s]$ and $\zeta = 0.3 [-]$. The full response is calculated as well for the purpose of comparison. A sketch of the approach of this reduction strategy is presented in Fig. 63 for the sake of clarity.

The test structure is nonlinear and is analogous to the one in section 4.4.1, which is presented in Fig. 34. A linear frame structure is resting on a nonlinear friction element, which connects the structure with the ground.

The big advantage of this so called ”universal” POD approach is that the time consuming procedure, i.e. the calculation of the snapshot matrix and following the transformation

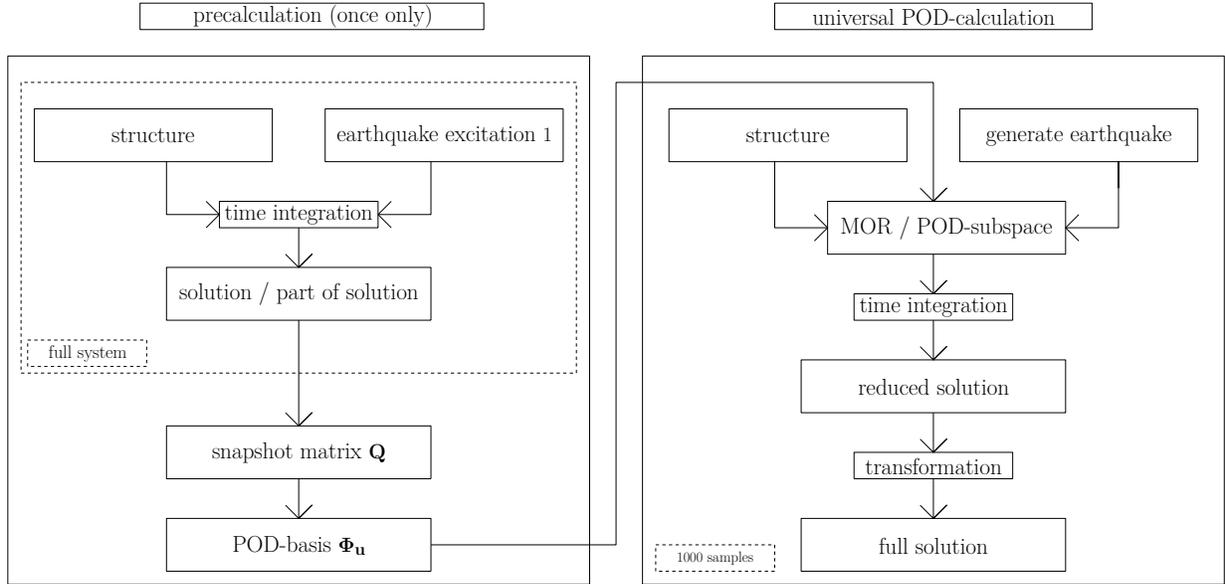


Figure 63: Calculation approach of the universal POD reduction with generated earthquakes

matrix $\Phi_{p,r}$, is performed only once for all computations concerning this structure. Especially when high dimensional models are generated an immense computational effort can be saved by the utilization of only one transformation matrix for all response calculations.

Taking into account that $\mu_h = E[h_{rel}]$ and $\sigma_h = E[(h_{rel} - \mu_h)^2]$ the error of the expectation $E_\mu = \mu_{h,full} - \mu_{h,POD}$ and the error of the standard deviation $E_\sigma = \sigma_{h,full} - \sigma_{h,POD}$ are calculated. Statistical evaluations is presented in Fig. 64.

The results reveal out that the reduced system is an accurate approximation of the results of the full system (in most cases more than adequately accurate for civil engineering problems). An angular phase shift of the POD solution in comparison to the full solution is observed as a result of the occurrence of frequencies appear that are not considered in the observation matrix (i.e. snapshot matrix). Often it is sufficient to choose a small time period at the beginning to assemble the observation matrix, but it is possible to define another time period within the excitation time period as well in order to improve the accuracy of the calculation. In this case an investigation of the frequency range dependent on time would make sense (i.e. wavelet transformation). Sometimes it can help to calculate the whole first excitation sample and spread the snapshots over the whole time period of the transient excitation. However, an improvement of the quality of the reduced solution is not guaranteed.

4.6.3 "Universal" POD reduction - Monte Carlo Simulation numerical example

Based on the statistical investigations of the previous section on a rather simple system, subsequently a numerical demonstration of the Monte Carlo procedure together with the POD strategy on a more complicated structure is presented. The dynamic structure is the "Las Vegas" building according to Fig. 50, where the "universal" MOR strategy is already presented with a set of six earthquake records. Elasto-plastic material properties

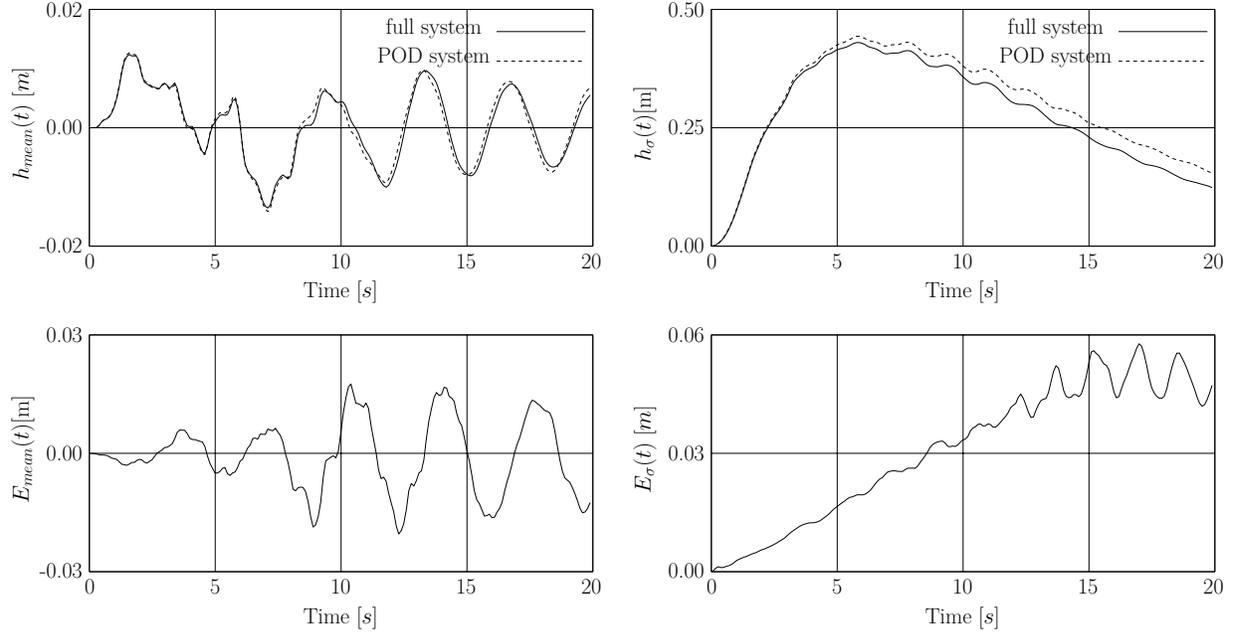


Figure 64: Results of the statistical evaluation 1000 sample transient excitations; Left top: Mean of the horizontal displacement μ_h ; Right top: standard deviation of the relative horizontal displacement σ_h ; Left bottom: Error of the mean relative horizontal displacement E_μ ; Right bottom: Error of the standard deviation of the relative horizontal displacement E_σ

are implemented in the beam elements according to Fig. 49. In order to ensure that the right number of plastic hinges is activated in the snapshot matrix in order to guarantee an accurate approximation of the full system with high probability a test run with ten earthquake samples is done. It turns out to chose the full responses to three earthquake samples in order to assemble the snapshot matrix. 120 snapshots are taken spread in equidistant time instants over the three response histories. The test run is presented in Appendix A.

Threshold The aim is to design the dimension of the side length of the quadratic cross section of the columns, b . The Monte Carlo Simulation procedure is done calculating 5×10^4 sample earthquakes within each design cycle. Each earthquake sample leads to a nonlinear response history of every DOF. Fig. 65 presents those DOF that are utilized for the calculation output. If the maximum output of the relative storey displacement in the first floor exceeds a defined value (treshold), it is assumed that the structure fails, i.e.

$$\max \|x_i\| \begin{cases} < x_{max} & \rightarrow \text{structure resists} \\ > x_{max} & \rightarrow \text{structure fails} \end{cases} \quad , \quad (98)$$

where $i = 1 \dots 7$ denotes the number of storey.

For an estimation of the threshold parameter a predetermined deformation curve for the columns is defined. An applied deformation in the floors leads to the assumed function of the bending moment within the column of the first storey, which is shown in Fig. 66. The

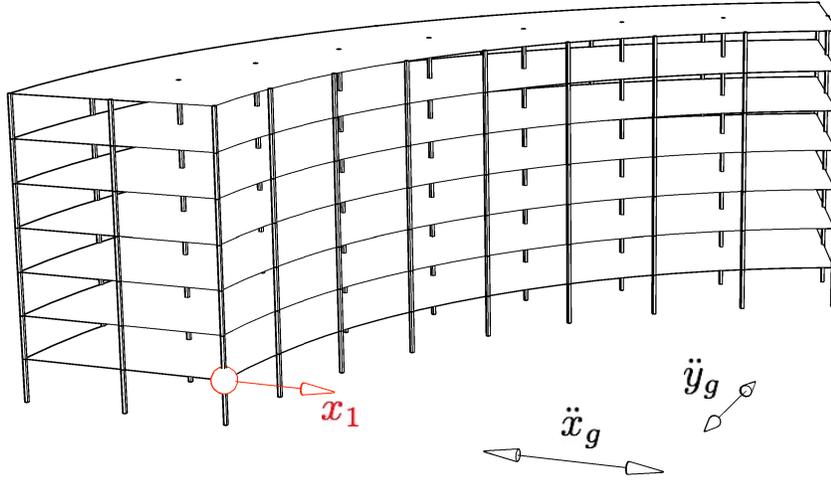


Figure 65: 3-D visualization of the building construction and direction of the earthquake excitation samples; degree of freedom for the response output x_1

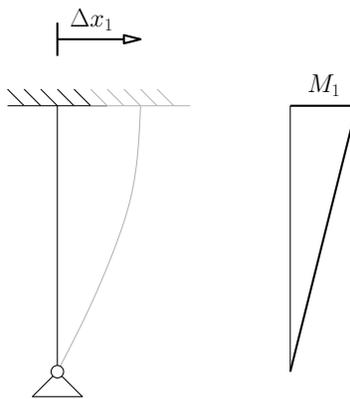


Figure 66: Applied deformation x_1 and corresponding function of the bending moment of the column in the first storey

maximum elastic deformation of the ground storey is then

$$\Delta x_{el} = \frac{f_{yd}h^2}{1.5Eb} \quad , \quad (99)$$

where h denotes the height of one storey. Consequently for a b of 0.25 an elastic limit for the ground storey a deformation x of 0.05 [m] can be estimated. A threshold for the ground storey deformation x of 0.1 [m] in x -direction is chosen. This corresponds with an additional plastic deformation of 100 % with respect to the maximum elastic range. Additionally it has to be mentioned that the maximum deformation in y -direction is about 1 % proportional to the maximum deformation in x -direction. Therefore, the part of the maximum deformation in y -direction is neglected for the Monte Carlo simulation procedures.

Numerical demonstration From the aspect of calculation speed the Newmark method is comparable to the reduced POD strategy for this particular example. The number of DOF is appropriate for the required storage, which is about 1.1 GB and is acceptable for modern computers. The specific type of nonlinearity allows not to recalculate and factorize the stiffness matrix at every time step but to use one stiffness matrix over a longer integration time period for the Newton Raphson iteration procedures. The stiffness matrix is calculated once based on the range, where the structure behaves linearly and therefore no yielding appears (i.e. due to the beginning ascending slope of the stress-strain curve). If there is no yielding no iteration is done, if the material yields the modified Newton Raphson iteration (see Table 2) is performed using the calculated stiffness matrix. Consequently only one stiffness matrix has to be calculated and factorized once before the calculation procedure. This simplification of the Newmark algorithm provides in this specific example a considerably time-saving algorithm to compare the reduced solution with the solution due to the full calculation. Thus the reduced solution is compared with the Newmark integrated solution of the full system. However, if more complicated material behavior is implemented a recalculation and factorization of the stiffness matrix can be necessary, in contrary to the central difference algorithm, where a recalculation as well as an factorization of the stiffness matrix is not performed. For the Monte Carlo calculations the combination of the Central difference method and the POD reduction has enormous advantages as this procedure needs only a minimum amount of computational storage (with a comparable time effort at this specific example). Therefore it is possible to run several time integration algorithms (i.e. calculation of responses to a sample earthquake) parallel on one computer if the processor has more than one core. The Monte Carlo Simulation is realized by activating five to 30 separated calculation procedures (threads) to speed up the computational time by a factor of about five to 30.

Firstly, a set of a priori Monte Carlo simulation runs is conducted. The earthquake samples are calculated according to section 4.6.1. The parameters for the calculation of the power spectral density applying the Kanai-Tajimi filter are $S_0 = 0.1 [m^2/s]$, $\omega_g = 15 [rad/s]$ and $\zeta = 0.3 [-]$. 10^3 samples per run are conducted using the "universal" MOR procedure. Ten runs are done varying b by 0.02 [m]. The starting value for b is 0.2, i.e. a range of $0.2 [m] \leq b \leq 0.38 [m]$ is performed. For every run a new sample set is generated. The peak ground acceleration of the sample earthquakes is comparable to those presented in Table 6.

For every run a new universal POD basis is calculated. 120 snapshots are taken from the response of the structure to three generated earthquakes applying the input parameters

presented in Fig. 62. The snapshots are spread equidistant over the the responses to the three earthquake excitations. Appendix B shows a small test run of ten generated earthquakes including the calculations of the full system applying the Newmark method and the reduced calculations applying the universal POD algorithm.

The result of the first Monte Carlo simulation is shown in Fig. 67. Because of the large mass of the floors, i.e. $d = 0.4 [m]$, the dimension b of the quadratic cross section of the columns gets considerably large if a low probability of failure is required. However, the presentation of this numerical example is not so much focused on practical results but more on the numerical strategies.

It can be seen that for larger dimensions of b , i.e. $b = 0.36 - 0.38 [m]$ a higher number of samples is required to provide more reliable results of the probability of failure. Therefore three runs with 10^4 sample earthquakes are performed. The results can be seen in Fig. 68. The results obtained by the first Monte Carlo run (10^3 samples per run) are pointed out blue and the results obtained by the second Monte Carlo run (10^4 samples per run) are pointed out red in Fig. 68.

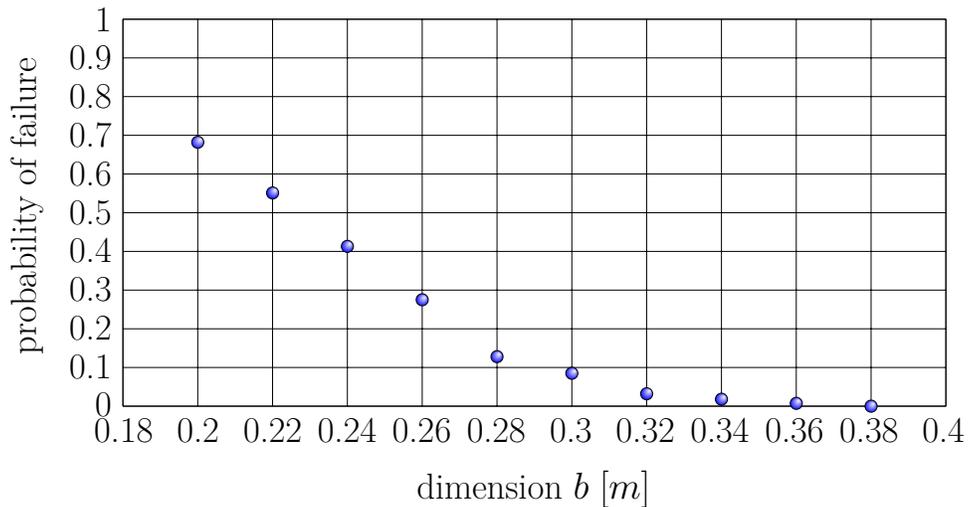


Figure 67: Results of the first Monte Carlo simulation with a threshold of $x_{max} = 0.1 [m]$ in the first floor; for every point 10^3 samples are performed

The results of the 10^4 to the 10^3 Monte Carlo sample run shows indeed differences as presented in Fig. 68. Especially for a value of $b = 0.37 [m]$ the number of samples of the first Monte Carlo run is way too small to present results, which are not zero. The second Monte Carlo run, where the number of samples is ten times higher, is able to provide a useful probability of failure of the structure. The conclusion is that a dimension b of $0.37 [m]$ has to be chosen if a probability of failure of under 0.001 and a dimension b of $0.36 [m]$ has to be chosen if a probability of failure of under 0.01, i.e. one %, is required.

4.6.4 Conclusion

The goal of this part is to provide a reduced Monte Carlo simulation strategy in order to realize the application of high dimensional models. The innovative focal point is that the full system has only to be calculated once in order to obtain the observation matrix and with

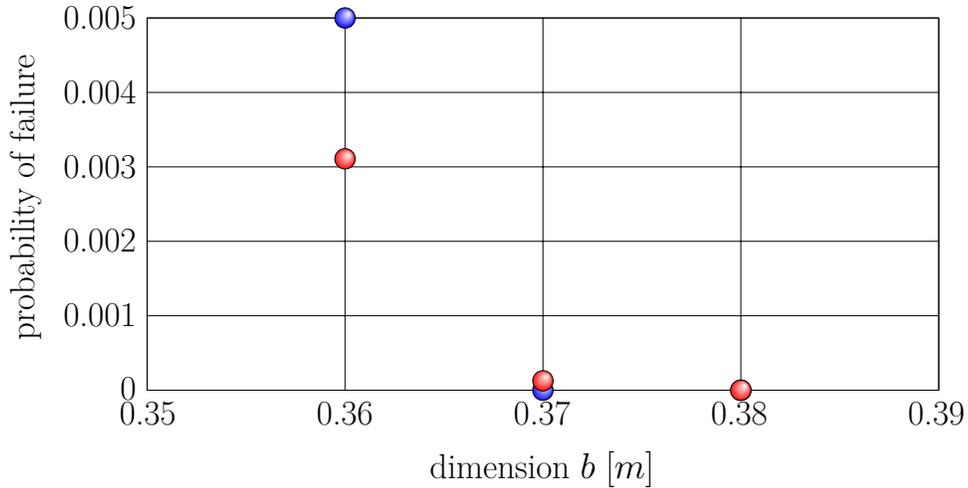


Figure 68: Results of the second Monte Carlo simulation with a threshold of $x_{max} = 0.1$ [m] in the first floor; for every blue point 10^3 samples are performed; for every red point 10^4 samples are performed

this information to assemble the deterministic transformation matrix containing the POD modes. With this transformation matrix the whole Monte Carlo simulation is conducted. Firstly the MC calculation procedure has been tested on a rather small and simple model, which has been used on other sections before, in order to be able to compare the statistical output of the full with the reduced system. Although the excitation samples differ from each other the reduction strategy is promising and the approximation error is negligible for civil engineering applications. Secondly the design of a more complicated structure is realized using the Monte Carlo Simulation procedure together with the POD method. Advantages of this method together with the practical implementation are carved out. Firstly eight Monte Carlo simulation runs with 10^3 samples varying the dimension of the quadratic cross section of the columns are performed. In order to produce reliable results concerning the probability of failure a considerably high number of Monte Carlo samples within one run is necessary. Therefore Monte Carlo simulation runs with 10^4 samples in a smaller range, which is chosen due to the result of the first Monte Carlo simulation runs, is performed. This is possible for high dimensional systems applying the "universal" MOR strategy as presented. Obviously the presented new strategy indeed creates the possibility to do artificial realizations on high dimensional structures in an expenditure of time, which is practically justifiable.

5 A new hybrid model order reduction strategy for impact problems

A novel hybrid MOR technique is presented, which includes the classical modal truncation and the nonclassical POD method. The analytic solution of a linear problem requires modal analysis, which creates the possibility to uncouple the DOF. Unfortunately if the number of DOFs becomes high the calculation of the eigenvalue solution becomes expensive in an disproportional way because the stiffness matrix must be inverted and in order to calculate

the exact response all eigenvectors must be evaluated. The second disadvantage can be avoided by a truncation of a large number of high frequency modes if the frequency content of the excitation affects only a small number of lower modes, which is very effective for earthquake excitation. However often this procedure requires much experience or an extensive investigation of the frequency content of the excitation. For impact problems, where a large frequency range is excited, the application of modal truncation, i.e. a cut off of higher frequency modes, cannot approximate the full solution without loss of the high frequency response, which can be responsible for maximum stresses and forces in the structure. Therefore this section present a new strategy which considers lower modes and selected higher frequency deterministic modes in order to increase accuracy within a minimum effort of time.

The impact load is realized by a simple idealization based on Ziegler (1998). The time period of contact is assumed to be zero, and thus the velocities of the two bodies are subjected to an abrupt change. Conservation of momentum leads under consideration of a collision factor, which defines the level of plasticity, to the velocities before and directly after the contact time instant. The level of plasticity is zero for following calculations, as a consequence a completely elastic impact is assumed. Consequently the impact load can be described by the solution of the free damped vibration subjected to an initial velocity vector $\dot{\mathbf{x}}_0$, where all components are zero except those, which are allocated to the spacial impact point within the FE discretized mesh. For more detailed information about this idealization of the impact problem the reader is referred to Ziegler (1998).

The aim is not to focus on the huge field of contact problems but much more on the generation of an excitation that covers a broad frequency band in order to test the new model order reduction method.

The homogenous linear damped set of equations of motion of the MDOF system is equivalent to Eq. (2) without force function

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{0} \quad (100)$$

and the full response function is due to the initial conditions $\mathbf{x}(t = 0) = \mathbf{0}$ and $\dot{\mathbf{x}}(t = 0) = \dot{\mathbf{x}}_0$, the sum of all decoupled homogenous solutions in the modal space (cf. Chopra (2001))

$$\mathbf{x}(t) = \sum_{i=1}^n \varphi_{\mathbf{m},i} q_{m,i} \quad , \quad q_{m,i} = e^{-\zeta_i \omega_{0,i} t} \begin{bmatrix} \dot{q}_{0,i} \sin \omega_{d,i} t \\ \omega_{d,i} \end{bmatrix} \quad , \quad (101)$$

where $\dot{\mathbf{q}}_0 = \Phi_{\mathbf{r}}^{-1} \dot{\mathbf{x}}_0$. This solution is equal to the responses to piecewise analytic functions presented in Eq. (16) and (18) with a zero vector as force function. An approximation of the global response of the structure can be realized by the truncation of k modes, where k is a small number, $k \ll n$ (i.e. modal truncation, see section 3.4.1). Although the inversion of the stiffness matrix is necessary the computational effort can be reduced significantly by the numerical evaluation of only a small number of modes. However, high dimensional movements, which are responsible for the main content of internal stresses and forces due to the impact load, cannot be captured by the low dimensional truncation. By taking more modes into account an accurate approximation of the exact response is not guaranteed because the mode number, where to truncate high frequency motion, is not known and can therefore only be estimated empirically.

Both the global lower frequency and the higher frequency modes, which are responsible for peak stresses and forces close to the impact area, must be considered for a more

accurate approximation. High frequency motions can be captured in the observation matrix, which is composed of the solution vectors in a small finite time period instantaneously after the impact. The response in this small time period is calculated numerically. As very effective method for this purpose the Newmark method for linear systems (see section 3.2.7) is chosen, which is an implicit time integration scheme but has an explicit formulation for linear systems (requiring a matrix inversion) and is unconditionally stable for an average acceleration approximation within one time step. Since this observation matrix $\mathbf{X}_s = [\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_s)]$ contains only high frequency information for a small time interval $t_0 \leq t \leq t_s$, the low frequency modes of the modal truncated matrix $\Phi_{m,r}$ (Eq. 73), which describes the global behavior, are added to the observation matrix

$$\mathbf{X}_h = [\mathbf{X}_s, \Phi_{m,r}] = \left(\begin{array}{cccc|cccc} x_1(t_1) & x_1(t_2) & \cdots & x_1(t_s) & \varphi_{m,11} & \varphi_{m,12} & \cdots & \varphi_{m,1k} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n(t_1) & x_n(t_2) & \cdots & x_n(t_s) & \varphi_{m,n1} & \varphi_{m,n2} & \cdots & \varphi_{m,nk} \end{array} \right). \quad (102)$$

This now so-called hybrid snapshot matrix \mathbf{X}_h contains local and global information and the POD is realized by the singular value decomposition of the hybrid observation matrix (see section 3.4.2). The left singular vectors are the hybrid POD modes and the singular values give information about the energy content of the corresponding hybrid POD mode. Since the singular values are arranged in an rectangular matrix in descending order, a regulated truncation of the hybrid POD modes is conducted, according to Eq. (80) and (84), where 99 percent of the total energy of the system is captured in the observation time period. The transformation matrix into the hybrid subspace is now $\mathbf{x} \approx \Phi_h \mathbf{q}_h$, where $\Phi_h \in \mathbb{R}^{n \times l}$, $l \ll n$ and $\mathbf{q}_h \in \mathbb{R}^{l \times 1}$ is the coordinate in the hybrid subspace.

The equation of motion in the hybrid subspace is

$$\mathbf{m}_h \ddot{\mathbf{q}}_h + \mathbf{c}_h \dot{\mathbf{q}}_h + \mathbf{k}_h \mathbf{q}_h = \mathbf{0}, \quad (103)$$

where \mathbf{m}_h , \mathbf{c}_h , $\mathbf{k}_h \in \mathbb{R}^{l \times l}$ are not diagonal matrices. The equation of motion is transformed into the state space according to Eq. (19) and this linear ODE is after adaptation of Eq. (23) to a zero force vector and under consideration of the initial condition $\mathbf{y}_0^T = [\mathbf{x}_0^T, \dot{\mathbf{x}}_0^T]^T$ and the modal transformation $\mathbf{z}(t) = \Phi \mathbf{y}(t)$ solved in the state space according to Eq. (27).

Subsequently, the combination of a numerical and an analytical procedure is introduced. This combination can be realized by using different numerical methods. The aim is to benefit from the advantages of the different methods in order to improve efficiency while preserving required accuracy conditions. As a consequence, e.g., the selection of the numerical time integration method to obtain the observation matrix depends on the size of the system. Therefore if dimensions become very high, the central difference approach will be more effective than the Newmark method. Thus it is up to the engineer to decide, which time integration algorithm is used to calculate the snapshot matrix.

5.1 Numerical demonstration

A three dimensional linear portal frame structure is generated. The FE structure is assembled by beam and shell/plate elements as presented in Fig. 69. The impact point is at the left corner in the first floor, the mass is hitting the structure at the impact point in

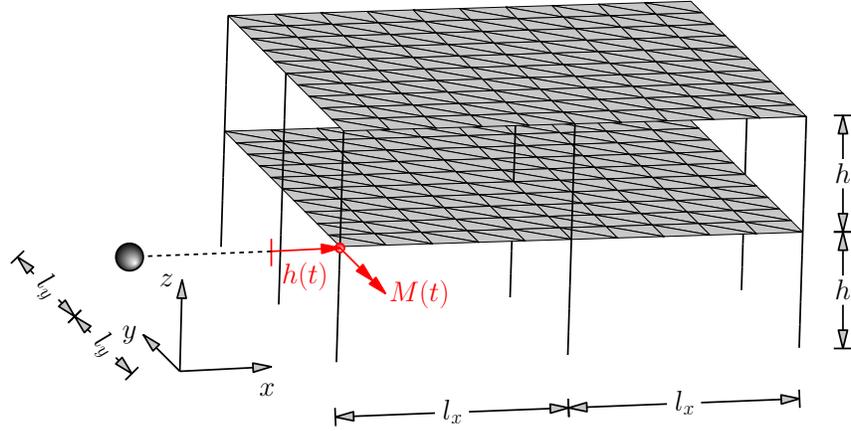


Figure 69: Portal frame structure with beam and plate/shell elements; $l_x = 10$, $l_y = 10$, $h = 5[m]$; number of DOF = 1911; the impact point as well as the horizontal displacement $h(t)$ and the bending moment $M(t)$ are highlighted in red

x -direction, as presented red marked in Fig. 69. An algorithm has been developed, which makes it possible to change the number of stories, the number of horizontal frames in x and y direction and the number of elements per story. As a consequence the new method can be tested and compared to classical methods dependent on the level of high dimensionality of the structure.

The first three structural global modes are depicted in Fig. 70. The hybrid observation matrix is assembled by 50 observations, which are calculated numerically by the Newmark method ($\Delta t = 0.001[s]$) and from the first ten normal modes. The first three hybrid POD modes are shown in Fig. 71.

The results, i.e. horizontal displacements $h(t)$ and the bending moment $M(t)$ (defined red marked in Fig. 69) are presented in Fig. 72 and 73, showing the full analytic, the modal truncated and the hybrid reduced response. The displacement is approximated more accurately by the hybrid method than by the method of modal truncation. The approximation of the bending moment is accurately by the hybrid method, where the method of modal truncation fails completely. In Fig. 74 the energy content of each POD mode is shown in relation to the entire energy of the system in the observation time period according to Eq. (84), representing the property of optimality. It is shown that taking more than 25 snapshots in an equidistant time period of 0.001 [s], i.e. a snapshot time period of 0.25 [s] the result does not improve noticeable.

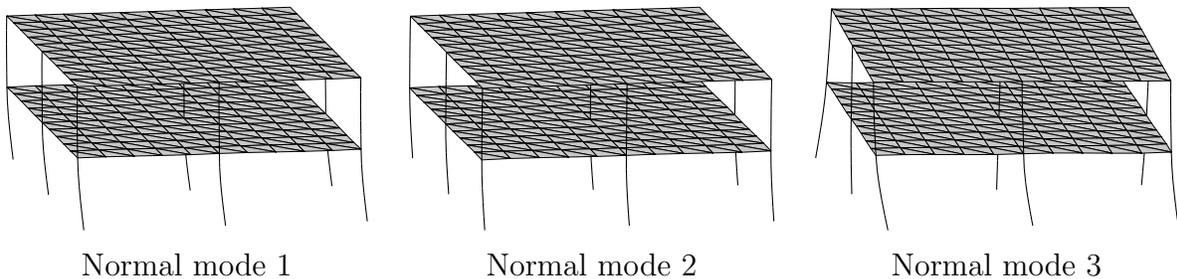


Figure 70: First 3 normal modes

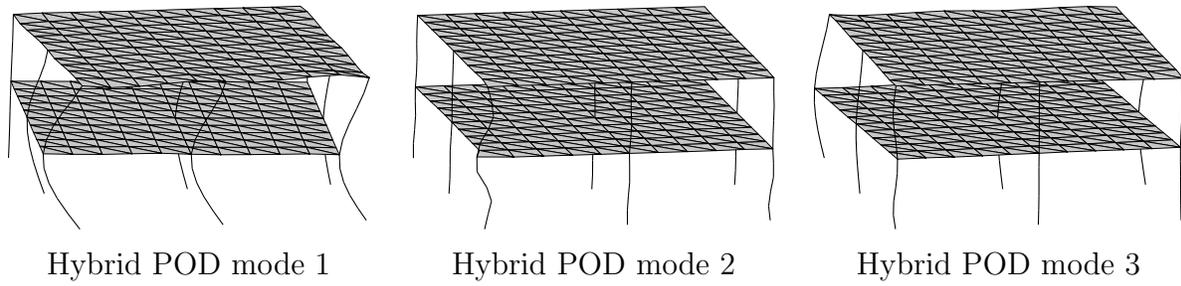


Figure 71: First 3 POD modes

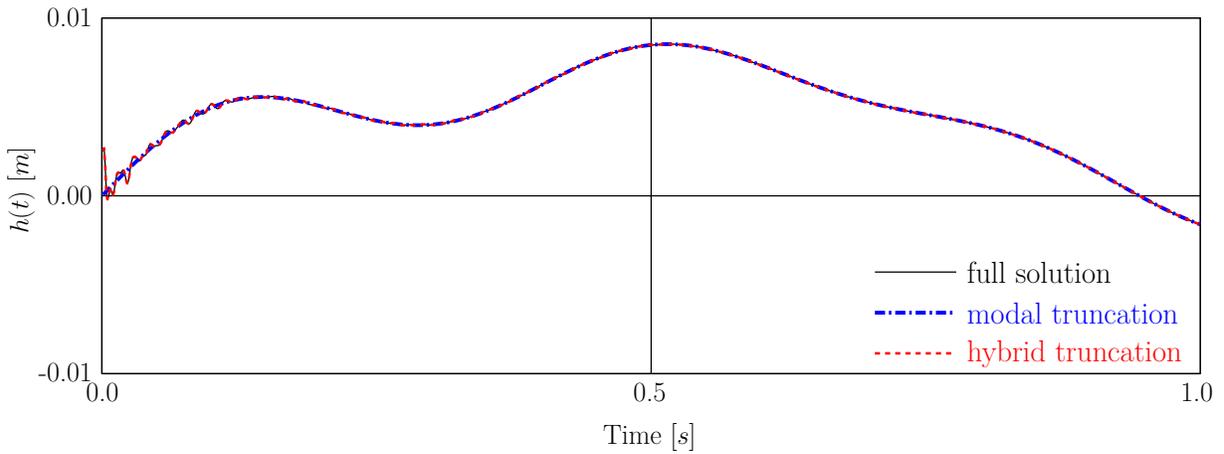


Figure 72: Displacement response $h(t)$ (the lines of the hybrid truncation solution and full response cover each other)

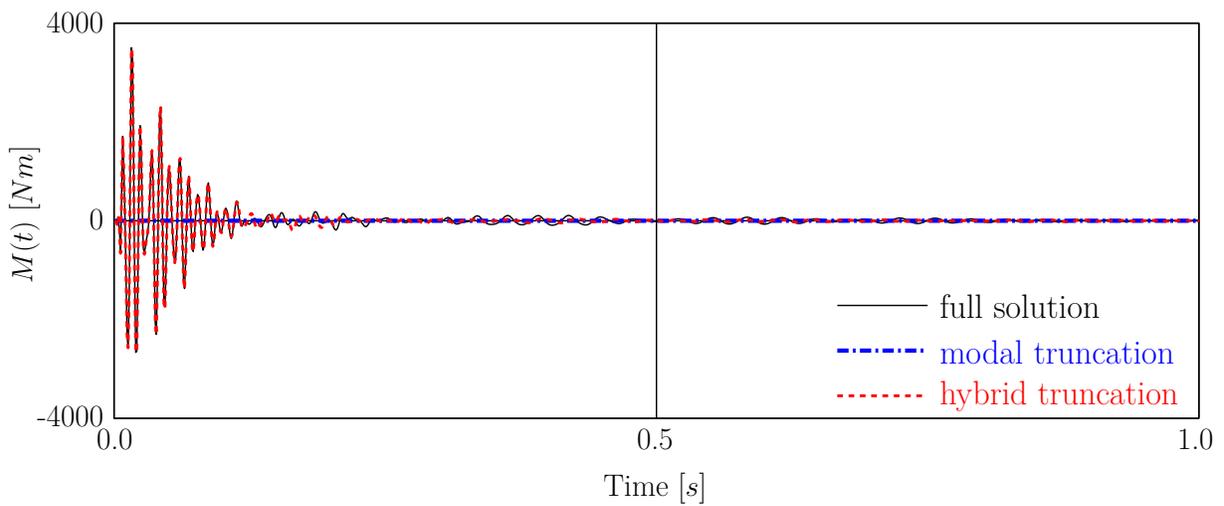


Figure 73: Bending moment response $M(t)$ (note: the lines of the hybrid truncation response and full solution cover each other but modal truncation fails completely)

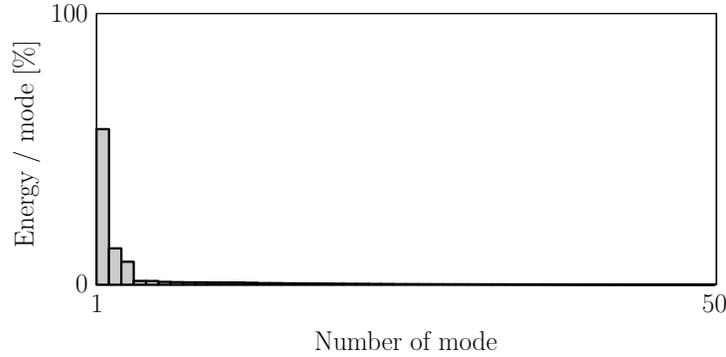


Figure 74: Energy content of the hybrid POD modes

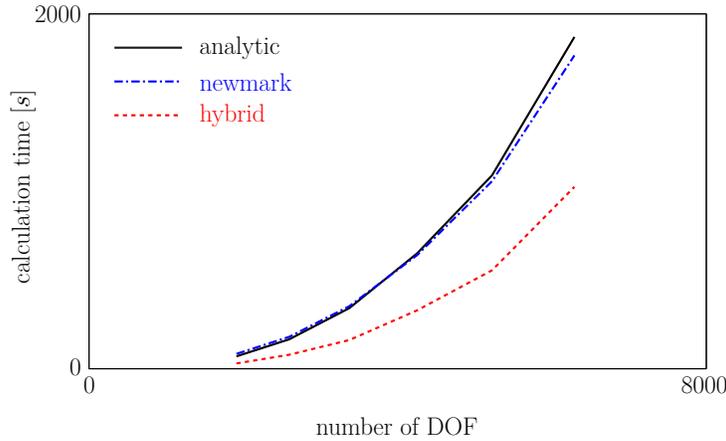


Figure 75: Computational time as a function of the numbers of DOF

5.2 Numerical efficiency and limits

Here the calculation times of the full analytic, the numerical Newmark and the new hybrid method is discussed. The efficiency with respect to the numbers of DOF is revealed in Fig. 75. It is shown that the hybrid truncation method reduces the computational effort compared to the classical methods by preserving a demanded level of accuracy in contrast to the modal truncation strategy. Generally the computational time is reduced by a factor of $1/2$.

The error of the reduced system is the difference of the full solution \mathbf{x} and the hybrid solution \mathbf{x}_h , i.e.

$$\mathbb{E} = \mathbf{x} - \mathbf{x}_h . \quad (104)$$

A scalar value for the error of the hybrid response is the first invariant of the covariance matrix $\Sigma_{\mathbb{E}} = E\{(\mathbb{E} - \boldsymbol{\mu})(\mathbb{E} - \boldsymbol{\mu})^T\}$ of the error of the complete result, where $\boldsymbol{\mu} = E\{\mathbb{E}\}$ is the expectation of the error (cf. section 4.3 as well). In Fig. 76 the error in form of the first invariant of the covariance matrix is presented as a function of the number of observations. As observations were taken every $0.001[s]$ the observation time period of $0.5[s]$ is equivalent to 500 observations.

According to the outcomes of Fig. 76 it can in the main be concluded that the error

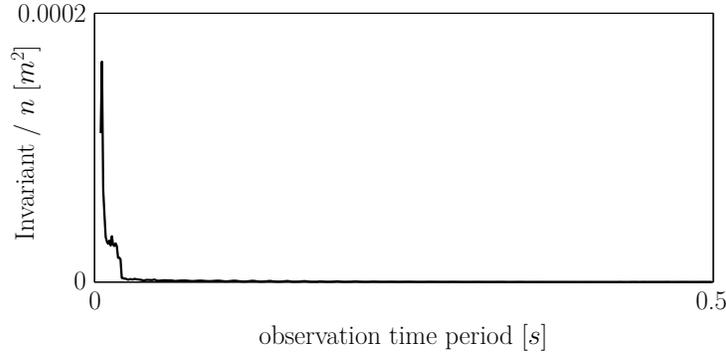


Figure 76: First invariant of the error covariance matrix as a function of the observation time period

decreases with increasing number of observations. However, a significant improvement of the solution when taking more than 25 observations in an observation time period of 0.025 seconds into account is not achieved.

5.3 Conclusion

A new MOR procedure for impact problems is presented, i.e. the so-called hybrid reduction method. Modal truncation captures the global low frequency content and the proper orthogonal decomposition describes local high frequency motions. It is shown that this hybrid reduction method is indeed able to approximate not only the displacement- but also the internal force response accurately where the classical method of modal truncation fails completely and the utilization of the analytic solution and numerical methods are much more time consuming.

6 General conclusion

The spacial discretization by the FE method in earthquake engineering and structural dynamics leads often to high dimensional linear and nonlinear systems. Therefore, the calculation of the response of a structure to an earthquake excitation applying numeric integration methods comes along with either expensive matrix operations (e.g. matrix factorizations) or to a large number of calculation loops, which depends on the type of numeric algorithm, i.e. implicit or explicit numeric integration. In structural dynamics the Newmark method (i.e. implicit numeric integrator) has established because of its property of unconditional stability (depends on the choice of two input parameters) and high accuracy. The downside is that a factorization of the stiffness matrix is, dependent on the problem, at least once or also repeatedly required in the calculation process. As a consequence the application of implicit numeric integration algorithms is limited concerning the number of DOF of the system, because a huge amount of storage is required. In this dissertation the application of explicit integration methods is applied that is not limited because of required storage but of calculation time. To overcome this problem MOR strategies based on the combination of the central difference algorithm and the POD method are presented that should provide accurate approximations of the responses to the full system in a fractional amount of time.

This dissertation deals with MOR reduction strategies applying the POD method. The presented MOR strategy is applied to simple earthquake excited structures and is consequently improved concerning more complicated structures and methodical changes. Additionally a new MOR strategy for linear impact problems applying the POD is presented.

The POD method is applied to a planar linear and nonlinear frame system excited by the El Centro earthquake record. The POD method is realized by integrating over a small part of the beginning of the excitation, i.e. the snapshot time period, in the physical coordinate and with this information the POD basis, i.e. the transformation matrix into the POD reduced subspace, is generated. Subsequently, the reduced system is integrated over the whole time period and then the coordinate in the reduced POD subspace is transformed back into the physical coordinate. This procedure is presented on the linear frame structure. Here the POD method is a useful alternative to the method of modal truncation. Additionally it has advantages to the method of modal truncation as it does not need a factorization of the stiffness matrix and is therefore not limited concerning number of DOF. Furthermore, nonlinear reacting friction bearings are added to the planar frame system in the basements. The application of the POD method produces accurate approximations of the full solution if nonlinear reactions are activated within the snapshot time period. Not only displacements but also moments and shear forces are calculated, which are approximated accurately as well applying the POD method. Error estimations are done dependent on the used number of POD modes and the length of the snapshot time period. The result is that only a small number of POD modes is necessary for an accurate approximation. If the snapshot time period is too short, the essential motion patterns cannot be described by the POD modes, which leads to inaccurate approximations.

The POD method is extended to a "universal" MOR strategy. The idea is that the POD basis is calculated by integrating over a small time period of one earthquake. Subsequently, this transformation matrix is not only used to solve the structure excited by this excitation but also the structure excited by another earthquake with similar properties. Therefore, in this dissertation the transformation matrix called "universal" POD basis or "universal" transformation matrix. This new strategy is now presented on a similar planar nonlinear frame system (nonlinearities are again caused by a friction element at the basement of the structure) on two main types of earthquake excitation records, i.e. near-fault and far-fault records. The far-fault excitation pair is the Fukushima foreshock record and the Fukushima main earthquake. The structure is integrated in a small time period in the middle of the Fukushima foreshock record, the POD basis is generated and subsequently the structure is integrated over the whole Fukushima foreshock time period in the reduced POD subspace. This procedure is equivalent to the procedure applied to the El Centro earthquake before. The approximation is accurate, consequently the "standard" POD procedure is again verified. The "universal" transformation matrix is now used to transform the structure excited by the Fukushima main earthquake into the reduced POD subspace, integrate over the whole time period and transform back into the physical coordinate. The approximation applying this strategy is not as accurate as for the standard POD procedure. This is because the intensity of the Fukushima foreshock record is smaller by a factor of thirty. Therefore, nonlinear reactions cannot be captured in the snapshot matrix and consequently not described at the response to the Fukushima main earthquake. An overestimation of about 10 % of the maximum displacement can be observed. Generally spoken this is still a useful approximation in civil engineering problems. The near-fault earthquake excitation

pair is presented by the Imperial Valley and the Northridge record. The "universal" POD basis is calculated by integrating over a small part in the middle of the Imperial Valley record. Subsequently the structure is integrated over the whole time period of the Imperial Valley record in the reduced space, i.e. the "standard" POD reduction procedure, and over the whole time period of the Northridge record, i.e. the "universal" POD reduction strategy. Not only by the "standard" but also by the "universal" POD strategy the full response is approximated accurately as these two records are similar concerning frequency content and intensity.

The "universal" POD strategy is now applied to a more complicated three-dimensional nonlinear building structure discretized by linear shell/plate elements (floors) and nonlinear beam elements (columns). The nonlinearities are caused by a bilinear stress-strain relation in axial direction of the beam elements. Six earthquake records are presented, i.e. the Bam earthquake record (2003), Northridge Rinaldi record (1994), Imperial Valley record (1979), Landers record (1992), Loma Prieta record (1989), North Palm Springs record (1986). The "universal" POD basis is calculated by integrating over the Northridge earthquake record. 300 snapshots are taken spread over the whole time period of the response to the Northridge earthquake record. Accurate approximations of the full response functions can be observed. Merely the approximation of the response to the Imperial Valley earthquake record shows small differences. This is caused by the fact that plastic hinges appear that are not developed in the Northridge earthquake. Therefore, these plastic reactions cannot be described in the reduced system.

A new Monte Carlo simulation technique is presented applying the "universal" POD reduction strategy. A small Monte Carlo test run is performed on the nonlinear planar considerably simple frame structure (same test structure presented for the "universal" MOR strategy on far-fault and near-fault records). The responses to 1000 earthquake samples are calculated in the physical coordinate. The POD basis is obtained by a small part of the response of one generated sample earthquake. Subsequently the POD approximated response to all sample earthquakes is calculated. The responses of full and the reduced systems are statistically evaluated and error estimations are done. The results of this optimized Monte Carlo simulation strategy are promising. Therefore this new Monte Carlo simulation strategy is demonstrated on the design of a nonlinear three-dimensional building construction, which was already applied for the presentation of the "universal" MOR strategy. The aim is to design the length of the cross section b of the beam elements. The threshold is chosen as $0.1 [m]$ of the elastic deformation in the first floor in x -direction. At first ten sample runs with 10^3 samples are done in order to estimate the parameter b . Subsequently, in order to assess small probabilities of failure, some sample runs with 10^4 samples are realized in a smaller range. The Monte Carlo simulations in the reduced space are performed by multithread algorithms in order to emphasize the advantages of the combination of the POD method and the central difference algorithm, i.e. small amount of storage and short computational time.

A "hybrid" MOR strategy is presented. Impact problems excite a wide frequency content. Where the main global displacements can be approximated considerably accurate by some low natural modes, stresses and forces near the impact point depend on high frequency modes. For high dimensional systems the calculation of the full linear response requires the solution of the eigenvalue problem and with it the factorization of the stiffness matrix and additionally an iterative procedure to compute all eigenvalues and eigenvectors, which re-

quires a lot of computational effort. Consequently, by a truncation of a few lower modes it is possible to describe stresses and forces near the impact point. Therefore a "hybrid" method is proposed, where a snapshot matrix is composed by numeric integration of the system directly after the impact time instant in order to describe high frequency motion. Additionally a few lower natural modes are added to this snapshot matrix in order to describe global low frequency motion. The strategy is tested on a linear three-dimensional structure. The results are promising. An accurate approximation of the displacements and the stresses and forces can be observed calculating only over a small snapshot time period directly after the impact time instant, where the method of modal truncation is not able to approximate the displacement history as accurate as the new "hybrid" method does and fails completely describing stresses and forces near the impact point.

References

- A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics, 2005.
- F. Bamer and C. Bucher. Application of the proper orthogonal decomposition for linear and nonlinear structures under transient excitations. *Acta Mechanica*, 223(12):2549–2563, 2012.
- M. C. C. Bampton and R. R. Craig. Coupling of substructures for dynamic analyses. *American Institute of Aeronautics and Astronautics*, 6(7):1313–1319, 1968.
- K. J. Bathe. *Finite Element Procedures*. Prentice Hall, 1995.
- R. E. Bellman. *Bellman, R. E. Introduction to Matrix Analysis*. New York: McGraw-Hill, 2nd edition, 1970.
- B. Bienkiewicz, Y. Tamura, H. J. Ham, H. Ueda, and K. Hibi. Proper orthogonal decomposition and reconstruction of multi-channel roof pressure. *Wind Engineering and Industrial Aerodynamics*, 54-55:369 – 381, 1995.
- I.N. Bronstein, K.A. Semendjajew, G. Musiol, and H. Mühlig. *Taschenbuch der Mathematik*. Harri Deutsch, 2001.
- C. Bucher. Probability-based optimal design of friction-based seismic isolation devices. *Structural Safety*, 31:500–507, 2009a.
- C. Bucher. *Computational Analysis of Randomness in Structural Mechanics*, volume 3 of *Structures and Infrastructures Series*. Taylor & Francis Group, 2009b.
- C. Bucher. Homepage Christian Bucher TU Vienna, August 2014. URL <http://info.tuwien.ac.at/bucher/Private/Welcome.html>.
- A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7):808 – 817, 2000.
- A. K. Chopra. *Dynamics of Structures, Theory and Applications to Earthquake Engineering*. Prentice Hall, 2001.
- A. K. Chopra and Chatpan Chintanapakdee. Comparing response of sdf systems to near-fault and far-fault earthquake motions in the context of spectral regions. *Earthquake Engineering And Structural Dynamics*, 30(12):1769–1789, 2001.
- R W. Clough and J. Penzien. *Dynamics Of Structures*. Computers & Structures, Inc., third edition edition, 1995.
- R. R. Craig and A. J. Kurdila. *Fundamentals of Strucutral Dynamics*. John Wiley & Sons, 2006.
- J.P. Cusumano, M.T. Sharkady, and B.W. Kimble. Spatial coherence measurements of a chaotic flexible-beam impact oscillator. *American Society of Mechanical Engineers, Aerospace Division (Publication) AD*, 33:13–22, 1993.

- E. J. Davidson. A method for simplifying linear dynamic systems. *Automatic Control, IEEE Transactions on*, 11(1):93–101, 1966.
- M. A. Dokainish and K. Subbaray. A survey of direct time-integration methods in computational structural dynamics-i. explicit methods. *Computers & Structures*, 32(6):1371–1386, 1989a.
- M. A. Dokainish and K. Subbaray. A survey of direct time-integration methods in computational structural dynamics-ii. implicit methods. *Computers & Structures*, 32(6):1387–1401, 1989b.
- B. F. Feeny and R. Kappagantu. On the physical interpretation of proper orthogonal modes in vibrations. *Journal of Sound and Vibration*, 211(4):607–616, 1998.
- P. Feldmann and R. Freund. Efficient linear circuit analysis by padé approximation via the lanczos process. *IEEE transactions on computer-aided design of integrated circuits and systems*, 14(5):639–649, 1995.
- Michael Greenberg. *Advanced Engineering Mathematics*. Springer, 2nd edition, 1998.
- John Guckenheimer and Philip Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Springer, 1983.
- R. C. Guyan. Reduction of stiffness and mass matrices. *American Institute of Aeronautics and Astronautics*, 3(2):380–380, 1965.
- E. Hairer, C. Lubich, and G. Wanner. Geometric numerical integration illustrated by the strömer-verlet method. *Acta Numerica*, pages 399–450, 2003.
- S. Han and B. Feeny. Application of proper orthogonal decomposition to structural vibration analysis. *Mechanical Systems and Signal Processing*, 17(5):989–1001, 2003.
- P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1996.
- H. Hotelling. Simplified calculation of principal components. *Psychometrika*, 1(1):27–35, 1935.
- R. Ierusalimschy, L. H. Figueiredo, and W. Celes. Lua 5.1 reference manual, July 2014. URL <http://www.lua.org/manual/5.1/manual.html>.
- R. Kappagantu and B. Feeny. An "optimal" modal reduction of a system with frictional excitation. *Journal of Sound and Vibration*, 224(5):863–877, 1999.
- R. Kappagantu and B. Feeny. Part 1: Dynamical characterization of a frictionally excited beam. *Nonlinear Dynamics*, 22:317–333, 2000a.
- R. Kappagantu and B. Feeny. Part 2: Proper orthogonal modal modeling of a frictionally excited beam. *Nonlinear Dynamics*, 23:1–11, 2000b.
- K. Karhunen. *Über lineare Methoden in der Wahrscheinlichkeitsrechnung*. PhD thesis, University of Helsinki, 1947.

- G. Kerschen and J. C. Golivani. Physical interpretation of the proper orthogonal modes using the singular value decomposition. *Journal of Sound and Vibration*, 249(5):849–865, 2002.
- G. Kerschen and J. C. Golivani. A model updating strategy of non-linear vibrating structures. *International Journal for Numerical Methods in Engineering*, 60:2147–2164, 2003.
- G. Kerschen, J. C. Golinval, A. F. Vakakis, and L. A. Bergman. The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview. *Nonlinear Dynamics*, 41(1):147–169, 2005.
- H. Kikitsu, Y. Okuda, M. Ohashi, and J. Kanda. Pod analysis of wind velocity field in the wake region behind vibrating three-dimensional square prism. *Wind Engineering and Industrial Aerodynamics*, 96:2093–2103, 2008.
- L. Knockaert and D. Zutter. Laguerre-svd reduced-order modelling. *IEEE Transactions on Microwave Theory and Techniques*, 48(9):1469–1475, 2000.
- L. Knockaert and D. De Zutter. Passive reduced order multiport modeling: the padé laguerre, krylov arnoldi svd connection. *Int. J. Electron. Commun. (AEÜ)*, 5:254–260, 1999.
- D. Kosambi. Statistics in function space. *Journal of the Indian Mathematical Society*, 1943.
- P. Koutsovasilis and M. Beitelschmidt. Comparison of model reduction techniques for large mechanical systems. *Multibody System Dynamics*, 20:111–128, 2008.
- Kyoshin-Network. Strong-motion seismograph networks (k-net, kik-net), August 2012. URL www.kyoshin.bosai.go.jp.
- J. W. Leech, P. Hsu, and E. W. Mack. Stability of a finite-difference method for solving matrix equations. *American Institute of Aeronautics and Astronautics*, 3:2172–2173, 1965.
- Y. C. Liang, H. P. Lee, S. P. Lim, W. Z. Lin, K. H. Lee, and C. G. Wu. Proper orthogonal decomposition and its applications - part 1: Theory. *Journal of Sound and Vibration*, 252(3):527–544, 2002.
- M. Loeve. Fonctions aleatoire de second ordre. *Revue Scientifique*, 48:195–206, 1946.
- N. M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division ASCE*, 3:67–94, 1959.
- A. Obadasioglu and M. Celik. Prima: Passive reduced-order interconnect macromodeling algorithm. *IEEE transactions on computer-aided design of integrated circuits and systems*, 17(8):645–654, 1998.
- K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:609–629, 1901.
- PEER. Pacific earthquake engineering research center, August 2012. URL <http://nisee.berkeley.edu>.

- L. T. Pillage and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer-Aided Design*, 9(4):352–366, 1990.
- Z. Q. Qu. *Model order reduction techniques with application in finite element analysis*. Springer London Limited, 2004.
- G. Rega and H. Troger. Dimension reduction of dynamical systems: Methods, models, applications. *Nonlinear Dynamics*, 41(1):1–15, 2005.
- T.G. Ritto, F.S. Buezas, and R. Sampaio. A new measure of efficiency for model reduction: Application to a vibroimpact system. *Journal of Sound and Vibration*, 330:1977–1984, 2010.
- J. B. Roberts and P. D. Spanos. *Random Vibration and Statistical Linearisation*. John Wiley & Sons, 1990.
- W. H.A. Schilders, H. A. Vorst, and J. Rommes. *Model Order Reducion, Theory, Research Aspects and Application*. Springer, 2008.
- M. Shinozuka. Monte carlo solution of structural dynamics. *Computers & Structures*, 2: 855–874, 1972.
- C. W. Silva. *Vibration, Fundamentals and Practice*. Taylor & Francis Group, 2007.
- L. Sirovich. Turbulence and the dynamic of coherent structures. part 1: coherent structures. *Quaterly of Applied Mathematics*, 45(3):561–571, 1987.
- F. Tubino, L. Carassale, and G. Solari. Seismic response of multi-supported structures by proper orthogonal decomposition. *Earthquake Engineering and Structural Dynamics*, 32: 1639–1654, 2003.
- A. S. Veletsos and C. E. Ventura. Modal analysis of non-classicaly damped linear systems. *Earthquake Engineering And Structural Dynamics*, 14(2):217–243, 1986.
- L. Verlet. "computer "experiments" on classical fluids. i. thermodynamical properties of lennardjones molecules". *Physical Review*, 159:98–103, 1967.
- S. Volkwein. *Model reduction using proper orthogonal decomposition*. Karl-Franzens-Universitaet Graz, Institut für Mathematik und Wissenschaftliches Rechnen, April 2008a.
- S. Volkwein. *Proper Orthogonal Decomposition and Singular Value Decomposition*. Karl-Franzens-Universitaet Graz, Institut für Mathematik und Wissenschaftliches Rechnen, 2008b.
- L. Wang, Y. Wang, L. Zhiqiang, and Y. Zhang. Estimation of the vortex shedding frequency of a 2-d building using correlation and the pod methods. *Wind Engineering and Industrial Aerodynamics*, 98:895–902, 2010.
- F. Ziegler. *Technische Mechanik der festen und flüssigen Körper*. SpringerWienNewYork, 3rd edition, 1998.

A Implementation Codes

In order not to interrupt the flow of information and to focus on the main scope of this dissertation the following section describes the background of the presented calculation procedures. All implementations have been done using the program slangTNG, which is a dynamic orientated program created in C++. Important parts of the library for this work are the "tmath"-library, where the main matrix operations as well as several eigenvalue solvers of high dimensional matrices are implemented not only for dense but also for sparse arrays. Also important for this dissertation is the FE library, which includes the implementation of linear and nonlinear beam elements, plate/shell elements tetraeder elements, the dynamic model of friction based isolation devices, etc. The program is free and can be downloaded from the homepage of Bucher (2014), where more detailed information about the extent of the libraries is presented as well. The script language is LUA, which is used because of its outstanding velocity. For information about this script language, which is also the leading script languages for games, and a comprehensive manual the reader is referred to the homepage of Ierusalimschy et al. (2014).

A.1 Implementation of piecewise analytic solution methods

A.1.1 Implementation of the response calculation of a structure to a piecewise constant force function

```
1 -----
2 ----- function vectorization of a diagonal matrix -----
3 -----
4
5 function diag(matrix)
6     vector = tmath.ZeroVector(matrix:Rows())
7     vector:SetOnes()
8     x = matrix*vector
9     return x
10 end
11
12 -----
13 ----- function create a vector with ones -----
14 -----
15 function ones(dimension_rows,dimension_cols)
16     local output = tmath.ZeroMatrix(dimension_rows,dimension_cols)
17     for i = 1 , dimension_rows do
18         for k = 1 , dimension_cols do
19             output[[i-1,k-1]] = 1
20         end
21     end
22     return output
23 end
24
25 -----
26 ----- function piecewise analytic response -----
27 -- input:
28 -- structure ... object with structural information
29 -- C ... damping matrix
30 -- F ... force matrix (ndof x timesteps)
31 -- delta_t ... time step
32 -- x_0 ... initial displacement
33 -- x_0_punkt ... initial velocity
34 -- truncation ... request for modal truncation (type: boolean)
35 -- output:
36 -- x ... solution matrix (ndof x timesteps)
37 -- functions called: diag(...), ones(...)
```

```

38 -----
39 -----
40
41 function analytic_solution(structure, C, F, delta_t, x_0, x_0_punkt, truncation)
42
43     local ndof = structure:GlobalDof()
44     local number_of_modes
45     if truncation == 0 then
46         number_of_modes = ndof
47     else
48         number_of_modes = 10
49     end
50
51     -- modal analysis
52     local M = structure:SparseMass()
53     local K = structure:SparseStiffness()
54
55     local eig_s , phi = K:Eigen(M,number_of_modes -1)
56
57     local M = M:Expand()
58     local K = K:Expand()
59
60     local modal_m = phi:Transpose() * M * phi
61     modal_m = diag(modal_m)
62     local modal_k = phi:Transpose() * K * phi
63     modal_k = diag(modal_k)
64     local modal_c = phi:Transpose() * C * phi
65     modal_c = diag(modal_c)
66
67     local omega_0 = (modal_k:CW() / modal_m):CW()^(1/2)
68     local modal_zeta = modal_c:CW() / (omega_0:CW()*modal_m*2)
69     local zeta = modal_zeta
70     local one_vector = ones(modal_zeta:Rows(),1)
71     local omega_d = omega_0 * (one_vector - modal_zeta:CW()^2):CW()^(0.5)
72
73     -- force
74     local f_modal = phi:Transpose() * F
75
76     local Q = tmath.ZeroMatrix(modal_m:Rows(),F:Cols())
77
78     local inv_phi_tr = phi:Transpose()*M
79     local q_0 = inv_phi_tr * x_0
80     local q_0_punkt = inv_phi_tr *x_0_punkt
81
82     for i = 1 , acc:Rows() do
83         local A = q_0 - f_modal:GetCols(i-1):CW() / (modal_m:CW() *omega_0:CW()^2)
84         local B = ( q_0_punkt + (omega_0:CW()*A):CW()*zeta ):CW() / omega_d
85
86         q_0 = f_modal:GetCols(i-1):CW() / (modal_m:CW()*omega_0:CW()^2) + (
87             (omega_0:CW()*zeta*delta_t*(-1)):CW():Exp() ):CW() * (A:CW()*
88                 (omega_d*delta_t):CW():Cos() + B:CW()* (omega_d*delta_t):CW():Sin())
89             q_0_punkt = ( (omega_0:CW()*zeta*delta_t*(-1) ):CW():Exp() ):CW()* ( (
90                 (omega_0:CW()*A):CW()*zeta*(-1) + omega_d:CW()*B ):CW()
91                 *(omega_d*delta_t):CW():Cos() - ( (omega_0:CW()*B):CW()*zeta + omega_d:CW()*A
92                 ):CW() * (omega_d*delta_t):CW():Sin() )
93
94         Q[{0,i-1}] = q_0
95     end
96
97     local x = phi*Q
98
99     return x
100 end

```

Implementation of the response calculation of a structure to a piecewise constant force function in the state space

```

1 -----
2 ----- solution in the state space for not diagonalizable matrices -----
3 -- input:
4 -- m ... mass matrix (sparse)
5 -- k ... stiffness matrix (sparse)
6 -- c ... damping matrix (sparse)
7 -- force ... force matrix (ndof x timesteps)
8 -- delta_t ... time_step
9 -- x_0 ... initial displacement
10 -- x_0_punkt ... initial velocity
11 -- output:
12 -- Q ... solution matrix (ndof x timesteps)
13 -----
14 -----
15 -- c Franz Bamer
16
17 function analytic_solution_state_space(m,k,c,force,delta_t,x_0,x_0_punkt)
18
19     local m = m:Expand()
20     local c = c:Expand()
21     local k = k:Expand()
22
23     -- generate systemmatrix G
24     G = tmath.ZeroMatrix(k:Rows()*2,k:Rows()*2)
25     G[{0,k:Rows()}] = tmath.Identity(k:Rows())
26     G[{k:Rows(),0}] = - tmath.Inverse(m) * k
27     G[{k:Rows(),k:Rows()}] = - tmath.Inverse(m) * c
28     local inv_G = tmath.Inverse(G)
29
30     local Q = tmath.ZeroMatrix(G:Rows(),force:Cols())
31
32     -- initial condition in the state space
33     local y_0 = tmath.ZeroVector(G:Rows())
34     y_0[0] = x_0
35     y_0[k:Rows()] = x_0_punkt
36
37     -- excitation in the state space
38     local inv_m = tmath.Inverse(m)
39     local g = tmath.ZeroMatrix(G:Rows(),force:Cols())
40     for i = 1 , force:Cols() do
41         g[{k:Rows(),i-1}] = inv_m*force:GetCols(i-1)
42     end
43
44     -- calculate incremental matrix e^{G*delta_t}
45     local eig = tmath.MatrixEigenUnsym(G*delta_t)
46     local Z = eig:Exponential()
47
48     for i = 1 , g:Cols() do
49         y_0 = Z*y_0 - inv_G*(tmath.Identity(G:Rows()) - Z) * g:GetCols(i-1)
50         Q[{0,i-1}] = y_0
51     end
52
53     Q = Q:GetRows(0,k:Rows())
54
55     return Q
56 end

```

A.2 Implementation of numeric time integration methods for linear systems

A.2.1 Implementation of the central difference algorithm for linear systems

```

1 -----
2 ----- central diff algorithm for linear systems -----
3 -- input:

```

```

4 -- K ... stiffness matrix (sparse)
5 -- M ... mass matrix (sparse)
6 -- C ... damping matrix (sparse)
7 -- force ... force matrix (ndof x timesteps)
8 -- x_0 ... initial displacement
9 -- x_0_punkt ... initial velocity
10 -- delta_t ... time step
11 -- output:
12 -- x_erg ... solution matrix (ndof x timesteps)
13 -----
14 -- c Franz Bamer
15
16 function central_diff_linear(K,M,C,force,x_0,x_0_punkt,delta_t)
17
18     local dt = 0.0001
19     local M = M:Expand()
20     local C = C:Expand()
21     local K = K:Expand()
22
23     --initial calculations
24     local U = x_0
25     local U_0_2_punkt = tmath.Inverse(M) * ( - C*x_0_punkt - K*x_0)
26     local U_minus1 = U - x_0_punkt*dt + U_0_2_punkt*dt^2 * 0.5
27
28     local k_dach = M / dt^2 + C / (2*dt)
29     local k_dach_inv = tmath.Inverse(k_dach)
30
31     --constants and storage reservations
32     local a = M / dt^2 - C / (2*dt)
33     local b = K - M*2 / dt^2
34     local t = delta_t
35     local i = 1
36     local x_erg = tmath.ZeroMatrix(U:Rows(),force:Cols())
37     local F = tmath.ZeroVector(force:Rows())
38     local R = tmath.ZeroVector(M:Rows())
39     local U1 = tmath.ZeroVector(M:Rows())
40
41     -- integration loop
42     while i < force:Cols() do
43
44         if (t >= delta_t-0.000001 and t <= delta_t+0.000001) then
45             F = force:GetCols(i-1)
46             t = 0
47             x_erg[{0,i-1}] = U
48             i = i + 1
49             print(i)
50         end
51         collectgarbage()
52
53         R = F - a*U_minus1 - b*U
54         U1 = k_dach_inv*R
55         U_minus1 = U
56         U = U1
57
58         t = t + dt
59
60         if (U[{3,0}]>10000) then
61             print('time step exceeds critical value ---> unstable')
62             break
63         end
64     end
65
66     return x_erg
67 end

```

A.2.2 Implementation of the Newmark algorithm for linear systems

```

1 -----
2 ----- newmark integration algorithm linear -----
3 -- input:
4 -- K ... stiffness matrix (sparse)
5 -- M ... mass matrix (sparse)
6 -- C ... damping matrix (sparse)
7 -- force ... force matrix (ndof x timesteps)
8 -- delta_t ... time step
9 -- x_0 ... initial displacement
10 -- x_0_punkt ... initial velocity
11 -- output:
12 -- x_erg ... solution matrix (ndof x timesteps)
13 -----
14 -- c Franz Bamer
15
16 function newmark(K,M,C,force,delta_t,x_0,x_0_punkt)
17
18     local K = K:Expand()
19     local M = M:Expand()
20     local C = C:Expand()
21
22     local gamma = 0.5
23     local beta = 0.25
24
25     local x = tmath.Matrix(force)
26     x:SetZero()
27
28     --initial calculations
29     local u_0 = x_0
30     local u_0_punkt = x_0_punkt
31     local u_0_2punkt = tmath.Matrix(u_0)
32     local k_dach = K + C * gamma/(beta*delta_t) + M * 1/(beta*delta_t^2)
33     local inv_k_dach = tmath.Inverse(k_dach)
34     local a = M * 1 / (beta*delta_t) + C * gamma/beta
35     local b = M * 1 / (2*beta) + C * delta_t * (gamma/(2*beta) - 1)
36
37     local f_minus_1 = tmath.ZeroVector(K:Rows())
38
39     -- integration loop
40     for i = 1 , force:Cols() do
41
42         local delta_p = force:GetCols(i-1) - f_minus_1
43         local delta_p_dach = delta_p + a * u_0_punkt + b * u_0_2punkt
44         local delta_u = inv_k_dach * delta_p_dach
45         local delta_u_punkt = delta_u * gamma/(beta*delta_t) - u_0_punkt * gamma/beta +
46             u_0_2punkt * delta_t * (1-gamma/(2*beta))
47         local delta_u_2punkt = delta_u * 1 / (beta * delta_t^2) - u_0_punkt * 1 /
48             (beta*delta_t) - u_0_2punkt * 1 / (2*beta)
49
50         local u_1 = u_0 + delta_u
51         local u_1_punkt = u_0_punkt + delta_u_punkt
52         local u_1_2punkt = u_0_2punkt + delta_u_2punkt
53
54         u_0 = u_1
55         u_0_punkt = u_1_punkt
56         u_0_2punkt = u_1_2punkt
57
58         f_minus_1 = force:GetCols(i-1)
59
60         x[{0,i-1}] = u_1
61         print(i)
62     end
63
64     return x
65 end

```

A.2.3 Numerical demonstration and test of the presented numeric time integration methods for linear systems

```
1 -----
2 ----- linear time integration testfile -----
3 -----
4 -- c Franz Bamer
5
6 -- request for recalculation
7 analytic = true
8 analytic_state_space = true
9 central = true
10 newmark_integration = true
11
12 dofile('function_structure_2d.tng')
13 dofile('function_generate_earthquake_force.tng')
14 dofile('function_rayleigh.tng')
15
16 dofile('function_central_diff_linear.tng')
17 dofile('function_analytic_solution.tng')
18 dofile('function_analytic_solution_state_space.tng')
19 dofile('function_newmark.tng')
20
21
22 dofile('file_input.tng')
23
24 -- generate structure object
25 structure =
26     structure_2d(H,L,anzahl_riegel,anzahl_stiel,anzahl_stockwerke,anzahl_rahmen,false)
27 -- built mass-, stiffness and damping matrix
28 M = structure: SparseMass()
29 K = structure: SparseStiffness()
30 C = rayleigh(K,M)
31
32 -- visualize structure
33 v = graph.Graph3D('structure')
34 tri = structure: Draw(3)
35 v: Triangles(tri)
36 v: Autoscale()
37 v: Rotate(10,1,0,0)
38 v: Render()
39 v: Collada('structure.dae')
40 structure_output = structure: Sketch(0,0,1,0,0)
41 tmath.CBDDrawLines(structure_output, "structure_linear.cb")
42
43 -- load transient excitation
44 earthquake = tmath.MatrixInput('Bam.txt')
45 timeplot = earthquake: GetCols(0)
46 acc_x = earthquake: GetCols(1)
47 acc_y = tmath.Matrix(acc_x)
48 acc_y: SetZero()
49 earthquake_force = generate_earthquake_force(acc_x,acc_y,M,structure)
50 delta_t = timeplot[1] - timeplot[0]
51
52 --initial conditions
53 x_0 = tmath.ZeroVector(M: Rows())
54 x_0_punkt = tmath.ZeroVector(M: Rows())
55
56 if central == true then
57     -- central difference integration
58     X_central_diff = central_diff_linear(K,M,C,earthquake_force,x_0,x_0_punkt,delta_t)
59     tmath.Output(X_central_diff,'X_central.txt')
60 else
61     X_central_diff = tmath.MatrixInput('X_central.txt')
62 end
63
64 if analytic == true then
65     -- piecewise analytic solution
```

```

65 x_analytic = analytic_solution(structure, C, earthquake_force, delta_t, x_0, x_0_punkt,
66 0)
66 tmath.Output(x_analytic,'x_analytic.txt')
67 else
68 x_analytic = tmath.MatrixInput('x_analytic.txt')
69 end
70
71 if analytic_state_space == true then
72 -- piecewise analytic solution in the state space
73 x_analytic_state_space =
74 analytic_solution_state_space(M,K,C, earthquake_force, delta_t, x_0, x_0_punkt)
74 tmath.Output(x_analytic_state_space,'x_analytic_state_space.txt')
75 else
76 x_analytic_state_space = tmath.MatrixInput('x_analytic_state_space.txt')
77 end
78
79 if newmark_integration == true then
80 x_newmark = newmark(K,M,C, earthquake_force, delta_t, x_0, x_0_punkt)
81 tmath.Output(x_newmark,'x_newmark.txt')
82 else
83 x_newmark = tmath.MatrixInput('x_newmark.txt')
84 end
85
86 -- ouptput: horizontal degree of freedom in the left corner
87 selection = structure:GetNodeIndex(anzahl_stiel)
88 selection_matrix = structure:GetAllDisplacements()
89 selection_matrix[{selection-1,0}] = 1
90 selection_vector = structure:ToDofDisplacements(selection_matrix)
91 index = 0
92 for i = 1 , selection_vector:Rows() do
93 if selection_vector[i-1] == 1 then
94 index = i
95 break
96 end
97 end
98
99 -- visualization of the response in the left corner
100 plot = graph.Graph("Response in the left corner", "Bright")
101 plot:AxisLabels("Time [s]", "displacement in x direction [m]")
102 plot:Plot(timeplot:Transpose(), X_central_diff:GetRows(index-1), 2, "central diff")
103 plot:Plot(timeplot:Transpose(), x_analytic:GetRows(index-1), 2, "analytic")
104 plot:Plot(timeplot:Transpose(), x_analytic_state_space:GetRows(index-1), 2, "displacement
105 analytic state space")
106 plot:Plot(timeplot:Transpose(), x_newmark:GetRows(index-1), 2, "newmark")
107
108 -- write output into text files
109 output_file = tmath.ZeroMatrix(2, timeplot:Rows())
110 output_file[{0,0}] = timeplot:Transpose()
111 output_central = output_file
112 output_central[{1,0}] = X_central_diff:GetRows(index-1)
113 tmath.Output(output_central:Transpose(), 'output_central_linear.txt')
114 output_analytic = output_file
115 output_analytic[{1,0}] = x_analytic:GetRows(index-1)
116 tmath.Output(output_analytic:Transpose(), 'output_analytic_linear.txt')
117 output_analytic_state_space = output_file
118 output_analytic_state_space[{1,0}] = x_analytic_state_space:GetRows(index-1)
119 tmath.Output(output_analytic_state_space:Transpose(), 'output_analytic_state_space_linear.txt')
120 output_newmark = output_file
121 output_newmark[{1,0}] = x_newmark:GetRows(index-1)
122 tmath.Output(output_newmark:Transpose(), 'output_newmark_linear.txt')

```

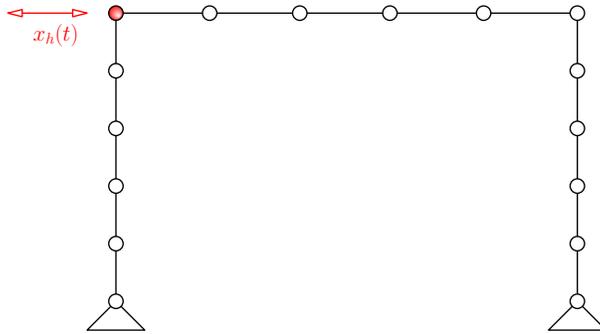


Figure 77: Linear test frame structure

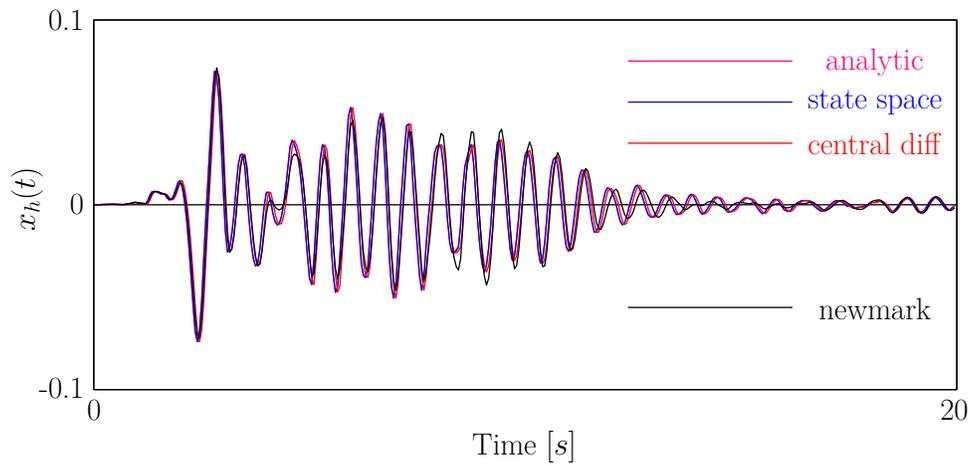


Figure 78: Response: Horizontal displacement of the left corner

A.3 Implementation of numeric time integration methods for nonlinear systems

A.3.1 Implementation of the central difference algorithm for nonlinear systems

```
1 -----
2 -----central diff algorithm for nonlinear systems -----
3 -- input:
4 -- structure ... object with structural information
5 -- M ... mass matrix (sparse)
6 -- C ... damping matrix (sparse)
7 -- force ... force matrix (ndof x timesteps)
8 -- x_0 ... initial displacement
9 -- x_0_punkt ... initial velocity
10 -- delta_t ... time resolution of the excitation
11 -- output:
12 -- x_erg ... response (ndof x timesteps)
13 -----
14 -- c Franz Bamer
15
16 function time_integratiion_central_diff_nonlinear(structure,M,C,force,x_0,x_0_punkt,delta_t)
17
18     local dt = 0.0001
19
20     --initial calculations
21     local U = x_0
22     structure:SetDofDisplacements(U)
23     structure:GlobalUpdate()
24     local U_0_2_punkt = - structure:GlobalResForce() - C:Dot(x_0_punkt)
25     local U_minus1 = U - x_0_punkt*dt + U_0_2_punkt*dt^2 * 0.5
26
27     local Keff = M:Add(C, 1/dt^2, 1/dt/2)
28
29     --constants and storage reservations
30     local a0 = 1/dt^2
31     local a1 = 1/(2*dt)
32     local a2 = 2/dt^2
33     local a3 = dt^2/2
34     local t = delta_t
35     local i = 1
36     local time_ = 0
37     local timeplot_ = tmath.ZeroMatrix(1,force:Cols())
38     local x_erg = tmath.ZeroMatrix(U:Rows(),force:Cols())
39     local F = tmath.ZeroVector(force:Rows())
40     local R = tmath.ZeroVector(M:Rows())
41     local U1 = tmath.ZeroVector(M:Rows())
42
43     -- integration loop
44     while i < force:Cols() do
45         if (t >= delta_t-0.000001 and t <= delta_t+0.000001) then
46             F = force:GetCols(i-1)
47             t = 0
48             x_erg[{0,i-1}] = U
49             i = i + 1
50             print(i)
51         end
52         collectgarbage()
53         R = F - structure:GlobalResForce() + M:Dot(U*2/dt^2 - U_minus1/dt^2) +
54             C:Dot(U_minus1/2/dt)
55         U1 = Keff:Solve(R)
56         structure:SetDofDisplacements(U1)
57         structure:GlobalUpdate()
58         U_minus1 = U
59         U = U1
60         t = t + dt
61         if (U[{3,0}]>10000) then
62             print('time step exceeds critical value ---> unstable')
63             break
64         end
65     end
66 end
```

```

63     end
64 end
65
66 return x_erg
67 end

```

A.3.2 Implementation of the Newmark algorithm for nonlinear systems

```

1 -----
2 ----- function nonlinear Newmark -----
3 -- input:
4 -- structure ... object with structural information
5 -- K ... stiffness matrix (sparse)
6 -- M ... mass matrix (sparse)
7 -- C ... damping matrix (sparse)
8 -- x_0 ... initial displacement
9 -- x_0_punkt ... initial velocity
10 -- delta_t ... time resolution of the excitation
11 -- force ... force matrix (ndof x timesteps)
12 -- output:
13 -- x_erg ... response (ndof x timesteps)
14 -----
15 -- c Franz Bamer
16
17 function newmark_nonlinear(structure,K,M,C,x_0,x_0_punkt,delta_t,force)
18
19     local X = tmath.ZeroMatrix(x_0:Rows(),force:Cols())
20
21     local dt = delta_t
22
23     local a0 = 4/dt^2
24     local a1 = 2/dt
25     local a2 = 4/dt
26     local a3 = 1
27     local a4 = 1
28     local a5 = 0
29     local a6 = dt/2
30     local a7 = dt/2
31
32     -- Effective "stiffness" for Newmark method
33     local Keff = K:Add(M, a0):Add(C,a1)
34     control.Interactive(true)
35     print('effective stiffness matrix generated')
36     control.Interactive(false)
37
38     local U = x_0
39     local V = x_0_punkt
40     local A = tmath.Matrix(V)
41     -- Newmark loop NOTE: this implements the Newton-Raphson iteration using the initial
42     -- effective stiffness within one time step
43     for i=0,force:Cols()-1 do
44
45         local R1 = force:GetCols(i) + M:Dot(A + V*a2 + U*a0) + C:Dot(V + U*a1)
46         local U1 = Keff:Solve(R1)
47         for k=0,5 do
48             structure:SetDofDisplacements(U1)
49             local R = R1 - structure:GlobalResForce() - M:Dot(U1*a0) - C:Dot(U1*a1)
50             local Rnorm = tmath.Norm(R)
51             if (Rnorm < 1) then break end
52             local DU = Keff:Solve(R)
53             U1 = U1+DU
54         end
55         print(i)
56         local V1 = U1*a1 - U*a1 - V
57         local A1 = V1*a1 - V*a1 - A
58         U = tmath.Matrix(U1)
59         V = tmath.Matrix(V1)

```

```

59     A = tmath.Matrix(A1)
60     structure:SetDofDisplacements(U)
61     structure:GlobalUpdate()
62     X[{0,i}] = U
63 end
64
65 return X
66 end

```

A.3.3 Numerical demonstration and test of the presented numeric time integration methods for nonlinear systems

```

1 -----
2 ----- nonlinear time integration tests -----
3 -----
4
5 -- request of the central difference calculation procedure
6 central = true
7 -- request of the newmark calculation procedure
8 newmark_integration = true
9
10 -- load function to create the structure object
11 dofile('function_structure_2d.tng')
12 -- load function to
13 dofile('function_generate_earthquake_force.tng')
14 dofile('function_rayleigh.tng')
15
16 dofile('function_time_integration_central_diff_nonlinear.tng')
17 dofile('function_newmark_nonlinear.tng')
18
19 dofile('file_input.tng')
20
21 -- create object with structural and geometrical information
22 structure =
23     structure_2d(H,L,anzahl_riegel,anzahl_stiel,anzahl_stockwerke,anzahl_rahmen,true)
24 -- create mass and stiffness matrix (sparse)
25 M = structure:SparseMass()
26 K = structure:SparseStiffness()
27 -- create Rayleigh damping matrix (sparse)
28 C = rayleigh(K,M)
29
30 -- graphical visualization of the structure
31 v = graph.Graph3D('structure')
32 tri = structure:Draw(3)
33 v:Triangles(tri)
34 v:Autoscale()
35 v:Rotate(10,1,0,0)
36 v:Render()
37
38 -- load transient excitation
39 earthquake = tmath.MatrixInput('imperial_valley.txt')
40 timeplot = earthquake:GetCols(0)
41 acc_x = earthquake:GetCols(1)
42 acc_y = tmath.Matrix(acc_x)
43 acc_y:SetZero()
44 earthquake_force = generate_earthquake_force(acc_x,acc_y,M,structure)
45 delta_t = timeplot[1] - timeplot[0]
46
47 --initial conditions
48 x_0 = tmath.ZeroVector(M:Rows())
49 x_0_punkt = tmath.ZeroVector(M:Rows())
50
51 if central == true then
52     -- central difference integration algorithm
53     X_central_diff_nonlinear = time_integration_central_diff_nonlinear(structure, M, C,
54         earthquake_force, x_0, x_0_punkt, delta_t)
55     tmath.Output(X_central_diff_nonlinear,'X_central_nonlinear.txt')

```

```

54 else
55     X_central_diff_nonlinear = tmath.MatrixInput('X_central_nonlinear.txt')
56 end
57
58 if newmark_integration == true then
59     -- newmark time integration algorithm
60     x_newmark_nonlinear = newmark_nonlinear(structure, K, M, C, x_0, x_0_punkt, delta_t,
        earthquake_force)
61     tmath.Output(x_newmark_nonlinear, 'x_newmark_nonlinear.txt')
62 else
63     x_newmark_nonlinear = tmath.MatrixInput('x_newmark_nonlinear.txt')
64 end
65
66 -- ouptput: horizontal degree of freedom in the left corner
67 selection = structure:GetNodeIndex(anzahl_stiel)
68 selection_matrix = structure:GetAllDisplacements()
69 selection_matrix[{selection-1,0}] = 1
70 selection_vector = structure:ToDoDisplacements(selection_matrix)
71 index = 0
72 for i = 1 , selection_vector:Rows() do
73     if selection_vector[i-1] == 1 then
74         index = i
75         break
76     end
77 end
78
79 -- visualization of the response
80 plot = graph.Graph("Response in the left corner", "Bright")
81 plot:AxisLabels("Time [s]", "displacement in x direction [m]")
82 plot:Plot(timeplot:Transpose(), X_central_diff_nonlinear:GetRows(index-1), 2, "central
        diff")
83 plot:Plot(timeplot:Transpose(), x_newmark_nonlinear:GetRows(index-1), 2, "newmark")

```

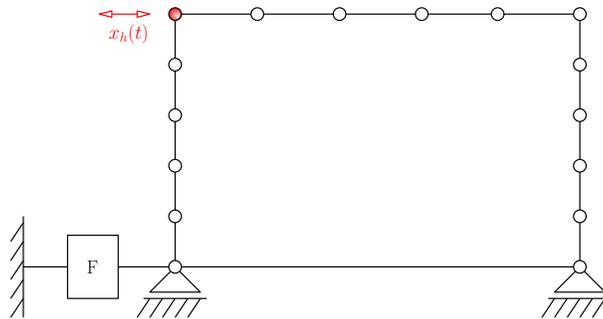


Figure 79: Nonlinear test structure with the friction isolator presented in section 4.3

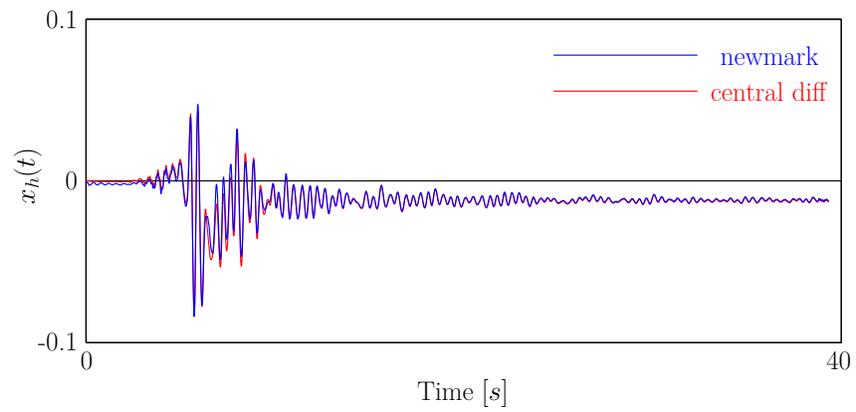


Figure 80: Response: Horizontal displacement of the left corner

A.4 Implementation of mechanical models

In order to be able to vary the number of DOF of the structures the implementation includes the possibility to change geometrical properties, e.g. number of stories, numbers of frames, etc. two implementation codes are presented.

A.4.1 Two dimensional frame structure

```
1 -----
2 ----- 2 dimensional frame structure -----
3 -- input parameters:
4 -- H ... height of one storey
5 -- L ... length of one frame
6 -- anzahl_riegel ... number of elements per beam
7 -- anzahl_stiel ... number of elements per column
8 -- anzahl_stockwerke ... number of storeys
9 -- anzahl_rahmen ... number of frames
10 -- nonlinear ... request for nonlinear friction elements (boolean)
11 -----
12 -- c Franz Bamer
13
14 function
15     structure_2d(H,L,anzahl_riegel,anzahl_stiel,anzahl_stockwerke,anzahl_rahmen,nonlinear)
16
17     -- define parameters for friction elements
18     local k0 = 4000000
19     local k1 = 400000
20     local fric = 10000
21
22     local s = fem.Structure("Rahmensystem")
23
24     -- create material (linearized concrete)
25     local mat = s.AddMaterial(1026, 'LINEAR_ELASTIC')
26     mat:SetData(tmeth.Matrix({{3.57 * 10000000000, 0.3, 2500}}))
27
28     -- create nonlinear friction element
29     local mat = fem.Material_FPS(1028)
30     mat:SetData(tmeth.Matrix({{k1, k0, fric, 1}}))
31     s:AddMaterial(mat)
32
33     local sec = s.AddSection(2048, "RECT", 0)
34     sec:SetData(tmeth.Matrix({{ b, b}}))
35
36     local sec = s.AddSection(2049, "RECT", 0)
37     sec:SetData(tmeth.Matrix({{ b, 1.5*b}}))
38
39     local sec = s.AddSection(2050, 'RECT', 0)
40     sec:SetData(tmeth.Matrix({{ b, 3*b}}))
41     sec:SetColor(tmeth.Matrix({{255,0,0,255}}))
42
43     -- create reference_node -----
44     local number_ref_node = 10000000000
45     s:AddNodes(tmeth.Matrix( {{ number_ref_node, -L/2, -L/2, 0 }} ))
46     local n = s.GetNode(10000000000)
47     n:SetAvailDof( tmeth.Matrix( {{ 0,0,0,0,0,0 }} ))
48     -----
49
50     -- create column elements
51     local nodes = tmeth.ZeroMatrix(anzahl_stiel*anzahl_stockwerke+1,2)
52     for k = 1, anzahl_rahmen+1 do
53         for i = 1, anzahl_stiel*anzahl_stockwerke+1 do
54             nodes[[i-1,0]] = (k-1)*L
55             nodes[[i-1,1]] = H / anzahl_stiel*(i-1)
56         end
57     for i = 1, nodes:Rows() do
58         s:AddNodes(tmeth.Matrix( {{ (k-1)*(anzahl_stiel*anzahl_stockwerke+1) + i,
59             nodes[[i-1,0]], nodes[[i-1,1]], 0}} ))
```

```

58 n=s:GetNode((k-1)*(anzahl_stiel*anzahl_stockwerke+1) + i) --fuer setavaildof zugriff
    auf den erstellten knoten
59 n:SetAvailDof(tmath.Matrix({{1, 1, 0, 0, 0, 1}})) -- ebenes Problem!
60 end
61 for i = 1, anzahl_stiel*anzahl_stockwerke do
62 s:AddElements("RECT", 1026, 2048, tmath.Matrix( {(k-1)*anzahl_stiel*anzahl_stockwerke
    + i, (k-1)*(anzahl_stiel*anzahl_stockwerke+1)+i,
    (k-1)*(anzahl_stiel*anzahl_stockwerke+1)+i+1, number_ref_node}))
63 end
64 end
65 local non = (anzahl_stiel*anzahl_stockwerke+1) * (anzahl_rahmen+1)
66 local noe = (anzahl_stiel*anzahl_stockwerke) * (anzahl_rahmen+1)
67
68 -- create beam elements
69 for k = 1, anzahl_rahmen do
70     for i = 1, anzahl_stockwerke do
71         s:AddNodes(tmath.Matrix( {{ non + i, (k-1)* L + L / anzahl_riegel, i*H, 0}} ))
72         n=s:GetNode(non + i) --fuer setavaildof zugriff auf den erstellten knoten
73         n:SetAvailDof(tmath.Matrix({{1, 1, 0, 0, 0, 1}})) -- ebenes Problem!
74         s:AddElements("RECT", 1026, 2049, tmath.Matrix( {{noe + i,
            (anzahl_stiel*anzahl_stockwerke+1)*(k-1) + i*anzahl_stiel+1, non+i,
            number_ref_node}))
75     end
76     non = non + anzahl_stockwerke
77     noe = noe + anzahl_stockwerke
78 end
79
80 local previous_node_first = non-(anzahl_stockwerke*anzahl_rahmen)+1
81 for j = 1, anzahl_rahmen do
82     for k = 1, anzahl_stockwerke do
83         for i = 1, anzahl_riegel-2 do
84             if i == 1 then
85                 previous_node = previous_node_first + (j-1)*anzahl_stockwerke+(k-1)
86             end
87             s:AddNodes(tmath.Matrix( {{ non+i, (j-1)*L + L / anzahl_riegel *(i+1), k*H, 0}} ))
88             n=s:GetNode(non+i) --fuer setavaildof zugriff auf den erstellten knoten
89             n:SetAvailDof(tmath.Matrix({{1, 1, 0, 0, 0, 1}})) -- ebenes Problem!
90             s:AddElements("RECT", 1026, 2049, tmath.Matrix( {{noe + i, previous_node, non+i,
                number_ref_node}))
91             previous_node = non+i
92         end
93         non = non + anzahl_riegel-2
94         noe = noe + anzahl_riegel-2
95     end
96 end
97
98 local previous_node = non - (anzahl_stockwerke*anzahl_rahmen*(anzahl_riegel-2)) +
    anzahl_riegel-2
99 local next_node = anzahl_stockwerke*anzahl_stiel+1 + anzahl_stiel + 1
100 for k = 1, anzahl_rahmen do
101     for i = 1, anzahl_stockwerke do
102         s:AddElements("RECT", 1026, 2049, tmath.Matrix( {{noe + i, previous_node, next_node,
            number_ref_node}))
103         previous_node = previous_node + anzahl_riegel-2
104         next_node = next_node + anzahl_stiel
105     end
106     next_node = next_node+1
107 end
108
109 if nonlinear == true then
110     local noe_nonlinear = 100000000
111     local previous_node = 1
112     local next_node = anzahl_stiel*anzahl_stockwerke+1 + 1
113     for i = 1, anzahl_rahmen do
114         s:AddElements("RECT", 1026, 2049, tmath.Matrix( {{noe_nonlinear + i, previous_node,
            next_node, number_ref_node}))
115         previous_node = next_node
116         next_node = next_node + anzahl_stiel*anzahl_stockwerke+1
117     end
118     noe_nonlinear = noe_nonlinear + anzahl_rahmen

```

```

119  -- add node for fps element
120  local number_fps_node = 500005
121  s:AddNodes(tmath.Matrix( {{ number_fps_node,    -L/2, 0, 0 }} ))
122  local n = s:GetNode(number_fps_node)
123  n:SetAvailDof( tmath.Matrix( {{ 0,0,0,0,0,0 }} ))
124  -- add fps element
125  s:AddElements("FPS", 1028, 2050,tmath.Matrix( {{ noe_nonlinear + 1, number_fps_node, 1,
126      number_ref_node }} ))
127
128  local node_bearing = 1
129  for i = 1, anzahl_rahmen + 1 do
130      n = s:GetNode(node_bearing)
131      n:SetAvailDof( tmath.Matrix( {{ 1,0,0,0,0,1 }} ))
132      node_bearing = node_bearing + anzahl_stiel*anzahl_stockwerke + 1
133  end
134  else
135      local node_bearing = 1
136      for i = 1, anzahl_rahmen + 1 do
137          n = s:GetNode(node_bearing)
138          n:SetAvailDof( tmath.Matrix( {{ 0,0,0,0,0,1 }} ))
139          node_bearing = node_bearing + anzahl_stiel*anzahl_stockwerke + 1
140      end
141  end
142  local ndof = s:GlobalDof()  ---- this method has to be executed before the calculation
143      procedure
144  return s
145 end

```

A.4.2 Three dimensional frame structure

In the same manner as for the two-dimensional frame structure a three-dimensional test object is generated, where the number of frames in x - and y -direction can be changed as the number of elements per column and the number of elements per frame in x - and y -direction:

```

1 -----
2 ----- generate 3d frame structure -----
3 -----
4 -- c Franz Bamer
5
6 -- define number of nodes for the columns
7 -- note: this function is only called in frame_structure_3d
8 function column_basis_nodes_definition(anzahl_rahmen,anzahl_riegel,L,T)
9     -- nodes from column elements to plate elements
10    local column_basis_nodes = tmath.ZeroMatrix((anzahl_rahmen+1)*(anzahl_plane+1),4)
11    local counter = 1
12    for j = 1 , anzahl_plane +1 do
13        for i = 1 , anzahl_rahmen + 1 do
14            column_basis_nodes[{counter-1,0}] = (anzahl_riegel*anzahl_rahmen
15                +1)*anzahl_riegel*(j-1) + anzahl_riegel*(i-1) + 1
16            column_basis_nodes[{counter-1,1}] = (i-1)*L
17            column_basis_nodes[{counter-1,2}] = (j-1)*T
18            column_basis_nodes[{counter-1,3}] = j
19            counter = counter + 1
20        end
21    end
22    return column_basis_nodes
23 end
24
25 -----
26 -- start main function to generate the structure -----
27 -- H ... heigth of one storey
28 -- L ... length of one frame in x direction
29 -- T ... length of one frame in y direction
30 -- anzahl_riegel ... number of nodes in x and y direction

```

```

31 -- anzahl_stiel ... number of elements per column
32 -- anzahl_sockwerke ... number of storeys
33 -- anzahl_rahmen ... number of frames in x direction
34 -- anzahl_plane ... number of frames in y direction
35 -- b ... side length of the cross section of the beam elements
36 -- h ... height of the shell plate elements
37 -----
38 function frame_structure_3_d(H, L, T, anzahl_riegel, anzahl_stiel, anzahl_stockwerke,
    anzahl_rahmen, anzahl_plane, b, h)
39
40     local s = fem.Structure("Rahmensystem 3d")
41
42     local mat = s:AddMaterial(1026, 'LINEAR_ELASTIC')
43     mat:SetData(tmeth.Matrix({{3.57 * 10000000000, 0.3, 2500}}))
44
45     local sec = s:AddSection(2048, "RECT", 0)
46     sec:SetData(tmeth.Matrix({{ b, b}}))
47
48     local sec = s:AddSection(2049, "RECT", 0)
49     sec:SetData(tmeth.Matrix({{ b, b}}))
50     sec:SetColor(tmeth.Matrix({{0,200,200,200}}))
51
52     local sec = s:AddSection(2, "SHELL", 0 --[[geometrisch linear--]] )
53     sec:SetData(tmeth.Matrix({{h}}))
54     sec:SetColor(tmeth.Matrix({{0,0,200,200}}))
55
56     local nodes_per_row = anzahl_riegel*anzahl_rahmen+1
57     local nodes_per_column = anzahl_riegel*anzahl_plane+1
58     local non_per_storey = nodes_per_row*nodes_per_column
59     local element_length = L/anzahl_riegel
60     local element_width = T/anzahl_riegel
61
62     local column_basis_nodes = column_basis_nodes_definition(anzahl_rahmen, anzahl_riegel,
        L, T)
63
64     local reference_nodes = tmeth.ZeroMatrix(anzahl_plane+1,1)
65     for plane = 1, reference_nodes:Rows() do
66         -- reference_node -----
67         local reference_node = 1000000000 + plane
68         s:AddNodes(tmeth.Matrix( {{ reference_node, -L/2, -L/2, -(plane-1)*T }} ))
69         local n = s:GetNode(reference_node)
70         n:SetAvailDof( tmeth.Matrix( {{ 0,0,0,0,0 }} ))
71         -----
72         reference_nodes[[plane-1,0]] = reference_node
73     end
74
75     local elements_per_row = nodes_per_row-1
76     local elements_per_column = nodes_per_column-1
77     local noe_per_storey = elements_per_row*elements_per_column*2
78
79     local non = 0
80     local noe = 0
81     -- plate elements
82     for stockwerke = 1, anzahl_stockwerke do
83         local counter = 1
84         local x_node = 0
85         for j = 1, nodes_per_column do
86             for i = 1, nodes_per_row do
87                 local y_node = (j-1)* element_width
88                 local number_of_node = (stockwerke-1)*non_per_storey + (j-1)*nodes_per_row + i
89                 --number_of_node = non + (j-1)*nodes_per_row + i
90                 s:AddNodes(tmeth.Matrix( {{ number_of_node, x_node, stockwerke*H, -y_node }} ))
91                 local n = s:GetNode(number_of_node) --fuer setavaildof zugriff auf den erstellten
                    knoten
92                 n:SetAvailDof(tmeth.Matrix({{1, 1, 1, 1, 1, 1}})) -- in jede richtung frei drehbar
93                 x_node = x_node + element_length
94                 counter = counter + 1
95             end
96         end
97         x_node = 0
98     end

```

```

98     local counter = 1
99     for j = 1 , elements_per_column do
100         for i = 1 , elements_per_row do
101             s:AddElements("TRIANGLE3N", 1026, 2, tmath.Matrix({{ noe + counter , (stockwerke -
                1)*non_per_storey + (j-1)*(elements_per_row+1) + i , (stockwerke -
                1)*non_per_storey + (j-1)*(elements_per_row+1) + i+1 , (stockwerke -
                1)*non_per_storey + j*(elements_per_row+1) + i }}))
102             counter = counter + 1
103             s:AddElements("TRIANGLE3N", 1026, 2, tmath.Matrix({{ noe + counter , (stockwerke -
                1)*non_per_storey + (j-1)*(elements_per_row+1) + i+1 , (stockwerke -
                1)*non_per_storey + j*(elements_per_row+1) + i , (stockwerke -
                1)*non_per_storey + j*(elements_per_row+1) + i+1 }}))
104             counter = counter + 1
105         end
106     end
107     local noe = noe + counter
108
109 end
110
111 local non = non_per_storey*anzahl_stockwerke
112 local noe = noe_per_storey*anzahl_stockwerke
113 local non_columns_per_storey = (anzahl_stiel-1)*column_basis_nodes:Rows()
114 local noe_columns_per_storey = (anzahl_stiel-2)*column_basis_nodes:Rows()
115
116 -- beam elements
117 local counter
118 for stockwerke = 1 , anzahl_stockwerke do
119
120     local column_length = H/anzahl_stiel
121     for columns = 1 , column_basis_nodes:Rows() do
122         for i = 1 , anzahl_stiel - 1 do
123             local number_of_node = non + (stockwerke-1)*non_columns_per_storey +
                (columns-1)*(anzahl_stiel-1) + i
124             s:AddNodes(tmath.Matrix( {{ number_of_node , column_basis_nodes[{{columns-1,1}},
                (stockwerke-1)*H + column_length*i , -column_basis_nodes[{{columns-1,2}}]}} ))
125             local n=s:GetNode(number_of_node) --fuer setavaildof zugriff auf den erstellten
                knoten
126             n:SetAvailDof(tmath.Matrix({{1, 1, 1, 1, 1, 1}})) -- in jede richtung frei drehbar
127         end
128     end
129     counter = 1
130     for columns = 1 , column_basis_nodes:Rows() do
131         for i = 1 , anzahl_stiel-2 do
132             s:AddElements("RECT", 1026, 2048, tmath.Matrix( {{ noe + counter ,
                non+(stockwerke-1)*non_columns_per_storey+(anzahl_stiel-1)*(columns-1) + i ,
                non + (stockwerke-1)*non_columns_per_storey +
                (anzahl_stiel-1)*(columns-1)+i+1 , reference_nodes[{{
                column_basis_nodes[{{columns-1,3}}]-1,0 }}] }}))
133             counter = counter + 1
134         end
135         s:AddElements("RECT", 1026, 2048, tmath.Matrix( {{ noe + counter , non
                +(stockwerke-1)*non_columns_per_storey+
                (anzahl_stiel-1)*(columns-1)+anzahl_stiel-1 , (stockwerke-1)*non_per_storey +
                column_basis_nodes[{{columns-1,0}}, reference_nodes[{{
                column_basis_nodes[{{columns-1,3}}]-1,0 }}] }}))
136         counter = counter + 1
137     end
138
139 end
140 noe = noe + counter
141 counter = 1
142 for stockwerke = 2 , anzahl_stockwerke do
143     for columns = 1 , column_basis_nodes:Rows() do
144         s:AddElements("RECT", 1026, 2048, tmath.Matrix( {{ noe + counter , non
                +(stockwerke-1)*non_columns_per_storey+ (anzahl_stiel-1)*(columns-1) + 1 ,
                (stockwerke-2)*non_per_storey + column_basis_nodes[{{columns-1,0}},
                reference_nodes[{{ column_basis_nodes[{{columns-1,3}}]-1,0 }}] }}))
145         counter = counter + 1
146     end
147 end

```

```

148 noe = noe + counter
149 local running_number_nodes = non_per_storey*anzahl_stockwerke +
    non_columns_per_storey*anzahl_stockwerke
150
151 for columns = 1 , column_basis_nodes:Rows() do
152     local number_of_node = running_number_nodes + columns
153     s:AddNodes(tmath.Matrix( {{ number_of_node, column_basis_nodes[columns-1,1]}, 0,
        -column_basis_nodes[columns-1,2]} } ))
154     local n=s:GetNode(number_of_node) --fuer setavaildof zugriff auf den erstellten knoten
155     n:SetAvailDof(tmath.Matrix({{0, 0, 0, 1, 1, 1}})) -- in jede richtung frei drehbar
        aber nicht verschiebbar (gelenkig gelagert)
156 end
157 counter = 1
158 for columns = 1 , column_basis_nodes:Rows() do
159     s:AddElements("RECT", 1026, 2048, tmath.Matrix( {{ noe + counter, non +
        (anzahl_stiel-1)*(columns-1) + 1, running_number_nodes + columns,
        reference_nodes[ column_basis_nodes[columns-1,3]-1,0 ] } } ))
160     counter = counter + 1
161 end
162
163 local ndof = s:GlobalDof() ---- always execute this method
164
165 return s
166 end
167
168
169 -----
170 ---- find dof for initial velocity and output -----
171 -----
172 function frame_structure_3_d_node_selection(structure, number_of_storey, number_column_x,
    number_column_y, anzahl_rahmen, anzahl_riegel, L, T)
173
174     column_basis_nodes = column_basis_nodes_definition(anzahl_rahmen,anzahl_riegel,L,T)
175     local nodes_per_row = anzahl_riegel*anzahl_rahmen+1
176     local nodes_per_column = anzahl_riegel*anzahl_plane+1
177     local non_per_storey = nodes_per_row*nodes_per_column
178
179     local number_column = (number_column_y - 1) * (anzahl_rahmen+1) + number_column_x
180
181     local selection = (number_of_storey-1)*non_per_storey + column_basis_nodes[
        number_column-1,0 ]
182
183     local auswahl_matrix = structure:GetAllDisplacements()
184     auswahl_matrix:SetZero()
185     local selection = structure:GetNodeIndex(selection)
186     auswahl_matrix[selection,0] = 1
187     local velo = structure:ToDofDisplacements(auswahl_matrix)
188     return velo
189 end
190
191 -----
192 -- find number of element for force output -----
193 -----
194 function frame_structure_3_d_element_selection(anzahl_riegel, anzahl_stiel,
    anzahl_stockwerke, anzahl_rahmen, anzahl_plane)
195
196     local nodes_per_row = anzahl_riegel*anzahl_rahmen+1
197     local nodes_per_column = anzahl_riegel*anzahl_plane+1
198     local noe = (nodes_per_row-1)*(nodes_per_column-1)*2
199     noe = noe*anzahl_stockwerke
200     noe = noe + anzahl_stiel
201     return noe
202 end

```

A.4.3 Three dimensional building construction - Las Vegas building

The three dimensional building construction is composed of linear triangle shell / plate elements and nonlinear beam elements. Nonlinearities are realized by a bilinear stress - strain curve in axial direction with strain hardening. In this work this structure is called Las Vegas building as the construction is motivated by the appearance of some hotels and casinos in Las Vegas. Two functions are presented: one function for the generation of the structure and one function for finding the desired DOF of the response after the time integration procedure.

```
1 -----
2 ----- Las Vegas building -----
3 -- input:
4 -- h_storey ... height of one storey
5 -- number_of_stories ... number of stories
6 -- output:
7 -- structure ... object with all structural information
8 -----
9 -- text files loaded:
10 -- load coordinates of nodes for one storey: nodes_plates.txt
11 -- load element identification: elements_plates.txt
12 -- load column nodes identification: column_elements.txt
13 -----
14 -- c Franz Bamer
15
16 function building_las_vegas_w_c(h_storey,number_of_stories)
17
18     local s = fem.Structure("building construction")
19     local nodes_plates = tmath.MatrixInput('structure_and_earthquake/ functions_structure/
20     3d_geometrical_data_round_w_c/ nodes_plates.txt')
21     local elements_plates = tmath.MatrixInput('structure_and_earthquake/
22     functions_structure/ 3d_geometrical_data_round_w_c/ elements_plates.txt')
23     local column_matrix = tmath.MatrixInput('structure_and_earthquake/
24     functions_structure/3d_geometrical_data_round_w_c/ column_elements.txt')
25
26     local mat = s:AddMaterial(302, "LINEAR_ELASTIC")
27     mat:SetData(tmath.Matrix({{3.57 * 10000000000, 0.3, 2500}})) -- linearized concrete
28
29     -- define plastic material
30     local mm = s:AddMaterial(8, "ELASTIC_PLASTIC_1D")
31     mm:SetData(tmath.Matrix({{2.1e11, .3, 7850, 2.4e8, .01}}))
32
33     local sec = s:AddSection(301, "SHELL", 0 --[[geometrisch linear--]])
34     sec:SetData(tmath.Matrix({{0.3}}))
35     sec:SetColor(tmath.Matrix({{0,0,200,200}}))
36
37     local sec = s:AddSection(2, "SHELL", 0 --[[geometrisch linear--]])
38     sec:SetData(tmath.Matrix({{0.2}}))
39     sec:SetColor(tmath.Matrix({{0,0,200,200}}))
40
41     local sec = s:AddSection(1, "SHELL", 0 --[[geometrisch linear--]])
42     sec:SetData(tmath.Matrix({{0.4}}))
43     sec:SetColor(tmath.Matrix({{200,200,200,200}}))
44
45     local sec = s:AddSection(3, "RECT", 0)
46     sec:SetData(tmath.Matrix({{0.25,0.25}}))
47     sec:SetColor(tmath.Matrix({{0,0,200,200}}))
48
49     for storey = 1, number_of_stories do
50         for i=1,nodes_plates:Rows() do
51             s:AddNodes(tmath.Matrix( {{ (storey-1)*nodes_plates:Rows() + i, x[i-1], y[i-1],
52             (storey-1)* h_storey }} ))
53             local n=s:GetNode(i) --fuer setavaildof zugriff auf den erstellten knoten
54             n:SetAvailDof(tmath.Matrix({{1, 1, 1, 1, 1, 1}})) -- in jede richtung frei drehbar
```

```

54     print('plates nodes', storey, i)
55     end
56
57     for i = 1, elements_plates:Rows() do
58         s:AddElements("TRIANGLE3N", 302, 1, tmath.Matrix({{
            (storey-1)*elements_plates:Rows() + i, (storey-1)*nodes_plates:Rows() +
            elements_plates[{i-1,1}], (storey-1)*nodes_plates:Rows() +
            elements_plates[{i-1,2}], (storey-1)*nodes_plates:Rows() +
            elements_plates[{i-1,3}]}}))
59     end
60
61     end
62
63     local non = nodes_plates:Rows()*number_of_stories
64     local noe = elements_plates:Rows()*number_of_stories
65     local non_per_column = 3
66     local length_column_element = h_storey / non_per_column
67     local number_reference_node = 1000000000
68     s:AddNodes(tmath.Matrix( {{ number_reference_node, 0, 0, 0 }} ))
69     local n=s:GetNode(number_reference_node)
70     n:SetAvailDof(tmath.Matrix({{0, 0, 0, 0, 0, 0}})) -- fest
71
72     local non_vorher
73     local noe_vorher
74     for k = 1 , number_of_stories-1 do
75         for i = 1 , column_matrix:Rows() do
76             non_vorher = non
77             noe_vorher = noe
78             s:AddNodes( tmath.Matrix( {{ non_vorher+i, x[ column_matrix[{i-1,1}]-1 ] , y[
                column_matrix[{i-1,1}]-1 ] , (k-1)*h_storey + length_column_element }} ))
79             local n=s:GetNode(non_vorher+i)
80             n:SetAvailDof(tmath.Matrix({{1, 1, 1, 1, 1, 1}}))
81             s:AddElements("RECT", 8, 3, tmath.Matrix({{ noe_vorher+i, (k-1)*nodes_plates:Rows() +
                column_matrix[{i-1,1}], non + i, number_reference_node }}))
82         end
83         non = non + column_matrix:Rows()
84         noe = noe + column_matrix:Rows()
85         for i = 1 , column_matrix:Rows() do
86             s:AddNodes( tmath.Matrix( {{ non+i, x[ column_matrix[{i-1,1}]-1 ] , y[
                column_matrix[{i-1,1}]-1 ] , (k-1)*h_storey + length_column_element*2 }} ))
87             local n=s:GetNode(non+i)
88             n:SetAvailDof(tmath.Matrix({{1, 1, 1, 1, 1, 1}}))
89             s:AddElements("RECT", 8, 3, tmath.Matrix({{ noe+i, non_vorher+i, non + i,
                number_reference_node }}))
90         end
91         noe = noe + column_matrix:Rows()
92         for i = 1 , column_matrix:Rows() do
93             s:AddElements("RECT", 8, 3, tmath.Matrix({{ noe+i, non+i, k*nodes_plates:Rows() +
                column_matrix[{i-1,1}], number_reference_node }}))
94         end
95         non = non + column_matrix:Rows()
96         noe = noe + column_matrix:Rows()
97     end
98
99
100     -- boundary conditions
101
102     for i = 1,nodes_plates:Rows() do
103         local n = s:GetNode(i)
104         n:SetAvailDof( tmath.Matrix( {{ 0,0,0,0,0,0 }} ))
105     end
106
107     local ndof = s:GlobalDof() -- this method has to be executed before the calculation
        procedure
108
109     return s
110 end

```

Additionally a function is written, which localizes the degrees of freedom for the response output. The localizing function of the Monte Carlo calculations is presented as representative example for this type of functions.

```

1 -----
2 ----- Las Vegas building response output -----
3 -- input:
4 -- structure ... object containing structural information
5 -- output:
6 -- select_vector ... vector with selected degrees of freedom
7 -----
8 -- note: localize the dof in x direction of column 6 in
9 --       every storey
10 -----
11 -- c Franz Bamer
12
13 function get_displacements(structure)
14
15     local auswahl_matrix = structure:GetAllDisplacements()
16     auswahl_matrix:SetZero()
17     local select1 = 96
18     local select2 = 181
19     local select3 = 266
20     local select4 = 351
21     local select5 = 436
22     local select6 = 521
23     local select7 = 606
24     select1 = structure:GetNodeIndex(select1) -- intern counter begins with 0 !!!!!!!
25     auswahl_matrix[{select1,0}] = 1
26     local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
27     for i = 1 , auswahl_vector:Rows() do
28         if auswahl_vector[i-1] == 1 then
29             select1 = i
30         end
31     end
32
33     auswahl_matrix:SetZero()
34     auswahl_vector:SetZero()
35     select2 = structure:GetNodeIndex(select2)
36     auswahl_matrix[{select2,0}] = 1
37     local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
38     for i = 1 , auswahl_vector:Rows() do
39         if auswahl_vector[i-1] == 1 then
40             select2 = i
41         end
42     end
43
44     auswahl_matrix:SetZero()
45     auswahl_vector:SetZero()
46     select3 = structure:GetNodeIndex(select3)
47     auswahl_matrix[{select3,0}] = 1
48     local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
49     for i = 1 , auswahl_vector:Rows() do
50         if auswahl_vector[i-1] == 1 then
51             select3 = i
52         end
53     end
54
55     auswahl_matrix:SetZero()
56     auswahl_vector:SetZero()
57     select4 = structure:GetNodeIndex(select4)
58     auswahl_matrix[{select4,0}] = 1
59     local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
60     for i = 1 , auswahl_vector:Rows() do
61         if auswahl_vector[i-1] == 1 then
62             select4 = i
63         end
64     end
65

```

```

66  auswahl_matrix:SetZero()
67  auswahl_vector:SetZero()
68  select5 = structure:GetNodeIndex(select5)
69  auswahl_matrix[{select5,0}] = 1
70  local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
71  for i = 1 , auswahl_vector:Rows() do
72      if auswahl_vector[i-1] == 1 then
73          select5 = i
74      end
75  end
76
77  auswahl_matrix:SetZero()
78  auswahl_vector:SetZero()
79  select6 = structure:GetNodeIndex(select6)
80  auswahl_matrix[{select6,0}] = 1
81  local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
82  for i = 1 , auswahl_vector:Rows() do
83      if auswahl_vector[i-1] == 1 then
84          select6 = i
85      end
86  end
87
88  auswahl_matrix:SetZero()
89  auswahl_vector:SetZero()
90  select7 = structure:GetNodeIndex(select7)
91  auswahl_matrix[{select7,0}] = 1
92  local auswahl_vector = structure:ToDoDisplacements(auswahl_matrix)
93  for i = 1 , auswahl_vector:Rows() do
94      if auswahl_vector[i-1] == 1 then
95          select7 = i
96      end
97  end
98
99  select_vector = tmath.ZeroVector(7)
100 select_vector[0] = select1
101 select_vector[1] = select2
102 select_vector[2] = select3
103 select_vector[3] = select4
104 select_vector[4] = select5
105 select_vector[5] = select6
106 select_vector[6] = select7
107
108 return select_vector
109 end

```

A.5 Earthquake generation

Implementation of the earthquake generation

```

1  -----
2  ----- earthquake sampling -----
3  -- input:
4  -- N ... number of sample points
5  -- dt ... time step (resolution)
6  -- output:
7  -- generated earthquake ... matrix output: first column time, second column excitation
8  -- history
9  -----
10 -- c Christian Bucher, CMSD-VUT January 2011, Franz Bamer
11 function generate(N,dt)
12     local NN=N/2
13     local ommax = math.pi/dt
14     local domega = ommax/NN
15     local omegas = tmath.Matrix(NN)
16     omegas:SetLinearRows(0, ommax)
17     local PSD = psd(omegas);

```

```

18 local var = tmath.Sum(PSD)*2*domega
19 local SQPS = (PSD*ommax):CW():Sqrt()
20 local re=stoch.Simulate(NN,1):CW()*SQPS
21 local im=stoch.Simulate(NN,1):CW()*SQPS
22 local x
23 x = spectral.IFT(re:AppendCols(im), domega)
24 local var1 = stoch.Sigma(x:Transpose())[0]^2
25 local t = tmath.Matrix(x)
26 for i = 1, t:Rows() do
27     t[{i-1,0}] = dt*(i-1)
28 end
29 local e = tmath.Matrix(t)
30 e:SetZero()
31 for i = 0, t:Rows()-1 do
32     e[i] = 4*(math.exp(-0.25*t[i]) - math.exp(-0.5*t[i]))
33 end
34 local a = tmath.Matrix(t)
35 a:SetZero()
36 a = e:CW() * x
37 local generated_earthquake = tmath.ZeroMatrix(a:Rows(),2)
38 generated_earthquake[{0,0}] = t
39 generated_earthquake[{0,1}] = a
40 collectgarbage()
41 return generated_earthquake
42 end

```

Line 12 calls the function, which calculates the power spectral density

```

1 -----
2 ----- power spectral density -----
3 -- input:
4 -- oms ... input vector (frequency content)
5 -- output:
6 -- ps ... power spectral density
7 -----
8 -- Christian Bucher, CMSD-VUT January 2011, Franz Bamer
9
10 function psd(oms)
11     local N = oms:Rows()
12     local ps = tmath.Matrix(N)
13     local zeta = 0.3
14     local S_0 = 0.1
15     local omegag = 15
16     for i=0, N-1 do
17         local omega = oms[i]
18         ps[i] = S_0 * (4*zeta^2*omegag^2 + omegag^4) / ((omegag^2-omega^2)^2 +
19             4*zeta^2*omegag^2*omega^2)
20         --ps[i] = 1./(1-1.8*omega^2+omegag^4);
21     end
22     collectgarbage()
23     return ps
24 end

```

Implementation of the calculation of the effective earthquake excitation on a structure

```

1 -----
2 ----- generate effective earthquake force -----
3 -- input:
4 -- acc_x ... excitation in x direction
5 -- acc_y ... excitation in y direction
6 -- M ... mass matrix (sparse)
7 -- structure ... object with structural information
8 -- output:
9 -- force ... matrix output: one force vector at every time point
10 -----
11 -- c Franz Bamer
12
13 function generate_earthquake_force(acc_x,acc_y,M,structure)
14
15     local force = tmath.ZeroMatrix(M:Rows(),acc_x:Rows())

```

```

16
17 local auswahl= structure:GetAllDisplacements()
18 auswahl:SetZero()
19 local acc_vec = tmath.ZeroVector(M:Rows())
20 for i = 1 , acc_x:Rows() do
21     for k = 1 , auswahl:Rows() do
22         auswahl[{:k-1,0}] = acc_x[i-1]
23         auswahl[{:k-1,1}] = acc_y[i-1]
24     end
25     acc_vec = structure:ToDoDisplacements(auswahl)
26     force[{:0,i-1}] = - M:Dot(acc_vec)
27 end
28 return force
29 end

```

A.6 Damping

```

1 -----
2 ----- function Rayleigh damping -----
3 -- input:
4 -- K ... stiffness matrix (sparse)
5 -- M ... mass matrix (sparse)
6 -- output:
7 -- C ... damping matrix (sparse)
8 -----
9 -- c Franz Bamer
10
11 function rayleigh(K,M)
12
13     local eval, evec = K:Eigen(M, 17)
14     local freq = eval:CW():Sqrt()/2/math.pi
15
16     local zeta1 = 0.04
17     local zeta2 = 0.04
18     local omega1 = freq[0]*2*math.pi
19     local omega2 = freq[1]*2*math.pi
20
21     local alpha = 2*(zeta1*omega1 - zeta2*omega2)/(omega1^2-omega2^2)
22     local beta = 2*omega1*zeta1 - alpha*omega1/2
23
24     local C = K:Add(M, alpha, beta)
25
26     return C
27 end

```

B Test run demonstration of the Monte Carlo Simulation for a three-dimensional building structure

According to section 4.6.3 a small test run of ten earthquake samples and the solution by application of the Newmark method for the calculation of the full system and the "universal" POD method for the calculation of the reduced system. The three-dimensional nonlinear structure is presented in Fig. 65. The response of the left corner (first floor) of the full and the reduced solution as well as the corresponding earthquake samples are presented in Fig. 81 - 90. The "universal" POD basis is calculated by taking 120 snapshots spread in equidistant time intervals over the responses to the first three earthquake excitation samples.

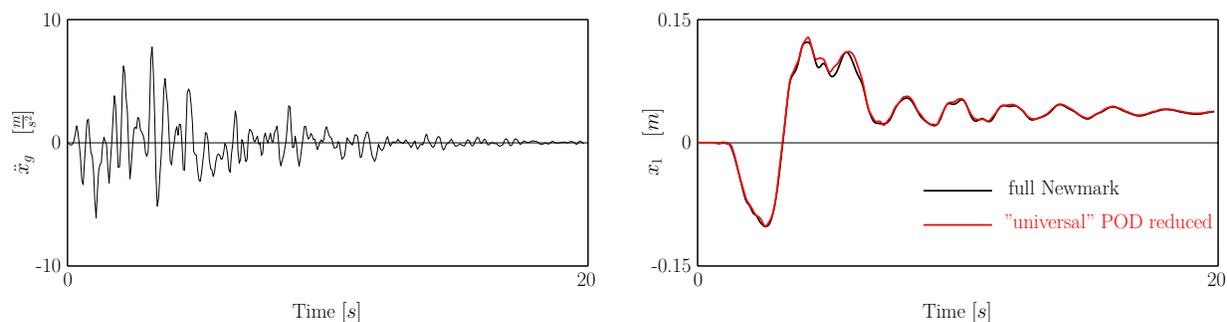


Figure 81: Left subplot: earthquake sample excitation 1; right subplot: response x_1 in the left corner (first floor)

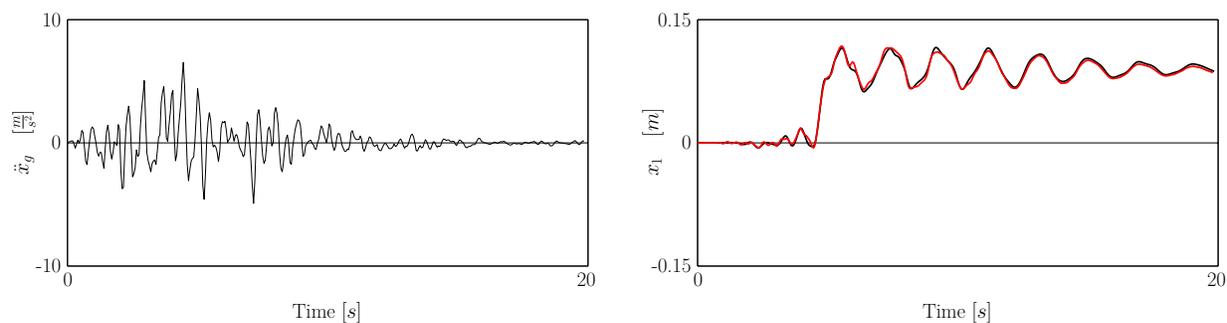


Figure 82: Left subplot: earthquake sample excitation 2; right subplot: response x_1 in the left corner (first floor)

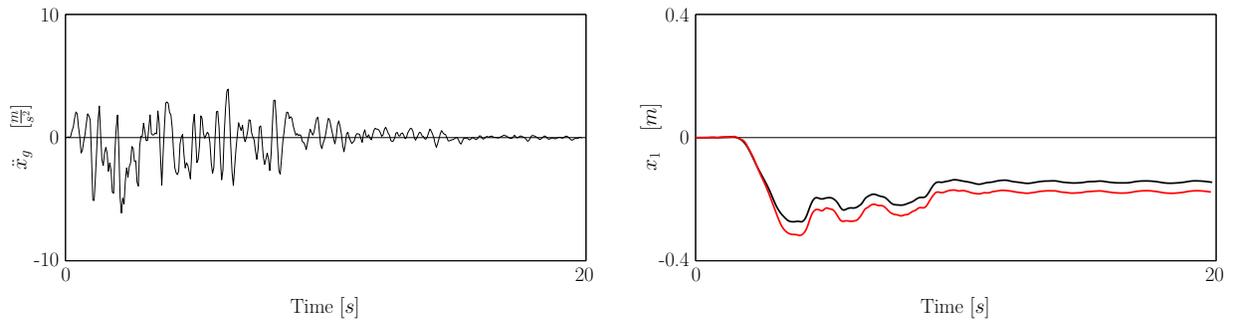


Figure 83: Left subplot: earthquake sample excitation 3; right subplot: response x_1 in the left corner (first floor)

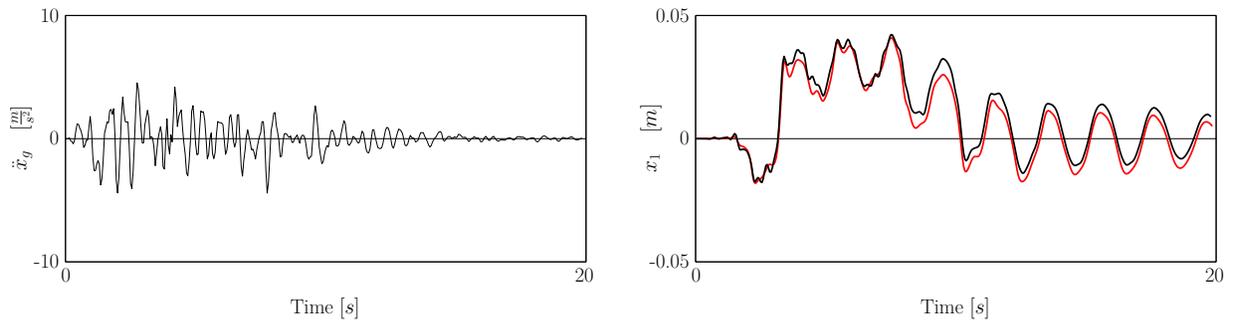


Figure 84: Left subplot: earthquake sample excitation 4; right subplot: response x_1 in the left corner (first floor)

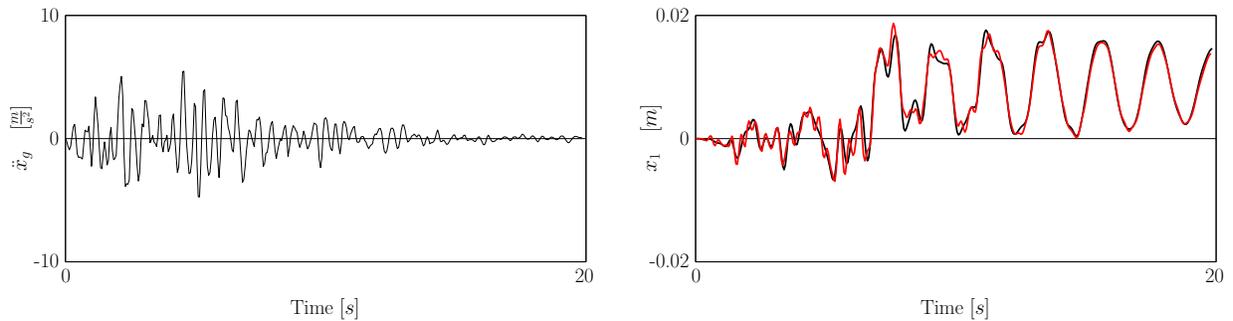


Figure 85: Left subplot: earthquake sample excitation 5; right subplot: response x_1 in the left corner (first floor)

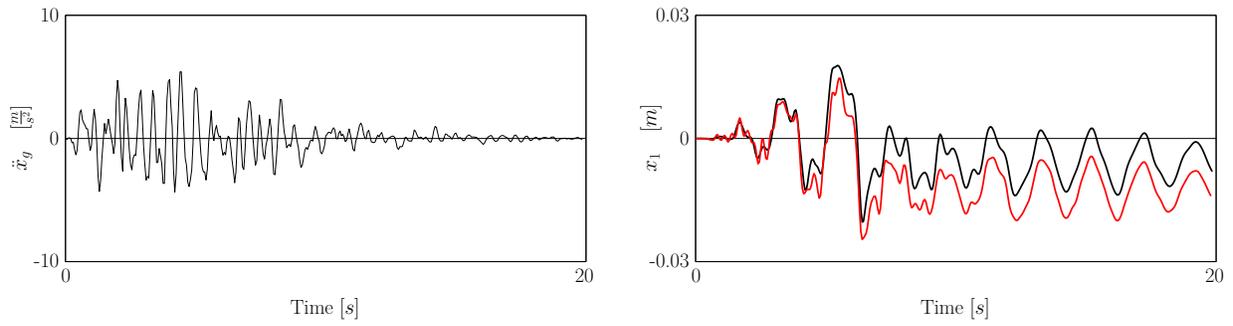


Figure 86: Left subplot: earthquake sample excitation 6; right subplot: response x_1 in the left corner (first floor)

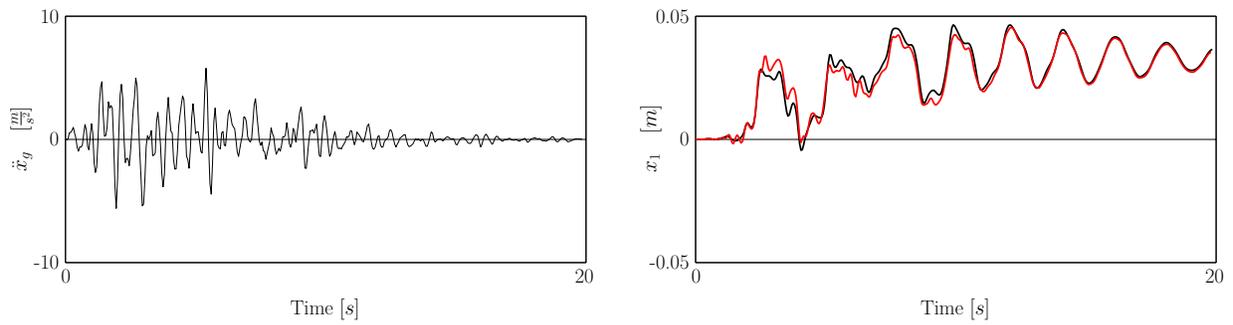


Figure 87: Left subplot: earthquake sample excitation 7; right subplot: response x_1 in the left corner (first floor)

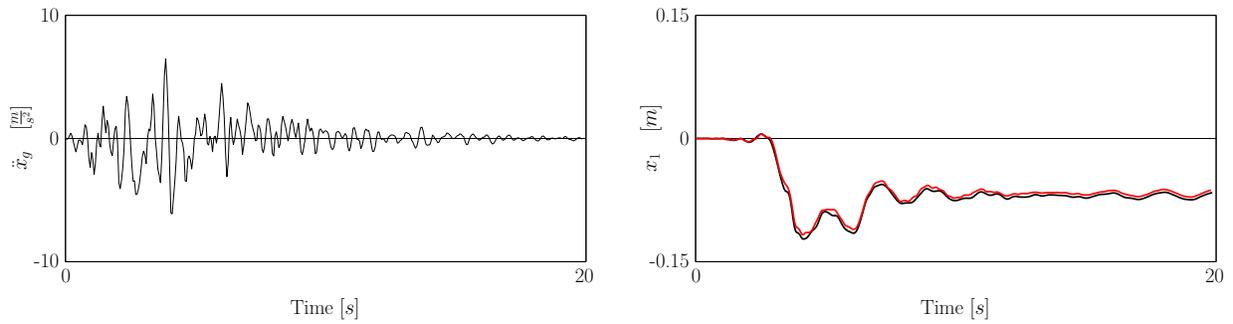


Figure 88: Left subplot: earthquake sample excitation 8; right subplot: response x_1 in the left corner (first floor)

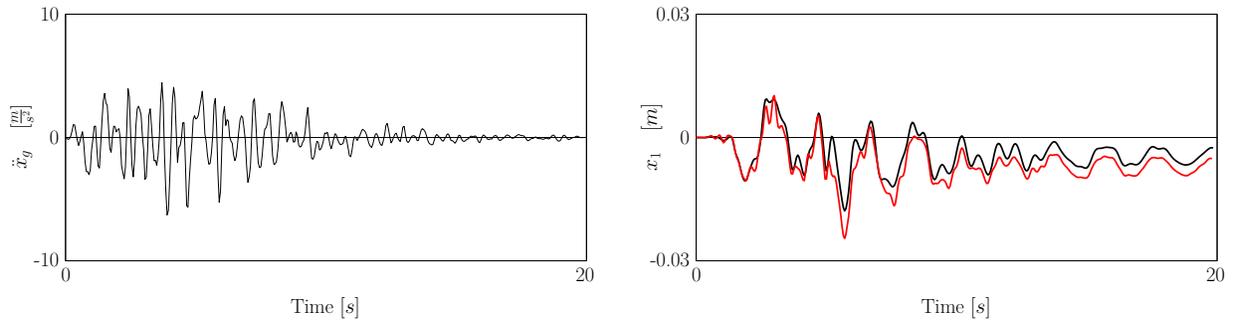


Figure 89: Left subplot: earthquake sample excitation 9; right subplot: response x_1 in the left corner (first floor)

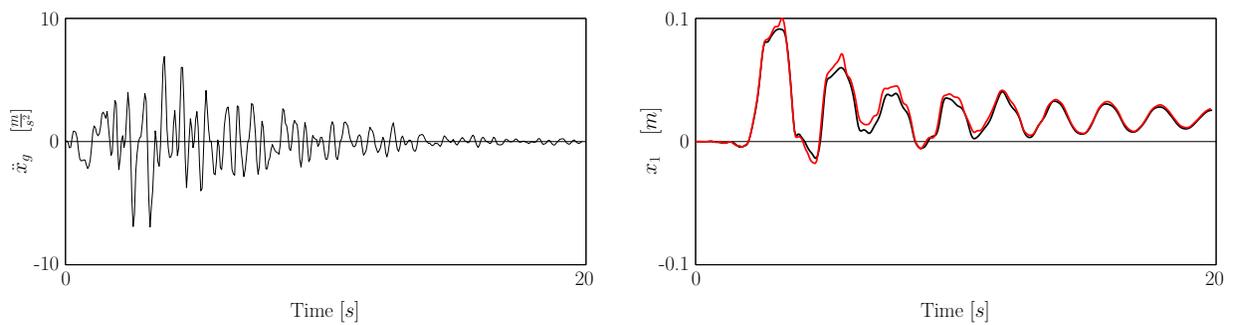


Figure 90: Left subplot: earthquake sample excitation 10; right subplot: response x_1 in the left corner (first floor)

List of Figures

1	Overview basic goal	1
2	Simple modeling	1
3	Examples for complicated non symmetric structures	2
4	Response to the small duration impulse (Clough and Penzien (1995))	11
5	Approximation of an excitation by a piecewise constant excitation function .	12
6	Realization of an excitation force by a piecewise linear force function	13
7	Central difference approximation, geometric assumptions for a SDOF system	17
8	Secant stiffness $\hat{\mathbf{K}}_{sec,i}$ and tangent stiffness $\mathbf{K}_{T,i}$	23
9	Newton Raphson iteration (cf. Chopra (2001))	24
10	Modified Newton Raphson iteration	25
11	Left subplot: transient excitation (earthquake), right subplot: calculation scheme	30
12	NS acceleration of the El Centro earthquake	30
13	Discretized three-story structure with five bays, $h_g(t)$ denotes the horizontal displacement of the first floor at the left corner (Bamer and Bucher (2012)) .	31
14	Horizontal displacements based on different solution procedures (Bamer and Bucher (2012))	34
15	Mean modal truncation error, $E_{mean,modal}(t)$ (Bamer and Bucher (2012)) . .	34
16	Mean POD reduction error, $E_{mean,POD}(t)$ (Bamer and Bucher (2012))	35
17	POD modes of the linear system (Bamer and Bucher (2012))	35
18	Invariant of the covariance matrix \mathbf{D} in the snapshot time period (Bamer and Bucher (2012))	36
19	Invariant of the covariance matrix \mathbf{D} in the whole observation period (Bamer and Bucher (2012))	36
20	Finite friction element (Bamer and Bucher (2012))	37
21	Planar frame structure with friction elements at the supports (Bamer and Bucher (2012))	37
22	Horizontal displacement left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))	39
23	Horizontal displacement - left bearing, full system compared to POD reduced system (Bamer and Bucher (2012))	40
24	Horizontal displacement - relative, full system compared to POD reduced system (Bamer and Bucher (2012))	40
25	Mean POD error, $E_{mean,POD}(t)$ (Bamer and Bucher (2012))	40
26	Bending moment left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))	41
27	Shear force left corner in the first floor - full system compared to POD reduced system (Bamer and Bucher (2012))	41
28	POD modes of the nonlinear system (Bamer and Bucher (2012))	41
29	Singular values of the corresponding POD modes (Bamer and Bucher (2012))	42
30	Invariant of \mathbf{D} and calculation time as a function of the used snapshots (Bamer and Bucher (2012))	43
31	Approach for universal POD modes (transformation matrix)	44
32	Approach for MOR with universal POD modes	45

33	Building construction	46
34	Left hand side: Finite element discretized structure, right hand side: friction isolator	47
35	Geographical visualization of the epicenters and the record station FKS16; note that this is a photo using <i>GoogleEarth</i> (https://earth.google.com) to illustrate the geographical distance	48
36	Top: East-west acceleration record of the Fukushima foreshock, bottom: Morlet wavelet transformation of the Fukushima foreshock accelerogram, note that time scales are different	49
37	Top: East-west acceleration record of the Fukushima main earthquake, bot- tom: Morlet wavelet transformation of the Fukushima main earthquake ac- celerogram, note that time scales are different	49
38	Top: Imperial Valley acceleration, bottom: Morlet wavelet transformation of the Imperial Valley acceleration	50
39	Top: Northridge acceleration, bottom: Morlet wavelet transformation of the Northridge acceleration	51
40	Displacements of the Fukushima foreshock full model and POD reduced model (equivalent to 4.3)	52
41	Universal POD modes	53
42	Displacements of the Fukushima earthquake full model and POD reduced model	54
43	Mean error of horizontal displacements - Fukushima foreshock	54
44	Mean error of horizontal displacements - Fukushima earthquake	54
45	Displacements of the Imperial Valley earthquake full model and POD reduced model (equivalent to 4.3)	55
46	Displacements of the Northridge earthquake full model and POD reduced model	56
47	Mean error of horizontal displacements - Imperial Valley earthquake	56
48	Mean error of horizontal displacements - Northridge earthquake	56
49	Stress strain behavior of a yielding material with strain hardening	59
50	3-D visualization of the building and direction of the earthquake excitation	59
51	Left side: construction plan of one storey; right side: FE discretized mesh of one storey	60
52	First 6 natural mode shapes of the linearized structure with the corresponding eigen angular frequencies and eigen frequencies	60
53	Harmonic ground acceleration	61
54	Left subplot: relative storey displacement x of the output node, right subplot: hysteresis - displacement x and moment M_y	61
55	Universal POD modes	62
56	Response to the Bam earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - displacement x and moment M_y	63
57	Response to the Northridge earthquake, left subplot: relative storey displace- ment x of the output node, right subplot: hysteresis - relative displacement x and moment M_y	63

58	Response to the Imperial Valley earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y	64
59	Response to the Landers earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y	64
60	Response to the Loma Prieta earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y	65
61	Response to the Palm Springs earthquake, left subplot: relative storey displacement x of the output node, right subplot: hysteresis - relative displacement x and moment M_y	65
62	Samples of generated earthquakes - ground acceleration $\ddot{x}_g(t)$; $S_0 = 0.1 [m^2/s]$, $\zeta = 0.3 [-]$, $\omega_g = 15 [rad/s]$	67
63	Calculation approach of the universal POD reduction with generated earthquakes	68
64	Results of the statistical evaluation 1000 sample transient excitations; Left top: Mean of the horizontal displacement μ_h ; Right top: standard deviation of the relative horizontal displacement σ_h ; Left bottom: Error of the mean relative horizontal displacement E_μ ; Right bottom: Error of the standard deviation of the relative horizontal displacement E_σ	69
65	3-D visualization of the building construction and direction of the earthquake excitation samples; degree of freedom for the response output x_1	70
66	Applied deformation x_1 and corresponding function of the bending moment of the column in the first storey	70
67	Results of the first Monte Carlo simulation with a threshold of $x_{max} = 0.1 [m]$ in the first floor; for every point 10^3 samples are performed	72
68	Results of the second Monte Carlo simulation with a threshold of $x_{max} = 0.1 [m]$ in the first floor; for every blue point 10^3 samples are performed; for every red point 10^4 samples are performed	73
69	Portal frame structure with beam and plate/shell elements; $l_x = 10$, $l_y = 10$, $h = 5[m]$; number of DOF = 1911; the impact point as well as the horizontal displacement $h(t)$ and the bending moment $M(t)$ are highlighted in red	76
70	First 3 normal modes	76
71	First 3 POD modes	77
72	Displacement response $h(t)$ (the lines of the hybrid truncation solution and full response cover each other)	77
73	Bending moment response $M(t)$ (note: the lines of the hybrid truncation response and full solution cover each other but modal truncation fails completely)	77
74	Energy content of the hybrid POD modes	78
75	Computational time as a function of the numbers of DOF	78
76	First invariant of the error covariance matrix as a function of the observation time period	79
77	Linear test frame structure	94
78	Response: Horizontal displacement of the left corner	94
79	Nonlinear test structure with the friction isolator presented in section 4.3	98

80	Response: Horizontal displacement of the left corner	99
81	Left subplot: earthquake sample excitation 1; right subplot: response x_1 in the left corner (first floor)	112
82	Left subplot: earthquake sample excitation 2; right subplot: response x_1 in the left corner (first floor)	112
83	Left subplot: earthquake sample excitation 3; right subplot: response x_1 in the left corner (first floor)	113
84	Left subplot: earthquake sample excitation 4; right subplot: response x_1 in the left corner (first floor)	113
85	Left subplot: earthquake sample excitation 5; right subplot: response x_1 in the left corner (first floor)	113
86	Left subplot: earthquake sample excitation 6; right subplot: response x_1 in the left corner (first floor)	114
87	Left subplot: earthquake sample excitation 7; right subplot: response x_1 in the left corner (first floor)	114
88	Left subplot: earthquake sample excitation 8; right subplot: response x_1 in the left corner (first floor)	114
89	Left subplot: earthquake sample excitation 9; right subplot: response x_1 in the left corner (first floor)	115
90	Left subplot: earthquake sample excitation 10; right subplot: response x_1 in the left corner (first floor)	115

List of Tables

1	Newton Raphson iteration within the time step t_i to t_{i+1} (cf. Chopra (2001))	23
2	Modified Newton Raphson iteration within the time step t_i to t_{i+1} (cf. Chopra (2001))	24
3	Information data of the Fukushima foreshock and the Fukushima main earthquake	48
4	Information data of the Imperial Valley earthquake and the Northridge earthquake	50
5	Calculation time [s]: Full calculation (central differences), POD calculation, universal POD calculation (calculation time of the snapshot matrix: 6 seconds)	57
6	Earthquake excitation list; Δt [s] resolution of the record data, T [s] duration of the record, d [km] distance from epicenter, M moment magnitude, PGA [m/s^2] peak ground acceleration	62