



Norwegian University of
Science and Technology

Digital Twins: Methods For Load Estimation Based on Sensor Time Histories

Mari Herskedal

Master of Science in Mechanical Engineering

Submission date: June 2018

Supervisor: Bjørn Haugen, MTP

Norwegian University of Science and Technology
Department of Mechanical and Industrial Engineering

//

Abstract

In structural dynamics, load identification based on measured responses is referred to as an “inverse” problem. In general, structures are dimensioned to withstand a given applied load. For structures subjected to varying dynamic loads, an exact load evaluation may be challenging. In order to properly monitor the structure during operation and thereby ensure safety, it is essential to know the magnitude of the applied load. Updated lifetime evaluations can then be done and maintenance needs may be met. For some structures, direct measurements of the applied loads are challenging or sometimes impossible. The loads may in such circumstances be obtained through inverse methods.

During recent years, inverse methods for load identification has gained an increased attention in the “Digital Twin” technology. In this technology, the structure, or physical object is represented by an identical virtual twin. Based on obtained sensor data of measured responses, it is desirable to estimate the applied load through effective inverse methods. Inverse methods have been proposed by scientists in different fields of engineering. Some of the studies have been summarized in this thesis. The majority of the methods have proved successful for linear systems, while a solution to nonlinear systems has not yet received as much attention. As for the digital twin technology, a robust procedure to obtain dynamic loads despite lack of system linearity is essential.

In this thesis, two of the presented load reconstruction methods have been numerically tested in terms of a digital twin solution. The results showed that both methods were successfully able to reconstruct the applied loads. However, parts of the reconstructed loads were interfered by noise which should be dealt with by using filtering techniques or regularization methods. No noticeable difference were observed for the load reconstruction using displacement, velocity or acceleration responses. Whether or not the methods can be applicable to nonlinear systems is yet to be investigated. Further research on the subject is recommended.

Sammendrag

Last identifikasjon basert på målte responser kalles et inversproblem i strukturell dynamikk. Strukturer er generelt dimensjonert for å motstå en bestemt påført last, men en eksakt last evaluering kan være utfordrende for strukturer som er utsatt for varierende dynamisk last. For å kunne monitorere en struktur under operative forhold og dermed opprettholde sikkerhet er det essensielt å vite størrelsen på den påførte lasten. Man kan da gjøre oppdaterte kalkulasjoner av forventet levetid, i tillegg til å planlegge vedlikehold. For noen strukturer kan en direkte måling av påført last være svært utfordrende eller umulig. Påført last kan i slike tilfeller bli funnet gjennom inverse metoder.

Inverse metoder for last identifikasjon har fått økt oppmerksomhet i forbindelse med “Digital Tvilling” teknologi. Denne teknologien går ut på at strukturen, eller det fysiske objektet, er representert av en identisk, virtuell tvilling. Ved å bruke sensorer til å måle strukturens responser, er det ønskelig å estimere påført last gjennom inverse metoder. Slike inverse metoder har blitt utviklet av forskere fra ulike ingeniørfagfelt. Noen av disse studiene er gjenfortalt i denne masteroppgaven. Mesteparten av metodene har vært vellykkede for lineære systemer, mens løsninger for ikke-lineære systemer har fått mindre oppmerksomhet. I forbindelse med digital tvilling teknologi er det ønskelig å utvikle en robust prosedyre for å finne dynamisk last uavhengig av mangel på systemets lineærhet.

I denne masteroppgaven er to av de presenterte last identifikasjonsmetodene blitt numerisk testet som en digital tvilling løsning. Resultatene viser at en riktig last var mulig å oppnå gjennom begge metodene. Støy i måledataene ble observert og bør hindres ved bruk av regularisering eller filtreringsteknikker. For last identifikasjon basert på forskyvnings-, hastighets- eller akselerasjonsmålinger ble ingen bemerkningsverdige forskjeller observert. Det gjenstår å se om metodene kan brukes for ikke-lineære systemer. Videre forskning er anbefalt.

Preface

This thesis is a part of a Master of Science at the Norwegian University of Science and Technology. The degree specialization is in Mechanical Engineering. The thesis was written during spring 2018 for the Department of Mechanical and Industrial Engineering.

I would like to thank my supervisor Associate Professor Bjørn Haugen for his great help and guidance with the project. I would also like to thank Terje Rølvåg for his help with the software used in the study, Fedem.

Trondheim, June 11th, 2018

A handwritten signature in cursive script that reads "Mari Herskedal". The signature is written in black ink and is positioned above a horizontal line.

Mari Herskedal

Table of Contents

Abstract	i
Sammendrag	ii
Preface	iii
Table of Contents	viii
List of Tables	ix
List of Figures	xv
Abbreviations	xvi
1 Introduction	1
1.1 Background and Motivation	1
1.2 Problem Description	2
1.3 Approach and Limitations	2
1.4 Thesis Structure	2
2 Literature Review	5
2.1 The Inverse Problem	5

2.2	The Ill-Posed Nature of the Inverse Problem	6
2.2.1	Kalman Filter	7
2.3	Force Reconstruction Strategies	9
2.4	Time Domain Methods	9
2.4.1	Convolution	11
2.4.2	Markov Parameters	14
2.5	The Frequency Domain Method	16
2.5.1	Force and Displacement Transmissibility	17
2.6	Determination of Force Location	22
3	Theory	23
3.1	Regularization Methods	23
3.1.1	Tikhonov Regularization	24
3.1.2	The Truncated Singular Value Decomposition Method	27
3.2	Filtering Techniques	29
3.2.1	Butterworth Filter	30
3.3	Linear SDOF Systems	31
3.4	Nonlinear SDOF Systems	33
3.5	Undamped SDOF systems	34
3.6	MDOF Systems	35
3.7	Direct Integration	36
3.7.1	The Central Difference Method	36
3.7.2	The Newmark- β Method	37
3.7.3	The Wilson- θ Method	37
3.8	Damping	38

3.8.1	Rayleigh Damping	39
3.8.2	Modal Damping	40
4	Experiment	43
4.1	Periodic Load Reconstruction	43
4.1.1	Model Setup	43
4.1.2	Iteration Process	46
4.1.3	Variation in Spring Stiffness	48
4.1.4	Variation in Structural Damping	51
4.1.5	Load Reconstruction And Noise Reduction	55
4.2	Impulse Load Reconstruction	60
4.2.1	Example	61
4.2.2	SDOF System	62
4.2.3	Simplified Excavator	73
4.2.4	Matrix Solution	74
5	Results	75
5.1	Periodic Load Reconstruction	75
5.1.1	Frequency Below Eigenfrequency	75
5.1.2	Frequency at Resonance	77
5.1.3	Frequency Above Eigenfrequency	78
5.2	Impulse Load Reconstruction	80
6	Discussion	83
6.1	Periodic Load Reconstruction	83
6.1.1	Noise Interference	84

6.1.2	Varying Properties	84
6.1.3	Linearity	85
6.2	Impulse Load Reconstruction	86
6.3	Software Performance	86
7	Concluding Remarks	87
7.1	Summary and Conclusions	87
7.2	Suggestions for Further Work	88
	Bibliography	91
	Appendix	95
A	Additional Information	97
A.1	Well-Posed Problems	97
A.2	Dynamic Programming Solution Using Markov Parameters	97
A.3	The Effect of an Increase in Spring Stiffness	100
A.4	Several Iterations	102
A.5	Readout Points SDOF system	104
A.6	Impulse Load Reconstruction	106
B	Python Code	107

List of Tables

4.1	Dynamic properties assigned to the excavator.	44
4.2	The eigenfrequencies of the excavator	45
4.3	Dynamic properties assigned to the digital twins.	46
4.4	Iteration parameters.	48
4.5	Relative damping for each chosen stiffness proportional	51
4.6	Dynamic properties assigned to SDOF mass-spring system.	62

List of Figures

2.1	Schematic representation of the forward and inverse problems. [1]	5
2.2	Implicit and explicit methods for nonlinear systems. [2]	10
2.3	The Dirac delta function. [3]	12
2.4	The convolution of two signals. [3]	12
2.5	An arbitrary dynamic system subjected to a dynamic force which causes a dynamic response.	18
2.6	The location of coordinates U, K and C on arbitrary dynamic systems [4].	20
3.1	Classical Tikhonov regularization for four different regularization parameters with corresponding percentage error of reconstructions. [5]	24
3.2	Generalized Tikhonov regularization for four different regularization parameters with corresponding percentage error of reconstructions. [5]	26
3.3	Idealized damped SDOF system.	31
3.4	Amplification ratio versus frequency ratio for a SDOF system of harmonic oscillations. [6]	35
3.5	Strategies for obtaining dynamic responses in MDOF systems. [7]	36
3.6	Damping ratio as a function of frequency. [8]	40
4.1	2-dimensional excavator.	44
4.2	Excavator model - Twin 1.	46

4.3	Excavator model - Twin 2.	46
4.4	Iteration process flow chart. Blue ellipse illustrates process start and end. Blue blocks illustrates the process steps. Blue diamond illustrates decision.	47
4.5	Estimated load in a 5 seconds time interval, using a spring stiffness of 10^5 N/m.	49
4.6	Estimated load at area of convergence, using a spring stiffness of 10^5 N/m.	49
4.7	Estimated load in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.	50
4.8	Estimated load at area of convergence, using a spring stiffness of 10^7 N/m.	50
4.9	Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0001.	52
4.10	Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0003.	52
4.11	Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0009.	53
4.12	Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0001.	53
4.13	Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0003.	54
4.14	Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0009.	54
4.15	First iteration.	55
4.16	Second iteration.	56
4.17	Third iteration.	56
4.18	Low-pass filter applied to the estimated load.	57
4.19	A filter with a time constant of 0.007.	57
4.20	A filter with a time constant of 0.02.	58
4.21	A filter with a time constant of 0.06.	58

4.22	Seven iterations using filter for load estimation of an applied load of frequency 3.	59
4.23	A filter with a time constant of 0.007 and an applied load frequency 13.0 Hz.	60
4.24	Assumed impulse load example.	61
4.25	Applied impulse load example.	62
4.26	Idealized SDOF system.	63
4.27	System model in Fedem.	63
4.28	Assumed impulse load using a time step of 0.01 seconds.	64
4.29	Assumed impulse load using a time step of 0.05 seconds.	64
4.30	Assumed impulse load using a time step of 0.1 seconds.	65
4.31	Displacement measurements assuming maximum impulse load at 0.01 seconds.	65
4.32	Displacement measurements assuming maximum impulse load at 0.05 seconds.	66
4.33	Estimated load using 100 readout points.	67
4.34	Estimated load with an upper and lower limit, using 100 readout points.	67
4.35	Estimated load using 20 readout points.	68
4.36	Estimated load with an upper and lower limit, using 20 readout points.	68
4.37	Displacement measurements assuming maximum impulse load at 0.1 seconds.	69
4.38	Displacement readout points.	69
4.39	Estimated load using 10 readout points.	70
4.40	Velocity measurements assuming maximum impulse load at 0.1 seconds.	71
4.41	Velocity readout points.	71
4.42	Acceleration measurements assuming maximum impulse load at 0.1 seconds.	72
4.43	Acceleration readout points.	72
4.44	Excavator spring force response from impulse load and applied load.	73

4.45	Spring force readout points.	74
5.1	Load comparison without filter for an applied load frequency below eigenfrequency.	76
5.2	Load comparison with filter for an applied load frequency below eigenfrequency.	76
5.3	Estimated load at resonance for a 5 seconds time interval.	77
5.4	Load comparison at resonance showing the low-frequent area.	78
5.5	Estimated load for an applied load frequency above eigenfrequency in a 5 seconds time interval	79
5.6	Estimated load at area of convergence for an applied load frequency above eigenfrequency.	79
5.7	Load comparison at area of convergence for an applied load frequency above eigenfrequency.	80
5.8	Applied impulse load versus estimated impulse load for the SDOF system.	81
5.9	Applied impulse load versus estimated impulse load for the excavator.	81
A.1	Estimated load at resonance in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.	100
A.2	Zoom at area of convergence.	101
A.3	Estimated load for an applied load frequency above eigenfrequency in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.	101
A.4	Zoom at area of convergence.	102
A.5	Estimated load at convergence after one iteration.	103
A.6	Estimated load at convergence after two iterations.	103
A.7	Estimated load after three iterations.	104
A.8	20 readout points.	105
A.9	100 readout points.	105

A.10 Estimated load versus applied load based on 20 readout points.	106
A.11 Estimated load versus applied load based on 100 readout points.	106
B.1 Python script.	107

Abbreviations

SDOF	=	Single Degree of Freedom
MDOF	=	Multiple Degrees of Freedom
FE	=	Finite Element
FEM	=	Finite Element Method
FRF	=	Frequency Response Function
TSVD	=	Truncated Singular Value Decomposition
SVD	=	Singular Value Decomposition
AI	=	Artificial Intelligence
TDGM	=	Time-Domain Galerkin Method

1. Introduction

1.1 Background and Motivation

Engineering structures subjected to dynamic loads of unknown magnitude has been the subject of several studies performed during recent years. Dynamic loads make important boundary conditions for a system and are therefore of great importance in regards to ensure safety and stability [9, 10]. There are several types of loads in structural dynamics, such as periodic, shock and random loads. The magnitude of these loads may be obtained through direct measurement [11]. However, for some applications such as tall buildings subjected to wind loads or bridges subjected to moving vehicles, it is challenging or even impossible to directly measure the dynamic loads [12, 13]. Hence, load identification methods have been developed in order to properly identify loads on structural systems.

Obtaining dynamic load is an inverse process and thereby a so-called “inverse” problem. Load identification methods are based on system properties and measured dynamic responses such as acceleration, displacement or strain measured at several accessible locations on the structure [13]. The methods are divided into two main categories, namely frequency domain methods and time-domain methods [14].

Solving inverse problems is highly relevant for a new and arising technology called the “Digital twin” technology. A digital twin is commonly described as a virtual, digital representation of a physical object [15]. This technology takes part in the Industrial Internet of Things where an increased and automated communication between objects happens without human interaction. The concept of twins for industrial use origins from NASA’s Apollo program where it was stated that “two identical space vehicles were built, allowing the engineers to mirror the conditions of the space vehicle during the mission, the vehicle remaining on earth being the twin [16, p. 63]”. This concept was later developed to include a virtual representation of the twin, resulting in the idea of a digital twin. Digital twin technology has been proved useful for several industrial purposes by different companies. One example is the US Air Force which have used digital models to monitor each aircraft [16].

For some applications, only a limited number of sensors can be placed on the structure [17]. It is desirable to optimize the number of sensors used for each object, in addition to choose optimal sensor locations and yet being able to reconstruct the applied loads. This is a challenge that must be overcome in the digital twin technology. As of today, many studies have proposed sufficient methods for force reconstruction of linear systems. The study of nonlinear dynamic structures is more complex and has not received as much attention.

1.2 Problem Description

A robust procedure for solving inverse problems is essential for the digital twin technology. The objective of this thesis has been to look into existing inverse methods and test some of the methods in regards to their usefulness in the digital twin technology. Finding working methods despite system linearity has been the target, in addition to find and develop an appropriate software tool.

1.3 Approach and Limitations

The study has been divided into two parts. Firstly, an overview of existing inverse methods has been presented in a literature review. Secondly, two load reconstruction experiments have been performed based on two of the methods presented in the first part. The experiments were performed on simple mass-spring systems and have only been tested numerically.

It is challenging to find a computer tool that proves successful for all proposed methods. In this thesis, a software called Fedem has been used for the force reconstruction. Fedem is an acronym for Finite Element Dynamics in Elastic Mechanisms and supports digital twin technology. The obtained results have been based on the capabilities of Fedem.

1.4 Thesis Structure

The thesis is structured as follows. Chapter 2 presents a literature review on proposed inverse methods for load reconstruction. Chapter 3 presents regularization methods and filtering techniques commonly used in the presented studies, followed by theory on basic structural dynamics. In chapter 4, two load reconstruction experiments are presented. The

experimental results are shown in chapter 5, followed by a discussion of the results in Chapter 6. Finally, concluding remarks and suggestions for further work will be presented in Chapter 7.

2. Literature Review

In this chapter, the inverse problem and its ill-posed nature is presented, followed by proposed methods for inverse force reconstruction. At last, studies on determination of force location has been briefly looked into.

2.1 The Inverse Problem

In structural dynamics, the theory used to solve the forward problem lies the foundation of the solution to the inverse problem [12]. Therefore, both problems have been looked into. The problems are illustrated in Figure 2.1. Identification of responses based on system boundary conditions and inputs is defined as the forward problem. The inverse problem has been divided into two classifications [1]:

1. Finding system inputs, based on given responses, boundary conditions and system model (the inverse identification problem).
2. Finding the system model, based on given inputs, responses and boundary conditions (the classical identification problem).

[1, p. 325]

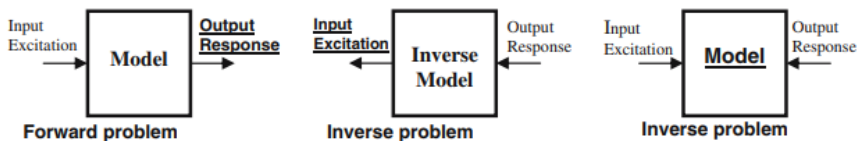


Figure 2.1: Schematic representation of the forward and inverse problems. [1]

Methods for force identification based on system responses have been divided into three categories; deterministic methods, stochastic methods and methods based on artificial intelligence (AI) [1, 18]. Deterministic methods strongly rely on experimental measurements

and their accuracy in regards to the inverse model identification. These methods are often challenging to apply when dealing with strongly nonlinear systems because an accurate nonlinear inverse model must be obtained. When using statistical models to recover the inputs, a statistical relation between the inputs and the outputs is established by taking measurements during operation. The methods based on AI are different kinds of algorithms resulting from a learning process which finds a relation between the inputs and the outputs [1].

In terms of system linearity, stochastic methods are not applicable for nonlinear systems. In nonlinear systems, the forces cannot be added because the relationship between stress and force is nonlinear. Therefore, the load history must be known as one cannot rely on probability distribution. Deterministic and AI methods are applicable for both linear and nonlinear systems. Further in this thesis, deterministic methods will be in focus.

2.2 The Ill-Posed Nature of the Inverse Problem

Inverse problems are highly subjected to measurement errors and are significantly more affected by noise than forward problems. This results in ill-posed solutions, meaning the solutions are often unstable and non-unique [19]. This is because the structure is limited by a finite number of points in which it can be measured [13]. Even the smallest errors in the identified inputs may cause considerable deviations in the results. Therefore, several techniques have been proposed to obtain a numerically stable solution and thereby a well-posed problem [18]. Well-posed problems are commonly defined in the sense of Hadamard as shown in Appendix A.1 [19].

J. Sanchez and H. Benaroya [20] have divided force reconstruction techniques to deal with the ill-posedness into three main categories, namely direct methods, regularization methods and probabilistic/statistical methods. Direct methods use physical or mathematical models to solve inverse problems. The general solution to the problem is often written as

$$u(x, t) = \sum_{n=1}^{\infty} \Phi_n(x) \Gamma_n(t) \quad (2.1)$$

for linear models, where $u(x,t)$ represents the displacement, Φ_n represents the mode shape and Γ_n represents the time variation. Using this solution is often referred to as modal superposition analysis. Direct methods often exhibit a certain degree of noise in the system and require regularization methods to eliminate that effect. Methods belonging to the third

category, probabilistic/statistical methods, are used to gain insight into the system's driving force. One such method is called adaptive estimation which will be looked into in the following section.

2.2.1 Kalman Filter

The Kalman filter is a common type of adaptive estimation similar to adaptive estimation systems used in control theory. In adaptive estimation, there is a measured signal of unknown parameters. By using estimates for the unknown parameters, there is another signal which estimates the state of the system. The unknown parameters are varied by a control law until the signal reaches convergence. The adaptive estimators update the unknown parameters by utilizing iterative control laws based on probabilistic or statistical methods [20].

The filter is widely used in several fields of science and engineering and can be derived by stochastic and deterministic methods. It is designed to recursively estimate the states of a dynamic system in regards to system outputs affected by noise and disturbance. The system is expressed in state-space form driven by a zero-mean white noise process. The use of the state-space form is due to its ability to handle multivariate and time-varying systems [21].

C.K. Ma et al. [22, 23, 24] have proposed several studies on impact force reconstruction of lumped-mass systems. They transformed the equation of motion into state-space equations, and used an input estimation algorithm for the force reconstruction. The algorithm consisted of a Kalman filter and a recursive least-squares algorithm which proved successful for one-dimensional single degree of freedom (SDOF) and multiple degree of freedom (MDOF) systems [22]. The method was also successfully used in linear numerical experiments on beam structures subjected to five different types of input forces. Equation 2.2 and 2.3 were used for the Kalman filter over the discretized time k [23]. A discretized Kalman filter is often divided into two steps; a prediction step (Equation 2.2) and a correction step (Equation 2.3) [25].

$$\begin{aligned}\tilde{\mathbf{X}}_{k|k-1} &= \Phi \tilde{\mathbf{X}}_{k-1|k-1} \\ \mathbf{P}_{k|k-1} &= \Phi \mathbf{P}_{k-1|k-1} \Phi^T + \Gamma \mathbf{Q} \Gamma^T\end{aligned}\tag{2.2}$$

$$\begin{aligned}
 \mathbf{S}_k &= \mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^T + \mathbf{R} \\
 \mathbf{K}_{a,k} &= \mathbf{P}_{k|k-1}\mathbf{H}^T\mathbf{S}_k^{-1} \\
 \mathbf{P}_{k|k} &= [\mathbf{I} - \mathbf{K}_{a,k}\mathbf{H}]\mathbf{P}_{k|k-1} \\
 \tilde{\mathbf{Z}}_k &= \mathbf{Z}_k - \mathbf{H}\tilde{\mathbf{X}}_{k|k-1} \\
 \tilde{\mathbf{X}}_{k|k} &= \tilde{\mathbf{X}}_{k|k-1} + \mathbf{K}_{a,k}\tilde{\mathbf{Z}}_k
 \end{aligned} \tag{2.3}$$

$\tilde{\mathbf{X}}$ and \mathbf{P} represent the state vector and the filters error covariance matrix respectively, \mathbf{S}_k and $\tilde{\mathbf{Z}}_k$ are the innovation covariance and the innovation respectively, \mathbf{K}_a is the Kalman gain which stabilizes the filter, $\mathbf{\Gamma}$ and $\mathbf{\Phi}$ are the input and state transition matrix respectively, \mathbf{Q} is the process noise covariance matrix, \mathbf{H} is the measurement matrix, \mathbf{R} is the measurement noise covariance matrix, \mathbf{Z} is the observation matrix and \mathbf{I} represent the identity matrix. [23][21]

Later, C.K. Ma and C.C. Ho extended the algorithm to be applicable for nonlinear structural systems. Consideration of nonlinear systems presented a more real-case scenario. The non-linearity was known to become stronger with an increasing response amplitude. In the study, the Newmark- β method was used to predict the dynamic responses of the system and an extended Kalman filter was introduced to the algorithm. The numerical experimental results showed convergence after only a few time steps given an adequate choice of tuning parameters. [24]

E.Lourens et al. proposed a study on an augmented Kalman filter which was used for force reconstruction in a deterministic-stochastic setting. In the study, a standard Kalman filter was applied to an augmented state-space model where the forces were added to an unknown state-space vector. The use of an augmented Kalman filter was compared to a traditional deterministic least-squares technique called Dynamic Programming. The two methods showed different strengths and weaknesses. The augmented Kalman filter provided more reliable results when collocated measurements were performed. However, as an optimal regularization parameter for a specific period must be calculated, the augmented Kalman filter was only applicable for off-line state estimation. [26]

Other common filtering techniques and regularization methods dealing with the ill-posedness of the inverse problem are presented in Chapter 3.

2.3 Force Reconstruction Strategies

Structural dynamic equilibrium equations may be solved in original degrees of freedom or modal degrees of freedom. In literature, three main solution strategies have been proposed to solve these equations [14]:

- a solution in time domain where the load and corresponding response development is pursued stepwise for a sufficiently long period of time, in which case time series of the structural response is obtained,
- an incremental stepwise state-space solution in time domain based on the Duhamel integral and applying the fluctuating load as a consecutive sequence of short impulses, or
- a solution where a Fourier transform is applied throughout the equilibrium equation and the problem is transferred into a frequency domain description, in which case a frequency domain spectral representation of the response is obtained.

[14, p. 229]

Originally, force identification problems were solved in the frequency domain, using the frequency response functions of the structures. Later on, solutions in the time domain became more prominent, following a more deterministic approach. There are also some examples of studies using combined deterministic-stochastic techniques [26].

As the state space method is a solution in time domain, it will be included in the time domain solution strategy further in this thesis. Hence, the two main solution strategies to be presented in the next sections are time domain methods and frequency domain methods.

2.4 Time Domain Methods

A time domain response reconstruction may be performed explicit or implicit. An explicit reconstruction is solely based on the known response history at time t_k . An implicit reconstruction accounts for the unknown future by assuming the response development or equilibrium condition by looking at the development between time step t_k and $t_k + \Delta t$ [14]. For the displacements \mathbf{X} and velocities $\dot{\mathbf{X}}$, the implicit-explicit solution may be written in the following form [2]:

$$\begin{aligned} \mathbf{X}(t_k + \Delta t) &= \mathbf{X}(t_k) + \Delta t \dot{\mathbf{X}}(t_k + \alpha \Delta t) \\ \dot{\mathbf{X}}(t_k + \Delta t) &= (1 - \alpha) \dot{\mathbf{X}}(t_k) + \alpha \dot{\mathbf{X}}(t_k + \Delta t) \end{aligned} \tag{2.4}$$

for a solution parameter α . The explicit and implicit solution may thereby be defined as follows [2].

- **Explicit:** it states the equilibrium at time t , with $\alpha = 0$. The displacement in the next step is obtained depending on the velocity and displacement of the previous step.
 - **Implicit:** it formulates the equilibrium at time $(t+\hat{t})$ with $\alpha = 1$. The displacement in the next step is obtained depending on the current time velocity and on the displacement of the previous step.
- [2, p. 33]

Figure 2.2 represents common time domain methods in nonlinear structural dynamics in regards to their implicit or explicit character.


α	Method	Solution type
0	Forward Difference Method Forward Euler	Explicit  Implicit
1/2	Half point rule Crank-Nicholson	
2/3	Galerkin	
1	Backward difference Method Backward Euler	

Figure 2.2: Implicit and explicit methods for nonlinear systems. [2]

The most common method for inverse force identification in the time domain is the state space method, mentioned in the previous section. It is a conditionally stable explicit time-stepping method which often uses a reduced model to find the responses of a structure. However, the method has some drawbacks due to a large discretization error when using a low sampling frequency or a long sampling duration. As opposed to the state space method's conditionally stable character, a method called the Newmark- β method is known for its unconditionally stable character. This method has been widely used in forward dynamic analyses, but seldom in inverse analyses. K. Liu et al. [27] proposed a study on force identification where the state space method was compared to the Newmark- β method.

In the study, the Newmark- β algorithm was transformed to an explicit form for the solution of the $\mathbf{Ax}=\mathbf{B}$ equations. For multiple sinusoidal and white noise excitations with 10 percent measurement noise in the responses, the method proved more successful for identification of force time histories as compared to the state space method.

The Newmark- β method and the state space method was also investigated in a dynamic force reconstruction study done by V. Jayalakshmi et al. [18], based on measured acceleration responses. Two time-domain algorithms were evaluated, the first was a direct method derived from the Newmark- β time marching scheme while the second was an inverse method using the state-space form of the same scheme. The latter proved more successful. A modification to the Tikhonov regularization was also successfully proposed to account for the ill-conditioning of the inverse problem. The Newmark- β method has been described in Section 3.7.2.

A new time domain method called the time-domain Galerkin method (TDGM) was presented in a study by J. Liu et al. [10]. By looking into other time-domain studies they identified the following shortcomings; an accurate identification of forces only applies for small time intervals and the methods show a weak anti-noise performance. In the proposed method, the time domain was discretized into a series of time elements. For each time element, a variety of shape functions were used to find the dynamic load, kernel function response and structural response by using the least-square fitting method. In the analysis, the TDGM was compared to the Green kernel function method and proved more successful in regards to the mentioned shortcomings.

2.4.1 Convolution

In time domain force reconstruction methods, the dynamic excitation forces and the dynamic responses are often related in the form of convolution [1]. The convolution integral of the convoluting signals $k(t)$ and $x(t)$ may be written as follows:

$$y(t) = k(t) * x(t) = \int_0^t k(t - \tau)x(\tau)d\tau \quad (2.5)$$

where $*$ represents the convolution of the signals, $y(t)$ and $x(t)$ represent the impulse response and input signal respectively, and $k(t)$ represents the kernel function. The kernel function represents the system's response to the Dirac delta $\delta(t)$ function illustrated in Figure 2.3. The digital form of the convoluting signals is illustrated in Figure 2.4. [28, 3]

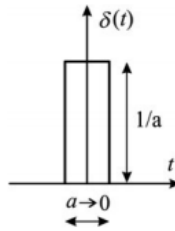


Figure 2.3: The Dirac delta function. [3]

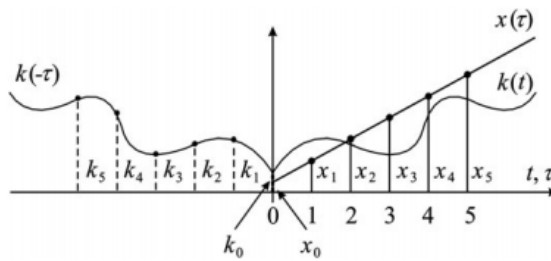


Figure 2.4: The convolution of two signals. [3]

Convolution problems may be rewritten as inverse problems, also called deconvolution. Such problems are often referred to as statistical inverse problems due to the presence of random noise [29].

For an arbitrary deconvolution problem, $\hat{x}(t)$ represents the unknown input signals while $k(t)$ and $y(t)$ are known signals. In the case of measured signals $k(t)$ and $y(t)$ represent the known impulse response of the system and the measured signal respectively. Deconvolution is then established by a reverse transformation of Equation 2.5. The deconvolution may be written in matrix form for $n = N-1$ spectral lines and a shift in spectral lines, Δ , as shown in Equation 2.6. For a damped system, some of the k -values in the lower left of the matrix will become zero as the response from early pulse forces eventually will die out. The deconvolution may also be written in terms of the Fourier transform as shown in Equation 2.7. [3]

$$\begin{bmatrix} \tilde{x}_0 \\ \tilde{x}_1 \\ \tilde{x}_2 \\ \cdot \\ \cdot \\ \tilde{x}_{N-1} \end{bmatrix} = \frac{1}{k_0 \Delta} \left(\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_{N-1} \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & \cdot & \cdot & 0 \\ k_1 & 0 & 0 & \cdot & \cdot & 0 \\ k_2 & k_1 & 0 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ k_{N-1} & k_{N-2} & k_{N-3} & \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_0 \\ \tilde{x}_1 \\ \tilde{x}_2 \\ \cdot \\ \cdot \\ \tilde{x}_{N-1} \end{bmatrix} \right) \quad (2.6)$$

$$\tilde{X}_n = \frac{IF[Re\tilde{X}(e^{j\omega}) + jIm\tilde{X}(e^{j\omega})]}{\Delta} \quad (2.7)$$

Time domain solution of deconvolution problems is especially suitable for early time problems because only a few components of the force is reconstructed. A large time window is required if deconvolution is to be solved using the frequency domain method which is global and not limited to the elapse of the wave propagation time. Hence, frequency domain deconvolution is unfitted for early time problems [30]. In several time domain deconvolution studies of linear systems, the Fourier transform is applied to Equation 2.5 which turns it into the frequency domain. Nevertheless, the resulting excitation force is expressed in the time domain, which makes it a time domain solution.

E. Jaquelin et al. [30] performed a deconvolution study on impact problems in the time domain with the use of the Fourier transform. The deconvolution technique was investigated, in addition to the problems that occurred. The study highlighted the measurement position's influence on the results, where a closer measurement performed a better reconstruction. The effect of the measurement position in addition to the general ill-posedness of the deconvolution problem were handled by introducing different regularization techniques. Adding a boundary condition forcing the solution to be non-negative was suggested, but this regularization was limited to only be applicable to impact forces. Instead, they used Tikhonov regularization and the TSVD method to get a stable solution for any force.

In a study by J. F. Doyle [31] a wavelet deconvolution method for impact force reconstruction was presented. The method was similar to the Fourier method, but overcame some of the challenges related to the frequency domain method. The Fourier method provides a one-to-one relation between frequencies and may therefore result in the loss of certain frequency data. In wave propagation problems there is a prominent time delay between the impact force event and the measured responses. Hence, a one-to-one relation between forces and responses cannot be established which results in a retarded convolution relation.

H. Kalhori et al. [32] studied on whether or not the transfer function in the convolution integral required an explicit establishment in order to reconstruct an impact force. The impact tests were performed on a steel-beam-reinforced concrete deck. The most successful results were obtained without the use of an explicit transfer function. However, the method always required regularization due to the ill-conditioned nature of the deconvolution.

In a study by B. Qiao et al. [33], a sparse deconvolution model was used for impact force reconstruction. The study looked into some of the drawbacks of the Tikhonov regularization method and the TSVD method. In regards to computational cost, these methods does not perform well for large-scale ill-posed inverse problems. A large-scale sparse deconvolution model was therefore constructed. The model was based on the primal-dual interior point method and managed to successfully reconstruct an impact force acting on a wind turbine blade.

2.4.2 Markov Parameters

Time domain techniques are often memory intensive due to the large number of inputs and outputs involved in addition to thousands of datapoints. This drawback was accounted for in a study by D. C. Kammer [34]. In the study, a time domain technique was used to estimate unit force pulses acting on a structure by estimating its so-called Markov parameters. The Markov parameters represent the systems response to unit force pulses at input locations and include the dynamic properties of the structure. Under the assumption of system linearity, forward Markov parameters were used to compute the inverse system Markov parameters. The forward system was written as follows:

$$\mathbf{y}(k) = \sum_{i=0}^k \mathbf{H}_i \mathbf{u}(k-i) \quad (2.8)$$

for the inputs \mathbf{u} , the outputs \mathbf{y} , the time step k and the following Markov parameters:

$$\mathbf{H}_0 = \mathbf{H} \quad \mathbf{H}_i = \mathbf{CA}^{i-1}\mathbf{D} \quad i = 1, 2, 3... \quad (2.9)$$

The corresponding inverse system was written as:

$$\mathbf{u}(k) = \sum_{i=0}^k \mathbf{h}_i \mathbf{y}(k-i) \quad (2.10)$$

given that the inverse system Markov parameters \mathbf{h}_i could be derived.

The linear expression in Equation 2.11 shows the relation between the inverse system Markov parameters and the forward system Markov parameters.

$$\mathbf{H}_0 \mathbf{h}_k = - \sum_{i=1}^k \mathbf{H}_i \mathbf{h}_{k-i} \quad (2.11)$$

for

$$\mathbf{h}_0 = \mathbf{H}_0^+. \quad (2.12)$$

Further, the Tikhonov regularization was used due to the ill-posed nature of the convolution matrix, resulting in a slightly altered well-posed system. The convolution then gave out a set of pseudo-forces which was applied back to the structure. The resulting structure response was closely matching the sensor data measured when the actual force was applied [34].

Later, Tadeusz Uhl [1] used the same principles for force reconstruction in the time domain. By using Markov parameters for the solution of the state space equation, Equation 2.8 and 2.9 was expressed in block-matrix form as follows,

$$\begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \cdot & \cdot & \cdot & \mathbf{H}_N \\ \mathbf{0} & \mathbf{H}_0 & \cdot & \cdot & \cdot & \mathbf{H}_{N-1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{0} & \mathbf{0} & \cdot & \cdot & \cdot & \mathbf{H}_0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_N \\ \mathbf{u}_{N-1} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{u}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{y}_N^0 \\ \mathbf{y}_{N-1}^0 \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{y}_0^0 \end{bmatrix} \quad (2.13)$$

with $(N + 1)^2$ matrix blocks of dimension $n_0 \times n_i$ in the upper-block-triangular matrix, \mathbf{H} . The input forces \mathbf{u} were solved by the least-squares equation shown in Equation 2.14, resulting in the inverse solution in Equation 2.15.

$$\min_u \|\mathbf{H}_0 \mathbf{u} - \hat{\mathbf{y}}\|_2^2 \quad (2.14)$$

$$\mathbf{u} = \mathbf{H}_0^+ \hat{\mathbf{y}}_0 \quad (2.15)$$

As Equation 2.15 produces an ill-posed result, the study introduced a restriction to the solution in terms of an *a priori* bound $\|\mathbf{L}_i \mathbf{u}\|_2$. The reformulation of the problem in the least-square sense became:

$$E = \min_u \left\{ \|\mathbf{H}_0 \mathbf{u} - \hat{\mathbf{y}}\|_2^2 + \lambda \|\mathbf{L}_i \mathbf{u}\|_2^2 \right\} \quad (2.16)$$

for a regularization parameter λ that weights the restriction on \mathbf{u} . The dynamic programming solution can be read in its complete form in Appendix A.2, and is shortly summarized here. The solution was based on the following function:

$$g(c) = \min_{f_j} E_n(y_0, u) \quad (2.17)$$

simulating the system from stage $j = n$ to $j = N$, with $y_n = c$ and optimal inputs u_j 's. When applying minimization and the principle of optimality, the complete solution for one step became:

$$g_{N-1}(c) = \frac{(y_{N-1} - c)^2 + L(y_N - H_0 c)^2}{(L + 1)^2 + [y_N - H_0 c - (y_N - H_0 c)/(L + 1)]^2} \quad (2.18)$$

In the study, the method was used for off-line reconstruction of contact forces for an operating rail vehicle. By basing the algorithm on Tikhonov's regularization, the problem became well-posed which resulted in a correct reconstruction of forces. The method is applicable for both linear and nonlinear models. However, more complex simulation methods are required for the latter in order to achieve correct results. [1]

2.5 The Frequency Domain Method

The frequency domain method often involves obtaining the frequency response functions (FRFs) of a system. The FRFs give a mathematical representation of the relationship between a systems inputs and outputs, and is often based on the Fourier transform.

A frequency domain solution of the forward problem, reconstruction of responses, is often preferred for excitation loads of stochastic character. This is because a time domain solution would require a time domain simulation of the load components at all nodes. Reconstruction in the frequency domain may be obtained in original or modal degrees of freedom, in which the latter is often far more convenient as the size of the relevant matrices

becomes considerably smaller [14].

If $\mathbf{Y}(\omega)$ represents the dynamic displacement amplitude, $\mathbf{Z}^{-1}(\omega)$ represents the inverse of the dynamic stiffness matrix and $\mathbf{F}(\omega)$ represents the external force amplitude, the following equation can be established for a periodic MDOF system.

$$\mathbf{Y}(\omega) = \mathbf{Z}^{-1}(\omega)\mathbf{F}(\omega) \quad (2.19)$$

Writing the equation in terms of the inverse of the dynamic stiffness matrix, called the receptance matrix or frequency response matrix, $\mathbf{H}(\omega)$, yields [13, 14]:

$$\mathbf{Y}(\omega) = \mathbf{H}(\omega)\mathbf{F}(\omega) \quad (2.20)$$

for

$$\mathbf{H}(\omega) = (-\mathbf{M}\omega^2 + \mathbf{C}i\omega + \mathbf{K})^{-1}. \quad (2.21)$$

Lage et al. [17] proposed an algorithm for force identification using the frequency domain method. The algorithm was based on a finite element (FE) model using the Bernoulli-Euler beam theory and the measured dynamic responses of the structure. The force position and quantification were evaluated, in addition to the effect of the measurement position of the sensors. In the study, the system model was assumed known. They presented three methods for force localization were one of them proved most efficient. Comparison of the experimental and numerical FRFs showed a considerably good match. However, a few high peaks, corresponding to the anti-resonances in the FRFs, occurred in the numerical results due to the lack of damping.

2.5.1 Force and Displacement Transmissibility

There have been several attempts for obtaining dynamic transmissibility between applied forces and reaction forces, and transmissibility between the corresponding dynamic displacements. The relation between the transmissibilities depends on the dynamic system to be investigated. For a single-degree-of-freedom (SDOF) system the dynamic force transmissibility equals the dynamic displacement transmissibility. Force transmissibility is defined as the ratio between the force amplitude transmitted to ground and the external force amplitude. Displacement transmissibility is defined as the ratio between the response displacement amplitude and the displacement amplitude at the foundation. For a multiple-degrees-of-freedom (MDOF) system the transmissibilities are not equal. [4]

The FE method (FEM) can be used to establish an expression for a transmissibility matrix in the frequency domain. Figure 2.5 illustrates an arbitrary dynamic system subjected to an external dynamic force of known value which causes a dynamic response of unknown value. The subscript K is used for the known values and the subscript U is used for the unknown values. If the forces are periodic they may be represented by the Fourier series. Otherwise, the Fourier integral should be used [7].

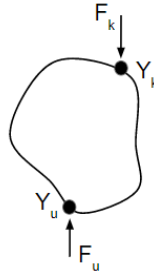


Figure 2.5: An arbitrary dynamic system subjected to a dynamic force which causes a dynamic response.

Y. E. Lage et al. [13, 4] have done several studies on the force transmissibility concept for MDOF systems. The following paragraphs show the derivation of the transmissibility matrix presented in two of their studies.

The system in Figure 2.5 has two nodes of interest; one where force is applied and one reaction node. By using the same subscripts as before, that is, U for unknown reaction forces and K for known applied forces, equation 2.20 becomes:

$$\begin{bmatrix} Y_K \\ Y_U \end{bmatrix} = \begin{bmatrix} H_{KK} & H_{KU} \\ H_{UK} & H_{UU} \end{bmatrix} \begin{bmatrix} F_K \\ F_U \end{bmatrix}. \quad (2.22)$$

By assuming zero displacement at the reaction point, that is $Y_U = 0$, the unknown reaction force may be expressed as:

$$F_U = -(H_{UU})^{-1} H_{UK} F_K \quad (2.23)$$

which results in the transmissibility matrix shown in Equation 2.24.

$$(T_f)_{UK} = -(H_{UU})^{-1}H_{UK} \quad (2.24)$$

If the assumption of zero displacement does not hold, the relation becomes:

$$F_U = (H_{UU})^{-1}(Y_U - H_{UK}F_K). \quad (2.25)$$

Inserting the transformation matrix established from equation 2.24 yields:

$$F_U = (T_f)_{UK}F_K + (H_{UU})^{-1}Y_U. \quad (2.26)$$

Equation 2.23 and 2.26 give two solutions for F_U . These give the solutions to the direct problem where the applied load is a known value. For the inverse problem, the dynamic stiffness matrix should be used, which leads to the following definition of the applied loads:

$$\mathbf{F}(\omega) = \mathbf{Z}(\omega)\mathbf{Y}(\omega) \quad (2.27)$$

By using the same subscripts as above, that is, K for the external force and U for the unknown displacement, in addition to a third subscript, C, which includes the remaining nodes as illustrated in Figure 2.6, the relation becomes:

$$\begin{bmatrix} F_K \\ F_U \\ F_C \end{bmatrix} = \begin{bmatrix} Z_{KK} & Z_{KU} & Z_{KC} \\ Z_{UK} & Z_{UU} & Z_{UC} \\ Z_{CK} & Z_{CU} & Z_{CC} \end{bmatrix} \begin{bmatrix} Y_K \\ Y_U \\ Y_C \end{bmatrix} \quad (2.28)$$

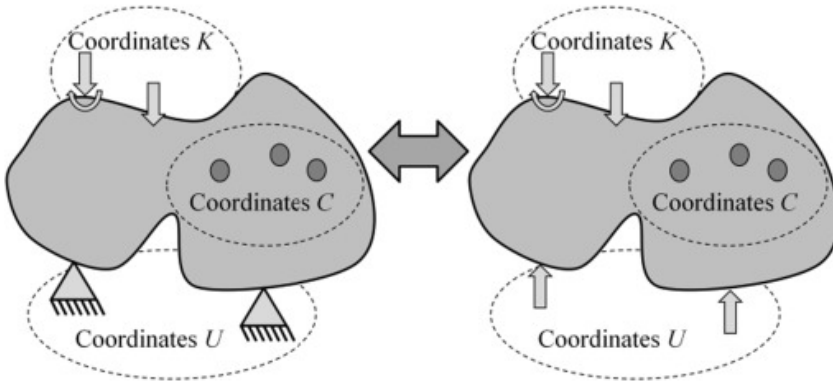


Figure 2.6: The location of coordinates U, K and C on arbitrary dynamic systems [4].

The equation may be simplified by including the subscripts K and C under the same subscript E, which gives:

$$\begin{bmatrix} F_E \\ F_U \end{bmatrix} = \begin{bmatrix} Z_{EE} & Z_{EU} \\ Z_{UE} & Z_{UU} \end{bmatrix} \begin{bmatrix} Y_E \\ Y_U \end{bmatrix}. \quad (2.29)$$

Assuming zero displacement at the supports yields:

$$F_E = Z_{EE}Y_E \quad (2.30)$$

$$F_U = Z_{UE}Y_E. \quad (2.31)$$

Solving for F_U by substituting Y_E results in:

$$F_U = Z_{UE}(Z_{EE})^{-1}F_E. \quad (2.32)$$

Finally, the transmissibility matrix becomes:

$$(T_f)_{UE} = Z_{UE}(Z_{EE})^{-1}. \quad (2.33)$$

In one of the studies, two types of forces were considered in regards to the transmissibility matrix [13]. The problems were defined as:

1. reaction forces estimation (direct problem) with the objective to calculate a set of unknown reactions using information from the known set of applied loads:

$$\mathbf{F}_U = (\mathbf{T}_f)_{UK} \mathbf{F}_K; \quad (2.34)$$

2. applied forces estimation (inverse problem) with the objective of calculating a set of applied forces using information from the known set of reaction forces:

$$\mathbf{F}_K = ((\mathbf{T}_f)_{UK})^+ \mathbf{F}_U, \quad (2.35)$$

[13, p. 3]

where the subscript + represents the pseudoinverse. The pseudoinverse is limited to only be applicable when the number of reactions is larger than or equal to the number of applied forces. The solutions to Equation 2.34 and 2.35 were based on obtaining the transmissibility matrix. The matrix was obtained both numerically and experimentally, and showed acceptably consistent results [13].

In a third study by Y. E. Lage et al. [35], the load vector in Equation 2.19 was also successfully reconstructed for a MDOF system by using the transmissibility concept. For a simple case where all the responses \mathbf{Y} in Equation 2.19 are known, corresponding to the finite element discretization, the forces could be obtained by solving Equation 2.36.

$$\mathbf{F}(\omega) = \mathbf{H}^{-1}(\omega) \mathbf{Y}(\omega) \quad (2.36)$$

However, this is seldom the case. For complex structures, structures with a very large number of coordinates or structures with inaccessible locations, the load vector cannot be calculated directly. Hence, the study proposed a two-step methodology to obtain the load vector based on the assumption that the system was completely known. In the first step, the number of external loads and their locations were identified. Here, an algorithm based on the response transmissibility concept was used to construct a transmissibility matrix correspondent to the dynamics of the system. The number and location of the external loads were known once the transmissibility matrix was constructed. In the second step, force amplitude regeneration was performed based on the displacement transmissibility found in the first step. The obtained frequency response function agreed well with the measured experimental values.

2.6 Determination of Force Location

For structural applications where the location of the dynamic load is unknown, there have been several proposed methods on how to obtain the location of the applied loads in addition to its magnitude.

Sifa Zheng et al. [9] have analyzed the coherence of the transfer function matrix, and performed numerical simulations where the coherence analysis has been compared with the minimum condition number method and the weighted condition number method. The object of the study was a heavy truck cab. The dynamic forces acting between the cab and suspension were identified during operation. The coherence analysis showed most successful while the weighed condition number method provided most inaccurate results.

Bridges subjected to moving vehicles has been the subject of several force reconstruction studies. In such a case, the position of the moving force is unknown. The moving vehicles adds additional masses on the bridge which affects the eigenfrequencies of the system and makes the mass distribution unknown. As a result, the system becomes non-linear. To solve such a system, both the system and force must be identified. The systems are often under-determined which makes it necessary to do some assumptions to stabilize them. L. Yu and T. H. T. Chan [36] studied on moving force reconstruction for bridge-vehicle systems where the bridge deck was modelled as an Euler beam. They used a frequency-time domain method with the assumption of linearity to reconstruct the moving force. The Fourier transform was performed in modal degrees of freedom. The study used two techniques for calculation of the pseudo-inverse; direct calculation and calculation via the singular value decomposition (SVD) technique. The SVD technique provided a reconstructed force with an acceptable accuracy.

3. Theory

In this chapter, some regularization methods and filtering techniques dealing with the ill-posed nature of the inverse problem will be presented. Next, basic structural dynamics will be presented for SDOF and MDOF systems respectively.

3.1 Regularization Methods

Regularization methods are commonly used to obtain numerically stable solutions to inverse problems. These methods often involve applying additional constraints to the system, often in terms of physical or mathematical boundaries. [20]

Sensitivity to noise is often observed by analyzing the singular values of the systems transfer function. A problem is ill-posed if the singular values gradually decay to zero. If there is a well-defined gap between the singular values, the problem is also rank deficient [20]. One solution may be to truncate the singular values, which is done in a method called the singular value decomposition (TSVD). This method and a method called Tikhonov regularization are the most traditional regularization methods [33]. For a regularized problem, these methods give an exact solution in a finite dimension and in a finite number of operations up to the rounded errors. However, they do not perform well for large-scale problems. The matrices obtained after discretization in large-scale problems are often sparse. A solution by Tikhonov regularization or the TSVD will result in a too large number of operations and the structure of the matrices may be affected [37].

Other proposed regularization methods are optimization methods which seek to minimize error estimates. These methods provide a single optimal solution to inverse problems by employing improvements to the least-squares method. Examples of such methods are the conjugate gradient method, also known as the iterative regularization method, and the Levenberg-Marquardt iterative regularization method. The latter method is based on the application of Newton's method to the least-squares problem. The solution is written as $x_{k+1} = x_k + s(x_k)$ for a solution to the problem x_k at iteration k , and a solution x_{k+1} at iteration $k + 1$. Defining the step function, $s(x_k)$, is the main objective, which ensures

that x_k converges to its optimal solution. This method has been used for both linear and non-linear systems. [20]

As the Tikhonov regularization method and the TSVD method are the most used methods, they will be presented the following sections.

3.1.1 Tikhonov Regularization

Given the discrete linear equation $\mathbf{m} = \mathbf{A}\mathbf{f} + \varepsilon$, the classical Tikhonov regularization is the vector $T_\alpha(\mathbf{m}) \in \mathbb{R}^n$ that minimizes the following expression:

$$\|AT_\alpha(\mathbf{m}) - \mathbf{m}\|^2 + \alpha\|T_\alpha(\mathbf{m})\|^2 \tag{3.1}$$

for a regularization parameter $\alpha > 0$ [5]. The vector may then be defined as:

$$T_\alpha(\mathbf{m}) = \arg \min_{\mathbf{z} \in \mathbb{R}^n} \{ \|A\mathbf{z} - \mathbf{m}\|^2 + \alpha\|\mathbf{z}\|^2 \} \tag{3.2}$$

The effect of the regularization parameter for a one-dimensional deconvolution problem is illustrated in Figure 3.1.

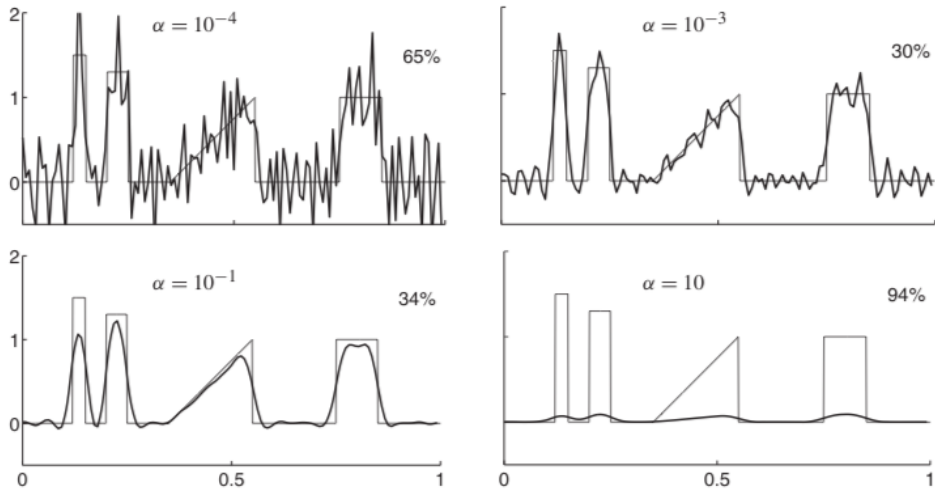


Figure 3.1: Classical Tikhonov regularization for four different regularization parameters with corresponding percentage error of reconstructions. [5]

For a generalized Tikhonov regularization, *a priori* information is given about the solution. Say the vector \mathbf{f} represents a continuous function f , with $f(s_j) = \mathbf{f}_j$. The derivative of the continuum can then be discretized by:

$$\frac{df}{ds}(s_j) \approx \frac{f(s_{j+1}) - f(s_j)}{\Delta s} = \frac{\mathbf{f}_{j+1} - \mathbf{f}_j}{\Delta s}, \quad (3.3)$$

resulting in the discrete differentiation matrix:

$$L = \frac{1}{\Delta s} \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & -1 & 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & & & & \cdot & & \\ \cdot & \cdot & \cdot & & & & & \cdot & \\ \cdot & \cdot & \cdot & & & & & & \cdot \\ 0 & & \cdot & \cdot & \cdot & 0 & -1 & 1 & 0 \\ 0 & & \cdot & \cdot & \cdot & 0 & 0 & -1 & 1 \\ 1 & & \cdot & \cdot & \cdot & 0 & 0 & 0 & -1 \end{bmatrix}. \quad (3.4)$$

Using the discrete differentiation matrix in the reconstruction computations yields for a generalized Tikhonov regularization as shown in Figure 3.2. The generalized reconstruction is smoother and has a more non-zero reconstruction at $\alpha = 10^{-2}$. [5]

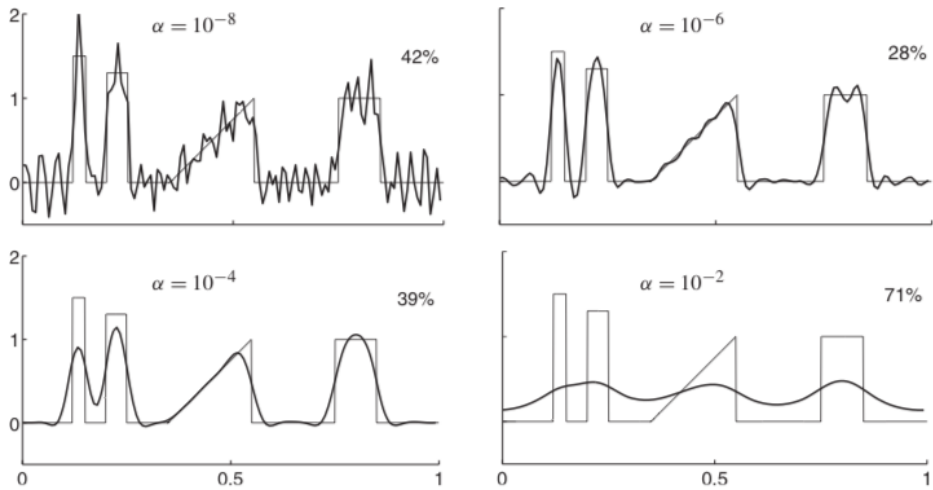


Figure 3.2: Generalized Tikhonov regularization for four different regularization parameters with corresponding percentage error of reconstructions. [5]

The expression for the generalized Tikhonov vector becomes:

$$T_{\alpha}(\mathbf{m}) = \arg \min_{\mathbf{z} \in \mathbb{R}^n} \left\{ \|\mathbf{Az} - \mathbf{m}\|^2 + \alpha \|\mathbf{Lz}\|^2 \right\}. \quad (3.5)$$

If the convolution between two signals is written in terms of the Fourier transform as follows,

$$\mathbf{Y} = \mathbf{KX}, \quad (3.6)$$

for a response signal \mathbf{Y} , an input signal \mathbf{X} and a kernel function \mathbf{K} , the Tikhonov regularization will then revolve around defining a smoothing norm, shown in Equation 3.7, and a trade off between the smoothing norm and the residual norm in Equation 3.6 [30].

$$\Omega(\mathbf{X}) = \|\mathbf{KX}\|^2 \quad (3.7)$$

The expression for the regularization then becomes:

$$\min_{\mathbf{X}} \left\{ \|\mathbf{KX} - \mathbf{Y}\|^2 + \alpha \Omega(\mathbf{X}) \right\}. \quad (3.8)$$

Choosing an optimal regularization parameter, α , is extremely challenging. On the one hand, a low value of α singles out the minimum of the residual norm and thereby results in an unstable solution. On the other hand, a great value of α filters out large frequencies resulting in a too smooth solution [30]. The Morozov's discrepancy principle can be used if the parameter is to be decided based on data noise level. The L-curve method is useful for balancing the norm of the smoothing term and the norm of the residual term. Only the latter method is applicable for the generalized regularization method [5].

3.1.2 The Truncated Singular Value Decomposition Method

The singular value decomposition (SVD) method has been commonly used for linear inverse problems. Theoretically, the decomposition gives a simple solution to the least squares problem [37].

The TSVD may be derived as follows. If $\mathbf{f} \in \mathbb{R}^n$, $\mathbf{m} \in \mathbb{R}^k$ and A is of size $k \times n$, the equation $A\mathbf{f} = \mathbf{m}$ may be solved by $A^+\mathbf{m}$ if A is written in the form of its SVD as $A = UDV^T$. $V \in \mathbb{R}^{k \times k}$ and $U \in \mathbb{R}^{n \times n}$ are orthogonal matrices and $D \in \mathbb{R}^{k \times n}$ is a diagonal matrix. The subscript $+$ denotes the pseudoinverse. The singular values are ordered from large to small, where r is defined as the largest index for a nonzero singular value as follows [5]:

$$r = \max\{j | 1 \leq j \leq \min(k, n), d_j > 0\} \quad (3.9)$$

for

$$d_1 > 0, d_2 > 0, \dots, d_r > 0, d_{r+1} = 0, \dots, d_{\min(k, n)} = 0. \quad (3.10)$$

The minimum norm solution of $A\mathbf{f} = \mathbf{m}$ is then given by $A^+\mathbf{m}$ as follows:

$$A^+\mathbf{m} = VD^+U^T\mathbf{m} \quad (3.11)$$

for

$$D^+ = \begin{bmatrix} 1/d_1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 1/d_2 & & & & & & & & \cdot \\ \cdot & & \cdot & & & & & & & \cdot \\ \cdot & & & \cdot & & & & & & \cdot \\ \cdot & & & & 1/d_r & & & & & \cdot \\ & & & & & 0 & & & & \cdot \\ \cdot & & & & & & \cdot & & & \cdot \\ \cdot & & & & & & & \cdot & & \cdot \\ \cdot & & & & & & & & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & & & \cdot & \cdot & \cdot & 0 \end{bmatrix} \in \mathbb{R}^{n \times k}. \quad (3.12)$$

If \mathbf{f} is written as a linear combination of $\mathbf{f} = \sum_{j=1}^n a_j V_j = V\mathbf{a}$, finding the coefficients a_1, \dots, a_n will give the minimum norm solution of \mathbf{f} [5]. Setting $\mathbf{m}' = U^T \mathbf{m}$ yields:

$$\begin{aligned} \|\mathbf{A}\mathbf{f} - \mathbf{m}\|^2 &= \|UDV^T \mathbf{a} - U\mathbf{m}'\|^2 \\ &= \|D\mathbf{a} - \mathbf{m}'\|^2 \\ &= \sum_{j=1}^r (d_j a_j - \mathbf{m}'_j)^2 + \sum_{j=r+1}^k (\mathbf{m}'_j)^2 \end{aligned} \quad (3.13)$$

when using the orthogonality of U , that is, $\|U\mathbf{z}\| = \|\mathbf{z}\|$ for any vector $\mathbf{z} \in \mathbb{R}^k$. The minimum is defined when $a_j = \mathbf{m}'_j/d_j$ for $j = 1, \dots, r$. The least-square solution of \mathbf{f} becomes:

$$\mathbf{f} = V \begin{bmatrix} d_1^{-1} \mathbf{m}'_1 \\ \cdot \\ \cdot \\ \cdot \\ d_r^{-1} \mathbf{m}'_r \\ a_{r+1} \\ \cdot \\ \cdot \\ \cdot \\ a_n \end{bmatrix}. \quad (3.14)$$

This solution fulfills Hadamard's existence and uniqueness condition, but not the continuity

condition shown in Appendix A.1. Therefore, the truncated SVD must be introduced [5]. The TSVD is defined as $A_\alpha^+ = VD_\alpha^+U^T$ for any $\alpha > 0$, where

$$D_\alpha^+ = \begin{bmatrix} 1/d_1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 1/d_2 & & & & & & & & \cdot \\ \cdot & & \cdot & & & & & & & \cdot \\ \cdot & & & \cdot & & & & & & \cdot \\ \cdot & & & & 1/d_{r_\alpha} & & & & & \cdot \\ & & & & & 0 & & & & \cdot \\ \cdot & & & & & & \cdot & & & \cdot \\ \cdot & & & & & & & \cdot & & \cdot \\ \cdot & & & & & & & & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & & & \cdot & \cdot & \cdot & 0 \end{bmatrix} \in \mathbb{R}^{n \times k} \quad (3.15)$$

and

$$r_\alpha = \min \left\{ r, \max \left\{ j \mid 1 \leq j \leq \min(k, n), d_j > \alpha \right\} \right\}. \quad (3.16)$$

By defining a reconstruction function \mathcal{L}_α as:

$$\mathcal{L}_\alpha(m) = VD_\alpha^+U^T m, \quad (3.17)$$

all three conditions for a well-posed problem holds. The reconstruction function, $\mathcal{L}_\alpha : \mathbb{R}^k \rightarrow \mathbb{R}^n$, is a well-defined, single-valued linear mapping with norm:

$$\|\mathcal{L}_\alpha\| = \|VD_\alpha^+U^T\| \leq \|V\| \|D_\alpha^+\| \|U^T\| = \|D_\alpha^+\| = d_{r_\alpha}^{-1} \quad (3.18)$$

implying continuity. [5]

3.2 Filtering Techniques

The ill-posedness of inverse problems may also be handled through filtering techniques. Digital filtering techniques are used to filtrate unwanted frequencies, such as high frequency

areas arising due to noise in the system inputs or outputs. Butterworth, Chebyshev, Cauer, and Bessel are the most common types of digital filters [38].

A low-pass filter is generally used to filter out high frequencies and is often based on the Fourier transform. The filter may be converted into high-pass, bandpass and bandstop filters by using appropriate transformations. For an ideal low-pass filter in a distortion-less system of harmonic input signals the following relationship may be established for an input and output signal of $x(t)$ and $y(t)$ respectively, and a signal multiplication factor k [38].

$$y(t) = kx(t - t_0) \quad (3.19)$$

By applying the Fourier transform, the transform function may be written as follows for a frequency variable $j\omega$ [38].

$$H(j\omega) = \frac{Y(j\omega)}{X(j\omega)} = ke^{-j\omega t_0} \quad (3.20)$$

For Butterworth, Chebyshev, Cauer, and Bessel filters the RLC network structure is bounded, meaning a maximum value exists for the filter transfer function. This is accounted for by assuming transfer functions of squared magnitude as shown in Equation 3.21. [38]

$$|H(j\omega)|^2 = \frac{1}{1 + |K(j\omega)|^2} \quad (3.21)$$

The function $K(j\omega)$ assumes low values between $\omega=0$ and the filter cutoff frequency ω_0 , and increases rapidly above the cutoff frequency [38]. The Butterworth filter is briefly described in the next section.

3.2.1 Butterworth Filter

Butterworth filters of N filter orders may be expressed as follows:

$$|H(j\omega)|^2 = \frac{1}{1 + \frac{\omega^2}{\omega_c^2} N}. \quad (3.22)$$

By introducing a complex frequency P and the so-called Butterworth polynomial, $B_n(P)$, the Butterworth filter may be simplified to:

$$H(P) = \frac{1}{B_n(P)} \quad (3.23)$$

for

$$B_n(P) = (P - P_{\infty 1})(P - P_{\infty 2}) \dots (P - P_{\infty N}). \quad (3.24)$$

A 1st order Butterworth filter is shown in Equation 3.25. [38]

$$H(P) = \frac{1}{1 + P} \quad (3.25)$$

3.3 Linear SDOF Systems

A SDOF mass-spring system is illustrated in Figure 3.3. The spring stiffness is defined by k , the viscous damping is defined by c and the mass of the specimen is defined by m . The displacements $u(t)$ and $x(t)$ illustrates the motion of the wall and the motion of the specimen respectively. For such a system subjected to an external force, the damping causes a resistance force F_d that will, in addition to the spring force F_s , work against the external force F_{ext} as illustrated in the figure.

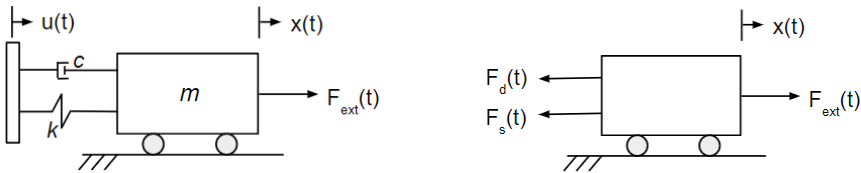


Figure 3.3: Idealized damped SDOF system.

By using Newton's second law of motion, the following dynamic equilibrium expression can be established for a damped mass-spring system.

$$F_{ext}(t) - F_s(t) - F_d(t) = m\ddot{x} \quad (3.26)$$

The viscous damper force F_d is proportional to the velocity of its elongation as shown in Equation 3.27.

$$F_d = c\dot{x} \quad (3.27)$$

The spring force F_s is proportional to its elongation as shown in Equation 3.28.

$$F_s = kx \quad (3.28)$$

Given a harmonic, perfectly resonant external load, F_{ext} can be expressed as follows.

$$F_{ext} = F_0 \sin(\omega_n t) \quad (3.29)$$

The displacement $x(t)$ is defined by the motion of the wall and the elongation of the spring as follows.

$$x(t) = u(t) + \frac{F_{spr}(t)}{k} \quad (3.30)$$

Based on these equations an expression for the unknown external force can be established.

$$F_{ext}(t) = m[\ddot{u}(t) + \frac{\ddot{F}_s(t)}{k}] + c[\dot{u} + \frac{\dot{F}_s}{k}] + ku + F_s \quad (3.31)$$

For a rigid body analysis it is reasonable to assume zero spring deflection. This will in turn result in a theoretically infinite stiffness, reducing equation 3.30 and 3.31 as follows.

$$x(t) \approx u(t) \quad (3.32)$$

$$F_{ext}(t) = m\ddot{u}(t) + c\dot{u} + ku + F_s \quad (3.33)$$

By connecting the expressions for the spring force from Equation 3.30 and 3.26, a third equation set can be established.

$$m\ddot{x} + kx = F_{ext} + ku - F_d \quad (3.34)$$

3.4 Nonlinear SDOF Systems

For nonlinear systems, the viscous damping force and the spring force are no longer assumed proportional to the velocity and displacement respectively. When dealing with nonlinear systems the forces are usually expressed as functions of higher order displacements and velocities [24].

For systems of nonlinear behavior the equation of motion becomes:

$$m\ddot{x} + f(x, \dot{x}, t) = 0. \quad (3.35)$$

There have been several studies on approximating the transient response of a system from first-order nonlinear equations. The methodology used is often based on the Runge-Kutta Methods and the Multistep Methods. Numerical integration methods used to solve second-order nonlinear differential equations are often based on step-by-step procedures. For second-order nonlinear equations at times t_i and $t_{i+1} = t_i + \Delta t_i$ the equation of motion becomes [7]:

$$\begin{aligned} m\ddot{x}_i + F_{di} + F_{si} &= F_{ext,i} \\ m\ddot{x}_{i+1} + F_{d(i+1)} + F_{s(i+1)} &= F_{ext,i+1}. \end{aligned} \quad (3.36)$$

For nonlinear systems, the spring force and damping force are of nonlinear behavior. The forces may be written with the following approximations:

$$\begin{aligned} \Delta F_{si} &\approx k_i \Delta u_i \\ \Delta F_{di} &\approx c_i \Delta \dot{u}_i \end{aligned} \quad (3.37)$$

where k_i and c_i equals the tangent slope at u_i and the tangent slope of F_d versus \dot{u} at curve \dot{u}_i respectively. These two tangent slopes should be evaluated at the start of each time step in a step-by-step integration. [7]

The acceleration at time t_i , \ddot{u}_i , may be calculated from Equation 3.38 or 3.39 for both linear and nonlinear systems.

$$\Delta \ddot{u}_i = \frac{4}{\Delta t_i^2} (\Delta u_i - \dot{u}_i \Delta t_i) - 2\ddot{u}_i \quad (3.38)$$

$$\begin{aligned}
u_{i+1} &= u_i + \Delta u_i \\
\dot{u}_{i+1} &= \dot{u}_i + \Delta \dot{u}_i \\
\ddot{u}_{i+1} &= \ddot{u}_i + \Delta \ddot{u}_i
\end{aligned}
\tag{3.39}$$

In order to obtain dynamic equilibrium for nonlinear systems, c and k should remain approximately constant over the time step Δt_i . Therefore, it is essential that the time step is small enough to minimize the difference between a linear solution and a nonlinear solution [7].

In a load reconstruction study by Sundermeyer [39], an iterative scheme for nonlinear systems was presented. The scheme was based on Equation 3.32, 3.33 and 3.34 for linear systems, but became applicable to nonlinear systems by considering the stiffness as $k(u)$. If $u(t)$, F_s , k and m were considered known values, the iterative representation for finding the unknown values $x(t)$ and F_{ext} was expressed schematically as:

$$\boxed{
\begin{aligned}
x_1(t) &= u(t) \\
F_{ext}^n &= m\ddot{x}_n + F_s \\
m\ddot{x}_{n+1} + kx_{n+1} &= F_{ext}^n + ku
\end{aligned}
}
\tag{3.40}$$

When including a damping term, $c(u)$, resulting in a damping force F_d , the extended iterative scheme becomes:

$$\boxed{
\begin{aligned}
x_1(t) &= u(t) \\
F_{ext}^n &= m\ddot{x}_n + F_d + F_s \\
m\ddot{x}_{n+1} + kx_{n+1} &= F_{ext}^n + ku + F_d
\end{aligned}
}
\tag{3.41}$$

3.5 Undamped SDOF systems

An undamped mass-spring system is highly sensitive to an external load with a frequency equal to the systems natural frequency. A systems natural frequency is defined as the point where the system is not subjected to an external force. Figure 3.4 illustrates the damping effect for the ratio between the forcing frequency, ω , and the systems natural frequency, ω_n .

For an undamped system where the ratio $\frac{\omega}{\omega_n}$ equals one, we have resonance. At this point, the dynamic amplification factor goes to infinity. [6]

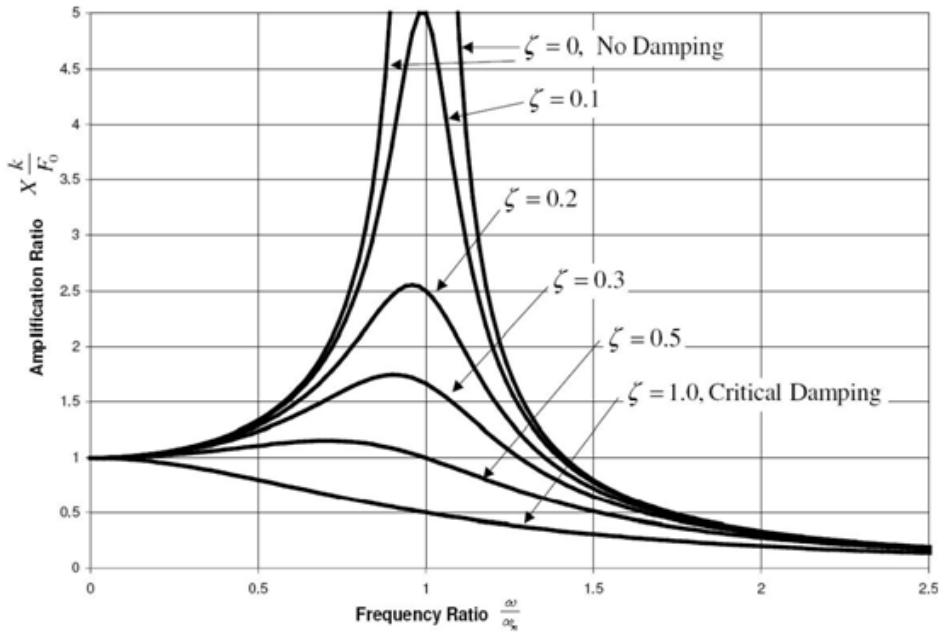


Figure 3.4: Amplification ratio versus frequency ratio for a SDOF system of harmonic oscillations. [6]

3.6 MDOF Systems

SDOF systems does not always provide sufficient description of a structures dynamic behavior. Therefore, real systems are more commonly analyzed in terms of MDOF [7]. The dynamic equilibrium expression for an MDOF system in matrix form is shown in Equation 3.42.

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{F}_{ext}(t) \quad (3.42)$$

\mathbf{M} , \mathbf{C} and \mathbf{K} represent the mass, viscous damping and stiffness matrices respectively. $\mathbf{F}_{ext}(t)$ equals the external load vector. $\mathbf{x}(t)$, $\dot{\mathbf{x}}(t)$ and $\ddot{\mathbf{x}}(t)$ equals the displacement, velocity and acceleration time history responses respectively.

There are generally three strategies for obtaining dynamic responses for MDOF systems as shown in Figure 3.5. The mode-superposition methods are only applicable for linear systems. When looking into nonlinear systems or systems with coupled damping, the direct integration method is more appropriate [7]. The direct integration method will be presented in the following section.

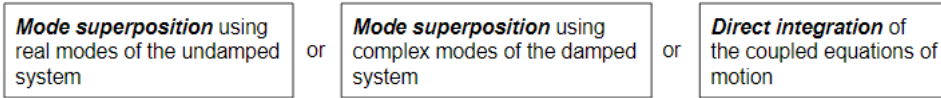


Figure 3.5: Strategies for obtaining dynamic responses in MDOF systems. [7]

3.7 Direct Integration

Three methods have been proposed for carrying out step-by-step numerical integration of second-order MDOF systems, namely the Central Difference Method, the Newmark- β method and the Wilson- θ Method. The two former methods are used for the direct integration of linear systems, while the latter method is applicable for some classes of nonlinear systems. [7]

3.7.1 The Central Difference Method

The Central Difference Method is commonly used in structural dynamics due to its simplicity and ability to be implemented into more complex algorithms. The foundation of the central difference algorithm is expressed in Equation 3.43, and its second derivative in Equation 3.44. [7]

$$\dot{\mathbf{x}}_n = \frac{\mathbf{x}_{n+1} - \mathbf{x}_{n-1}}{2h} \quad (3.43)$$

$$\ddot{\mathbf{x}}_n = \frac{\{\mathbf{x}_{n+1} - 2\mathbf{x}_n + \mathbf{x}_{n-1}\}}{h^2} \quad (3.44)$$

By substituting the two mentioned equations to the equation of motion (Equation 3.42), the following expression can be established:

$$\left(\frac{1}{h^2}\mathbf{M} + \frac{1}{2h}\mathbf{C}\right)\mathbf{x}_{n+1} + \left(\mathbf{K} - \frac{2}{h^2}\mathbf{M}\right)\mathbf{x}_n + \left(\frac{1}{h^2}\mathbf{M} - \frac{1}{2h}\mathbf{C}\right)\mathbf{x}_{n-1} = \mathbf{f}_n \quad (3.45)$$

The algorithm is stable if the time step h is smaller than a critical time step h_{cr} . This requires a certain qualitative knowledge of the response frequencies for the system in order to select an appropriate time step. [7]

3.7.2 The Newmark- β Method

The Newmark- β Method is known for its unconditionally stable character. It is implicit and based on Taylor series approximation. The linear method is based on Equation 3.46 and 3.47.

$$\dot{\mathbf{x}}_{n+1} = \dot{\mathbf{x}}_n + \underbrace{[(1-\gamma)\ddot{\mathbf{x}}_n + \gamma\ddot{\mathbf{x}}_{n+1}]}_{\text{weighted average}} h \quad (3.46)$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \dot{\mathbf{x}}_n h + \underbrace{\left[\frac{(1-2\beta)\ddot{\mathbf{x}}_n + 2\beta\ddot{\mathbf{x}}_{n+1}}{2}\right]}_{\text{weighted average}} \frac{h^2}{2} \quad (3.47)$$

By substituting the two equations with the equation of motion (Equation 3.42), the expression for the Newmark- β method becomes:

$$\mathbf{M}\ddot{\mathbf{x}}_{n+1} + \mathbf{C}\dot{\mathbf{x}}_{n+1} + \mathbf{K}\mathbf{x}_{n+1} = \mathbf{f}_{n+1} = \mathbf{f}(t_{n+1}) \quad (3.48)$$

3.7.3 The Wilson- θ Method

The Wilson- θ Method is based on the assumption that each component \ddot{u}_r of the acceleration vector $\ddot{\mathbf{u}}$ is linear in relation to time during an extended time step $s_i = \Theta\Delta t_i$. Equation 3.49 and 3.50 results from the assumption of linearity as follows:

$$\Delta_s \dot{\mathbf{x}}(t_i) = s_i \ddot{\mathbf{x}}(t_i) + \frac{s_i}{2} \Delta_s \ddot{\mathbf{x}}(t_i) \quad (3.49)$$

$$\Delta_s \mathbf{x}(t_i) = s_i \dot{\mathbf{x}}(t_i) + \frac{s_i^2}{2} \ddot{\mathbf{x}}(t_i) + \frac{s_i^2}{6} \Delta_s \ddot{\mathbf{x}}(t_i) \quad (3.50)$$

where

$$\begin{aligned}\Delta_s \mathbf{x}(t_i) &= \mathbf{x}(t_i + s_i) - \mathbf{x}(t_i) \\ \Delta_s \dot{\mathbf{x}}(t_i) &= \dot{\mathbf{x}}(t_i + s_i) - \dot{\mathbf{x}}(t_i) \\ \Delta_s \ddot{\mathbf{x}}(t_i) &= \ddot{\mathbf{x}}(t_i + s_i) - \ddot{\mathbf{x}}(t_i)\end{aligned}\tag{3.51}$$

When solving for $\Delta_s \ddot{\mathbf{u}}(t_i)$ and $\Delta_s \dot{\mathbf{u}}(t_i)$ and substituting into the equation of motion (3.42), the following equation can be established:

$$\mathbf{M} \Delta_s \ddot{\mathbf{u}}(t_i) = \mathbf{C}(t_i) \Delta_s \dot{\mathbf{u}}(t_i) + \mathbf{K}(t_i) \Delta_s \mathbf{u}(t_i) = \hat{\mathbf{f}}(t_i + s_i) - \mathbf{f}(t_i)\tag{3.52}$$

for

$$\hat{\mathbf{f}}(t_i + s_i) = \mathbf{f}(t_i) + \Theta[\mathbf{f}(t_i + \Delta t_i) - \mathbf{f}(t_i)]\tag{3.53}$$

3.8 Damping

In structural dynamics it is known that for any linear elastic system of free unloaded motion the oscillating motion will eventually stop. Thus, such systems are always subjected to a certain amount of damping forces. Damping forces may originate from stress-strain fluctuations in the material, friction in joints, resistance due to the surrounding environment, etc [14]. The exact nature of damping in dynamic structures is often unknown. Therefore, structures are generally assumed to exhibit viscous damping [7]. For simple structures of nonlinear behavior, the damping results from the development of viscous forces proportional to the velocity. For complex structures of same behavior, the viscosity term is often time-dependent [2].

The expression for the generalized damping matrix is shown in Equation 3.54 for a physical damping matrix \mathbf{C} , a damping matrix of principal coordinates \mathbf{C} and a modal matrix of undamped normal modes Φ . [7]

$$\mathbf{C} = \Phi^T \mathbf{C} \Phi\tag{3.54}$$

In nonlinear structural dynamics, a physical damping matrix should be used. This applies for the direct integration strategy. A physical modal damping matrix may be attained by

using Rayleigh damping or modal damping [7]. Every elastic part in an arbitrary mechanism of nonlinear behavior are typically subjected to Rayleigh damping. In addition, every joint exhibit discrete damping. The combined effects result in a complex damping matrix.

3.8.1 Rayleigh Damping

Rayleigh damping defines a system damping matrix of proportional damping. If \mathbf{K} represents an initial tangent stiffness matrix or \mathbf{C} is modified with each change in stiffness for a nonlinear system, the Rayleigh damping could be written as shown in Equation 3.55 [7].

$$\mathbf{C} = a_0\mathbf{M} + a_1\mathbf{K} \quad (3.55)$$

The two constants, a_0 and a_1 , define the modal damping factors for two selected modes.

In generalized form, the Rayleigh damping matrix becomes:

$$\mathbf{C} = \Phi^T \mathbf{C} \Phi = \text{diag}(a_0 + a_1\omega_r^2)M_r = \text{diag}(C_r) = \text{diag}(2\zeta_r\omega_r M_r) \quad (3.56)$$

for a system damping ratio, ζ_r , defined as:

$$\zeta_r = \frac{1}{2} \left(\frac{a_0}{\omega_r} + a_1\omega_r \right) \quad (3.57)$$

The damping ratio as a function of change in frequency is illustrated in Figure 3.6. For lower frequencies, the Rayleigh damping will mainly be defined by a mass-proportional model ($\alpha_1 = 0$), while as larger frequencies are mainly defined by a stiffness-proportional model ($\alpha_0 = 0$) [40].

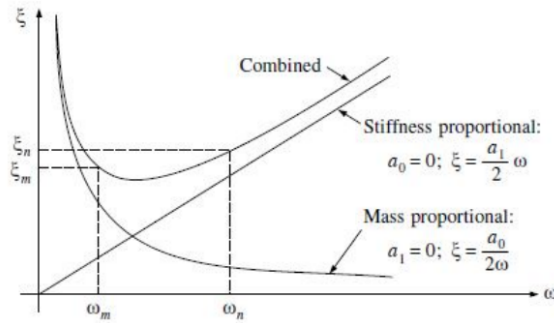


Figure 3.6: Damping ratio as a function of frequency. [8]

3.8.2 Modal Damping

A modal damping matrix is useful when a limited number of lower-frequency modes make important elements for calculation of structural responses. Equation 3.58 shows the expression for the modal damping matrix, \mathbf{C} . [7]

$$\mathbf{C} = \sum_{r=1}^{N_c} \frac{2\zeta_r \omega_r}{\mathbf{M}_r} (\mathbf{M}\phi_r)(\mathbf{M}\phi_r)^T \quad (3.58)$$

The above equation applies for modal damping ϕ_r up to N_c frequency modes. For higher-frequency modes ($N_c + 1, N_c + 2, \dots$) the damping matrix modifies to [7]:

$$\mathbf{C} = a_1 \mathbf{K} + \sum_{r=1}^{N_c-1} \frac{2\hat{\zeta}_r \omega_r}{\mathbf{M}_r} (\mathbf{M}\phi_r)(\mathbf{M}\phi_r)^T \quad (3.59)$$

for

$$\begin{aligned} a_1 &= \frac{2\zeta_{N_c}}{\omega_{N_c}} \\ \hat{\zeta}_r &= \zeta_r - \zeta_{N_c} \frac{\omega_r}{\omega_{N_c}} \end{aligned} \quad (3.60)$$

The modal damping will then be defined as

$$\zeta_r = \begin{cases} \text{value specified,} & r = 1, 2, \dots, N_c \\ \zeta_{N_c} \frac{\omega_r}{\omega_{N_c}}, & r = N_c + 1, N_c + 2, \dots, N \end{cases} \quad (3.61)$$

4. Experiment

In this chapter, two tests on simple mass-spring systems have been performed. The tests were based on previously discussed time domain load reconstruction methods. The reconstructed loads were of periodic and impulse character respectively, and the experiments were performed in Fedem. All units have been expressed in terms of SI units.

4.1 Periodic Load Reconstruction

The periodic load reconstruction experiment was based on Sundermeyer's force reconstruction study presented in Section 3.4. Unlike Sundermeyer's study, this study also looked into the effect from structural damping. Thus, the extended iterative scheme presented in Equation 3.41 has been used as a basis for the study.

The experiment was performed on a simplified excavator. It is challenging to directly measure the load carried by an excavator, which makes it a good subject for an inverse load reconstruction experiment. During the experiment, all properties were of linear behavior. Excavators are typically of non-linear behavior in relation to position, but may be considered linear during sufficiently small time intervals. Sundermeyer's study is not restricted to a certain linearity and should perform equally for linear and nonlinear systems.

4.1.1 Model Setup

The excavator was modelled in a 2-dimensional environment as illustrated in Figure 4.1. The frame of the excavator was modelled as a 2 meter long beam element connected to ground in one end. The joint was given a free rotation in y-direction and fixed in all other directions. An external load acting in the negative z-direction was applied to a triad on the opposite end of the frame as shown. The frame was supported by a spring illustrated with purple color. The spring was connected to the beam by a triad and to ground by a fixed joint.

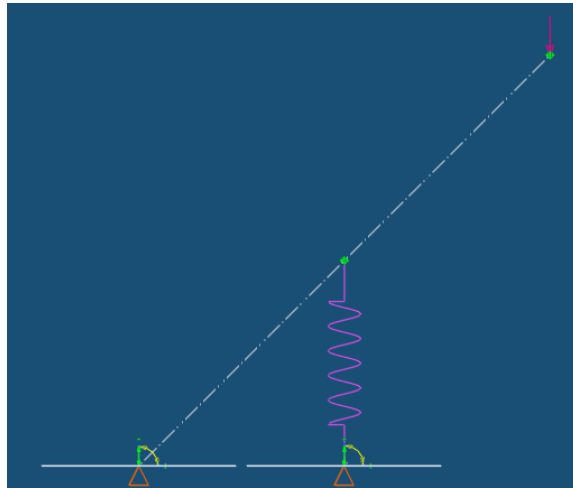


Figure 4.1: 2-dimensional excavator.

The excavator was given the dynamic properties listed in Table 4.1. Some of these values have been changed during the experiments to emphasize their influence in the load reconstruction. However, the ones listed may be considered as standard values used in all simulations unless stated otherwise. The system's eigenfrequencies are listed in Table 4.2.

Table 4.1: Dynamic properties assigned to the excavator.

Properties	Value
Spring stiffness [N/m]	10^5
Constant spring deflection	0
Stiffness proportional damping	0.0001
Mass proportional damping	0
Scaling of dynamic properties:	
Stiffness [N/m]	100
Mass [kg]	0

Table 4.2: The eigenfrequencies of the excavator

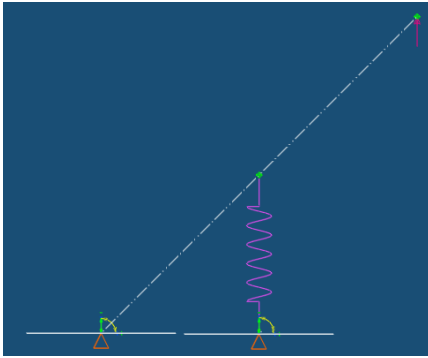
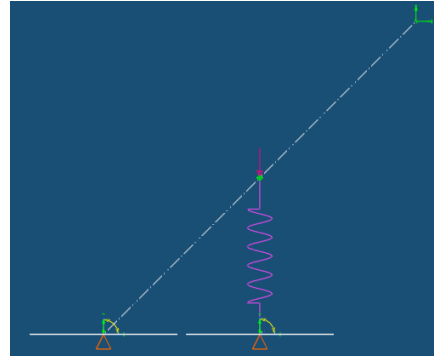
Eigenfrequencies [Hz]
8.21767E+00
5.17489E+01
1.99566E+02
3.19812E+02
4.84409E+02
8.82548E+02
9.50520E+02
1.70020E+03
1.73288E+03
2.75460E+03

In order to perform the iterative experiment, three independent and equal models of the simplified excavator were made. If the experiment was performed on a physical excavator in stead of a virtual, only two models would be required, representing two digital twins. A third model was included in this experiment to represent the physical excavator. The physical excavator would in reality be equipped with sensors measuring the dynamic responses. The dynamic responses would be given in terms of dynamic spring deflection and dynamic spring force as these parameters are usually measurable for the structure in question. For simplicity, this process was done in the third model. Thus, the spring deflection and spring force were given directly from the Fedem simulation. The third model will be referred as the “physical excavator”.

The responses attained from the physical excavator were used as inputs to the digital twins, referred to as Twin 1 and Twin 2. In Twin 1, the spring was given a pre-defined stress free length equal to the spring deflection from the physical excavator. To avoid additional spring influence in the system, the spring was given a considerably larger stiffness than the spring used in physical excavator. In Twin 2, the spring was not assigned any stiffness and was solely acting based on the spring force exported from the physical excavator. The dynamic properties deviating from the ones used for the physical excavator are listed in Table 4.3. For a given applied load, these properties remained the same through each experiment. The applied load was considered unknown in Twin 1 and Twin 2, as the load was to be reconstructed in the experiment. The excavator models used in Twin 1 and Twin 2 are equal to the one presented earlier, and are shown in Figure 4.2 and 4.3 respectively.

Table 4.3: Dynamic properties assigned to the digital twins.

	Twin 1	Twin 2
Spring stiffness [N/m]	10^9	0
Scaling of dynamic properties:		
Stiffness [N/m]	1000	100

**Figure 4.2:** Excavator model - Twin 1.**Figure 4.3:** Excavator model - Twin 2.

4.1.2 Iteration Process

Twin 1 and Twin 2 were subjects for the iteration process. The process evolved around obtaining values from a triad located at the top of the excavator frame at the expected position of the external load. The triad had two measurements of interest; displacement and force. The triad displacement has been termed “position in z-direction” and the triad force has been termed “estimated force”. Figure 4.4 illustrates the iteration process, explained in terms of the parameters listed in Table 4.4. In the first iteration, Twin 1 and Twin 2 read the cylinder deflection and cylinder force from the physical excavator respectively. These parameters were used in every iteration, but were only initialized in the first iteration. When running the simulation, the output from Twin 1, u_s , was given as an input in Twin 2. Twin 2 estimated an applied load that was compared with the previously estimated load. If the loads corresponded, convergence was reached and the iteration process would stop. Otherwise, iteration would continue until correspondence was achieved. All time integrations in Fedem were calculated by the Newmark- β algorithm [41].

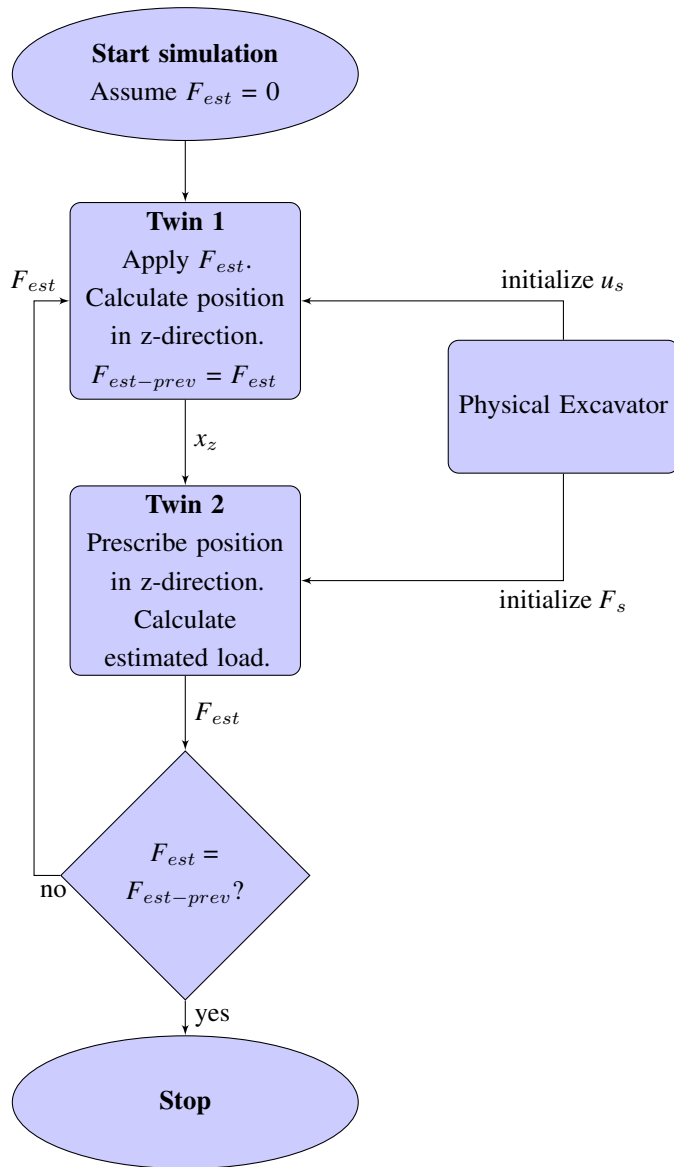


Figure 4.4: Iteration process flow chart. Blue ellipse illustrates process start and end. Blue blocks illustrates the process steps. Blue diamond illustrates decision.

Parameter	Description	Unit
u_z	Cylinder deflection	mm
x_s	Position in z-direction	mm
F_s	Cylinder force	N
F_{est}	Estimated load	N
$F_{est-prev}$	Previously estimated load	N

Table 4.4: Iteration parameters.

During the experiment, the effect from varying properties like spring stiffness and structural damping was looked into for certain applied load frequencies. For this purpose, only one iteration was run each time. The load applied to the physical excavator was given a constant amplitude of 100 N during the entire experiment while the effect of change in frequency was looked into. Load estimation was performed for applied load frequencies below the first eigenfrequency, at resonance and above the first eigenfrequency. All simulations were run with a time step of 0.001 seconds and a total time period of 5 seconds. A relatively long time period was chosen due to the interference of noise during the first few seconds.

4.1.3 Variation in Spring Stiffness

The effect from changing the spring stiffness in the physical excavator was looked into. An external load frequency of 3.0 Hz was used in the physical excavator, and two simulations were run with a spring stiffness of 10^5 N/m and 10^7 N/m respectively. A relatively large increase in stiffness was chosen to make the change more prominent. The estimated load from the simulation using a spring stiffness of 10^5 N/m is shown in Figure 5.5 and 4.6. The estimated load from the simulation using a spring stiffness of 10^7 N/m is shown in Figure 4.7 and 4.8.

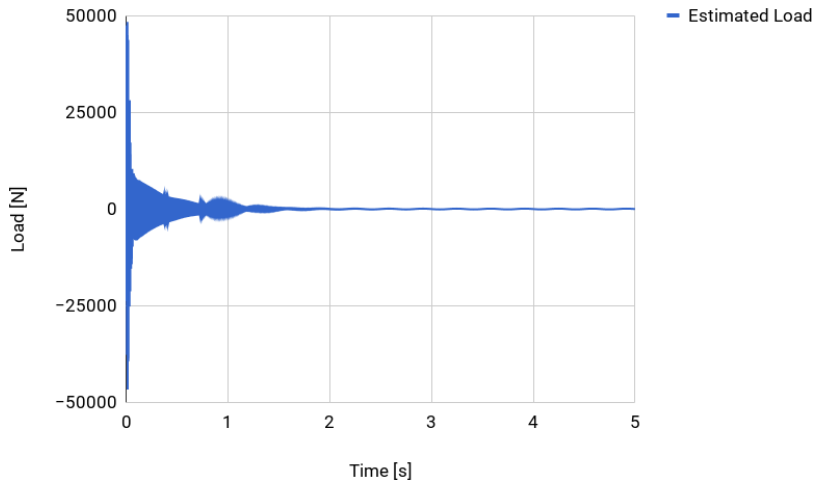


Figure 4.5: Estimated load in a 5 seconds time interval, using a spring stiffness of 10^5 N/m.

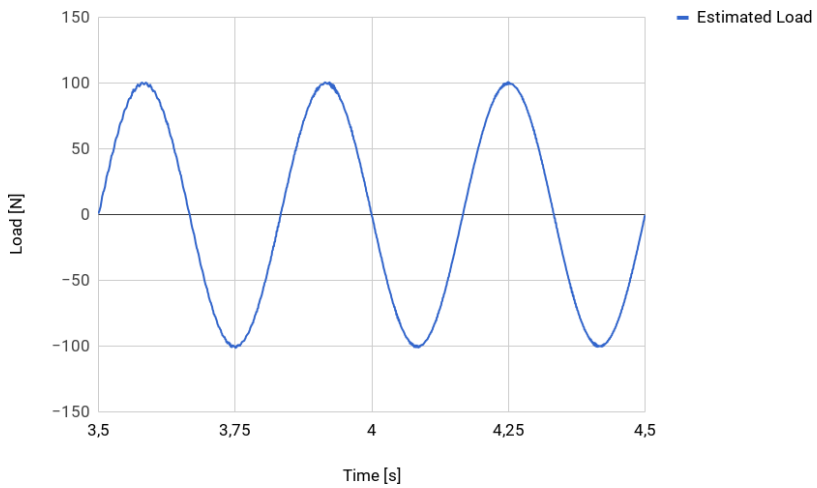


Figure 4.6: Estimated load at area of convergence, using a spring stiffness of 10^5 N/m.

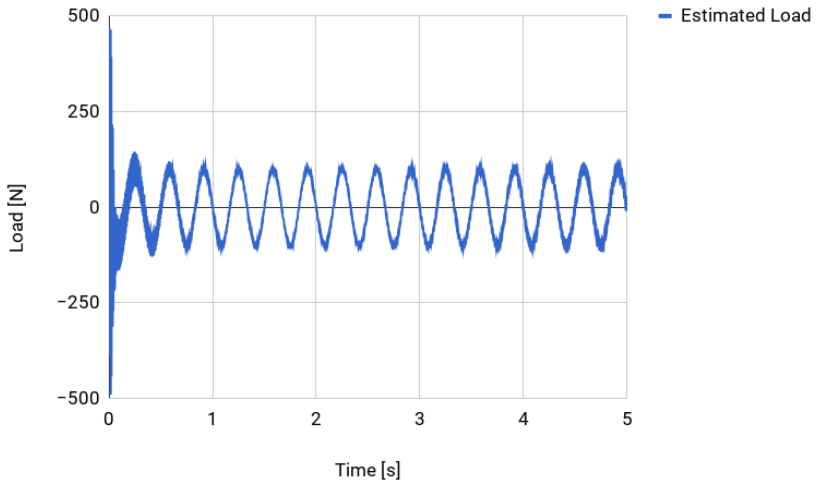


Figure 4.7: Estimated load in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.

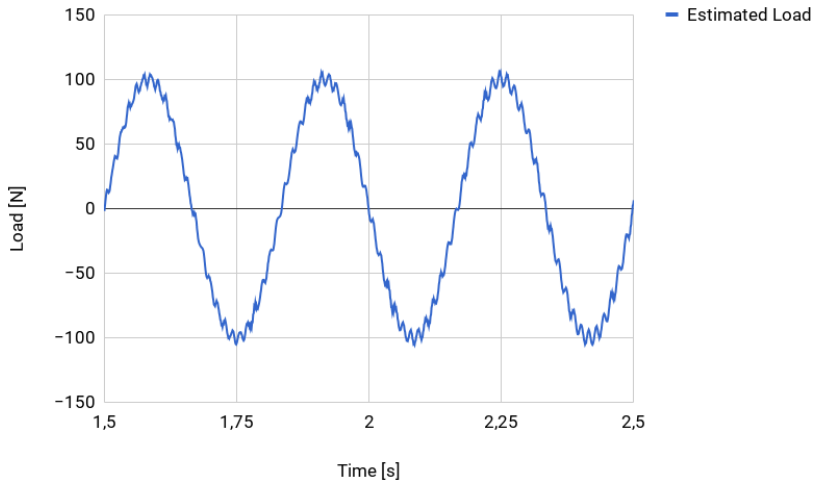


Figure 4.8: Estimated load at area of convergence, using a spring stiffness of 10^7 N/m.

As observed, the second simulation had a shorter high frequency area in the beginning of the simulation but generally an increase in noise in the entire curve. This was also observed for applied loads of higher frequencies as shown in Appendix A.3. The former simulations

gave a smoother graph after convergence.

4.1.4 Variation in Structural Damping

The effect of increasing the structural damping was investigated in order to find an appropriate amount of damping which gave the most accurate estimated load. The structure was assumed to exhibit Rayleigh damping. The relative damping was therefore given by Equation 3.55 presented in the theory presented in the previous chapter. Due to the structure's relatively large eigenfrequency, the Rayleigh damping was mainly defined by the stiffness proportional area in Figure 3.6. The value for the stiffness proportional was therefore changed in the excavator and the digital twins for each simulation. The relative damping for each chosen stiffness proportional and for an α_0 equals zero gave the values presented in Table 4.5.

Table 4.5: Relative damping for each chosen stiffness proportional

Stiffness Proportional	Relative damping
0.0001	0.00258
0.0003	0.00774
0.0009	0.0232

The variation in structural damping at the point of convergence for an applied load of frequency 3.0 Hz is shown in Figure 4.9, 4.10 and 4.11. The same time interval was chosen for each curve to emphasize the effect from an increased damping proportional.

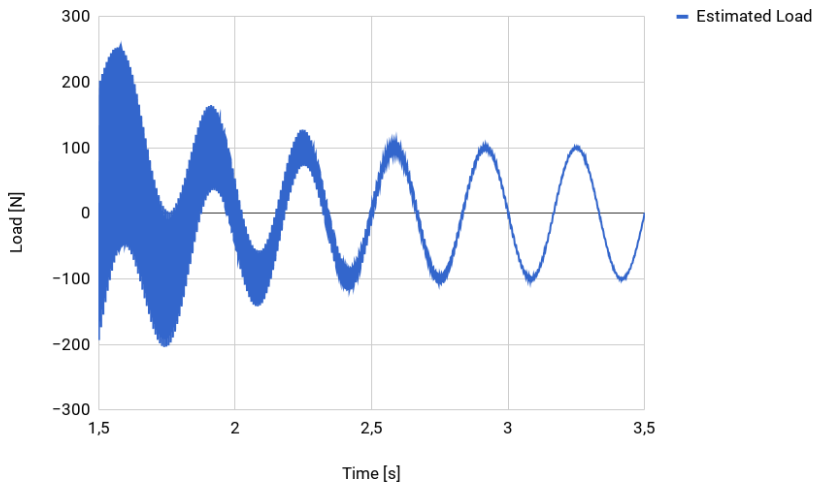


Figure 4.9: Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0001.

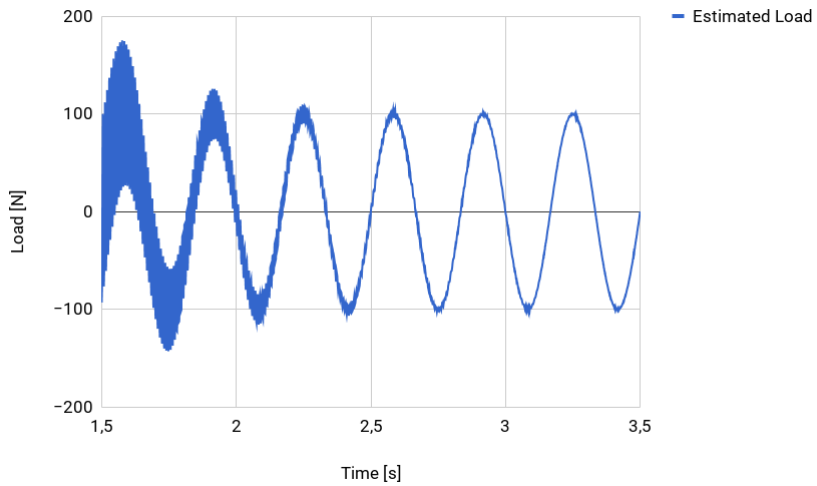


Figure 4.10: Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0003.

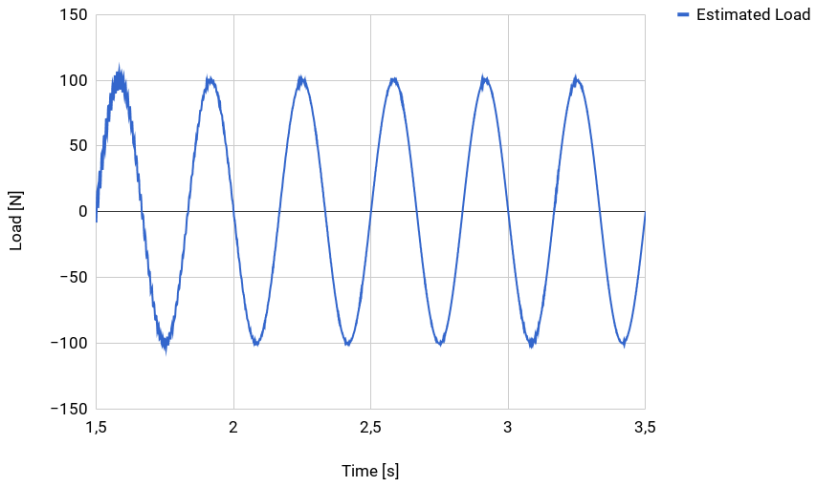


Figure 4.11: Load reconstruction for an applied load frequency 3.0 HZ and a stiffness proportional equal to 0.0009.

The same simulations were performed for an applied load of frequency 13.0Hz in order to see if the structural damping had a different effect for a higher frequency. The point of convergence for the estimated loads of same time interval are shown in Figure 4.12, 4.13 and 4.14.

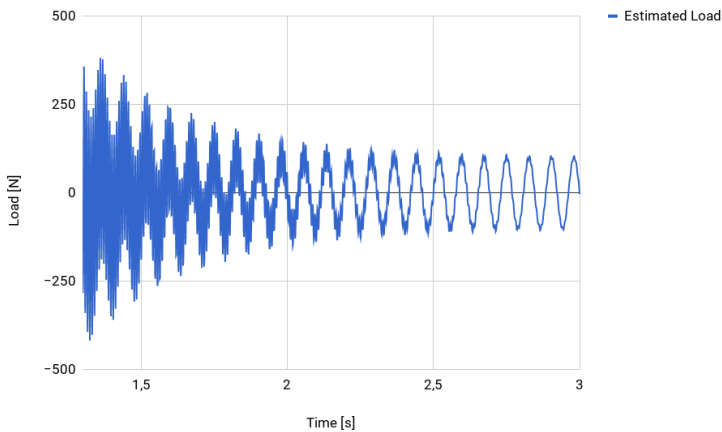


Figure 4.12: Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0001.

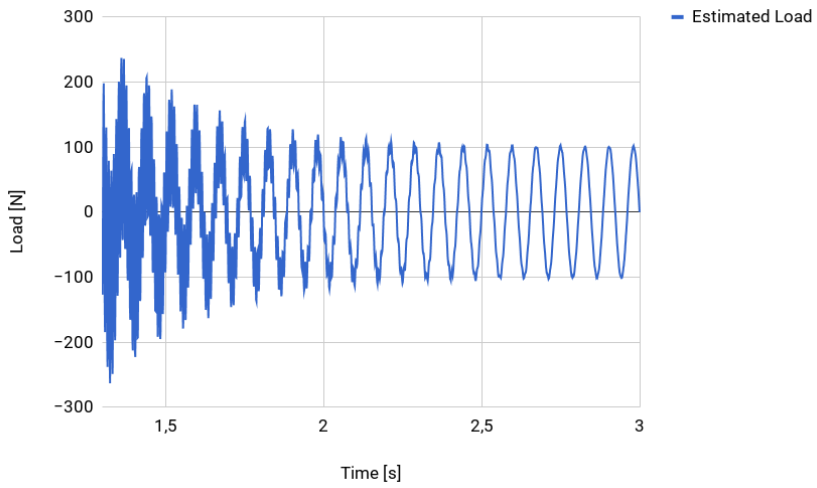


Figure 4.13: Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0003.

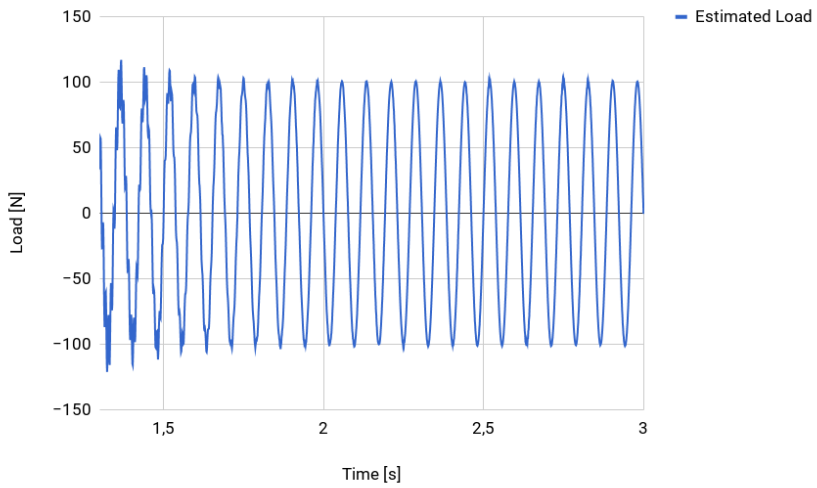


Figure 4.14: Load reconstruction for an applied load frequency 13.0 HZ and a stiffness proportional equal to 0.0009.

4.1.5 Load Reconstruction And Noise Reduction

Load reconstruction was performed using the following applied load frequencies; below eigenfrequency, at resonance and above eigenfrequency. The reconstructed load versus the applied load for each simulation is shown in the next chapter.

A large amount of noise was observed in the estimated load curves. This is illustrated in Figure 4.15, 4.16 and 4.17 for three iterations of an applied load of frequency 3.0 Hz. The blue area in the graphs were due to areas of high frequency. As shown, there is a tendency of an increasing noise at the beginning of each curve. Due to the large amount of noise, further iterations failed to solve.

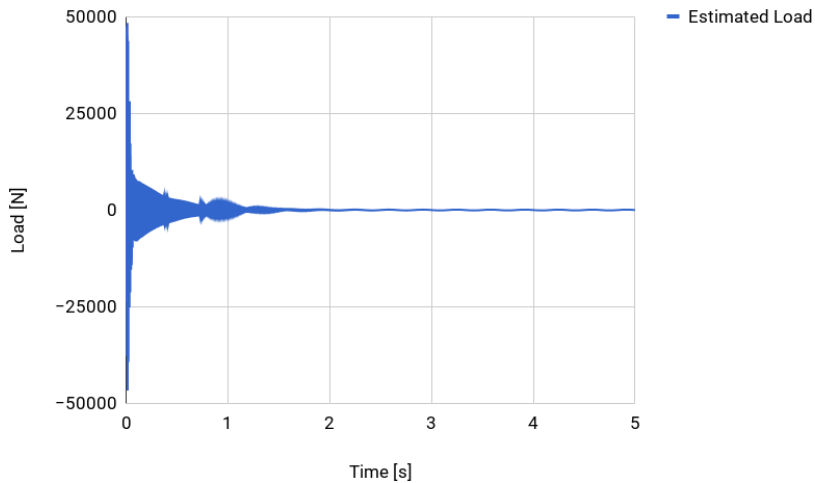


Figure 4.15: First iteration.

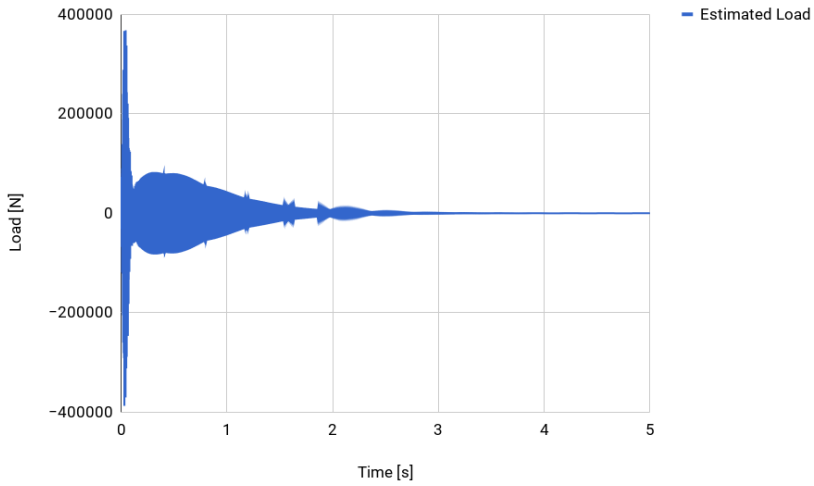


Figure 4.16: Second iteration.

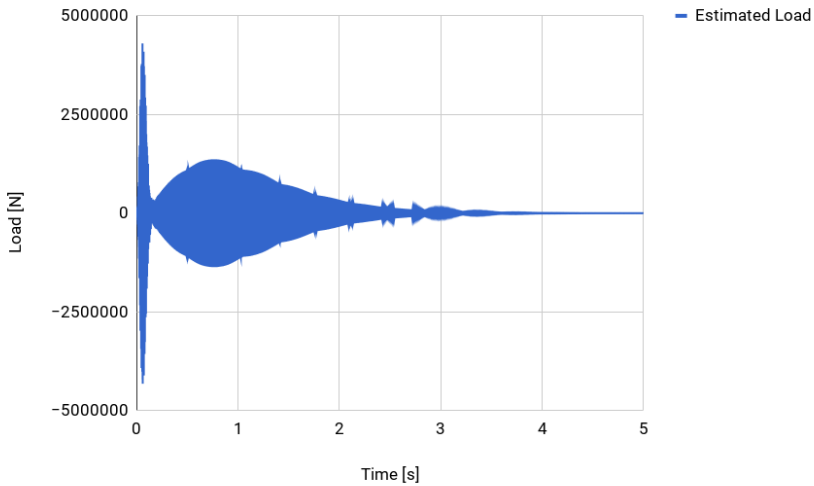


Figure 4.17: Third iteration.

A low-pass filter was applied to the estimated load in an attempt to reduce the large amount of noise in the system. The low-pass filter equal to a first order Butterworth filter was made in the control editor and is schematically presented in Figure 4.18. The amplification factor,

K , was set to 1 in all iterations. The time constant, T , was decided based on the bandwidth in the high frequency areas.

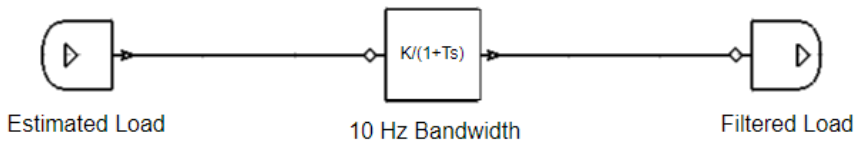


Figure 4.18: Low-pass filter applied to the estimated load.

Three different values of T were used for the filter in order to emphasize the difference of a larger value as compared to a smaller value. The chosen values were $T = 0.007$, $T = 0.02$ and $T = 0.06$ resulting in the estimated loads shown in Figure 4.19, 4.20 and 4.21 respectively.

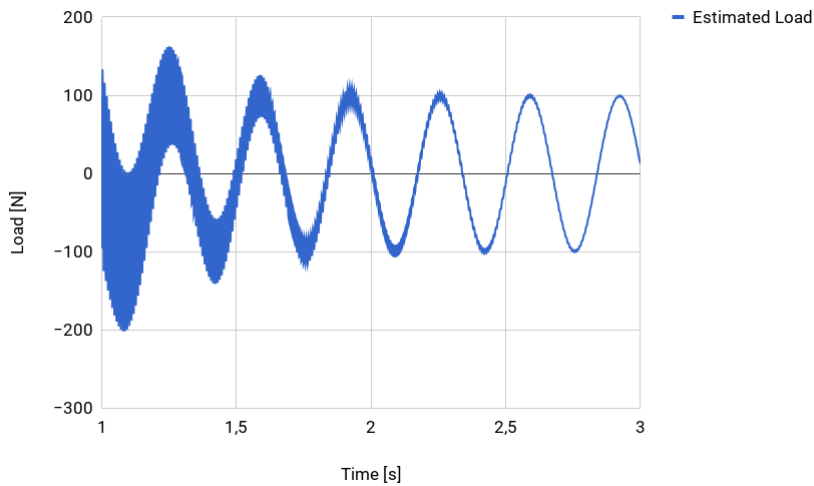


Figure 4.19: A filter with a time constant of 0.007.

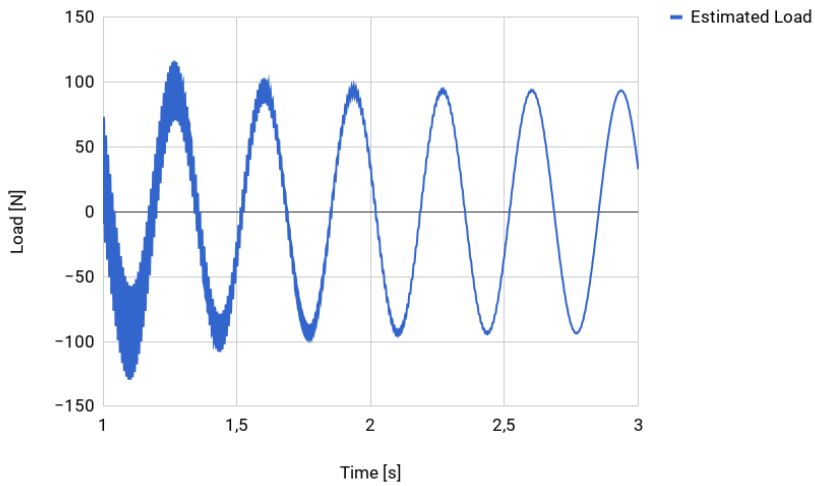


Figure 4.20: A filter with a time constant of 0.02.

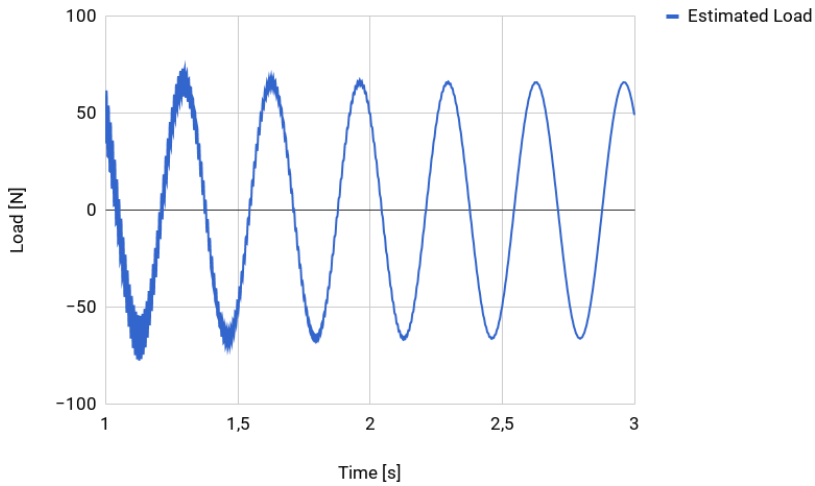


Figure 4.21: A filter with a time constant of 0.06.

The larger the value of T , the faster convergence was achieved. However, a large value of T also seemed to lower the low frequency area. Based on the tested filters, the one with T equal to 0.007 was used further. It did not filtrate all high frequencies but an exact force

reconstruction was achieved. The use of filter made it possible to perform several iterations than before. This was done in order to see how the noise evolved in each iteration. Seven iterations using filtrated estimated loads were performed as illustrated in Figure 4.22.

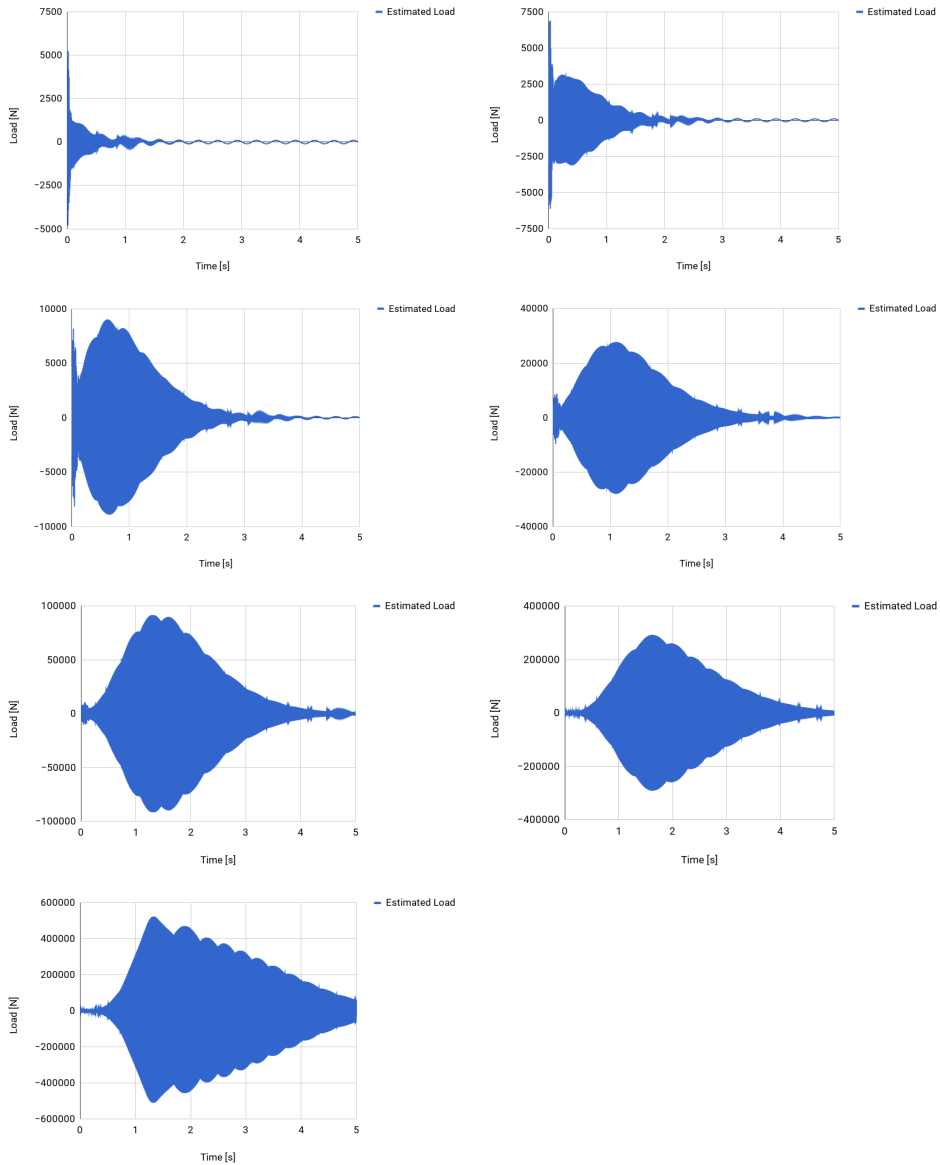


Figure 4.22: Seven iterations using filter for load estimation of an applied load of frequency 3.

Using a time constant of 0.007 was also tested on an applied load of frequency 13.0 Hz. The estimated filtrated load is shown in Figure 4.23. As shown, the time constant that proved successful for a lower frequency did not perform well for a higher frequency. The filter affected the low frequency area, and an amplitude of 100 N was not reached.

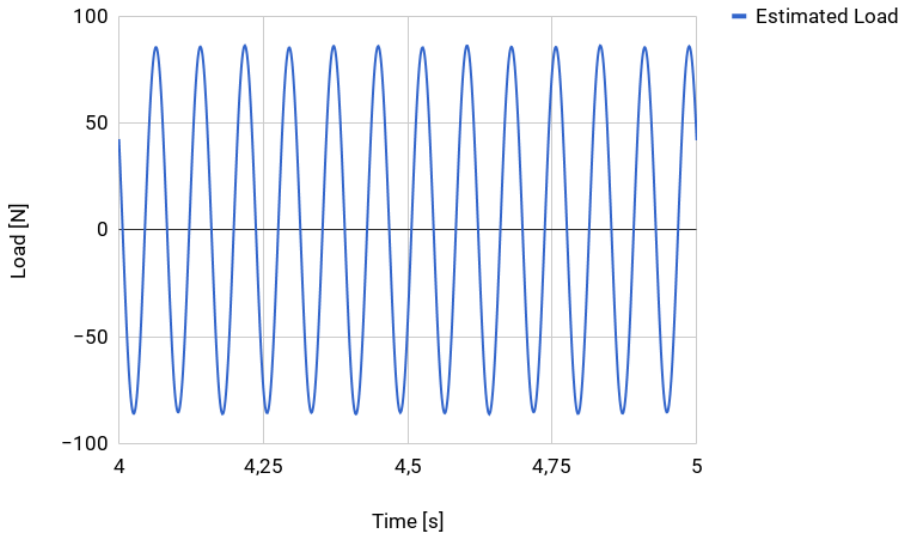


Figure 4.23: A filter with a time constant of 0.007 and an applied load frequency 13.0 Hz.

Choosing an optimal time constant was challenging as it also seemed to affect the low frequency area. It was therefore hard to tell whether or not the estimated load at convergence was affected by the filter. A time constant of 0.007 proved successful for a low frequency and was therefore used in the results presented in the next chapter. All other frequencies were not filtered as finding an optimal time constant was challenging.

4.2 Impulse Load Reconstruction

The deconvolution method based on superposition, proposed by T. Uhl in section 2.4.2 was used for an impulse load reconstruction. The method is mainly applicable to linear systems. It was tested on a SDOF mass-spring system and on the excavator presented in the previous section.

The method was based on obtaining the Markov parameters, \mathbf{H} , and the displacements, \mathbf{x} , in order to solve Equation 4.1 for an applied load, \mathbf{F} .

$$[\mathbf{H}][\mathbf{F}] = [\mathbf{x}] \quad (4.1)$$

4.2.1 Example

In the proposed study, the applied load was assumed to be one impulse load and the Markov parameters were obtained from the impulse response. The impulse load assumption was supported by the concept of normal linear interpolation. Say the system was subjected to an applied load of maximum value 1 N between 0.1 and 0.3 seconds. By using normal linear interpolation, the three impulse loads shown in Figure 4.24 may be used to reconstruct the applied load. If the impulse loads were numbered from 1 to 3, starting with the upper impulse, each impulse may be defined as $N_1 f_1$, $N_2 f_2$ and $N_3 f_3$ respectively. The total load would then be given by $\sum f_i N_i$ as shown in Figure 4.25.

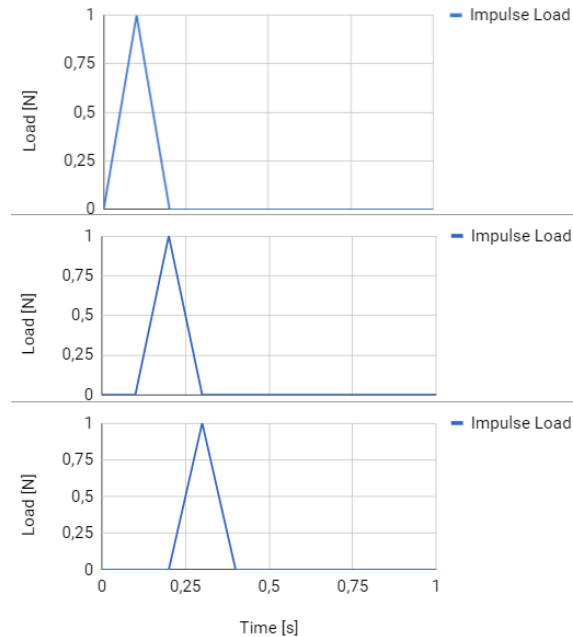


Figure 4.24: Assumed impulse load example.

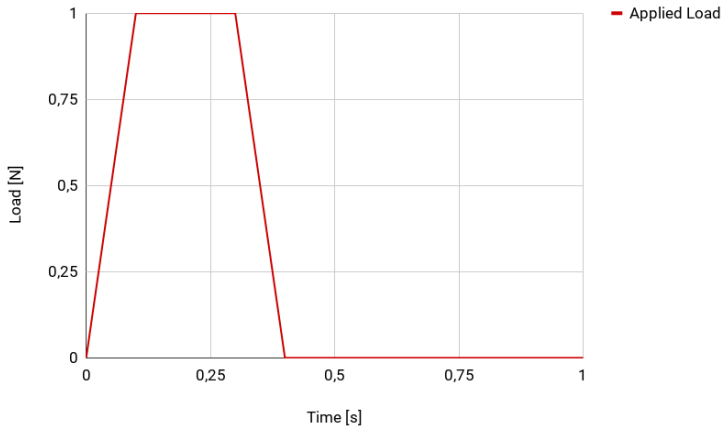


Figure 4.25: Applied impulse load example.

4.2.2 SDOF System

The first impulse load reconstruction experiment was performed on a simple SDOF mass-spring system. The idealized system is shown in Figure ?? and the corresponding model created in Fedem is shown in Figure 4.27. The system model consisted of two nodes; one fixed node and one node with free movement in x-direction. The latter node was subjected to a mass of 10kg and an external impulse load acting in x-direction. The two nodes were connected by a spring and a damper, both of linear behavior. The applied impulse load was considered unknown during the experiment, and was only used to evaluate consistency with the final reconstructed load in the results presented in the following chapter. All simulations were run with a time increment of 0.005 seconds and a total time period of 20 seconds. The system was given the properties listed in Table 4.6 and had an eigenfrequency of 0.159155 Hz.

Table 4.6: Dynamic properties assigned to SDOF mass-spring system.

Property	Magnitude
Spring stiffness [N/m]	10
Damping coefficient	5.0

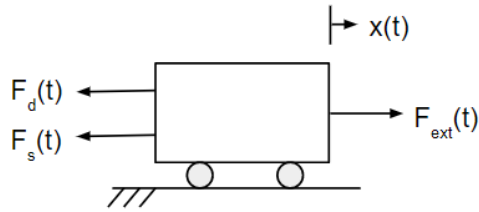


Figure 4.26: Idealized SDOF system.

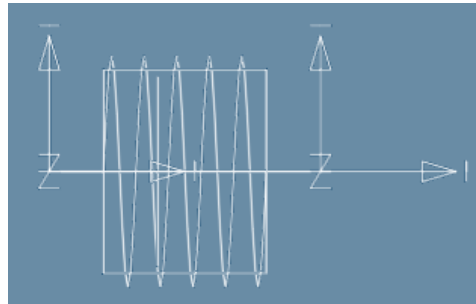


Figure 4.27: System model in Fedem.

The displacements in Equation 4.1 may be obtained from a strain gauge placed on a physical object, measuring the position in x -direction. The time variation in velocity and acceleration may also be generated from the strain gauge measurements. In this experiment, this was solved numerically. The responses used to reconstruct the applied load were obtained from Fedem. The response time histories resulting from the applied load will further be termed sensor response, and the response time histories resulting from the assumed impulse load will be termed impulse response.

During the experiment, different assumed impulse loads and number of readout points were tested. The assumed loads were made as a poly line force in Fedem, with a maximum load of 1 N at a time equal to the first readout point. Time steps of 0.01 seconds, 0.05 seconds and 0.1 seconds for a total time period of 1 second were tested, resulting in 100, 20 and 10 readout points respectively. The assumed impulse loads for each time step is shown in Figures 4.28, 4.29 and 4.30 respectively.

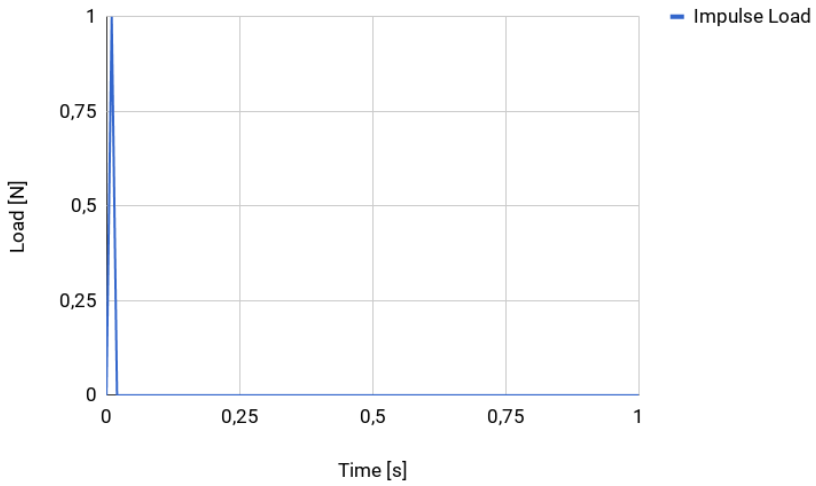


Figure 4.28: Assumed impulse load using a time step of 0.01 seconds.

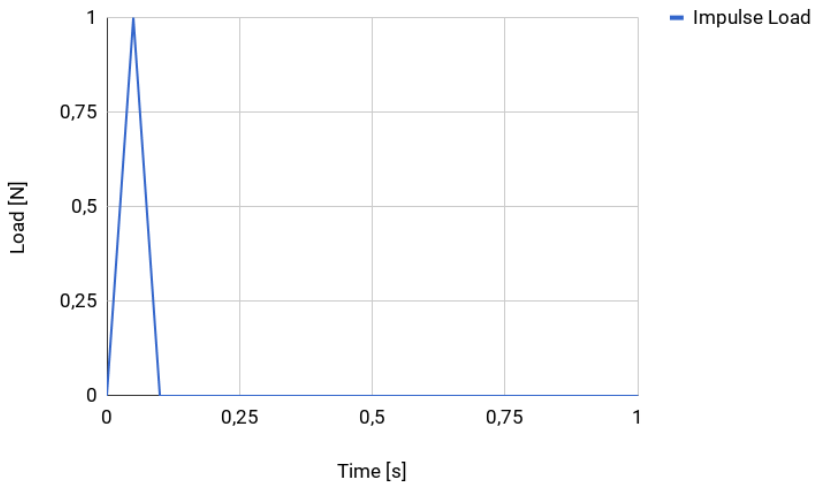


Figure 4.29: Assumed impulse load using a time step of 0.05 seconds.

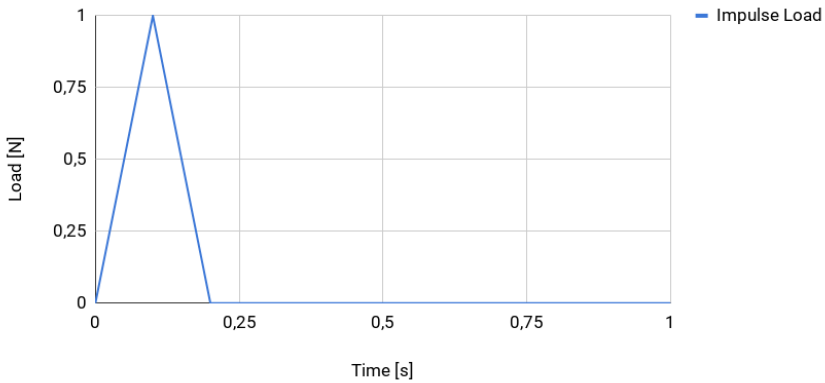


Figure 4.30: Assumed impulse load using a time step of 0.1 seconds.

Firstly, the estimation using 100 readout points and 20 readout points will be presented. The displacement measurements using an assumed impulse load with maximum value at 0.01 seconds and 0.05 seconds are shown in Figures 4.31 and 4.32 respectively. The corresponding readout points in a time interval of 1 second is shown in Appendix A.5.

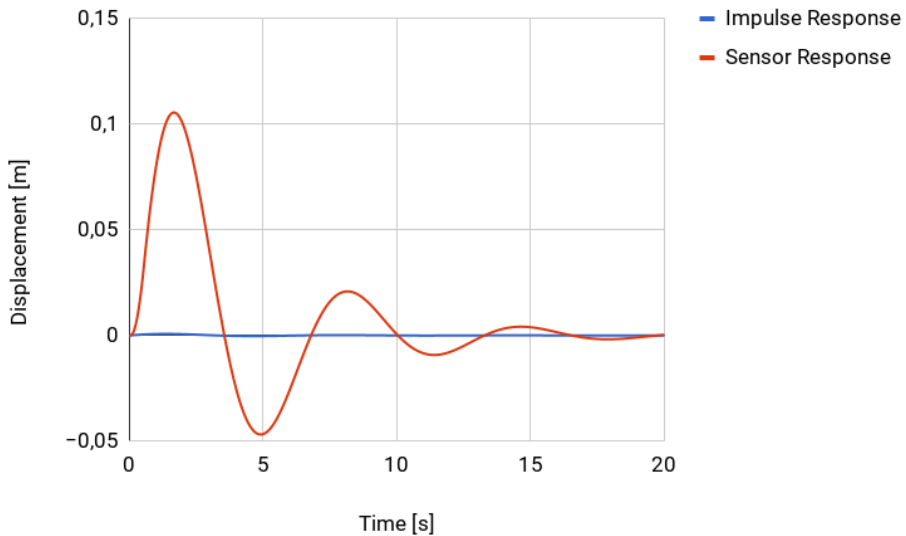


Figure 4.31: Displacement measurements assuming maximum impulse load at 0.01 seconds.

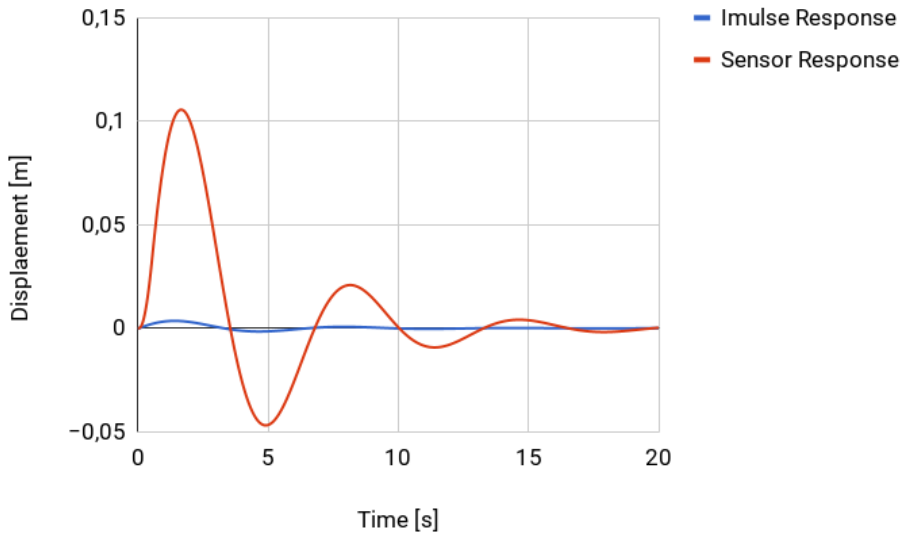


Figure 4.32: Displacement measurements assuming maximum impulse load at 0.05 seconds.

The estimated load using 100 readout points is shown in Figure 4.33. As shown, a large amount of noise arises at the end of the curve. An upper and lower bound of 5 N was therefore introduced in order to investigate the area with less noise interference, as shown in Figure 4.34. The figure shows a reconstructed load up to approximately 0.2 seconds, before the reconstruction is interfered with random noise. Instead of using regularization methods or filtering techniques, a smaller amount of readout points was tested. Thus, the number of readout points was reduced to 20 and the assumed impulse load was changed correspondingly.

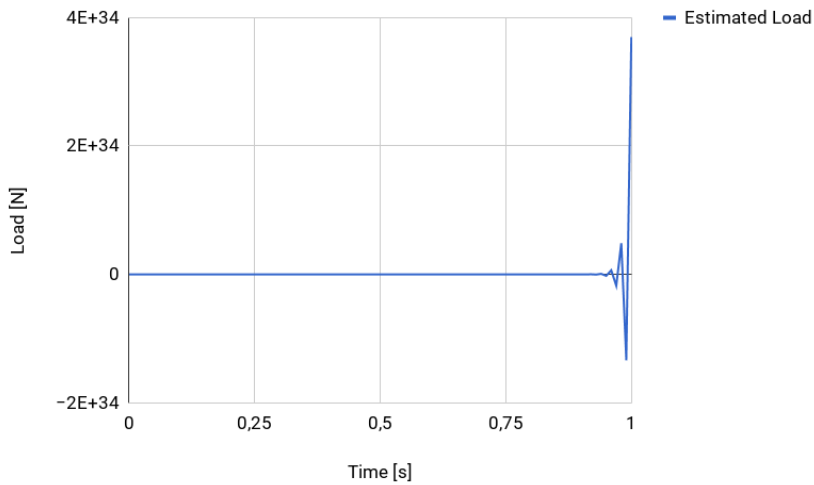


Figure 4.33: Estimated load using 100 readout points.

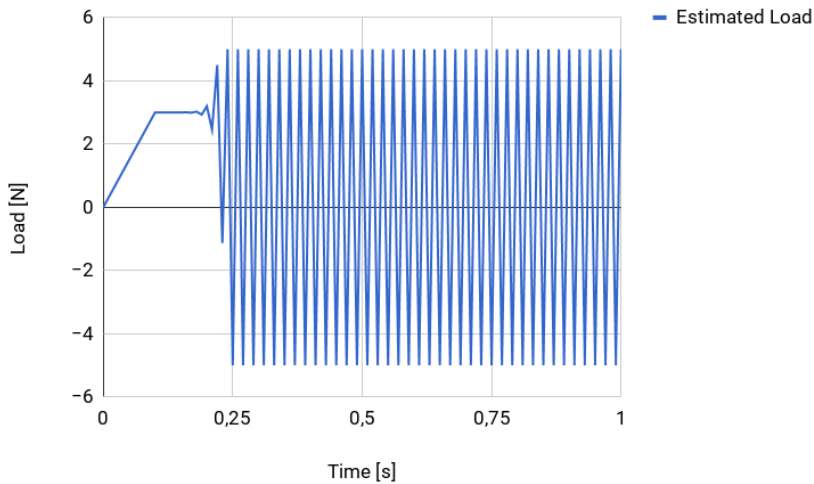


Figure 4.34: Estimated load with an upper and lower limit, using 100 readout points.

The estimated load using 20 readout points is shown in Figure 4.35. Compared to the previous load reconstruction, less noise interference was observed at the end of the curve. However, an upper and lower bound was still needed, resulting in the estimated load shown

in Figure 4.36. As shown, a larger part of the load was reconstructed, but the curve was still affected by noise.

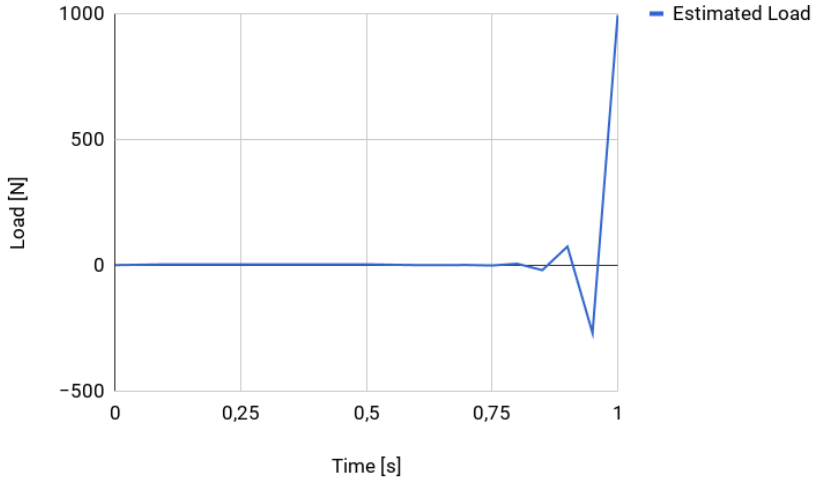


Figure 4.35: Estimated load using 20 readout points.

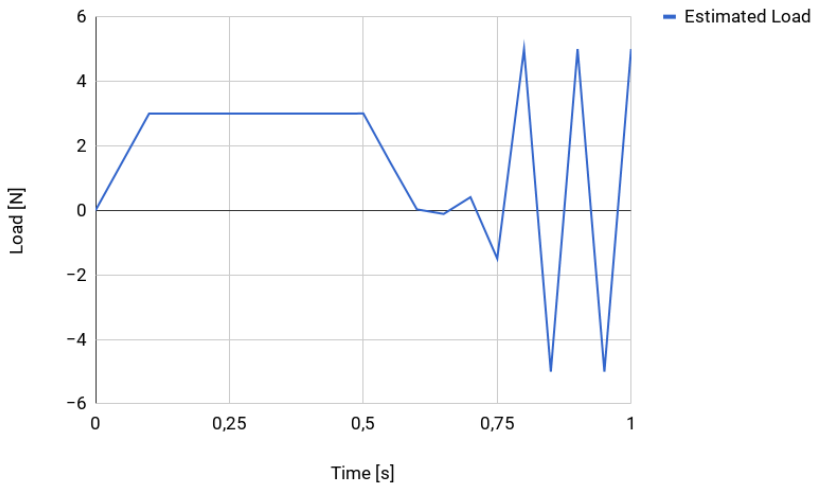


Figure 4.36: Estimated load with an upper and lower limit, using 20 readout points.

In an attempt to get a perfectly reconstructed load without noise interference, the number of readout points was reduced to 10. Correspondingly, the assumed impulse load had a maximum value at 0.1 seconds. The measured displacement responses are shown in Figure 4.37 and the readout points are shown in Figure 4.38.

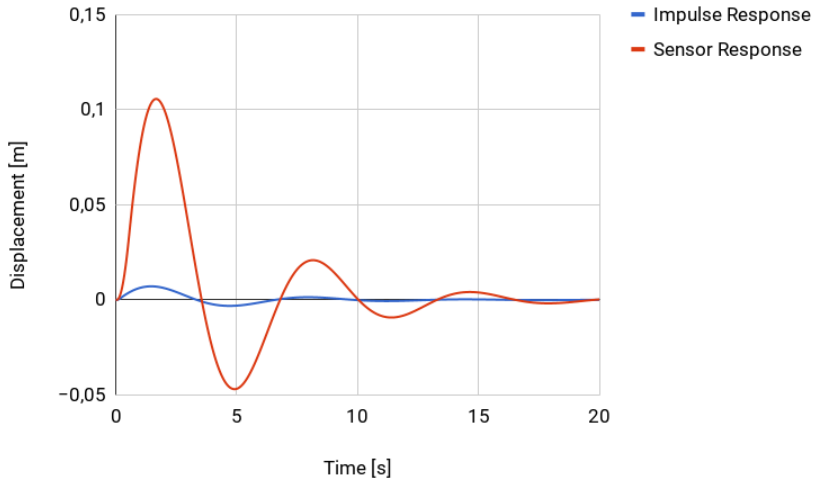


Figure 4.37: Displacement measurements assuming maximum impulse load at 0.1 seconds.

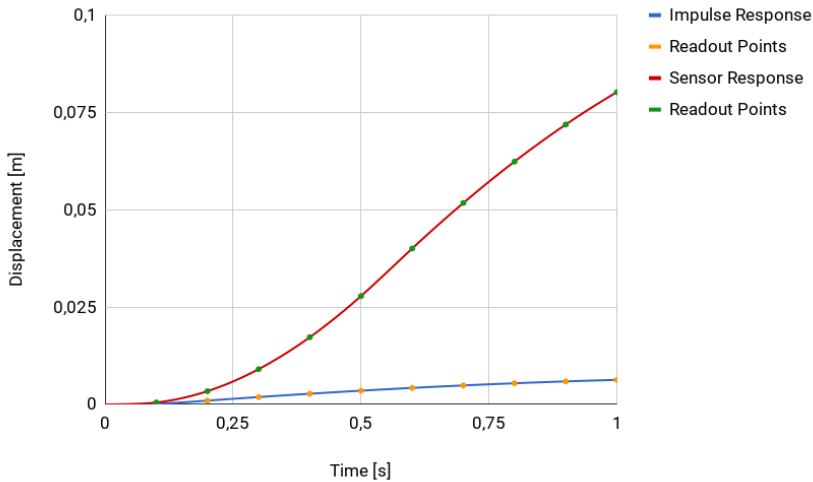


Figure 4.38: Displacement readout points.

Figure 4.39 shows the reconstructed load based on 10 readout points. As shown, no noise interference was observed. The estimated load has been compared to the applied load in the following chapter.

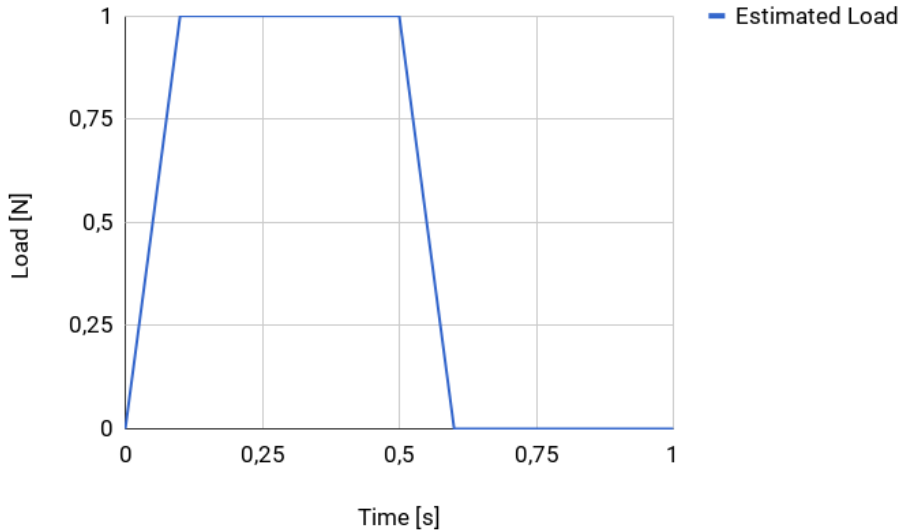


Figure 4.39: Estimated load using 10 readout points.

Load reconstruction was also performed based on velocity and acceleration responses to investigate whether or not there were any deviations when using different measurement parameters. The obtained velocity responses are shown in Figure 4.40, and the readout points for a time interval of 1 seconds are shown in Figure 4.41.

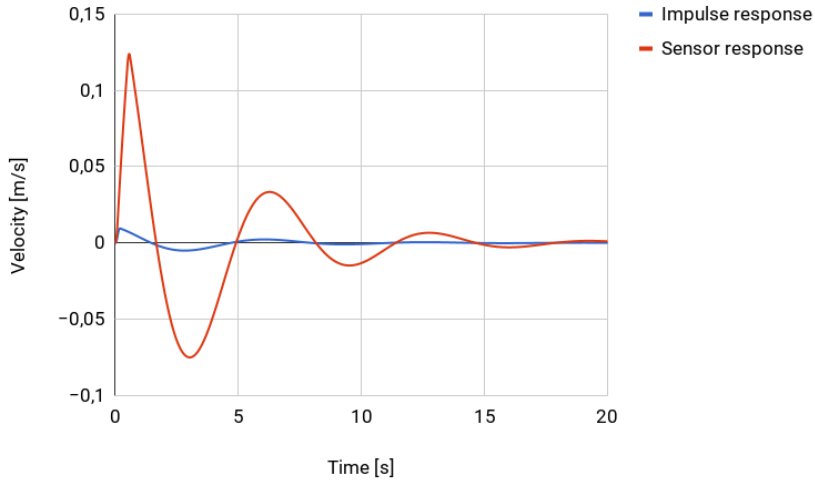


Figure 4.40: Velocity measurements assuming maximum impulse load at 0.1 seconds.

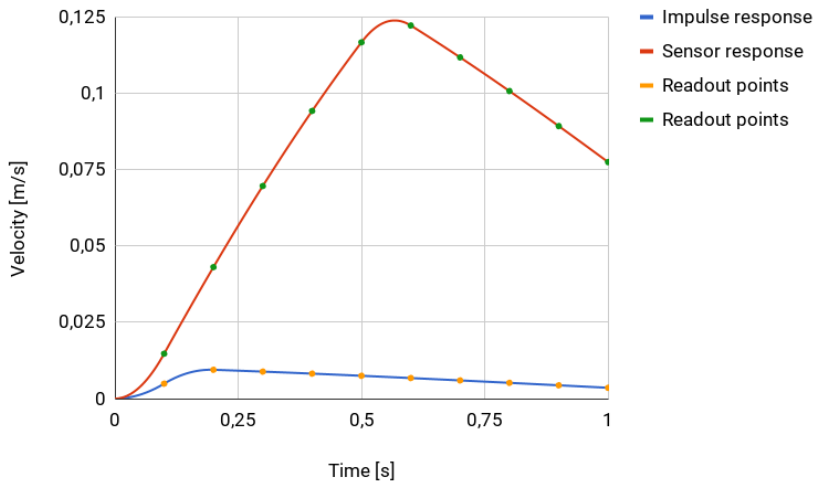


Figure 4.41: Velocity readout points.

The measured acceleration responses are shown in Figure 4.42, and the readout points for a 1 second time period is shown in Figure 4.43.

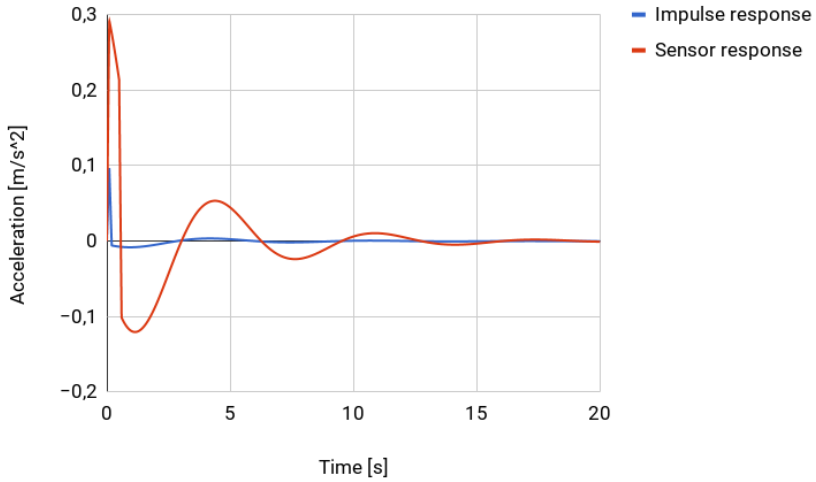


Figure 4.42: Acceleration measurements assuming maximum impulse load at 0.1 seconds.

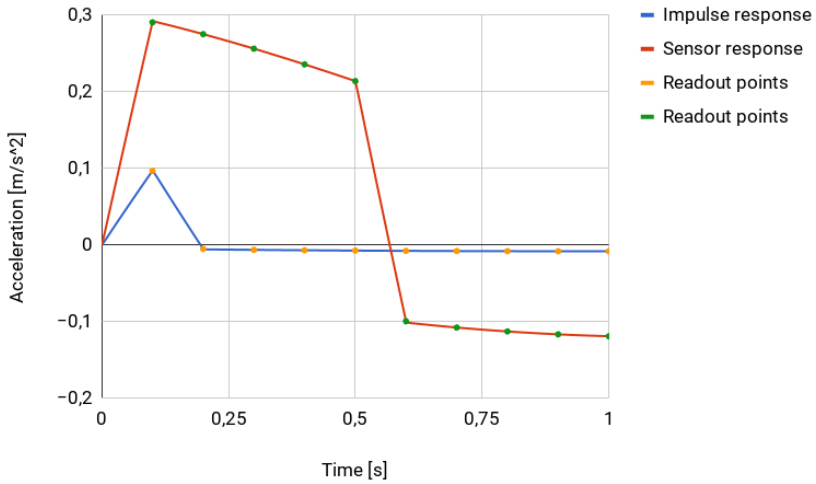


Figure 4.43: Acceleration readout points.

Load reconstruction based on displacement, velocity or acceleration using 10 readout points was identical. Thus, the estimated load in Figure 4.39 was representative for all.

4.2.3 Simplified Excavator

The use of Markov parameters was also tested on the 2-dimensional excavator used in the periodic load reconstruction experiment. The excavator was given the same properties as in the periodic load reconstruction experiment, and are listed in Table 4.1. The spring forces resulting from the applied load and the impulse load were used as dynamic responses to solve Equation 4.1. The impulse load was assumed to have a maximum value of 1 N at 0.1 seconds. The Markov parameters and sensor responses were chosen for a total time period of 1 second, resulting in 10 readout points. The spring responses are illustrated in Figure 4.44, and the readout points are illustrated in Figure 4.45.

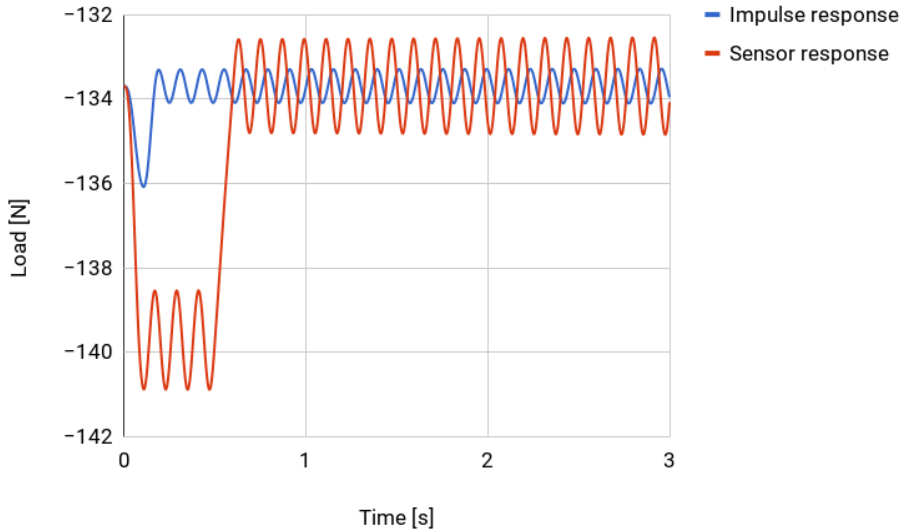


Figure 4.44: Excavator spring force response from impulse load and applied load.

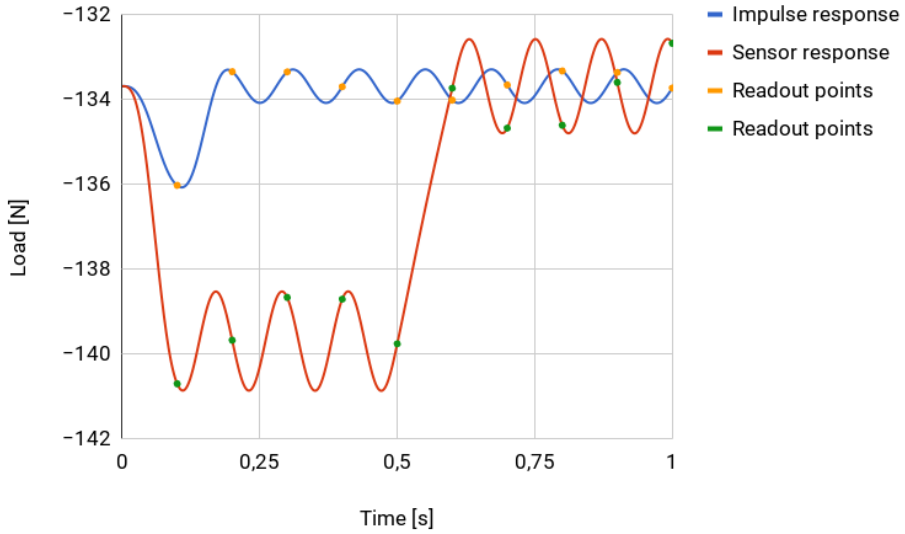


Figure 4.45: Spring force readout points.

The estimated load has been compared with the applied load in the following chapter.

4.2.4 Matrix Solution

A Python script was used to solve Equation 4.1 for the unknown forces based on the response measurements from Fedem for the SDOF system and the simplified excavator. The script is shown in Appendix B.

5. Results

In this chapter, the estimated periodic load and the estimated impulse load are compared with the applied load in each case.

5.1 Periodic Load Reconstruction

The results from the periodic load reconstruction were based on a chosen spring stiffness of 10^5 N/m in the physical excavator and a stiffness proportional of 0.0001 for the physical excavator and the digital twins. The actual estimated loads had an opposite direction than the applied load. The estimated loads were therefore multiplied by -1 in order to make them consistent with the applied loads.

5.1.1 Frequency Below Eigenfrequency

The estimated load versus the applied periodic load after one iteration is shown in Figure 5.1. As shown, the load seemed to reach convergence after about 3 seconds, however, at maximum and minimum it was still somewhat affected by noise. Even though the low-pass filter applied in the previous chapter did not prove sufficient for all frequencies, it gave accurate results for the frequency below eigenfrequency given an appropriate choice of time constant. When applying the low-pass filter with a time constant of 0.007, the noise at convergence was eliminated as shown in Figure 5.2. An earlier convergence was also observed. The filtrated load converged at about 2.75 seconds. The reconstructed load at convergence perfectly corresponded to the applied load with an amplitude of 100 N and a frequency of 3.0 Hz. A correct load was achieved after one iteration. Therefore, no further iterations were necessary. In fact, further iterations gave an increasing degree of inaccurate results as can be seen in Appendix A.4.

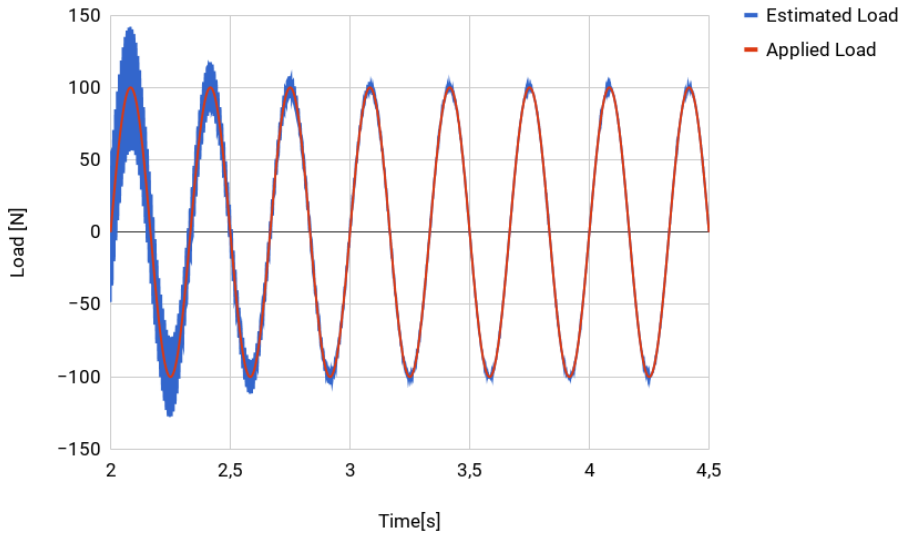


Figure 5.1: Load comparison without filter for an applied load frequency below eigenfrequency.

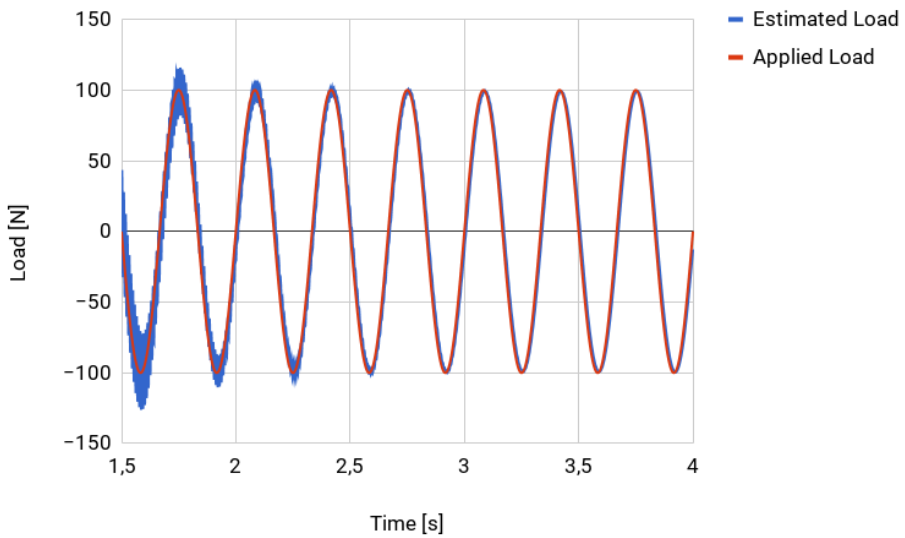


Figure 5.2: Load comparison with filter for an applied load frequency below eigenfrequency.

5.1.2 Frequency at Resonance

The results from the load reconstruction at a 5 seconds time interval for an applied load of frequency equal to the first eigenfrequency are shown in Figure 5.3. There was a large amount of noise in the curve during the first 1.5 seconds. When looking at the low frequency area in Figure 5.4 one can see that the frequency of 8.21767 Hz of the applied load was reached, but with a small shift to the right. The amplitude of the load did not converge, hence, the estimated load did not correspond to the applied load.

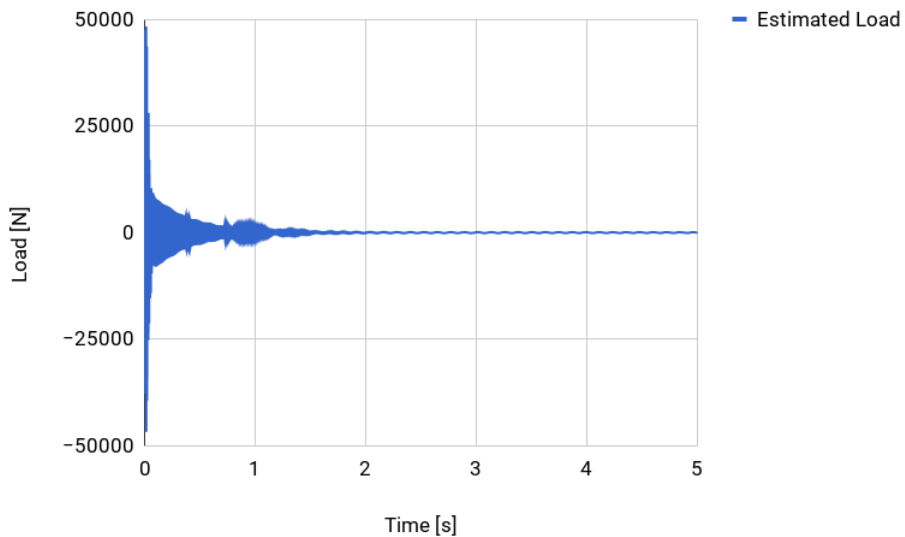


Figure 5.3: Estimated load at resonance for a 5 seconds time interval.

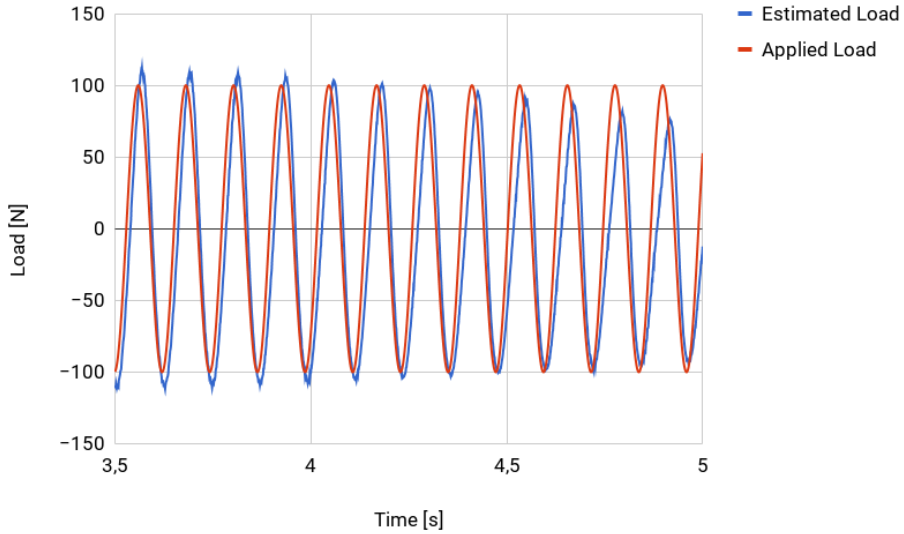


Figure 5.4: Load comparison at resonance showing the low-frequency area.

5.1.3 Frequency Above Eigenfrequency

The estimated load for an applied load frequency above eigenfrequency is shown in Figure 5.5. As shown, there was a high frequency area due to noise in the curve during the first 1.5 seconds. The high frequencies gradually decreased and the load almost reached convergence at about 3.0 seconds as illustrated in Figure 5.6. When comparing the estimated load to the applied load in Figure 5.7, an almost perfect correlation was observed. However, there were some high frequency peaks at the maximum and minimum values of the estimated load due to noise. The estimated load frequency seemed to perfectly correspond.

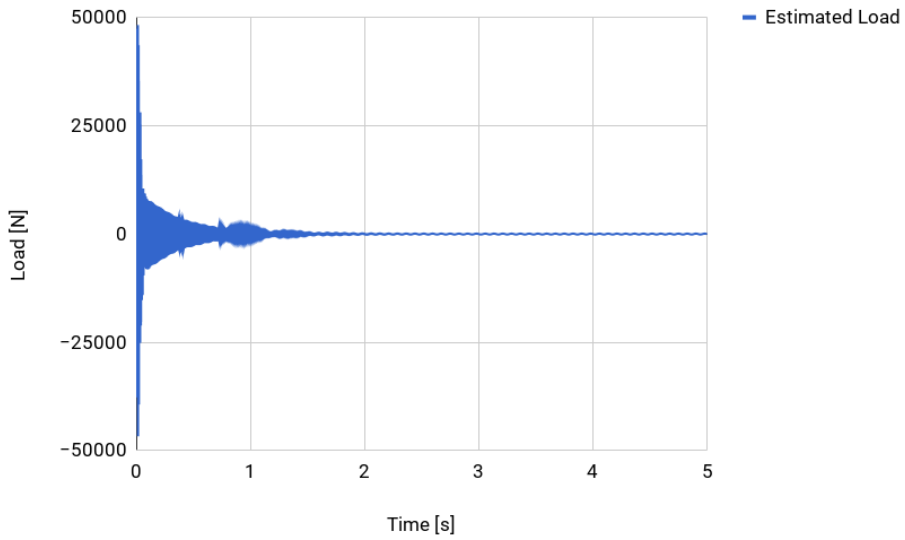


Figure 5.5: Estimated load for an applied load frequency above eigenfrequency in a 5 seconds time interval

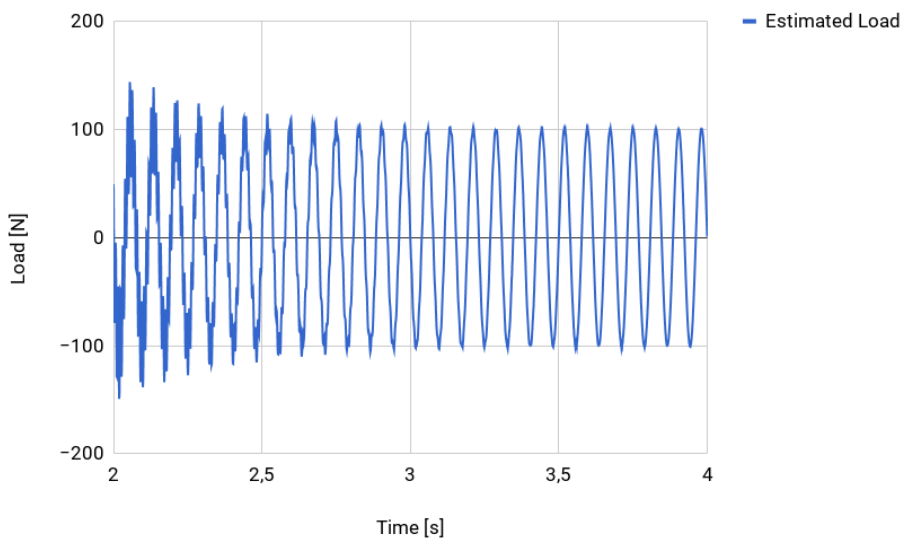


Figure 5.6: Estimated load at area of convergence for an applied load frequency above eigenfrequency.

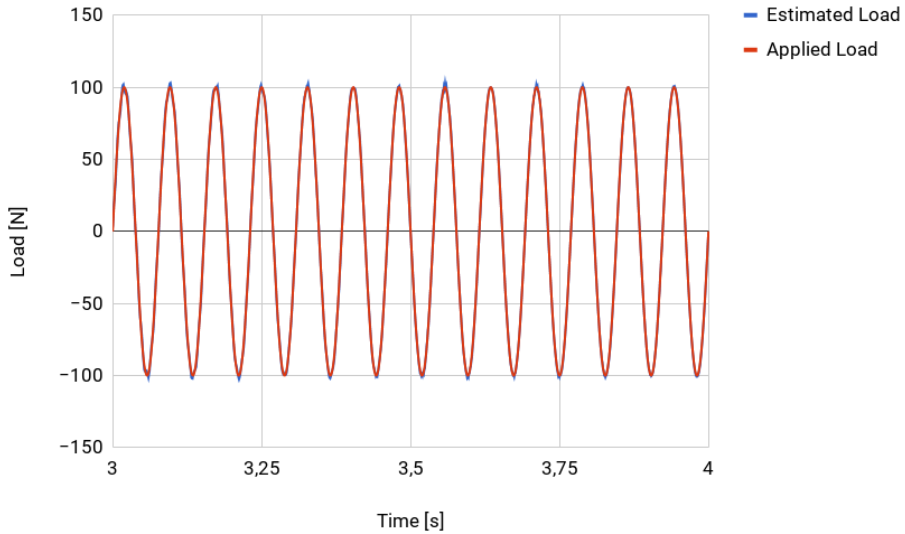


Figure 5.7: Load comparison at area of convergence for an applied load frequency above eigenfrequency.

5.2 Impulse Load Reconstruction

The estimated load for the SDOF system was equal for the displacement, velocity and acceleration measurements. The applied load versus the estimated load in all cases when using 10 readout points at a 1 second time interval is shown in Figure 5.8. The resulting estimated loads versus the applied loads when using 20 and 100 readout points are shown in Appendix A.6.

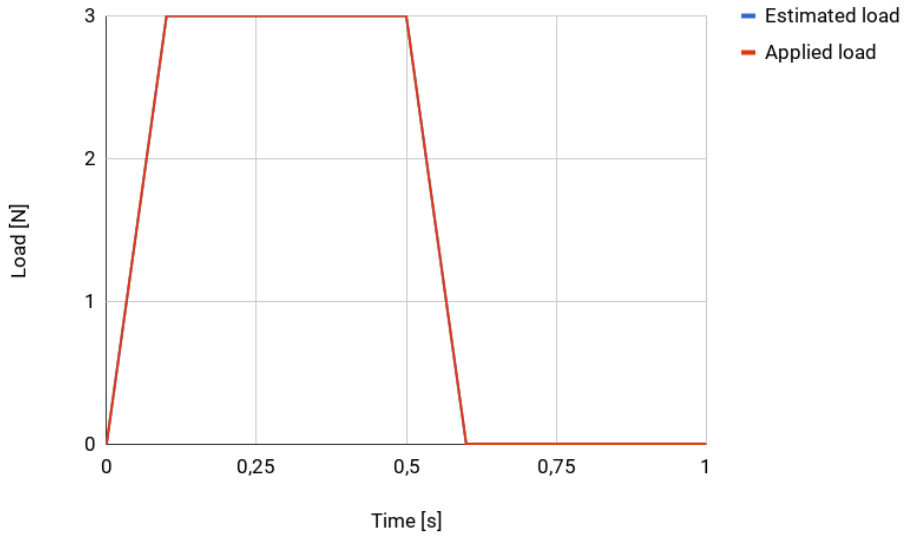


Figure 5.8: Applied impulse load versus estimated impulse load for the SDOF system.

The applied load versus the estimated load for the MDOF system is shown in Figure 5.9.

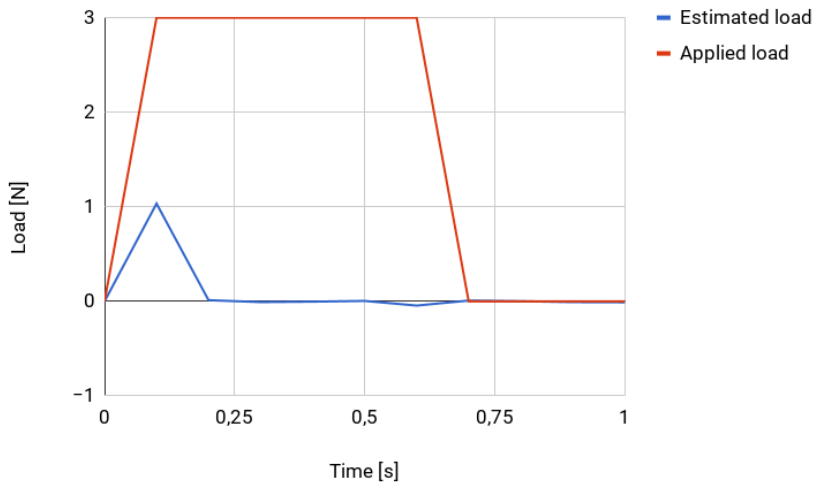


Figure 5.9: Applied impulse load versus estimated impulse load for the excavator.

6. Discussion

In this chapter, the results presented in the previous chapter will be discussed in addition to the methodology used to obtain the results. The two experiments will be discussed separately, followed by a brief evaluation of the software performance.

6.1 Periodic Load Reconstruction

The results from the periodic load reconstruction showed a considerably good match in regards to the applied load after only one iteration. This was expected as all properties were of linear behavior. The opposite direction of the estimated load can be explained by Newton's third law. Multiplying the obtained results by -1 was therefore reasonable as the applied load was to be reconstructed.

For the applied load frequency below eigenfrequency, an exact load was achieved using a low-pass filter in the area of convergence. Achieving an exact load was expected, as an applied load of a relatively low frequency does not bring much energy into the system. The low-pass filter seemed to provide sufficient filtration in the low frequency area of the curve. As for the high frequency area, it was not capable of filter out all the high frequencies.

At the point of resonance, the system experienced the most possible energy as, at this frequency, the dynamic amplification factor goes to infinity. The estimated load did not converge, still, the attained amplitudes did not deviate entirely from the expected amplitude of 100 N. The frequencies of the estimated load and the applied load seemed to correspond except from a small shift in the data.

The results from an applied load of frequency above the first eigenfrequency were slightly influenced by noise at the maximum and minimum values of the amplitude. This resulted in a slightly larger amplitude than expected. The estimated load frequency and the applied load frequency were identical. Hence, one can say that the load was successfully reconstructed.

Sundermeyers method proved successful for reconstruction of a frequency below and above eigenfrequency. Even though the load estimation at resonance did not entirely match the

applied load, the deviation was not too large. This experiment was performed numerically, which may have led to “perfect” data measurement responses. If a real physical excavator had been used instead, the measurements may have been more influenced by noise or other parameters, and thereby made the reconstruction more challenging.

6.1.1 Noise Interference

An increase in energy, resulting in noise interference, was observed during the first 1.75 seconds in all curves before the estimated load stabilized. The applied low-pass filter proved weak for the iteration method. Despite the choice in time constant, the filter did not filter out all the high frequencies in the beginning of the graph. When running a simulation with a relatively large time constant, the filter seemed to also be dampening the low frequency area to a certain degree. Thus, the estimated load at convergence became too low. The time constant of 0.007 seemed to work well for load estimation of an applied load frequency below eigenfrequency, but it damped the low frequency area for the frequency above eigenfrequency. Choosing an optimal time constant proved challenging with no prior knowledge of the magnitude of the applied load.

When performing several iterations to illustrate the changing pattern of the noise, there was a trend of an increase in noise during each iteration process. The noise seemed to expand like a “wave” through the simulation time period after each iteration before it seemed to stabilize and converge again. However, the amplitude of the new convergence area did not equal the applied load. Hence, one should avoid the “noise wave” when performing several iterations. The presence of noise should therefore be dealt with. The existence of noise is likely due to the ill-conditioning of the inverse problem. Other filtering techniques or regularization methods should be tested before moving forward with the study.

6.1.2 Varying Properties

As opposed to Sundermeyers method, this experiment also included the damping term, and the effect of variation in structural damping was looked into. Generally, the structural damping used for the digital twins should represent the structural damping in the physical system as good as possible. Considering damping is essential in order to stabilize a system for an applied dynamic load, as it may be both physically and numerically challenging to have a system of not sufficient damping. For a system with no damping at all, all energy due to an applied load remains in the system and the responses may continue in eternity. It is therefore important to consider damping. A large amount of damping may in many cases be

beneficial for the system in order to avoid harming components. In regards to the numerical problem, the numerical noise in the system increases when there is not sufficient damping present. This was illustrated in the experiment when changing the stiffness proportional. Hence, the numerical noise may to a certain degree be handled by choosing an appropriate amount of damping. A large amount of damping may thereby lead to an earlier convergence if noise is not reduced by other methods. It was proved that the damping was mainly defined by the stiffness proportional area as a change in stiffness proportional had an influence in the results.

For steel constructions the relative damping should generally be 5-10% of the systems critical damping. The different relative damping used in this experiment was only 0.3%, 0.7% and 2.3% of critical damping. The former was used as basis through the experiment. These damping values were too low and did not provide sufficient damping to the system. As this experiment was performed numerically it did not seem to have any effect on the results. However, the stiffness proportional could definitely be increased in this experiment which again would result in less numerical noise. Even though an increased damping may provide an earlier convergence, filtering or regularization methods should also be used in future experiments in order to attain stable results.

In addition to the structural damping, the spring stiffness in the digital twins should also represent the one used in the physical excavator in order to get a situation as close to the real situation as possible. In the experiment, the effect of increasing the spring stiffness was looked into. It resulted in arising high frequencies in the low frequency area. Hence, it was an increase in energy in the system. A spring stiffness of 10^5 N/m and 10^7 N/m were both affected by noise during the first few seconds of the simulation. However, the results using the lowest stiffness got an estimated load which accurately corresponded to the original applied load. Using a larger stiffness produced generally more noise, also at the end of the simulation, and a smooth curve was not achieved.

6.1.3 Linearity

Sundermeyers method is not restricted to a certain linearity. In Sundermeyers study, the inverse method was successfully tested on non-linear systems. However, for a non-linear system, one iteration is most likely not sufficient to reconstruct the external load. Several iterations should then be performed in order to get stable results.

6.2 Impulse Load Reconstruction

The results showed that the estimated load corresponded perfectly with the applied load for the SDOF mass-spring system when 10 readout points were used. T. Uhl's method showed a potential drawback when it came to choose an optimal amount of readout points and an appropriate impulse load assumption. The method required a certain amount of readout points, but still not too many. 10 readout points turned out to be an optimal amount for the system in question. When performing load reconstruction with 20 and 100 readout points, the reconstruction was not complete due to noise interference. It should be mentioned that Tikhonov regularization was used in T. Uhl's study, but it was not considered in this experiment.

Whether the responses were obtained in terms of displacement, velocity or acceleration made no observable difference in regards to the load reconstruction when 10 readout points were used. Whether or not it would have made any difference in regards to noise using a larger amount of readout points was not looked into.

For the excavator experiment, the load reconstruction failed and only one impulse corresponding to the assumed impulse load for obtaining the Markov parameters was possible to obtain. The load reconstruction was based on 10 readout points. Whether or not it was possible to reconstruct the load using more or less readout point was not looked into.

Methods for choosing an optimal amount of measurement positions and an optimal amount of readout points for a given time interval was not considered in this thesis. However, it is a matter that should be investigated further. In order to optimize this choice, one should have prior information about the load and the eigenfrequencies in order to capture the time variations of the response.

6.3 Software Performance

The software, Fedem, proved sufficient for the load reconstruction experiments performed in this thesis. All simulations were solved within a short time period due to an efficient model reduction function used in the software. Only one load was applied to each system. Whether or not Fedem is equally capable of performing load reconstruction on systems subjected to several loads has not been investigated.

7. Concluding Remarks

In this thesis, the aim was to look into proposed methods for load reconstruction and find methods that could be used in the digital twin technology. Here, the study will be summarized and main findings will be presented, followed by suggestions for further work.

7.1 Summary and Conclusions

In this theses, several proposed load reconstruction methods have been summarized in a literature review. The methods were classified into solutions in the time domain and solutions in the frequency domain. The former methods were often based on the concept of deconvolution and were useful for all systems, despite system linearity. The latter methods were often based on the Fourier transform and thereby restricted to linear systems. Due to the ill-posed nature of the inverse problem, common regularization methods and filtering techniques were presented. The most common regularization technique was the Tikhonov regularization which was used in many of the presented studies. Basic structural dynamics for SDOF and MDOF systems was looked into, in addition to structural damping effects.

Based on two of the proposed methods, two load reconstruction experiments with solutions in the time domain were performed using a software called Fedem. The experiments were based on the work of Sundermeyer and T. Uhl respectively. The experiments and results from each experiment will be explained separately.

Based on the work of Sundermeyer, an extended iterative scheme for MDOF systems was used to reconstruct a periodic load acting on a simplified excavator. The load was obtained through measured responses in terms of cylinder force and cylinder displacement using two virtual representations of the excavator, thus, two digital twins. Applied loads with frequencies below, above and equal to the first eigenfrequency of the system were tested. From the results, the following trends were observed:

- For a frequency below eigenfrequency, the estimated load equals the external load after only one iteration.

- For a frequency equal to the first eigenfrequency, convergence was not achieved and an exact applied load was not established. However, the estimated load and the applied load did not deviate entirely.
- For a frequency above eigenfrequency, an exact reconstruction was achieved.
- During the iterative process, there was an increasing high-frequency area during the first few seconds of the curves. The applied low-pass filter did not prove successful.
- Increasing the structural damping led to a faster convergence and less noise interference.

In the second experiment, a method based on the use of the so-called Markov parameters presented by T. Uhl was used for an impulse load reconstruction. The load acting on a SDOF mass-spring system was reconstructed using 10 readout points from measured responses in terms of displacement, velocity or acceleration. The Markov parameters were obtained from the responses resulting from an assumption that the applied load was equal to one impulse with maximum value at the time equal to the first readout point. The applied impulse load was successfully reconstructed. No notifiable difference was noticed using the different responses. The method showed a drawback in terms of choosing an optimal amount of readout points. A complete load reconstruction failed when using more than 10 readout points due to noise interference. Tikhonov regularization was used in T. Uhl's study to eliminate noise. Regularization was not used in this experiment.

The software, Fedem, was successfully able to reconstruct the applied load in each experiment.

7.2 Suggestions for Further Work

The experiments performed in this thesis were based on 2-dimensional models, and were only performed numerically. Whether or not the same results are possible to achieve using 3-dimensional models and physical objects with digital twins should be looked into.

Both experiments highlighted the ill-posed nature of the inverse problem and the need for regularization methods or filtering techniques. Common methods used for inverse problems have been presented in theory and may be applicable to the experimental systems used in this thesis. As Tikhonov regularization proved successful in T. Uhl's study, it is likely that it is capable of eliminating the noise observed for force reconstruction of the SDOF

mass-spring system. In regards to T. Uhl's study, finding an optimization to the choice of Markov parameters is also suggested.

Bibliography

- [1] T. Uhl, “The inverse identification problem and its technical application,” *Archive of Applied Mechanics*, vol. 77, pp. 325–337, 2007.
- [2] S. Oller, *Nonlinear Dynamics of Structures*. Springer International Publishing, Cham, 2014, pp. 33.
- [3] E. Layer and K. Tomczyk, *Signal Transforms in Dynamic Measurements*. Springer International Publishing, Cham, 2015, pp. 169-188.
- [4] Y. Lage, M. Neves, N. Maia, and D. Tcherniak, “Force transmissibility versus displacement transmissibility,” *Journal of Sound and Vibration*, vol. 333(22), pp. 5708–5722, 2014.
- [5] J. Mueller and S. Siltanen, *Linear and Nonlinear Inverse Problems with Practical Applications*. Philadelphia, PA: SIAM, Society for Industrial and Applied Mathematics, 2012, pp. 63-82.
- [6] M. Wahab, *Dynamics and vibration : an introduction*. Chichester, England: John Wiley, 2008, pp. 326.
- [7] R. Craig and A. Kurdila, *Fundamentals of Structural Dynamics*, 2nd ed. John Wiley, 2006, pp.195-199, 211-213, 296, 171-177, 500.
- [8] R. Clough and J. Penzien, *Dynamics of Structures*, 2003.
- [9] S. Zheng, L. Zhou, X. Lian, and K. Li, “Technical note: Coherence analysis of the transfer function for dynamic force identification,” *Mechanical Systems and Signal Processing*, vol. 25(6), pp. 2229–2240, 2011.
- [10] J. Liu, X. Meng, C. Jiang, X. Han, and D. Zhang, “Time-domain galerkin method for dynamic load identification,” *International Journal for Numerical Methods in Engineering*, vol. 105(8), pp. 620–640, 2015.

-
- [11] W. Guan, C. Wang, D. Chen, and J. Gou, "Overview of the latest method and development direction for dynamic load identification," *Applied Mechanics and Materials*, vol. 556-562, pp. 2573–2576, 2014.
- [12] X. Sun, J. Liu, X. Han, C. Jiang, and R. Chen, "A new improved regularization method for dynamic load identification," *Inverse Problems in Science and Engineering*, vol. 22(7), pp. 1062–1076, 2013.
- [13] Y. Lage, N. Maia, and M. Neves, "Force magnitude reconstruction using the force transmissibility concept," *Shock and Vibration*, pp. 1–9, 2014.
- [14] E. Strømmen, *Structural Dynamics*, 2nd ed. Springer International Publishing, 2014, pp. 229-247, 355.
- [15] M. Grieves. (2014) Digital twin: Manufacturing excellence through virtual factory replication. [Online]. Available: http://innovate.fit.edu/plm/documents/doc_mgr/912/1411.0.Digital.Twin.White.Paper.Dr.Grieves.pdf
- [16] S. Boschert and R. Roland, *Industry 4.0 - Digital Twin - The Simulation Aspect*, in *Mechatronics Futures*, D.Bradley and P.Hehenberger, Springer International Publishing, 2016, pp.59-74.
- [17] Y. Lage, N. Maia, M. Neves, and A. Ribeiro, *A force identification approach for multiple-degree-of-freedom systems*, in *Conference Proceedings of the Society for Experimental Mechanics Series*, 4th ed., T. Proulx, Ed. Springer, 2011, pp. 53-61.
- [18] V. Jayalakshmi, K. Lakshmi, and A. M. Rao, "Dynamic force reconstruction techniques from incomplete measurements," *Journal of Vibration and Control*, pp. 1–24, 2018. [Online]. Available: <http://journals.sagepub.com/doi/pdf/10.1177/1077546317752709>
- [19] S. Kabanikhin, *Inverse and ill-posed problems : theory and applications*. Berlin: De Gruyter, 2012, pp. 22-24.
- [20] J. Sanchez and H. Benaroya, "Review of force reconstruction techniques," *Journal of Sound and Vibration*, vol. 33(14), pp. 2999–3018, 2014.
- [21] R. Doraiswami, C. Diduch, and M. Stevenson, *Identification of Physical Systems*. Chichester, UK: John Wiley & Sons, Ltd, 2014, pp.231-288.
- [22] C. Ma, P. Tuan, D. Lin, and C. Liu, "A study of an inverse method for the estimation of impulsive loads," *International Journal Of Systems Science*, vol. 29(6), pp. 663–672, 1998.

-
- [23] C. Ma, J. Chang, and D. Lin, "Input forces estimation of beam structures by an inverse method," *Journal of Sound and Vibration*, vol. 259(2), pp. 387–407, 2003.
- [24] C. Ma and C. Ho, "An inverse method for the estimation of input forces acting on non-linear structural systems," *Journal of Sound and Vibration*, vol. 275(3-5), pp. 953–971, 2004.
- [25] F. Naets, J. Cuadrado, and W. Desmet, "Stable force identification in structural dynamics using kalman filtering and dummy-measurements," *Mechanical Systems and Signal Processing*, vol. 50-51, pp. 235–248, 2015.
- [26] E. Lourens, E. Reynders, G. D. Roeck, G. Degrande, and G. Lombaert, "An augmented kalman filter for force identification in structural dynamics," *Mechanical Systems and Signal Processing*, vol. 27, pp. 446–460, 2012.
- [27] K. Liu, S. Law, X. Zhu, and Y. Xia, "Explicit form of an implicit method for inverse force identification," *Journal of Sound and Vibration*, vol. 333 (3), pp. 730–744, 2014.
- [28] A. Deb, S. Roychoudhury, and G. Sarkar, *Analysis and Identification of Time-Invariant Systems, Time-Varying Systems, and Multi-Delay Systems using Orthogonal Hybrid Functions*. Springer International Publishing, Cham, 2016, pp. 167-168.
- [29] M. Pricop-Jeckstadt, "Nonlinear tikhonov regularization in hilbert scales with balancing principle tuning parameter in statistical inverse problems," *Inverse Problems in Science and Engineering*, pp. 1–32, 2018.
- [30] E. Jacquelin, A. Bennani, and P. Hamelin, "Force reconstruction: analysis and regularization of a deconvolution problem," *Journal of Sound and Vibration*, vol. 265(1), pp. 81–107, 2003.
- [31] J. Doyle, "A wavelet deconvolution method for impact force identification," *Experimental Mechanics*, vol. 37(4), pp. 403–408, 1997.
- [32] H. Kalhori, L. Ye, S. Mustapha, J. Li, and B. Li, "Reconstruction and analysis of impact forces on a steel-beam-reinforced concrete deck," *Experimental Mechanics*, vol. 56(9), pp. 1547–1558, 2016.
- [33] B. Qiao, X. Zhang, J. Gao, R. Liu, and X. Chen, "Sparse deconvolution for the large-scale ill-posed inverse problem of impact force reconstruction," *Mechanical Systems and Signal Processing*, vol. 83, pp. 93–115, 2017.
- [34] D. Kammer, "Input force reconstruction using a time domain technique," *Journal of Vibration and Acoustics*, vol. 120(4), pp. 868–874, 1998.
-

-
- [35] Y. Lage, N. Maia, M. Neves, and A. Ribeiro, “Force identification using the concept of displacement transmissibility,” *Journal of Sound and Vibration*, vol. 332(7), pp. 1674–1686, 2013.
- [36] L. Yu and T. Chan, “Moving force identification based on the frequency-time domain method,” *Journal of Sound and Vibration*, vol. 261(2), pp. 329–349, 2003.
- [37] M. Kern, *Numerical methods for inverse problems*. iSTE : Wiley, 2016, pp.52-94.
- [38] D. Schlichthärle, *Digital Filters: Basics and Design*, 2nd ed. Springer Berlin Heidelberg, 2011, pp.5-27.
- [39] Sundermeyer, “Iterative load calculations in flexible mechanisms,” *Technical document from Caterpillar (there is a patent on the theme)*, 2002 [Not published].
- [40] C. Cruz and E. Miranda, “Evaluation of the rayleigh damping model for buildings,” *Engineering Structures*, vol. 138, pp. 324–336, 2017.
- [41] Fedem Technology AS, *Fedem Release 7.2 Theory Guide*.

Appendix

A. Additional Information

A.1 Well-Posed Problems

Well-posed problems are often defined in terms of Hadamard as follows [19]:

Let an operator A map a topological space Q into a topological space F ($A : Q \rightarrow F$). For any topological space Q , let $\mathcal{O}(q)$ denote a neighbourhood of an element $q \in Q$. Throughout what follows, $D(A)$ is the domain of definition and $R(A)$ is the range of A .

The problem $Aq = f$ is *well-posed* on the pair of topological spaces Q and F if the following three conditions hold:

1. for any $f \in F$ there exists a solution $q_e \in Q$ to the equation $Aq = f$, i.e., $R(A) = F$ (the existence condition);
2. the solution q_e to the equation $Aq = f$ is unique in Q (the uniqueness condition);
3. for any neighbourhood $\mathcal{O}(q_e) \subset Q$ of the solution q_e to the equation $Aq = f$, there is a neighbourhood $\mathcal{O}(f) \subset F$ of the right-hand side f such that for all $f_\delta \in \mathcal{O}(f)$ the element $A^{-1}f_\delta = q_\delta$ belongs to the neighbourhood $\mathcal{O}(q_e)$, i.e., the operator A^{-1} is continuous (the stability condition).

[19, p. 24]

A.2 Dynamic Programming Solution Using Markov Parameters

T. Uhl [1] made the following algorithm to solve inverse problems using Markov parameters:

The dynamic programming solution to the inverse identification problem is based on minimization of the following function:

$$g(c) = \min_{f_j} E_n(y_0, u) \quad (\text{A.1})$$

This function represents the minimum value of E_n , given the formula (Equation 2.16 in section 2.4.2) starting at any stage $j = n$ with $y_n = c$ and simulating the system to $j = N$ with the optimal inputs u_j 's. The important item to emphasize are that c is considered to be arbitrary and that n can represent any value between 1 and N . Applying the principle of optimality leads to the following recurrence formula:

$$g_{n-1}(c) = \min_{g_{n-1}} [(y_{n-1} - c)^2 + Lu_{n-1}^2 + g_n(H_0c + u_{n-1})] \quad (\text{A.2})$$

The minimum at any point is determined by selecting the decision variable (f_{n-1}) that minimizes the cost function (A.1) and the remaining cost resulting from the previous value of the decision variable (the third term in (A.2)). The decision variable in the previous step will be the result in the next state, $H_0c + u_{n-1}$. The solution should be obtained by starting from the end of the process $n = N$ and working backward to $n = 1$. At the end point, $n = N$, the minimum value of E is given by:

$$g_N(c) = \min_{g_N} [(y_N - c)^2 + Lu_N^2] \quad (\text{A.3})$$

The force u_N that minimizes this expression is $u_N = 0$, which gives:

$$g_N(c) = (y_N - c)^2 \quad (\text{A.4})$$

This is interpreted to mean that, if there has only one stage and $y_N = c$, then the error is only related to y_N . This is the advantage of starting from the end point; the optimum solution is easy to determine at this point. To solve the inverse identification task, there is only one variable to be determined, the initial state, denoted by c . This value can be found by considering the recursive formula:

$$g_{N-1}(c) = \min_{g_{N-1}} [(y_{N-1} - c)^2 + Lu_{N-1}^2 + g_N(H_0c + u_{N-1})] \quad (\text{A.5})$$

Since g_N has been determined from (A.1) for an arbitrary argument, Eq. (A.2) becomes:

$$g_{N-1}(c) = \min_{g_{N-1}} [(y_{N-1} - c)^2 + Lu_{N-1}^2 + (y_N - H_0 c + u_{N-1})^2] \quad (\text{A.6})$$

Minimizing (A.6), the optimum value of the excitation (the force u^*) can be obtained from the following:

$$u_{N-1}^* = \frac{(y_N - H_0 c)}{(L + 1)} \quad (\text{A.7})$$

Substituting the expression (A.7) for u_{N-1}^* in (A.6) gives:

$$g_{N-1}(c) = \frac{(y_{N-1} - c)^2 + L(y_N - H_0 c)^2}{(L + 1)^2 + [y_N - H_0 c - (y_N - H_0 c)/(L + 1)]^2} \quad (\text{A.8})$$

At this point we have a complete solution for one step. If the iteration process started at $N - 1$ with $y_{N-1} = c$, then the optimal choice for force u_{N-1} is calculated using (A.5).

[1, p. 329-330]

A.3 The Effect of an Increase in Spring Stiffness

Figure A.1 and A.2 show the estimated load for a spring stiffness of 10^7 in the main file, and an external load frequency of 8.21767Hz.

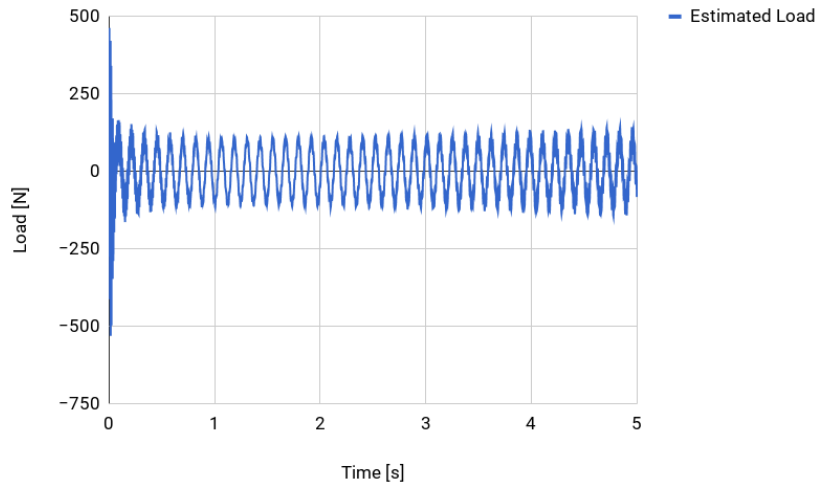


Figure A.1: Estimated load at resonance in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.

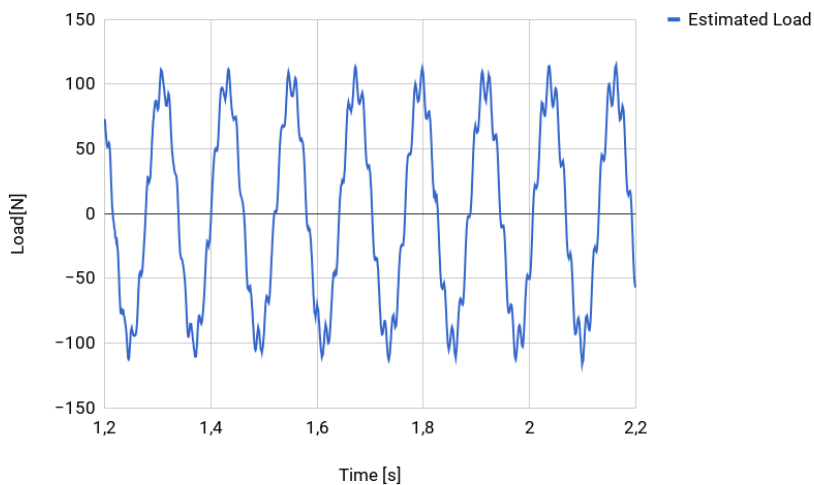


Figure A.2: Zoom at area of convergence.

Figure A.3 and A.4 show the estimated load for a spring stiffness of 10^7 in the main file, and an external load frequency of 13.0Hz.

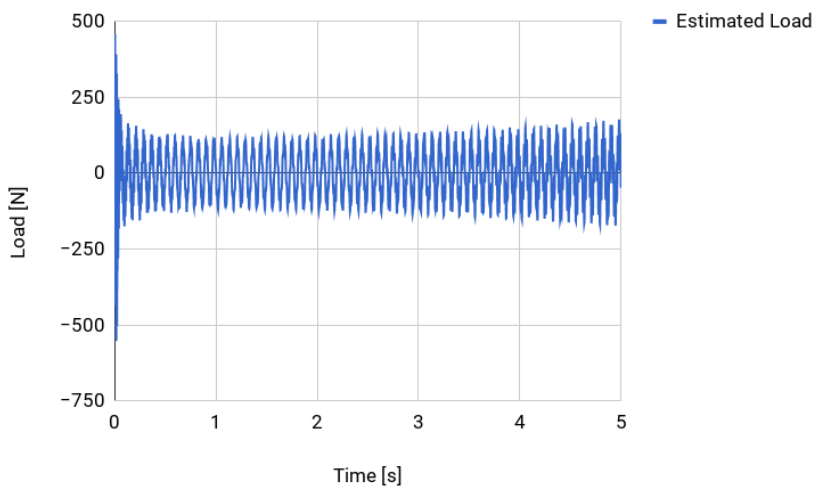


Figure A.3: Estimated load for an applied load frequency above eigenfrequency in a 5 seconds time interval, using a spring stiffness of 10^7 N/m.

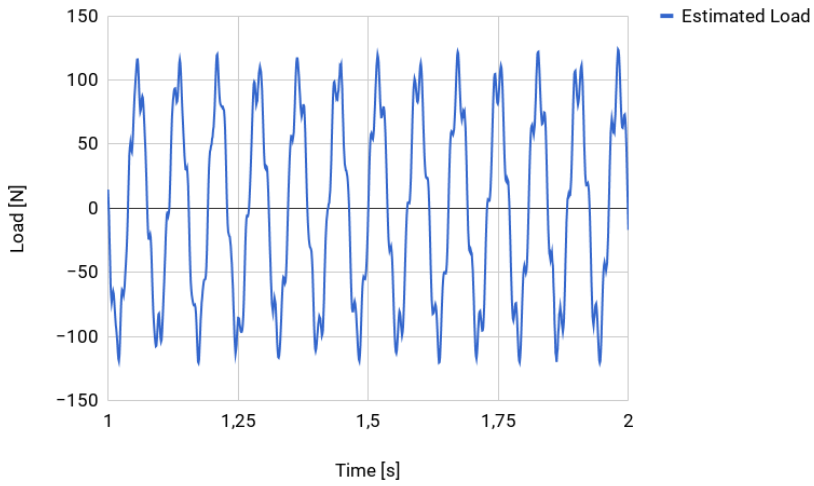


Figure A.4: Zoom at area of convergence.

A.4 Several Iterations

The point of convergence or closest to convergence in the chosen time period for an applied load of frequency 3.0Hz is shown in Figure A.5, A.6 and A.7.

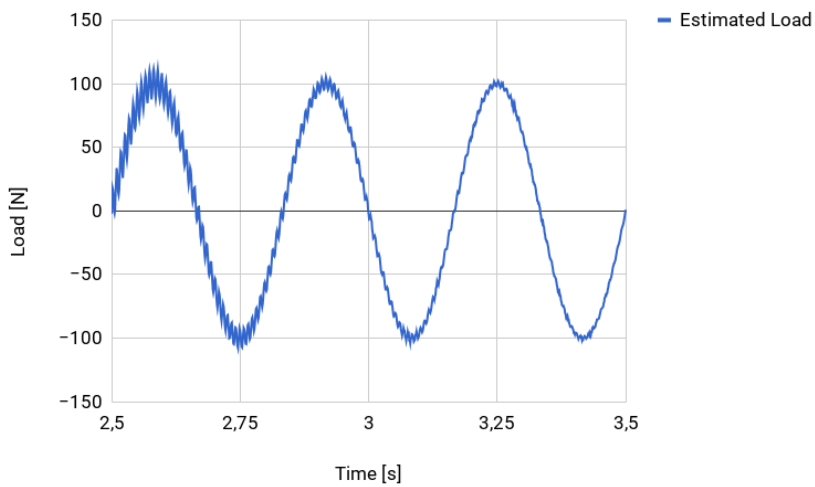


Figure A.5: Estimated load at convergence after one iteration.

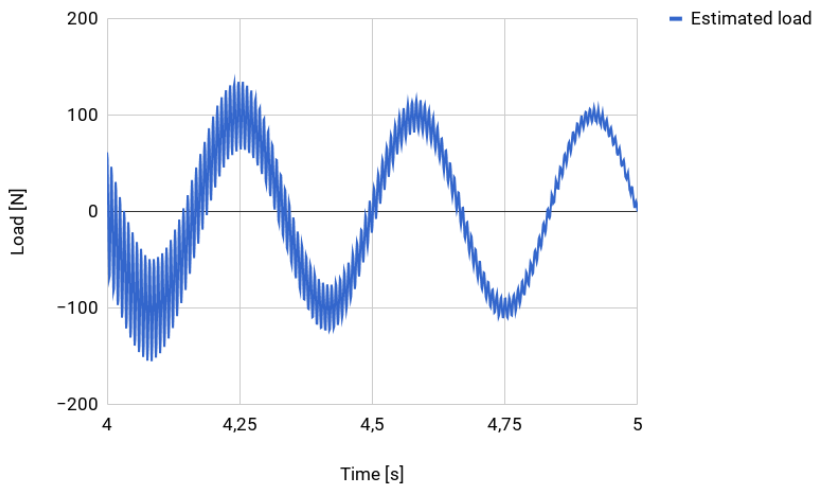


Figure A.6: Estimated load at convergence after two iterations.

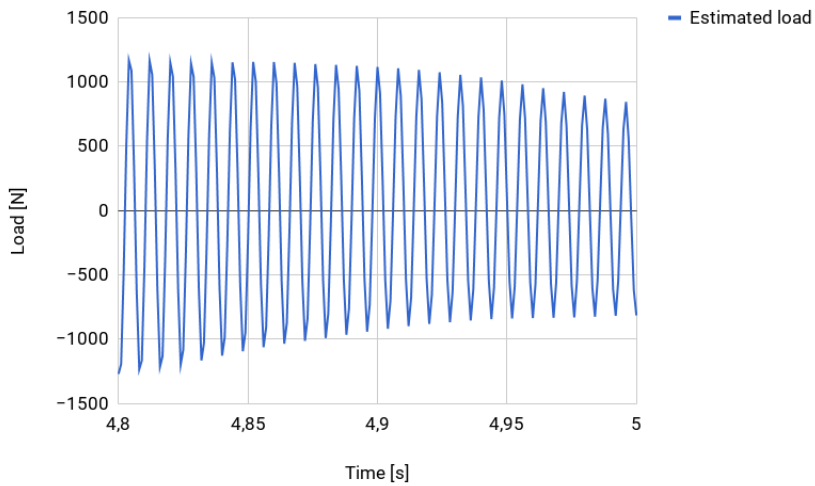


Figure A.7: Estimated load after three iterations.

A.5 Readout Points SDOF system

Force reconstruction with the use of 20 and 100 measurement readout points is shown in Figure A.8 and A.9 respectively.

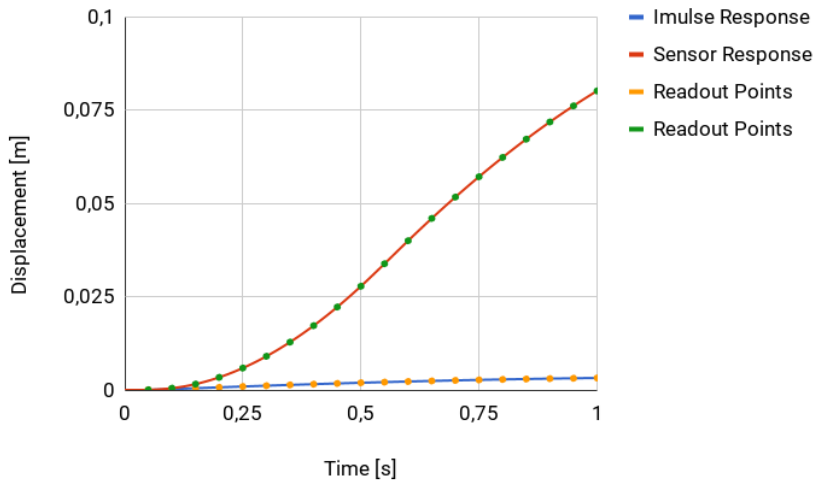


Figure A.8: 20 readout points.

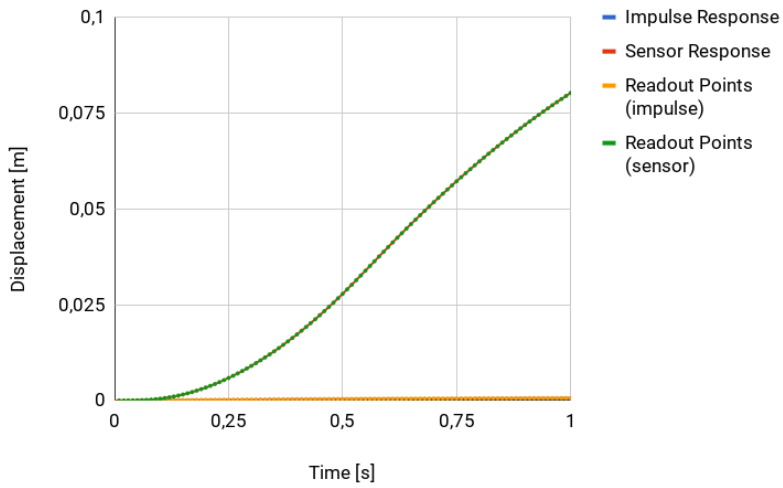


Figure A.9: 100 readout points.

A.6 Impulse Load Reconstruction

Figures A.10 and A.11 shows the resulting estimated loads and the applied loads when using 20 and 100 readout points respectively.

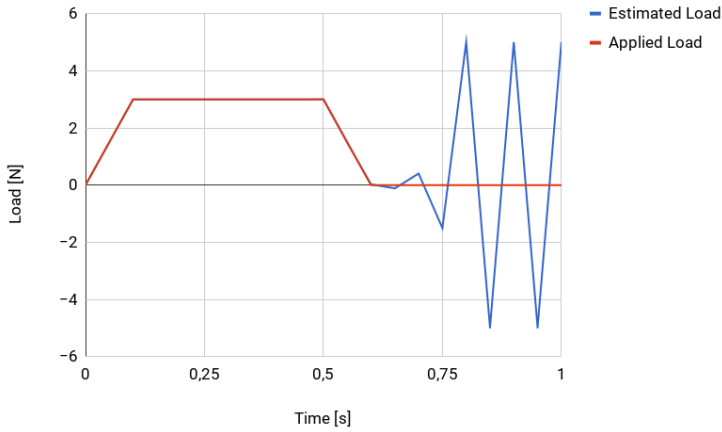


Figure A.10: Estimated load versus applied load based on 20 readout points.

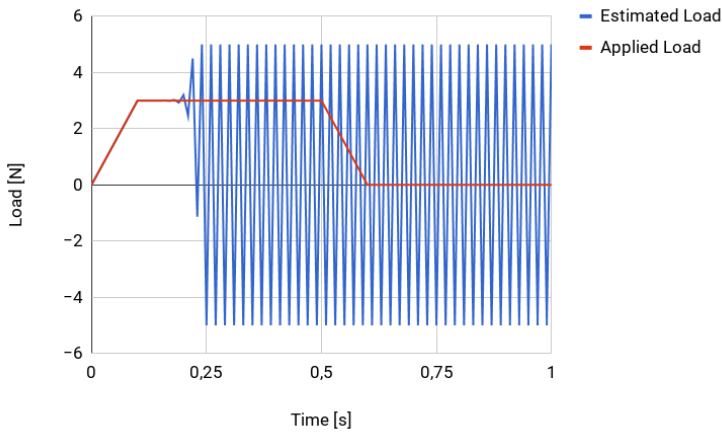


Figure A.11: Estimated load versus applied load based on 100 readout points.

B. Python Code

The code used to solve the estimated load using Markov parameters is shown in Figure B.1. The \mathbf{H} matrix consists of the Markov parameters attained from the impulse load. The \mathbf{x} matrix consists of the displacement read out points from the applied load. All values were obtained from Fedem, and deviated in each simulation, depending on the structure and the applied load.

```
8
9 import numpy as np
10 import cmath
11 import math
12 import matplotlib.pyplot as plt
13
14 sensorVec = np.array([x9, x8, x7, x6, x5, x4, x3, x2, x1, x0])
15
16 MarkovMatrix = np.array([[h0, h1, h2, h3, h4, h5, h6, h7, h8, h9],
17                          [0.0, h0, h1, h2, h3, h4, h5, h6, h7, h8],
18                          [0.0, 0.0, h0, h1, h2, h3, h4, h5, h6, h7],
19                          [0.0, 0.0, 0.0, h0, h1, h2, h3, h4, h5, h6],
20                          [0.0, 0.0, 0.0, 0.0, h0, h1, h2, h3, h4, h5],
21                          [0.0, 0.0, 0.0, 0.0, 0.0, h0, h1, h2, h3, h4],
22                          [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, h0, h1, h2, h3],
23                          [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, h0, h1, h2],
24                          [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, h0, h1],
25                          [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, h0]])
26
27
28 forceVec = np.linalg.solve(MarkovMatrix,sensorVec)
29
30 revForceVec = forceVec[::-1]
31
32 plt.plot(revForceVec)
33
34
35 print("==== Finished =====")
```

Figure B.1: Python script.