STRUCTURAL DAMAGE DETECTION USING AMBIENT VIBRATIONS

By

NADER NABIL AZIZ TADROS

B.S. of Civil Engineering, Zagazig University, Egypt, 2000

A THESIS

Submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Civil Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2014

Approved by:

Major Professor
Dr. Hani G Melhem

Copyright

NADER NABIL AZIZ TADROS

2014

Abstract

The objective of this research is to use structure ambient random vibration response to detect damage level and location. The use of ambient vibration is advantageous because excitation is caused by service conditions such as normal vehicle traffic on a highway bridge, train passage on a railroad bridge, or wind loads on a tall building. This eliminates the need to apply a special impact or dynamic load, or interrupt traffic on a bridge in regular service.

This research developed an approach in which free vibration of a structure is extracted from the response of this structure to a random excitation in the time domain (acceleration versus time) by averaging out the random component of the response. The result is the free vibration that includes all modes based on the sampling rate on time. Then this free vibration is transferred to the frequency domain using a Fast Fourier Transform (FFT). Variations in frequency response are a function of structural stiffness and member end-conditions. Such variations are used as a measure to identify the change in the structural dynamic properties, and ultimately detect damage.

A physical model consisting of a $20 \times 20 \times 1670$ -mm long steel square tube was used to validate this approach. The beam was tested under difference supports conditions varying from a single- to three-span continuous configuration. Random excitation was applied to the beam, and the dynamic response was measured by an accelerometer placed at various locations on the span. A numerical model was constructed in ABAQUS and the dynamic response was obtained from the finite element model subjected to similar excitation as in the physical model. Numerical results were correlated against results from the physical model, and comparison was made between the different span/support configurations. A subsequent step would be to induce damage that simulates loss of stiffness or cracking condition of the beam cross section, and that would be reflected as a change in the frequency and other dynamic properties of the structure.

The approach achieved good results for a structure with a limited number of degrees of freedom. Further research is needed for structures with a larger number of degrees of freedom and structures with damage in symmetrical locations relative to the accelerometer position.

Table of Contents

List of Figures

List of Tables

Acknowledgements

"I can do all things through Christ who strengtheneth me" (Philippians 4:13)

I would like to express my deep thanks to my advisor Dr. Hani G. Melhem, without whose continued support, guidance, valued time and encouragement I wouldn't be here as a graduate research assistant in the Civil Engineering Department at Kansas State University

I would like also to give a sincere acknowledgement to Dr. Asad Esmaeily for his support, patience and time spent with me during this research and other valuable Civil Engineering courses.

Also, I would like to thank Dr. Kyle Riding for serving on my committee. His advice and support are highly appreciated. I would also like to thank Dr. Robert Stokes, the Civil Engineering Department Interim Head, for department financial support to make this work done, Mr. Martin Courtois for his guidance as an ETDR consultant, and Mr. Ryan Benteman for constructing the physical model.

Thanks to all faculty who taught me very useful courses in my course work, administrative staff who are always doing their best to make the paper-work and other administrative issues to be achieved smoothly, the international student center staff for their guidance and support, and everyone who spend his/her time to support and encourage ambitious graduate students regarding more useful and helpful topics for the prosperity of the world.

Last but not least, I am giving a very special thank you to my God for his care, blessings and for giving me this great opportunity to be here at Kansas State University, and also all my family members one by one for their prayers, supports, encouragements and patience.

Dedication

This research is dedicated to all my family members for their prayers, love, support, patience and continuous encouragements.

Chapter 1 - INTRODUCTION

1.1 Introduction

Non-destructive testing (NDT) refers to a wide group of analysis techniques used in science and industry to evaluate the properties of a material, component, or system without causing damage. Nondestructive examination or evaluation (NDE), and nondestructive inspection (NDI) are also commonly used to describe this technology. Because NDT does not permanently alter the inspected article, it is a highly valuable technique that saves money and time in product evaluation, troubleshooting, and research. Common NDT methods include ultrasonic, magnetic-particle, liquid penetrant, radiographic, remote visual inspection (RVI), eddy-current testing, and low coherence interferometry. NDT is commonly used in Civil and Structural Engineering, Mechanical Engineering, Electrical Engineering, Forensic Engineering, Systems Engineering, Aeronautical Engineering, Medicine, and Art.

It is noted that the majority of the defects found in structures or machines are identified by visual inspection; however, visual inspection cannot detect small or internal flaws. Flaws found visually can be located near the fracture toughness or strength limit threshold of the material, and visual inspection is unable to accurately assess residual capacity of the structure. Therefore, NDE is a desirable method.

Generally, NDE methods can be categorized as active or passive; and surface, near surface, or volumetric. Active techniques are those defined as introducing some form of energy into or onto the specimen, and a change of the input energy is expected if a flow is presented. Vibration analysis, ultrasonic, magnetics, and radiography are examples of this technique. Passive techniques monitor the specimen without induced energy. Instead, the condition of the specimen is evaluated under ambient loading and possibly with a visual enhancing liquid over the surface. A defect is determined by any irregular response of the specimen. Visual examination, acoustic emission, and noise analysis are examples of this techniques. As the name implies, surface methods are those that are limited to detecting flaws in the specimen near or at the surface (e.g. eddy current, magnetics, penetrant methods, etc.). Volumetric methods are capable of finding voids and cracks throughout the thickness of the specimen. They include vibration, ultrasonic, and radiography.

Most active NDE methods are more elaborate processes of determining degradation of a structure, and would require an added cost for equipment and training. The benefit of NDE is superior information about the structure's overall condition. If implemented in the design phase of the system, proper NDE methods can increase the life of its components. NDE should reduce the overall costs associated with the life of the structure by detecting flaws when they are small thus limiting downtime for repair of larger problems.

1.2 Advantages and Applications of NDE on Structures

Many benefits are achieved when NDE is used on civil structures, especially when accessibility limitations are present on structure elements, in addition to the capability of keeping the tested location without any damage and the simplicity of processing the required tests compared to other destructive tests. NDE could be used to temporarily or permanently evaluate and monitor structures.

The following are some Non-destructive tests (NDTs) that are commonly used in structural engineering practice:

- Schmidt Hammer to evaluate reinforced concrete strength.
- Concrete ultrasonic tester to measure crack depth.
- Ground penetrating radar for rebar detection.
- Digital coating thickness gauge to measure painting thickness of steel members
- Steel ultrasonic tester to measure steel member thickness and detect internal defects in welded areas.

1.3 Objective of this Research

The objective of this research is to develop an approach in which free vibration of a structure is extracted from response of structure to random excitation in the time domain (acceleration versus time) by averaging out the random component of the response. The result is the free vibration that includes all modes based on the sampling rate on time. Then this free vibration is transferred to the frequency domain using a Fast Fourier Transform (FFT). Variations in frequency response are a function of structural stiffness and member end-conditions. Such variations are used as a measure to identify the change in the structural dynamic

properties, and ultimately detect damage. Bridges are the most common structures that will benefit from this procedure due to their importance, traffic loads, and the required Structural Health Monitoring (SHM) for their performance.

The main advantages of ambient vibration compared to other non-destructive testing procedures are:

- No traffic disruption
- Utilization of vibration occurring on a bridge
- Limited preparation and testing time
- Lower cost and simple when equipment is available
- Using ordinary traffic
- Measuring the structure acceleration
- No need for specific loads
- Using dynamic data
- No damage is inflicted to the structure

The ultimate goal of this effort is to use the response of structure to a random excitation results from traffic, wind, and/or earthquakes to evaluate and monitor the structure performance by recording the structure vibration response due to the mentioned reasons and evaluate this vibration signal to determine any changes in the structure performance due to aging, different and repetitive loading conditions, the impact of climate and other environmental issues.

Chapter 2 - LITERATURE REVIEW – FREE VIBRATION FOR MULTI-DEGREE OF FREEDOM SYSTEMS

Free vibration relates to the motion of a structure without any dynamic excitation, external forces or support motion. It is initiated by distributing the structure from its equilibrium position by some initial displacement and/or by imparting some initial velocities.

This chapter on free vibration of Multi-Degree of Freedom (MDF) is divided into two parts. In Part (2.1), the notation of natural frequencies and natural modes of vibration of a structure have been developed; These concepts play a central role in the dynamic and earthquake analysis of linear systems.

Part (2.2) is a description of the use of these vibration properties to determine the free vibration response of systems. Undamped systems are analyzed first, followed by a discussion of the difference between the free vibration response of systems with classical damping and that of systems with nonclassical damping. The analysis procedure is extended to systems with classical damping, recognizing that such systems possess the same natural modes as the undamped system.

2.1 Natural Vibration Frequencies and Modes

2.1.1 Systems without Damping

Free vibration of linear MDF systems is governed by

$$m\ddot{u} + c\dot{u} + ku = p(t) \tag{2.1}$$

where,

m **:** mass

$\ddot{u}$ **:** acceleration

c **:** damping

$\dot{u}$ **:** velocity

k **:** stiffness

u **:** displacement

with $p(t) = 0$ for free vibration (no force). For systems without damping, Equation (2.1) becomes

$$m\ddot{u} + ku = 0 \tag{2.2}$$

Equation (2.2) represents N homogeneous differential equations that are coupled through the mass matrix, the stiffness matrix, or both matrices; N is the number of DOFs. It is desired to find the solution u(t) of Equation (2.2) that satisfies the initial condition at $t = 0$:

$$u = u(0) \ \& \ \dot{u} = \dot{u}(0) \tag{2.3}$$

Figure (2.1) shows the free vibration of a two story shear frame. The story stiffness and lumped masses at the floors are noted, and the free vibration is initiated by the deflection shown by curve *a* in Figure (2.1b). The resulting motion u_j of the two masses is plotted in Figure (2.1d) as a function of time; T_1 will be defined later.
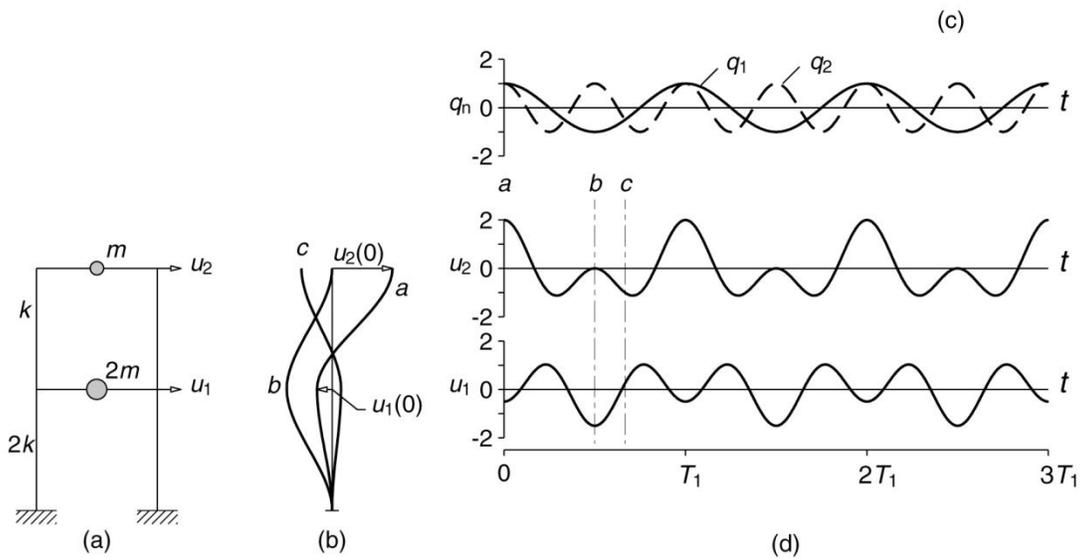
Figure 2.1 Free vibration of an undamped system due to arbitrary initial displacement: (a) two-story frame; (b) deflected shapes at time instants *a, b,* and *c*; (c) modal coordinates q_n (*t*); (d) displacement history. (Chopra, 2000)

5

Deflected shapes of the structure at selected time instants *a, b,* and *c* are also shown. The displacement–time plot for the jth floor starts with the initial conditions $u_j(0)$ and $\dot{u}_j(0)$; the $u_j(0)$ are identified in Fig. 2.1b and $\dot{u}_j(0) = 0$ for both floors. The motion of each mass or floor is not a single harmonic motion as in Single Degree of Freedom (SDF) systems, and the frequency of the motion cannot be defined. Furthermore, the deflected shape (i.e., the ratio u_1 / u_2) varies with time, as evidenced by differing deflected shapes *b* and *c,* which also differ from the initial deflected shape *a.*

An undamped structure would undergo simple harmonic motion without change of deflected shape if free vibration is initiated by appropriate distribution of displacements in the various DOFs. As shown in Fig. 2.2 and 2.3, two characteristic deflected shapes exist for this two–DOF system such that if it is displaced in one of these shapes and released, it will vibrate in simple harmonic motion, maintaining the initial deflected shape. Both floors reach their extreme displacements at the same time and pass through the equilibrium position at the same time. Each characteristic deflected shape is called a *natural mode of vibration* of an MDF system.

Observe that the displacements of both floors are in the same direction in the first mode but in opposite directions in the second mode. The point of zero displacement, called a *node*, does not move at all (Fig. 2.3); as the mode number *n* increase, the number of nodes increases accordingly.
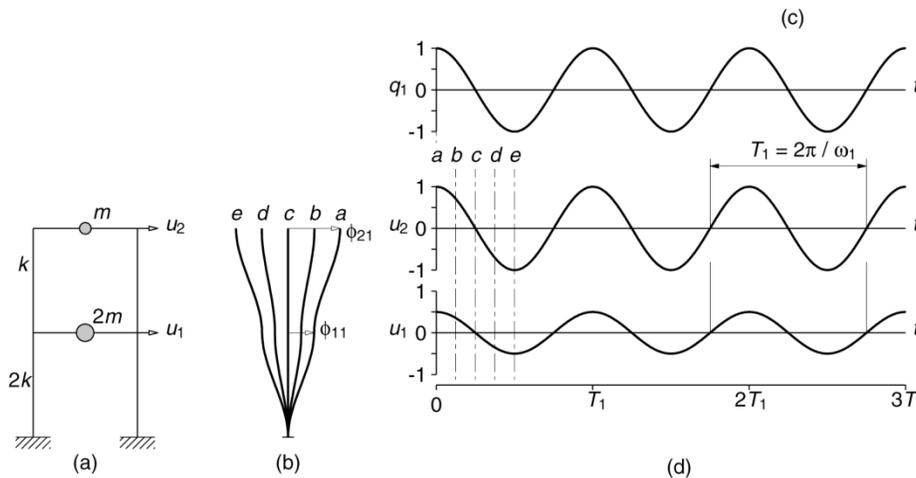
Figure 2.2 Free vibration of an undamped system in its first natural mode of vibration: (a) two-story frame; (b) deflected shapes at time instants *a, b, c, d,* and *e;* (c) modal coordinate $q_1(t)$; (d) displacement history. (Chopra, 2000)

6
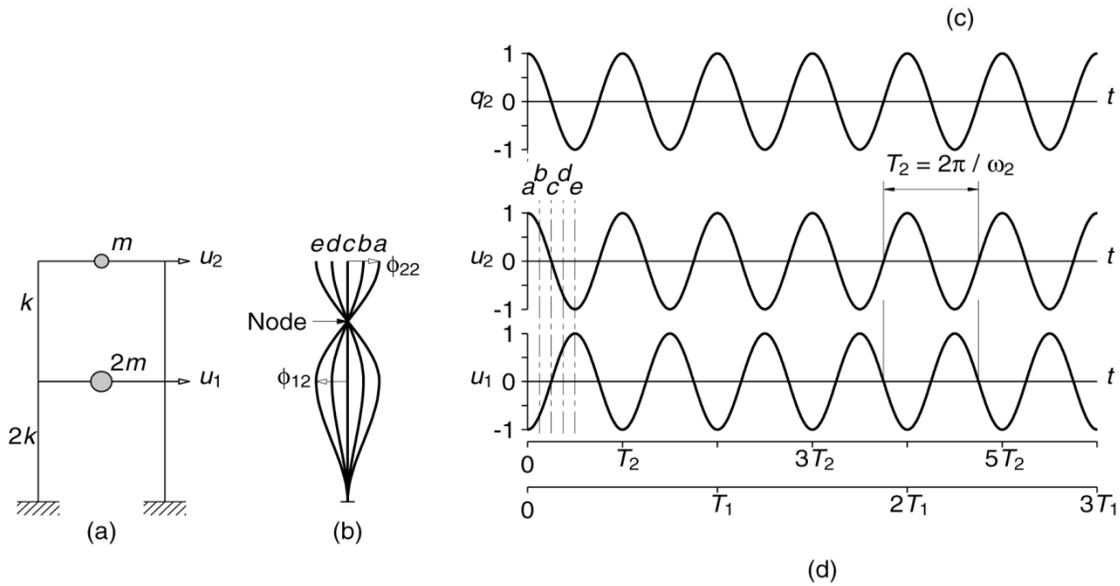
Figure 2.3 Free vibration of an undamped system in its second natural mode of vibration: (a) two-story frame; (b) deflected shapes at the time instants *a, b, c, d*, and *e;* (c) modal coordinate $q_2(t)$; (d) displacement history. (Chopra, 2000)

A *natural period of vibration T_n* of a MDF system is the time required for one cycle of the simple harmonic motion in one of these natural modes. The corresponding *natural circular frequency of vibration* is ω_n and the *natural cyclic frequency of vibration* is f_n, where

$$T_n = \frac{2\pi}{\omega_n} \tag{2.4}$$

$$f_n = \frac{1}{T_n} \tag{2.5}$$

Figures 2.2 and 2.3 show the two natural periods T_n and natural frequencies ω_n ($n = 1, 2$) of the two story building vibrating in its natural modes $\emptyset_n = \langle \emptyset_{1n} \quad \emptyset_{2n} \rangle^T$. The smaller of the two natural vibration frequencies is denoted by ω_1, and the larger natural vibration frequency is donated by ω_2. Correspondingly, the longer of the two natural vibration periods is denoted by T_1 and the shorter one as T_2.

2.1.2 Natural Vibration Frequencies and Modes

This section introduces an eigenvalue problem whose solution gives the natural frequencies and modes of a system. Free vibration of an undamped system is one of its natural vibration modes (graphically displayed in Figure 2.2 and 2.3 for a two-DOF system), can be described mathematically by

$$u(t) = q_n(t) \emptyset_n \tag{2.6}$$

where the deflected shape $\emptyset_n$ does not vary with time. The time variation of the displacement is described by the simple harmonic function

$$q_n(t) = A_n \cos \omega_n t + B_n \sin \omega_n t \tag{2.7}$$

where A_n and B_n are constants of integration that can be determined from the initial conditions that initiate the motion. Combining Equations (2.6) and (2.7) gives

$$u(t) = \emptyset_n (A_n \cos \omega_n t + B_n \sin \omega_n t) \tag{2.8}$$

where ω_n and $\emptyset_n$ are unknown.

Substituting this form of u(t) in Equation (2.2) gives

$$[-\omega_n^2 m \emptyset_n + k \emptyset_n] q_n(t) = 0 \tag{2.9}$$

This equation can be satisfied in one of two ways. Either $q_n(t) = 0$, that implies that u(t) = 0 for which there is no motion of the system (this is the so-called trivial solution), or the natural frequencies ω_n and modes $\emptyset_n$ must satisfy the following algebraic equation:

$$k \emptyset_n = \omega_n^2 m \emptyset_n \tag{2.10}$$

which provides a useful condition. This algebraic equation is called the *matrix eigenvalue problem*. When necessary it is called the real eigenvalue problem. The stiffness and mass matrices k and m are known; the problem is to determine the scalar ω_n^2 and vector $\emptyset_n$.

To indicate the formal solution to Equation (2.10), it is written as

$$[k - \omega_n^2] \emptyset_n = 0 \tag{2.11}$$

which can be interpreted as a set of *N* homogeneous algebraic equations for the *N* elements $\emptyset_{jn}$ (j = 1, 2, ….., *N*). This set always has the trivial solution $\emptyset_n = 0$, which is not useful because it implies no motion. It has nontrivial solution if

$$\det[k - \omega_n^2] = 0 \tag{2.12}$$

When the determinate is expanded, a polynomial of order *N* in ω_n^2 is obtained. Equation (2.12) is known as the *characteristic equation* or *frequency equation*. This equation has *N* real and positive roots for ω_n^2 because m and k, the structure mass and stiffness matrices, are symmetric

and positive definite. The positive definite property of k is assured for all structures supported in a way that prevents rigid-body motion. Such is the case for civil engineering structures of interest, but not for unrestrained structures such as aircraft in flight. The positive definite property of m is also assured because the lumped masses are nonzero in all DOFs retained in the analysis after the DOFs with zero lumped mass have been eliminated by static condensation.

The N roots ω_n^2 of Equation (2.12) determine the N natural frequencies ω_n (n = 1, 2, .., N) of vibration. These roots of the characteristic equation are also known as *eigenvalues, characteristic values,* or *normal values.* When a natural frequency ω_n is known, Equation (2.11) can be solved for the corresponding vector $\emptyset_n$ to within a multiplicative constant. The eigenvalue problem does not fix the absolute amplitude of the vectors $\emptyset_n$, only the shape of the vector given by the relative values of the N displacements $\emptyset_{jn}$ (j = 1, 2, ….., N). Corresponding to the N natural vibration frequencies ω_n of an N-DOF system, there are N independent vectors $\emptyset_n$ which are known as *natural modes of vibration,* or *natural mode shapes of vibration.* These vectors are also known as *eigenvalues, characteristic vectors,* or *normal modes.*

In summary, a vibrating system with N DOFs has N natural vibration frequencies ω_n (n = 1, 2, …., N), arranged in sequence from smallest to largest ($\omega_1 < \omega_2 < \cdots < \omega_N$); corresponding natural periods T_n ; and natural modes $\emptyset_n$. The term *natural* is used to qualify each of these vibration properties to emphasize the fact that these are natural properties of the structure in free vibration, and they depend only on its mass and stiffness properties. The subscript *n* denotes the mode number and the first mode (n = 1) is also known as the fundamental mode.

2.1.3 Modal and Spectral Matrices

The N eigenvalues and N natural modes can be assembled compactly into matrices. Let the natural mode $\emptyset_n$ corresponding the natural frequency ω_n have elements $\emptyset_{jn}$, where *j* indicates the DOFs. The N eigenvectors then can be displayed in a single square matrix, each column of which is a natural mode:

$$\Phi = \begin{bmatrix} \emptyset_{jn} \end{bmatrix} = \begin{bmatrix} \emptyset_{11} & \emptyset_{12} & \cdots & \emptyset_{1N} \\ \emptyset_{21} & \emptyset_{22} & & \emptyset_{2N} \\ \vdots & & \ddots & \vdots \\ \emptyset_{N1} & \emptyset_{N2} & \cdots & \emptyset_{NN} \end{bmatrix}$$

The matrix ϕ is called the *modal matrix* for the eigenvalue problem, Equation (2.10). The N eigenvalues ω_n^2 can be assembled into a diagonal matrix Ω^2, which is known as the *spectral matrix* of the eigenvalue problem, equation (2.10):

$$\Omega^2 = \begin{bmatrix} \omega_1^2 & & & \\ & \omega_2^2 & & \\ & & \ddots & \\ & & & \omega_N^2 \end{bmatrix}$$

Each eigenvalue and eigenvector satisfies Equation (2.10), which can be written as the relation

$$k\,\emptyset_n = m\,\emptyset_n\,\omega_n^2 \tag{2.13}$$

By using the modal and spectral matrices, all relations $(n = 1, 2, \ldots, N)$ can be assembled into a single matrix equation:

$$k\phi = m\phi\Omega^2 \tag{2.14}$$

Equation (2.14) provides a compact presentation of the equations relating all eigenvalues and eigenvectors.

2.1.4 Orthogonality of Modes

The natural modes corresponding to different natural frequencies can be shown to satisfy the following orthognality conditions. When $\omega_n \neq \omega_r$,

$$\emptyset_n^T\,k\,\emptyset_r = 0 \tag{2.15}$$
$$\emptyset_n^T\,m\,\emptyset_r = 0 \tag{2.16}$$

These important properties can be proven as follow: The nth natural frequency and mode satisfy Equation (2.10); premultiplying it by $\emptyset_r^T$, the transpose of $\emptyset_r$, gives

$$\emptyset_r^T\,k\,\emptyset_n = \omega_n^2\,\emptyset_r^T\,m\,\emptyset_n \tag{2.17}$$

Similarly, the rth natural frequency and mode satisfy Equation (2.10); thus $k\,\emptyset_r = \omega_r^2\,m\,\emptyset_r$. Premultiplying by $\emptyset_n^T$ gives

$$\emptyset_n^T\,k\,\emptyset_r = \omega_r^2\,\emptyset_n^T\,m\,\emptyset_r \tag{2.18}$$

The transpose of the matrix on the left side of Equation (2.17) will equal the transpose of the matrix on the right side of the equation; thus

$$\emptyset_n^T\,k\,\emptyset_r = \omega_n^2\,\emptyset_n^T\,m\,\emptyset_r \tag{2.19}$$

wherein we have utilized the symmetry property of the mass and stiffness matrices. Subtracting Equation (2.18) from Equation (2.19) gives

$$(\omega_n^2 - \omega_r^2)\, \emptyset_n^T\, m\, \emptyset_r = 0 \tag{2.20}$$

Equation (2.16) is true when $\omega_n^2 \neq \omega_r^2$ which for systems with positive natural frequencies implies that $\omega_n \neq \omega_r$. Substituting (2.16) in (2.18) indicates that Equation (2.15) is true when $\omega_n \neq \omega_r$. This completes a proof for the orthogonality relations of Equations (2.15) and (2.16).

We have established the orthogonality relations between modes with distinct frequencies (i.e., $\omega_n \neq \omega_r$). If the frequency equation (2.10) has a *j*-fold multiple root (i.e., the system has one frequency repeated *j* times) it is always possible to find *j* modes associated with this frequency that satisfy Equations (2.15) and (2.16). If these *j* modes are included with the modes corresponding to the other frequencies, a set of *N* modes is obtained which satisfies Equations (2.15) and (2.16) for $n \neq r$.

The orthogonality of natural modes implies that the following square matrices are diagonal:

$$K \equiv \phi^T\, k\, \phi \tag{2.21}$$
$$M \equiv \phi^T\, m\, \phi \tag{2.22}$$

where the diagonal elements are

$$K_n = \emptyset_n^T\, k\, \emptyset_n \tag{2.23}$$
$$M_n = \emptyset_n^T\, m\, \emptyset_n \tag{2.23}$$

Since *m* and *k* are positive definite, the diagonal elements of *K* and *M* are positive. They are related by

$$K_n = \omega_n^2\, M_n \tag{2.24}$$

This can be demonstrated from the definition of K_n and M_n as follows: Substituting equation (2.10) in (2.23) gives

$$K_n = \emptyset_n^T\, (\omega_n^2\, m\, \emptyset_n) = \omega_n^2\, (\emptyset_n^T\, m\, \emptyset_n) = \omega_n^2\, M_n \tag{2.25}$$

2.1.5 Interpretation of Modal Orthogonality

In this section we developed physically motivated interpretations of the orthogonality properties of natural modes. One implication of modal orthogonality is that the work done by the nth-mode inertia forces in going through the rth-mode displacements is zero. To demonstrate this result, consider a structure vibrating in the nth mode with displacements

$$u_n(t) = q_n(t)\emptyset_n \tag{2.26}$$

The corresponding accelerations are $\ddot{u}_n(t) = \ddot{q}_n(t)\,\emptyset_n$ and the associated inertia forces are

$$(f_I)_n = -m\,\ddot{u}_n(t) = -m\,\emptyset_n\,\ddot{q}_n(t) \tag{2.27}$$

Next, consider displacements of the structure in its rth natural mode:

$$u_r(t) = q_r(t)\,\emptyset_r \tag{2.28}$$

The work done by the inertia forces of Equation (2.27) in going through the displacements of Equation (2.28) is

$$(f_I)_n^T\,u_r = -(\emptyset_n^T\,m\,\emptyset_r)\,\ddot{q}_n(t)\,\ddot{q}_r(t) \tag{2.29}$$

Which is zero because of the modal orthogonality relation of Equation (2.16). This completes the proof.

Another implication of the modal orthogonality properties is that the work done by the equivalent static forces associated with displacements in the nth mode in going through the rth mode displacements equal to zero. These forces are

$$(f_S)_n = k\,u_n(t) = k\,\emptyset_n\,q_n(t) \tag{2.30}$$

and the work they do in going through the displacements of Equation (2.28) is

$$(f_S)_n^T\,u_r = (\emptyset_n^T\,k\,\emptyset_r)q_n(t)\,q_r(t) \tag{2.31}$$

which is zero because of the modal orthogonality relation of Equation (2.15). This completes the proof.

2.1.6 Normalization of Modes

As mentioned earlier, the eigenvalue problem, Equation (2.10), determines the natural modes to only within a multiplicative factor. If the vector $\emptyset_n$ is a natural mode, any vector proportional to $\emptyset_n$ is essentially the same natural mode because it also satisfies Equation (2.10). Scale factor are sometimes applied to natural modes to standardize their elements associated with various DOFs. This process is called *normalization.* Sometimes it is convenient to normalize each mode so that its largest element is unity. Other times it may be advantageous to normalize each mode so that the element corresponding to a particular DOF, say the top floor of a multistory building, is unity. In theoretical discussions and computer programs it is common to normalize modes so that the M_n have unit values. In this case

$$M_n = \emptyset_n^T\,m\,\emptyset_n = 1 \tag{2.32}$$

$$\phi^T\,m\,\phi = I \tag{2.33}$$

12

where I is the identity matrix with unit values along the main diagonal. Equations (2.32) and (2.33) states that the natural modes are not only orthogonal but are normalized with respect to m. They are then called a mass *orthonormal set*. When the modes are normalized in this manner, equations (2.23) and (2.22) become

$$K_n = \emptyset_n^T k \, \emptyset_n = \omega_n^2 M_n = \omega_n^2 \tag{2.34}$$

$$K = \phi^T k \, \phi = \Omega^2 \tag{2.35}$$

2.1.7 Modal Expansion of Displacement

Any set of N independent vectors can be used as a basis for representing any other vector of order N. In the following sections the natural modes are used as such a basis. Thus, a modal expansion of any displacement vector u has the form

$$u = \sum_{r=1}^{N} \emptyset_r \, q_r = \phi_q \tag{2.36}$$

where q_r are scalar multipliers called *modal coordinates* or *normal coordinates* and $q = \langle q_1 \quad q_2 \quad ... \quad q_n \rangle^T$. When the $\emptyset_r$ are known, for a given u it is possible to evaluate the q_r by multiplying both sides of equation (2.36) by $\emptyset_n^T \, m$:

$$\emptyset_n^T \, m \, u = \sum_{r=1}^{N} (\emptyset_n^T \, m \, \emptyset_r) \, q_r \tag{2.37}$$

Because of the orthogonality relation of Equation (2.16), all terms in the summation above vanish except the $r = n$ term; thus

$$\emptyset_n^T \, m \, u = (\emptyset_n^T \, m \, \emptyset_n) \, q_n \tag{2.38}$$

The matrix products on both sides of this equation are scalars. Therefore,

$$q_n = \frac{\emptyset_n^T \, m \, u}{\emptyset_n^T \, m \, \emptyset_n} = \frac{\emptyset_n^T \, m \, u}{M_n} \tag{2.39}$$

The modal expansion of the displacement vector u, equation (2.36), is employed in Section (2.2.1) to obtain solutions for the free vibration response of undamped systems. It also plays a central role in the analysis of forced vibration response and dynamic response of MDF systems.

2.2 Free Vibration Response

2.2.1 Solution of Free Vibration Equation: Undamped Systems

This section revives discussion of the problem posed by Equations (2.2) and (2.3) and find its solution. For the example structure of Figure (2.1a), such a solution was shown in Figure (2.1b). The differential Equation (2.2) to be solved had led to the matrix eigenvalue problem of Equation (2.10). Assuming that the eigenvalue problem has been solved for the natural frequencies and modes, the general solution of Equation (2.2) is given by a superposition of the response in individual modes given by Equation (2.8). Therefore

$$u(t) = \sum_{n=1}^{N} \emptyset_n \left(A_n \cos \omega_n t + B_n \sin \omega_n t \right) \tag{2.40}$$

where A_n and B_n are $2N$ constants of integration. To determine these constants, we will also need the equation for the velocity vector, which is

$$\dot{u}(t) = \sum_{n=1}^{N} \emptyset_n \omega_n \left(-A_n \sin \omega_n t + B_n \cos \omega_n t \right) \tag{2.41}$$

Setting $t = 0$ in Equations (2.40) and (2.41) gives

$$u(0) = \sum_{n=1}^{N} \emptyset_n A_n \tag{2.42}$$

$$\dot{u}(0) = \sum_{n=1}^{N} \emptyset_n \omega_n B_n \tag{2.43}$$

With the initial displacement $u(0)$ and initial velocity $\dot{u}(0)$ known, each of these two equation sets represents N algebraic equations in the unknown A_n and B_n, respectively. Simultaneous solution of these equations is not necessary because they can be interpreted as a modal expansion of the vectors $u(0)$ and $\dot{u}(0)$. Following Equation (2.36), we can write

$$u(0) = \sum_{n=1}^{N} \emptyset_n q_n(0) \tag{2.44}$$

$$\dot{u}(0) = \sum_{n=1}^{N} \emptyset_n \dot{q}_n(0) \tag{2.45}$$

where, analogous to Equation (2.39), $q_n(0)$ and $\dot{q}_n(0)$ are given by

$$q_n(0) = \frac{\emptyset_n^T \, m \, u(0)}{M_n} \tag{2.46}$$

$$\dot{q}_n(0) = \frac{\emptyset_n^T \, m \, \dot{u}(0)}{M_n} \tag{2.47}$$

Equations (2.42), (2.43), (2.44) and (2.45) are equivalent, implying that $A_n = q_n(0)$ and $B_n = \dot{q}_n(0)/\omega_n$. Substituting these in Equation (2.40) gives

$$u(t) = \sum_{n=1}^{N} \emptyset_n \left[q_n(0) \cos \omega_n t + \frac{\dot{q}_n(0)}{\omega_n} \sin \omega_n t \right] \tag{2.48}$$

or, alternatively,

$$u(t) = \sum_{n=1}^{N} \emptyset_n \, q_n(t) \tag{2.49}$$

where

$$q_n(t) = q_n(0) \cos \omega_n t + \frac{\dot{q}_n(0)}{\omega_n} \sin \omega_n t \tag{2.50}$$

is the time variation of modal coordinates, which is analogous to the free vibration response of SDF systems. Equation (2.48) is the solution of the free vibration problem. It provides the displacement u as a function of time due to initial displacement $u(0)$ and velocity $\dot{u}(0)$. Assuming that the natural frequencies ω_n and modes $\emptyset_n$ are available, the right hand side of Equation (2.48) is known with $q(0)$ and $\dot{q}(0)$ defined by Equations (2.46) and (2.47).

2.2.2 Free Vibration of Systems with Damping

When damping is included, the free vibration response of the system is governed by Equation (2.1) with $p(t) = 0$

$$m\ddot{u} + c\dot{u} + ku = 0 \tag{2.51}$$

It is desired to find the solution $u(t)$ of Equation (2.51) that satisfies the initial conditions

$$u = u(0) \ \& \ \dot{u} = \dot{u}(0) \tag{2.52}$$

at $t = 0$. A procedure to obtain the desired solution will be developed in Section (2.2.3) for certain forms of damping that are reasonable models for many real structures. In this section the solution is presented for a specific system that enables us to understand qualitatively the effects of damping on the free vibration of MDF systems.

For this purpose we express the displacement u in terms of the natural modes of the system without damping; thus Equation (2.36) is substituted in Equation (2.51):

$$m \, \phi \, \ddot{q} + c \, \phi \, \dot{q} + k \, \phi \, q = 0 \qquad (2.53)$$

Premultiplying by ϕ^T gives

$$M \, \ddot{q} + C \, \dot{q} + K \, q = 0 \qquad (2.54)$$

where the diagonal matrices M and K were defined in Equations (2.21), (2.22) and

$$C = \phi^T \, c \, \phi \qquad (2.55)$$

The square matrix C may or may not be diagonal, depending on the distribution of damping in the system. If C is diagonal, Equation (2.54) represents N uncoupled differential equations in modal coordinates q_n, and the system is said to have *classical damping* because classical modal analysis is applicable to such systems. These systems possess the same natural modes as those of the undamped system. Systems with damping such that C is nondiagonal are said to have *nonclassical damping*. These systems are not amenable to classical modal analysis, and they do not possess the same natural modes as undamped system.

2.2.3 Solution of Free Vibration Equations: Classically Damped Systems

This section represents a formal solution for free vibration of systems with classical damping due to initial displacements and/or initial velocities is presented. For this form of damping the natural modes are unaffected by damping. Therefore, the natural frequencies and modes of the system are first computed for the system without damping; the effect of damping on the natural frequencies is considered in the same manner as for a SDF system. This becomes apparent by dividing Equation (2.56) governing $q_n(t)$ by M_n to obtain Equation (2.57)

$$M_n \, \ddot{q}_n + C_n \, \dot{q}_n + K_n \, q_n = 0 \qquad (2.56)$$

$$\ddot{q}_n + 2 \, \xi_n \, \omega_n \, \dot{q}_n + \omega_n^2 \, q_n = 0 \qquad (2.57)$$

This equation is of the same form as Equation (2.58) when dividing Equation (2.51) by m

$$\ddot{u} + 2 \, \xi \, \omega_n \, \dot{u} + \omega_n^2 = 0 \qquad (2.58)$$

where $\omega_n = \sqrt{k/m}$ and $\xi = \dfrac{c}{2 \, m \, \omega_n}$

governing the free vibration of a SDF system with damping for which the solution is equation (2.59)

$$u\,(t) = e^{-\xi \, \omega_n \, t} \left[u(0) \cos \omega_D \, t + \frac{\dot{u}(0) + \xi \, \omega_n \, u(0)}{\omega_D} \sin \omega_D \, t \right] \qquad (2.59)$$

where

$$\omega_D = \omega_n \sqrt{1 - \xi^2}$$

Adapting this result, the solution for Equation (2.57) is given by

$$q_n(t) = e^{-\xi_n \omega_n t} \left[q_n(0) \cos \omega_{nD} t + \frac{\dot{q}_n(0) + \xi_n \omega_n q_n(0)}{\omega_{nD}} \sin \omega_{nD} t \right] \qquad (2.60)$$

where the *n*th natural frequency with damping is

$$\omega_{nD} = \omega_n \sqrt{1 - \xi_n^2}$$

The displacement response of the system is then obtained by substituting Equation (2.60) for $q_n(t)$ in Equation (2.49):

$$u(t) = \sum_{n=1}^{N} \emptyset_n e^{-\xi_n \omega_n t} \left[q_n(0) \cos \omega_{nD} t + \frac{\dot{q}_n(0) + \xi_n \omega_n q_n(0)}{\omega_{nD}} \sin \omega_{nD} t \right] \qquad (2.61)$$

This is the solution of the free vibration problem for a MDF system with classical damping. It provides the displacement u as a function of time due to initial displacement $u(0)$ and velocity $\dot{u}(0)$. Assuming that the natural frequencies ω_n and modes $\emptyset_n$ of the system without damping are available together with the modal damping ratios ξ_n, the right hand side of Equation (2.61) is known with $q_n(0)$ and $\dot{q}_n(0)$ defined by Equations (2.46) and (2.47).

Damping influences the natural frequencies and periods of vibration of a MDF system. Therefore the effect of damping on the natural frequencies and periods of a MDF system is negligible for damping ratios ξ_n below 20%, a range that includes most practical structures.

In a MDF system with classical damping undergoing free vibration in the *n*th natural mode, the displacement amplitude at any DOF decreases with each vibration cycle. The rate of decay depends on the damping ratio ξ_n in that mode, in a manner similar to SDF systems. Thus the ration of two response peaks separated by *j* cycles of vibration is related to the damping ratio with appropriate change in notation.

Consequently, the damping ratio in a natural mode of a MDF system can be determined, in principle, from a free vibration test following the procedure for a SDF system. In such a test the structure would be deformed by pulling on it with a cable that is then suddenly released, thus causing the structure to undergo free vibration about its static equilibrium position. A difficulty in such tests is to apply the pull and release in such a way that the structure will vibrate in only one of its natural modes. For this reason this test procedure is not an effective means to

determine damping except possibly for the fundamental mode. After the response contribution of the higher modes have damped out, the free vibration is essentially in the fundamental mode, and the damping ratio for this mode can be computed from the decay rate of vibration amplitudes.

2.3 Applications of NDT to Special Structures Using Other Methods

A brief description of some applications is given below.

2.3.1 Damage Detection in Concrete by Fourier and Wavelet Analysis

Melhem and Kim (2003) investigated the effectiveness of vibration-based methods in damage detection of a typical highway structure. Two types of full-scale concrete structures subjected to fatigue loads are studied: (1) Portland cement concrete pavements on grade; and (2) a simply supported prestressed concrete beams. Fast Fourier Transform (FFT) and Continuous Wavelet Transform (CWT) are used in the analysis of the structures' dynamic response to impact, and results from both techniques are compared. Both FFT and CWT can identify which frequency components exist in a signal. However, only the wavelet transform can show when a particular frequency occurs. Results of this research are such that FFT can detect the progression of damage in the beam but not in the slab. In contrast, the CWT analysis yielded a clear difference between the initial and damaged states for both structures. These findings confirm the conclusion of previous studies conducted on small-scale specimens that wavelet analysis has a great potential in the damage detection of concrete. The study also demonstrates that the approach is applicable to full-scale component of sizes similar or close to actual in-service structures.

2.3.2 Structural Damage Detection using Signal Pattern-Recognition

Qiao et al. (2008) applied a signal-based pattern-recognition procedure for structural damage detection with a limited number of input/output signals. The method is based on extracting and selecting the sensitive features of the structure response to form a unique pattern for any particular damage scenario, and recognizing the unknown damage pattern against the known database to identify the damage location and level (severity). In this study, two types of transformation algorithms are implemented separately for feature extraction: (1) Continuous Wavelet Transform (CWT); and (2) Wavelet Packet Transform (WPT). Three pattern-matching

algorithms are also implemented separately for pattern recognition: (1) correlation, (2) least square distance, and (3) Cosh spectral distance. To demonstrate the validity and accuracy of the procedure, experimental studies are conducted on a simple three story steel structure. The results show that the features of the signal for different damage scenario can be uniquely identified by these transformations, and correlation algorithms can best perform pattern recognition to identify the unknown damage pattern. The proposed method can also be used to possibly detect the type of damage.

2.3.3 Damage Detection on Bridge

Omenzetter et al. (2004) identified the unusual events in multi-channel bridges by monitoring strain data using wavelet transform and outlier analysis. The strain data was recorded during continuous, long-term operation of a multi-sensor SHM system installed on a full-scale bridge. Outlier detection in multivariate data was to find and localize abnormal, sudden events in the strain data and wavelet transform was used to separate the abrupt strain changes from slowly varying ones. The method was successfully tested using known events recorded during construction of the bridge and later effectively used for detection of anomalous post-construction events.

Omenzetter and Brownjohn (2006) proposed and examined the application of concepts of time series analysis to the processing of data from a continuously operating SHM system installed in a major bridge structure. The recorded static strain data was modeled using ARIMA models. The coefficients of the ARIMA models were identified on-line using an extended Kalman filter. The method was first applied to strains recorded during bridge construction, when structural changes corresponded to known significant events such as cable tensioning. Then the method was used to analyze signals recorded during the post-construction period when the bridge was in service. The results show that the method can provide information on structural performance under normal environmental and operational conditions.

Ding and Li (2007) proposed an online structural health monitoring method for long-term suspension bridge performance using wavelet packet transform (WPT). The method was based on the wavelet packet energy spectrum (WPES) variation of structural ambient vibration responses. As an example application, the WPES-based health monitoring system was used on

the Runyang Suspension Bridge to monitor the responses of the bridge in real-time under various types of environmental conditions and mobile loads. As for the vibration monitoring of the bridge, a total of 27 uni-axial servo type accelerometers were installed at the nine sections of the bridge deck. In each deck section, one lateral accelerometer directly recorded the lateral response, and vertical acceleration of the deck section was obtained by averaging the accelerations measured by the two vertical accelerometers located in the upriver and downriver cross sections, respectively. The analysis showed that actual environmental conditions, including temperature and traffic loadings, were in excellent agreement with the measured WPES. Changes in environmental temperature had a long-term trend influence on the WPES, while the effect of traffic loadings on the measured WPES of the bridge presented instantaneous changes.

Zhang (2007) presented a statistical damage identification procedure for bridge health monitoring. It was assumed that the structure, in both healthy and unknown conditions, was continuously monitored and the dynamic responses under ambient excitations were available. The damage features were extracted based on time series analysis combining auto-regressive and auto-regressive with exogenous input prediction models. The structural condition was evaluated in a stastical way based on the damage possibilities that were derived from a quite large number of data samples to minimize the effect of the variability in data acquisition process and in structural properties on the damage assessment. The validity of the proposed damage identification procedure was demonstrated by numerical studies using a 3-span continuous girder bridge under random ground excitations. Reasonable damage severities for beam structures as well as realistic noise levels were simulated. The results show that the damage identification procedure has great potential to detect structural damage at early stage, in which the structural modal frequency changes are almost imperceptible.

2.3.4 Crack Detection on Beam and Plate

Wang and Deng (1999) detected the crack on beam and plate structures based on wavelet analysis of spatially distributed structural response measurements. Simulated deflection signals of a beam containing a transverse crack and the displacement response of a plate with a through-thickness crack were used. Wavelet transforms were performed on these signals to obtain the

wavelet coefficients along the span of the structures. The crack location was detected by observing a sudden change, such as a spike, in the distribution of the wavelet coefficients. The magnitude of the spike in the wavelet analysis was maximum when the measurement point was next to the damage location.

Douka et al. (2003) identified the location and size of the crack in a cantilever beam based on wavelet analysis. The fundamental vibration mode of a cracked cantilever beam was analyzed using continuous wavelet transform. The location of the crack was determined by the sudden change in the spatial variation of the transformed signal at the site of the crack. To estimate the size of the crack, an intensity factor was defined which related the size of the crack to the corresponding wavelet coefficients. It was shown that the intensity factor changed with crack depth according to a second order polynomial law and therefor, it could be used as an indicator for crack extent. The viability of the proposed method was investigated both analytically and experimentally in a cantilever beam containing a transverse surface crack. Douka et al. (2004) also applied the same method to detect the crack in plates.

Chang and Chen (2005) detected the locations and the sizes of multi-cracks in a beam by spatial wavelet analysis. The crack type was open crack and was represented as a rotational spring. Frist, the mode shapes of free vibration and natural frequencies of the multi-cracked beam were obtained. Then the mode shapes were analyzed by wavelet transformation. The positions of the cracks were observed as a sudden change in the plot of wavelet coefficients. The natural frequencies were used to predict the depth of the cracks through the characteristic equation. The limitation of this method is that there are two peaks near the boundaries in the wavelet plot, and the crack cannot be detected when it is near the boundaries.

Poudel et al. (2007) employed high resolution images for damage detection on a simply supported prismatic steel beam. A high-speed digital video camera was used to recode the free vibration displacement of the beam which was excited by imposing an initial displacement near the mid-span from the left support. The camera had a CMOS (complementary Metal Oxide semiconductor) sensor with 1280×1024 resolution and a 10 bit A/D converter. Its frame rate ranges was from 100 to 2000 frames/s. The displacement data with high spatial resolution were then used to obtain the mode shapes and the mode shape difference function between the reference and damage states of the structure. The spatial signal in terms of mode shape difference function was decomposed by wavelet transformation to display the changes due to

cracking damage. The appropriate range of wavelet scale was determined by the spatial frequency bandwidths of the mode shape difference functions. The maximum modulus and sign change of phase angle in the wavelet coefficients indicated the changes at the damaged locations.

Yan et al. (2004) detected crack damage in a honeycomb sandwish plate by using two structural vibration damage feature indexes: natural frequency and WPT energy index. The finite element dynamic model of a honeycomb sandwish plate was presented using different mesh division for the surface plate and the sandwich plate to accurately express the crack damage status (locations, length and direction) of the plate. In order to acquire the experimental dynamic response of the plate, two piezo-patches with the size of $25 \times 15 \times 0.28$ mm were bonded on the surface of the plate. One of them acted as an actuator and the other acted as a sensor. The natural frequencies of the undamaged plate were experimentally measured to verify the numerical model. Based on the dynamic model verified by the experiment, the damage feather indexes for various crack damage status were numerically computed. Then the crack damage status was determined by comparing the damage feature indexes obtained from the numerical and experimental results. The authors found that using structural natural frequency might not be suitable for detecting crack damage less than 10%, even up to 20%, of the total size of a plate-like structure; however energy spectrum of wavelet transform signals of structural dynamic response had higher sensitivity to crack damage, it could exhibit structural damage status for a crack length close to 2% of the dimension of a plate-like structure. They also found that structural damage information was often contained in some high order modes of a structure, and more vibration modes should be included in a structural dynamic model for detection of a small damage.

Biemans et al. (2001) applied the piezoceramic sensors for crack propagation monitoring in aluminum plate specimens with a crack initiated by spark erosion. The plates were instrumented with piezocerami devices bonded in a symmetrical configuration on both sides of the crack. One of the piezoceramics was used as an actuator and excited by a sine sweep and Gaussian white noise signals in order to exploit broadband excitation. The plates were subjected to static and dynamic tensile loading. The growing crack was monitored by the remaining piezoceramic sensors. The response strain data was analyzed using a number of time, frequency, and wavelet domain statistical parameters. The results show that low frequency broadband excitation offers a possible means of on-line detection of cracks in metallic structures.

2.3.5 Damage Detection on Mechanical Structure

Seibold and Weinert (1996) localized the cracks in rotating machinery based on time domain analysis. A bank of extended Kalman filters (EKFs) was designed to localize a crack based on measured displacements picked up during normal operation of the rotor. Each filter was turned to a different damage hypothesis, i.e., in this case the specific crack location. By calculating the probabilities of the different hypotheses, the crack could be localized and its depth could be determined. The method was applied for the localization of a crack in a simulated rotor on hinged supports and for the identification of the crack depth in a rotor test rig.

Li et al. (1998) applied neural networks to the detection of motor bearing conditions based on the frequency features of bearing vibration. Five basic frequencies related to rolling bearing dynamic movement were extracted by fast Fourier transform (FFT) technique. The basic frequency amplitude vectors were constructed to represent different bearing vibrations. These vectors were created from the power spectrum of the vibration signal and consisted of the five basic frequencies with varying amplitudes based on the defect present. The network consisted of five input measurements corresponding to the amplitudes of the five basic frequencies of interest, ten hidden nodes, and three output fault detectors (bearing looseness, defects on the inner raceway, and defects on the rolling elements). The network was tested using the data generated by MOTORISM. The results show that neural network can be an effective agent in the detection of various motor bearing faults through the measurement and the interpretation of motor bearing vibration signals.

Kar and Mohanty (2006) applied the multi-resolution Fourier transform (MFT) of vibration and current signals for gearbox health monitoring. One and two teeth were artificially removed in one gear of the gearbox to stimulate actual fault condition. When the gearbox was operated under several loads, the vibration signals were acquired from the tail-end bearing of the gearbox, and simultaneously the current drawn by an induction motor is acquired and monitored. MFT combined the characteristics of short-term Fourier transform (STFT) and wavelet transform (WT) by convolving a moving window with the signal at a particular scale. Discrete wavelet transform (DWT) with an orthogonal wavelet of 'db8' was used to scale the vibration and current signals. Then a Hanning window with 256 data points and 50% overlap was applied to the scaled signal to find the MFT coefficients. It was inferred that MFT coefficients of vibration and current signals could predict a consistent trend in the energy level processed by the gear

mesh frequencies with an increase in the severity of the defects, and hence could be a useful tool in gearbox health monitoring.

Staszewski and Tomlinson (1994) applied the wavelet transform to the problem of the detection of a broken tooth in a spur gear. The fault detection algorithm was based on pattern recognition analysis. Features of the pattern were the modulus of the wavelet transform. Spectral analysis and an orthogonal transform were used to compress feature elements. The Statistical concept of similarity analysis was used to compare patterns obtained from the normal (no fault) condition and not normal (fault) condition. It is shown that the Mahalanobis distance between contours plots of the modulus of the wavelet transform can be used as a fault detection symptom; it increases monotonously with the fault advancement. Visual inspection of the modulus and phase of the wavelet transform were used to localize the fault.

Wang and McFadden (1995, 1996) used the wavelet transform to detect abnormal transient generated by gear damage. The early damage to a gear tooth usually caused a variation in the associated vibration signal over a short time, initially less than one tooth meshing period, taking the form of modulated or unmodulated oscillation. In later stages, the duration of the abnormal variation became longer, lasting more than one tooth meshing period. Other distributed faults, such as eccentricity and wear, might cover the most part of the whole revolution of the gear. Changes in the vibration signals therefore could be analyzed to provide an indicator of gear condition. When the size and shape of a wavelet were exactly the same as a section of a signal, the transform gave a maximum absolute value of wavelet coefficients. Therefore, the abnormal signal caused by a gear fault could be displayed by the wavelet transform, which could be regarded as a procedure for comparing the similarity of the signal and the chosen wavelet.

Liao et al. (2004) developed a novel technique for monitoring the gearbox condition based on the self-organizing feature maps (SOFM) network. Seven time-domain features parameters, namely standard deviation, Kurtosis, root mean square value, absolute mean value, crest factor, clearance factor, and impulse factor were extracted from industrial gearbox vibration signals measured under different operating conditions. Trained with the SOFM network and visualized using the U-matrix method, the feature data were mapped into a two-dimensional space and formed clustering regions, each indicative of a specific gearbox work condition. Therefore the gearbox operating condition with fatigue crack or a broken tooth compared with

24

the normal condition was clearly identified. Furthermore, with the trajectory of the image points for the feature data in two-dimensional space, the vibration of gearbox condition was observed visually and the development of early-stage gearbox failure was monitored in time.

Ozturk and Yesilyurt (2008) presented the use of a scalogram and its mean frequency variation for monitoring the progression of pitting failure in gear system. Real gear vibrations were obtained from a test rig utilizing a two-stage industrial gearbox. Pits representing different degrees of fault severity were simulated on a few tooth surfaces. The continuous wavelet transform was used to obtain a scalogram and its mean frequency variation. It was found that the presence of the seeded pits was not clearly revealed by the scalogram in the early phases of fault progression. When the severity (or number) of pits was further increased, the scalogram exhibited fault symptoms as an increase in energy density. In contrast, the mean frequency variation showed the presence of the fault even when there was only a single pit. The resulting fault displayed itself as a localized deviation which repeated itself every pinion rotation. Increasing the number of pits caused correspondingly stronger fault symptoms and sharper fault localization at the same gear positions.

Chapter 3 - THEORETICAL BACKGROUND

3.1 Historical Overview

Vibration analysis has been used for many years, and in many ways. In ancient times sounding of clay artifacts revealed cracks and voids (Heyns 1996). Shaking coconuts to test ripeness is an example of vibration analysis without rigorous concern for instrumentation or sensitivity.

Sir Isaac Newton, largely credited for explaining the basis of modern dynamics as we know it today. His formulation of laws of motion were first stated in 1687 in his *Philosophiae Naturalis Principa Mathematica* (Fertis 1995). Newton reached these conclusions by expanding on work established by Galileo. Galileo had made careful observation of bodies in free fall, motion on an inclined plane, and motion of the pendulum, but this work was hindered by the low sensitivity of measuring devices (Meriam and Kraige 1997). In Galileo's era, little precision was developed for measuring time. Some original observation of the pendulum were tracked using his own pulse as a timepiece (Ramirez 1985).

Currently, measuring time is conducted with much more precision. One second is defined as 9,192,631,770 transition between two specified, hyperfine levels of a cesium 133 atom (Ramirez 1985). Science demands such precision when researching or observing motion. Most applications of vibration analysis require precise measurement of time, and limited influence from data acquisitioning hardware and software. This precision is needed for proper correlation between data acquired and degradation to be established.

The fundamental behind using vibration analysis is that wearing – due to fatigue loading, corrosion, crack growth, etc… - of a specimen will alter the specimen's ability to impede motion due to vibration signals. The state of the specimen – including supports, flaws, loading, and internal stresses – at the time of applying a vibration wave – active excitement of specimen – establishes a signature for the specimen. The signature reflects the natural frequency of a system at time of analysis. The signature should not change unless the specimen's parameter change.

Vibration could be obtained in a form of Time vs. Displacement, Velocity or Acceleration. Nowadays, one of the most common ways to record the system vibration is by using accelerometer that sending the recorded signals to a software over the base station in a form of Time vs. Acceleration which is identified as Time Domain. Acceleration is preferred

and used in spite of Displacement and Velocity because it is usually hard to determine a specific datum for the displacement to be measured from and the acceleration is the second derivative from the displacement and the first derivative from the velocity which gives a better resolution when recording the vibration.

3.2 Time Domain

Time domain is the analysis of mathematical functions, physical signals or time series of dynamic, economic or environmental data, with respect to time. In the time domain, the signal or function's value is known for all real numbers, for the case of continuous time, or at various separate instants in the case of discrete time. An oscilloscope is a tool commonly used to visualize real-world signals in the time domain. A time-domain side shows how a signal changes with time (Figure 3.1), whereas a frequency-domain side shows how much of the signal lies within each given frequency band over a range of frequencies (Figure 3.2).
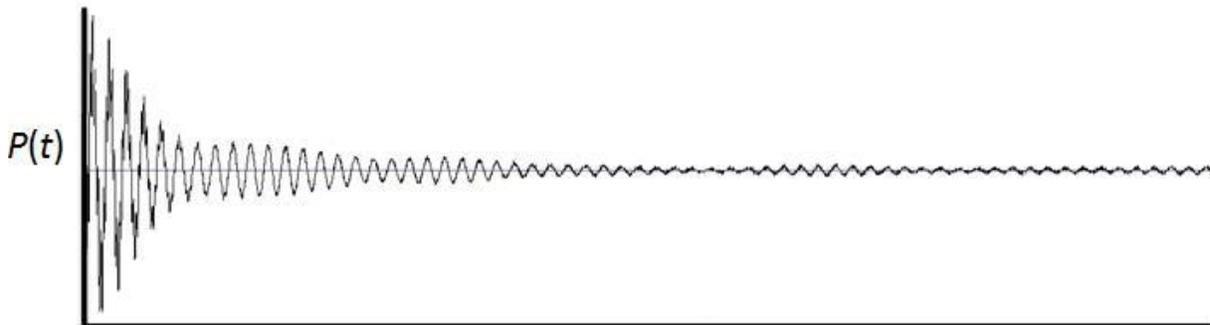
Figure 3.1 Time Domain

The use of the contrasting terms time domain and frequency domain developed in US communication engineering in the late 1940s, with the terms appearing together without definition by 1949 (Lee et al 1949) when an analysis uses the second or one of its multiples as a unit of measurement, then it is in the time domain. When analysis concerns the reciprocal units such as Hertz, then it is in the frequency domain.

3.3 Frequency Domain

In engineering and statistics, the frequency domain refers to the analysis of mathematical functions or signals with respect to frequency, rather than time (Broughton and Bryan 2008).

Simply, a time-domain graph shows how a signal changes over time, whereas a frequency-domain graph shows how much of the signal lies within each given frequency band over a range of frequencies. A frequency-domain representation can also include information on the phase shift that must be applied to each sinusoid in order to be able to recombine the frequency components to recover the original time signal.

A given function or signal can be converted between the time and frequency domains with a pair of mathematical operators called a transform. An example is the Fourier transform, which converts the time function into a frequency domain function. The 'spectrum' of frequency components is the frequency domain representation of the signal. The inverse Fourier transform converts the frequency domain function back to a time function. A spectrum analyzer is the tool commonly used to visualize real-world signals in the frequency domain.

Signal processing also allows representations or transforms that result in a joint time-frequency domain, with the instantaneous frequency being a key link between the time domain and the frequency domain.

3.3.1 Magnitude and Phase

In using the Laplace, Z-, or Fourier transforms, the frequency spectrum is complex, describing the magnitude and phase of a signal, or of the response of a system, as a function of frequency. In many applications, phase information is not important. By discarding the phase information it is possible to simplify the information in a frequency domain representation to generate a frequency spectrum or spectral density. A spectrum analyzer is a device that displays the spectrum, while the time domain frequency can be seen on an oscilloscope.

The power spectral density (PSD) is a frequency-domain description that can be applied to a large class of signals that are neither periodic nor square-integral, to have a power spectral density, a signal needs only to be the output of a wide-sense stationary random process.

3.3.2 Different Frequency Domains

Although "the" frequency domain is spoken of in the singular, there are a number of different mathematical transforms which are used to analyze time functions and are referred to as

"frequency domain" methods. These are the most common transforms, and the fields in which they are used:

- Fourier series – repetitive signals, oscillating systems
- Fourier transform – non-repetitive signals, transients
- Laplace transform – electronic circuits and control systems
- Z-transform – discrete signals, digital signal processing

More generally, one can speak of the transform domain with respect to any transform. The above transforms can be interpreted as capturing some form of frequency, and hence the transform domain is referred to as a frequency domain.

3.3.3 Discrete frequency domain

The Fourier transform of a periodic signal only has energy at a base frequency and its harmonics. Another way of saying this is that a periodic signal can be analyzed using a discrete frequency domain. Dually, a discrete-time signal gives rise to a periodic frequency spectrum. Combining these two, if we start with a time signal which is both discrete and periodic, we get a frequency spectrum which is both periodic and discrete. This is the usual context for a discrete Fourier transform.
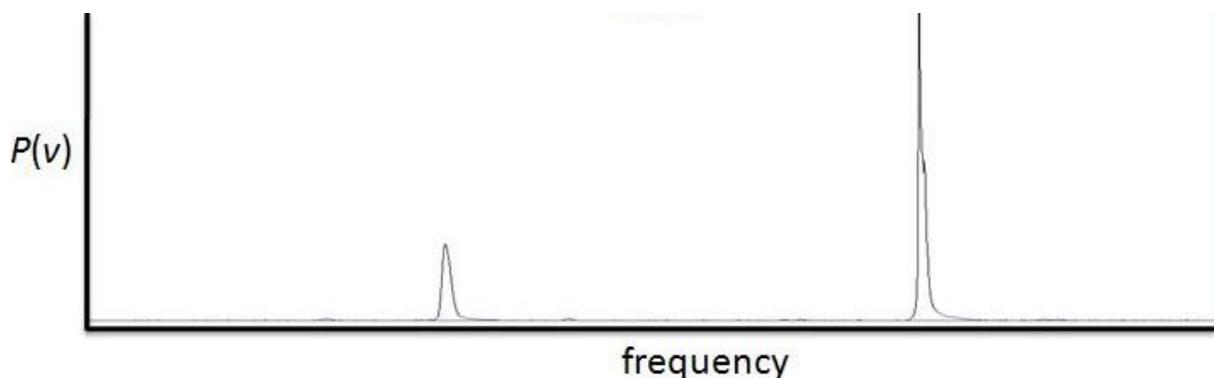
Figure 3.2 Frequency Domain

For this research, to obtain the system frequency – the signature – which is identified as Frequency Domain from Time Domain, Fourier Transform is the used transform that converts the Time Domain to Frequency Domain and vice versa.
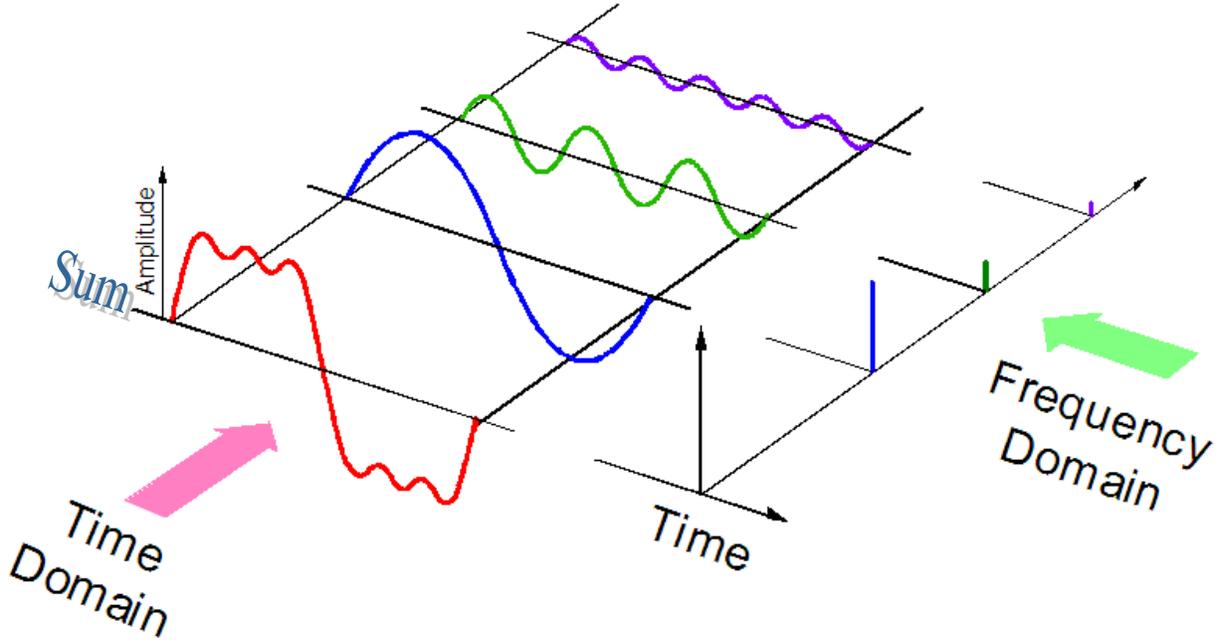
Figure 3.3 Descriptive graph of difference between Time Domain and Frequency Domain for a signal

3.4 Fourier Transforms

The Fourier transform (named after its discoverer, the French mathematician Jean-Baptiste Joseph Fourier) is a frequently-based transform widely used in analysis of linear systems. It decomposes a signal into sine waves of different frequencies which sum to the original waveform, and also distinguishes such different frequency sine waves and their respective amplitudes.

3.4.1 Fourier Series

A Fourier series is an expansion of a periodic function *f(x)* in terms of an infinite sum of sines and cosines. Fourier series make use of the orthogonality relationships of the sine and cosine functions. The computation and study of Fourier series is known as harmonic analysis and is extremely useful as a way to break up an *arbitrary* periodic function into a set of simple terms that can be plugged in, solved individually, and then recombined to obtain the solution to the original problem or an approximation to it to whatever accuracy is

desired or practical. Examples of successive approximations to common functions using Fourier series are illustrated in Figure 3.4.
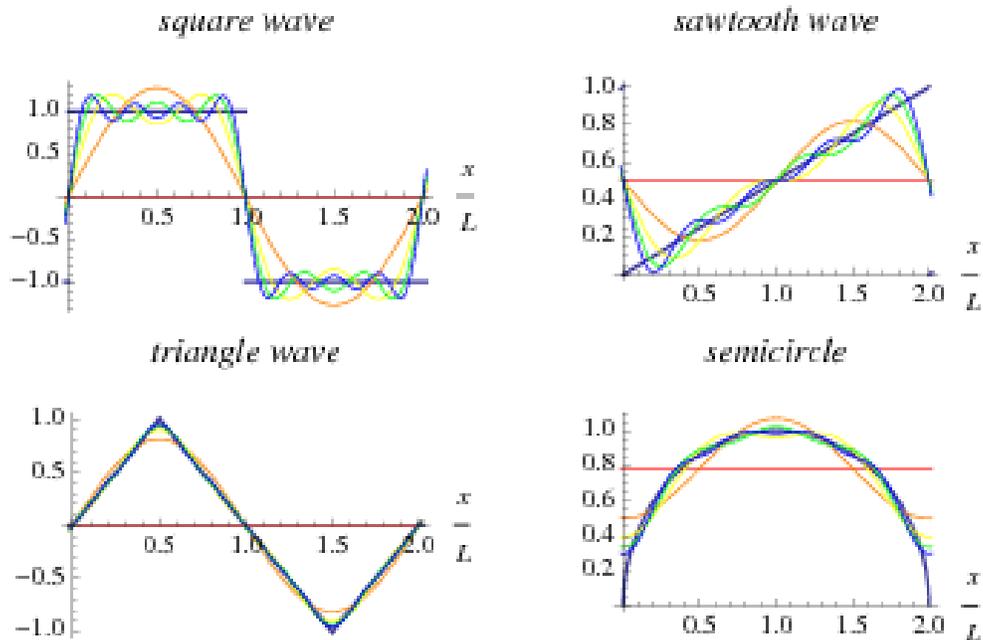
Figure 3.4 Successive approximations to common functions using Fourier series

In particular, since the superposition principle holds for solutions of a linear homogeneous ordinary differential equation, if such an equation can be solved in the case of a single sinusoid, the solution for an arbitrary function is immediately available by expressing the original function as a Fourier series and then plugging in the solution for each sinusoidal component. In some special cases where the Fourier series can be summed in closed form, this technique can even yield analytic solutions. Any set of functions that form a complete orthogonal system have a corresponding generalized Fourier series analogous to the Fourier series. For example, using orthogonality of the roots of a Bessel function of the first kind gives a so-called Fourier-Bessel series.

The computation of the (usual) Fourier series is based on the integral identities

$$\int_{-\pi}^{\pi} \sin(m\,x)\sin(n\,x)\,dx = \pi\delta_{mn} \tag{3.1}$$

31

$$\int_{-\pi}^{\pi} \cos(m\,x)\cos(n\,x)\,dx = \pi\delta_{mn} \qquad (3.2)$$

$$\int_{-\pi}^{\pi} \sin(m\,x)\cos(n\,x)\,dx = 0 \qquad (3.3)$$

$$\int_{-\pi}^{\pi} \sin(m\,x)\,dx = 0 \qquad (3.4)$$

$$\int_{-\pi}^{\pi} \cos(m\,x)\,dx = 0 \qquad (3.5)$$

for $m,\ n \neq 0$, where δ_{mn} is the Kronecker delta.

Using the method for a generalized Fourier series, the usual Fourier series involving sines and cosines is obtained by taking $f_1(x){=}cos\ x$ and $f_2\ (x){=}sin\ x$. Since these functions form a complete orthogonal system over $[-\pi,\pi]$, the Fourier series of a function $f(x)$ is given by

$$f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos(n\,x) + \sum_{n=1}^{\infty} b_n \sin(n\,x), \qquad (3.6)$$

Where,

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\,dx \qquad (3.7)$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\cos(n\,x)\,dx \qquad (3.8)$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\sin(n\,x)\,dx \qquad (3.9)$$

And $n = 1, 2, 3, \ldots$. Note that the coefficient of the constant term a_0 has been written in a special form compared to the general form for generalized in order to preserve symmetry with the definitions of a_n and b_n.

A Fourier series converges to the function $\bar{f}$ (equal to the original function at points of continuity or to the average of the two limits at points of discontinuity)

$$f = \begin{cases} \frac{1}{2}\left[\lim\limits_{x \to x_0^-} f(x) + \lim\limits_{x \to x_0^+} f(x)\right] & \text{for } -\pi < x_0 < \pi \\ \frac{1}{2}\left[\lim\limits_{x \to \pi^+} f(x) + \lim\limits_{x \to \pi^-} f(x)\right] & \text{for } x_0 = -\pi, \pi \end{cases} \tag{3.10}$$

if the function satisfies so-called Dirichlet boundary conditions. Dini's test gives a condition for the convergence of Fourier series.

As a result, near points of discontinuity, or "ringing" known as the Gibbs phenomenon, can occur, as shown in Figure 3.5.
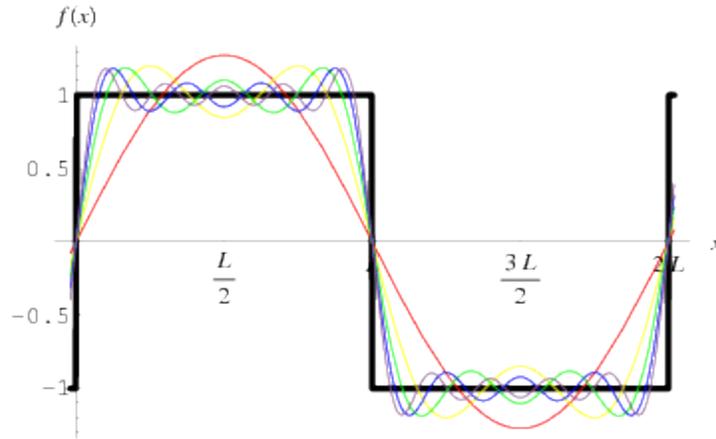
Figure 3.5 Gibbs phenomenon

For a function $f(x)$ periodic on an interval $[-L,L]$ instead of $[-\pi,\pi]$, a simple change of variables can be used to transform the interval of integration from $[-\pi,\pi]$ to $[-L,L]$, let

$$x = \frac{\pi x'}{L} \tag{3.11}$$

$$dx = \frac{\pi dx'}{L} \tag{3.12}$$

Solving for x' gives $x'=Lx/\pi$, and plugging this in gives

$$f(x') = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos\left(\frac{n\pi x'}{L}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi x'}{L}\right), \tag{3.13}$$

Therefore,

$$a_0 = \frac{1}{L}\int_{-L}^{L} f(x')\, dx' \tag{3.14}$$

$$a_n = \frac{1}{L} \int_{-L}^{L} f(x') \cos\left(\frac{n\pi x'}{L}\right) dx' \qquad (3.15)$$

$$b_n = \frac{1}{L} \int_{-L}^{L} f(x') \sin\left(\frac{n\pi x'}{L}\right) dx' \qquad (3.16)$$

Similarly, the function is instead defined on the interval [0, 2L], the above equations simply become

$$a_0 = \frac{1}{L} \int_{0}^{2L} f(x') \, dx' \qquad (3.17)$$

$$a_n = \frac{1}{L} \int_{0}^{2L} f(x') \cos\left(\frac{n\pi x'}{L}\right) dx' \qquad (3.18)$$

$$b_n = \frac{1}{L} \int_{0}^{2L} f(x') \sin\left(\frac{n\pi x'}{L}\right) dx' \qquad (3.19)$$

In fact, for $f(x)$ periodic with period $2L$, *any* interval $(x_0,\ x_0 + 2L)$ can be used, with the choice being one of convenience or personal preference (Arfken 1985, p. 769).

The coefficients for Fourier series expansions of a few common functions are given in Beyer (1987, pp. 411-412) and Byerly (1959, p. 51). One of the most common functions usually analyzed by this technique is the square wave. The Fourier series for common functions are summarized in the table below.

Table 3.1 Fourier series for common functions

Function	$f(x)$	Fourier series
Fourier series--sawtooth wave	$x/2l$	$\frac{1}{2} - \frac{1}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \sin\left(\frac{n\pi x}{L}\right)$
Fourier series--square wave	$2\left[H\left(\frac{x}{L}\right) - H\left(\frac{x}{L} - 1\right)\right] - 1$	$\frac{4}{\pi} \sum_{n=1,3,5,\dots}^{\infty} \frac{1}{n} \sin\left(\frac{n\pi x}{L}\right)$
Fourier series--triangle wave	$T(x)$	$\frac{8}{\pi^2} \sum_{n=1,3,5,\dots}^{\infty} \frac{(-1)^{(n-1)/2}}{n^2} \sin\left(\frac{n\pi x}{L}\right)$

If a function is even so that $f(x)=f(-x)$, then $f(x)sin(nx)$ is odd. (This follows since sin (nx) is odd and an even function times an odd function is an odd function.) Therefore, $b_n=0$ for all n. Similarly, if a function is odd so that $f(x)=-f(-x)$, then $f(x)cos(nx)$ is odd. (This follows since cos (nx) is even and an even function times an odd function is an odd function.) Therefore, $a_{n=0}$ for all n.

The notion of a Fourier series can also be extended to complex coefficients. Consider a real-valued function $f(x)$. Write

$$f(x) = \sum_{n=-\infty}^{\infty} A_n \, e^{inx} \tag{3.20}$$

Now examine

$$\int_{-\pi}^{\pi} f(x) \, e^{-imx} \, dx = \int_{-\pi}^{\pi} \left(\sum_{n=-\infty}^{\infty} A_n \, e^{inx} \right) e^{imx} \, dx \tag{3.21}$$

$$= \sum_{n=-\infty}^{\infty} A_n \int_{-\pi}^{\pi} e^{i\,(n-m)x} \, dx \tag{3.22}$$

$$= \sum_{n=-\infty}^{\infty} A_n \int_{-\pi}^{\pi} \{\cos\,[(n-m)x\,] + i \sin[(n-m)x]\} \, dx \tag{3.23}$$

$$= \sum_{n=-\infty}^{\infty} A_n \, 2\pi\delta_{mn} \tag{3.24}$$

$$= 2\pi A_m, \tag{3.25}$$

So

$$A_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)e^{-inx} \, dx \tag{3.26}$$

The coefficients can be expressed in terms of those in the Fourier series

$$A_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)[\cos(n\,x) - i \sin(n\,x)] \, dx \tag{3.27}$$

$$\left.
\begin{array}{ll}
\dfrac{1}{2\pi} \displaystyle\int_{-\pi}^{\pi} f(x)\,[\cos(n\,x) + i\,\sin(|n|\,x)]\,dx & n < 0 \\[2em]
\dfrac{1}{2\pi} \displaystyle\int_{-\pi}^{\pi} f(x)\,dx & n = 0 \\[2em]
\dfrac{1}{2\pi} \displaystyle\int_{-\pi}^{\pi} f(x)\,[\cos(n\,x) - i\,\sin(|n|\,x)]\,dx & n > 0
\end{array}
\right\} \tag{3.28}$$

$$=
\left\{
\begin{array}{ll}
\dfrac{1}{2}\,(a_n + i\,b_n) & for\ n < 0 \\[1em]
\dfrac{1}{2}\,a_0 & for\ n = 0 \\[1em]
\dfrac{1}{2}\,(a_n - i\,b_n) & for\ n > 0
\end{array}
\right. \tag{3.29}$$

For a function periodic in [-L/2, L/2], these become

$$f(x) = \sum_{n=-\infty}^{\infty} A_n\, e^{i\,(2\pi n x/L)} \tag{3.30}$$

$$A_n = \frac{1}{L} \int_{-L/2}^{L/2} f(x)\, e^{-i(2\pi n x/L)}\, dx \tag{3.31}$$

These equations are the basis for the extremely important Fourier transform, which is obtained by transforming A_n from a discrete variable to a continuous one as the length $L \longrightarrow \infty$

3.4.2 Continuous Fourier Transform

The Fourier transform is a generalization of the complex Fourier series in the limit as $L \longrightarrow \infty$. Replace the discrete A_n with the continuous $F(k)\ dk$ while letting $n/L \longrightarrow k$. Then change the sum to an integral, and the equations become

$$f(x) = \int_{-\infty}^{\infty} F(k)\, e^{2\pi i k x}\, dk \tag{3.32}$$

$$F(k) = \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i k x}\, dx \tag{3.33}$$

Here,

$$F(k) = \mathbb{F}_x[f(x)](k) \tag{3.34}$$

$$= \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} \, dx \tag{3.35}$$

is called the *forward* (-*i*) Fourier transform, and

$$F(x) = \mathbb{F}_x^{-1}[f(x)](k) \tag{3.36}$$

$$= \int_{-\infty}^{\infty} F(k) e^{2\pi i k x} \, dk \tag{3.37}$$

is called the *inverse* (+*i*) Fourier transform. The notation $\mathbb{F}_x[f(x)](k)$ is introduced in Trott (2004, p. xxxiv), and $\hat{f}(k)$ and $\check{f}(x)$ are sometimes also used to denote the Fourier transform and inverse Fourier transform, respectively (Krantz 1999, p. 202).

Note that some authors (especially physicists) prefer to write the transform in terms of angular frequency $\omega = 2\pi\nu$ instead of the oscillation frequency ν. However, this destroys the symmetry, resulting in the transform pair

$$H(\omega) = \mathbb{F}[h(t)] \tag{3.38}$$

$$= \int_{-\infty}^{\infty} h(t) e^{-i\omega t} \, dt \tag{3.39}$$

$$h(t) = \mathbb{F}^{-1}[H(\omega)] \tag{3.40}$$

$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} H(\omega) e^{i\omega t} \, d\omega \tag{3.41}$$

To restore the symmetry of the transforms, the convention

$$g(y) = \mathbb{F}[f(t)] \tag{3.42}$$

$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-iyt} \, dt \tag{3.43}$$

$$f(t) = \mathbb{F}^{-1}[g(y)] \tag{3.44}$$

$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(y) e^{iyt} \, dy \tag{3.45}$$

is sometimes used (Mathews and Walker 1970, p. 102).

In general, the Fourier transform pair may be defined using two arbitrary constants a and b as

$$F(\omega) = \sqrt{\frac{|b|}{(2\pi)^{1-a}}} \int_{-\infty}^{\infty} f(t)e^{ib\omega t}\, dt \tag{3.46}$$

$$f(t) = \sqrt{\frac{|b|}{(2\pi)^{1+a}}} \int_{-\infty}^{\infty} F(\omega)e^{-ib\omega t}\, d\omega \tag{3.47}$$

Unfortunately, a number of other conventions are in widespread use. For example,$(0,1)$ is used in modern physics, $(1,-1)$ is used in pure mathematics and systems engineering, $(1,1)$ is used in probability theory for the computation of the characteristic function, $(-1,1)$ is used in classical physics, and $(0, -2\pi)$ is used in signal processing. In this work, following Bracewell (1999, pp. 6-7), *it is always assumed that a=0 and b=-2π unless otherwise stated.* This choice often results in greatly simplified transforms of common functions such as 1, cos $(2\pi k_0 x)$, etc.

Since any function can be split up into even and odd portions *E(x) and O(x)*,

$$f(x) = \frac{1}{2}[f(x) + f(-x)] + \frac{1}{2}[f(x) - f(-x)] \tag{3.48}$$

$$= E(x) + O(x), \tag{3.49}$$

Fourier transform can always be expressed in terms of the Fourier cosine transform and Fourier sine transform as

$$\mathbb{F}_x[f(x)](k) = \int_{-\infty}^{\infty} E(x)\cos(2\pi kx)dx - i \int_{-\infty}^{\infty} O(x)\sin(2\pi kx)\, dx \tag{3.50}$$

A function *f (x)* has a forward and inverse Fourier transform such that

$$f(x) = \begin{cases} \int_{-\infty}^{\infty} e^{2\pi ikx}\left[\int_{-\infty}^{\infty} f(x)e^{-2\pi ikx}\, dx\right] dk & \text{for } f(x)\text{continuous at } x \\ \frac{1}{2}[f(x_+) + f(x_-)] & \text{for } f(x)\text{discontinuous at } x, \end{cases} \tag{3.51}$$

provided that

1. $\int_{-\infty}^{\infty}|f(x)|\, dx\ exists.$

2. There are a finite number of discontinuities.

3. The function has bounded variation. A sufficient weaker condition is fulfillment of the Lipschitz condition

(Ramirez 1985, p. 29). The smoother a function (i.e., the larger the number of continuous derivatives), the more compact its Fourier transform.

The Fourier transform is linear, since if $f(x)$ and $g(x)$ have Fourier transforms $F(k)$ and $G(k)$, then

$$\int [a\,f(x) + b\,g(x)]e^{-2\pi ikx}\,dx = a\int_{-\infty}^{\infty} f(x)e^{-2\pi ikx}\,dx + b\int_{-\infty}^{\infty} g(x)e^{-2\pi ikx}\,dx \qquad (3.52)$$

$$= a\,F(k) + b\,G(k). \qquad (3.53)$$

Therefore,

$$\mathbb{F}\,[a\,f(x) + b\,g(x)] = a\,\mathbb{F}\,[\,f(x) + b\,\mathbb{F}\,[\,g(x)\,] \qquad (3.54)$$

$$= a\,F(k) + b\,G(k). \qquad (3.55)$$

The Fourier transform is also symmetric since $F(k) = \mathbb{F}_x\,[f(x)]\,(k)$ implies $F(-k) = \mathbb{F}_x\,[f(-x)]\,(k)$

Let $f * g$ denote the convolution, then the transforms of convolutions of functions have particularly nice transforms,

$$\mathbb{F}\,[f * g] \qquad = \mathbb{F}\,[f]\,\mathbb{F}\,[\,g\,] \qquad (5.56)$$

$$\mathbb{F}\,[fg] \qquad = \mathbb{F}\,[f\,] * \mathbb{F}\,[\,g\,] \qquad (3.57)$$

$$\mathbb{F}^{-1}\,[\mathbb{F}\,(f)\,\mathbb{F}\,(g)] \qquad = f * g \qquad (3.58)$$

$$\mathbb{F}^{-1}\,[\mathbb{F}\,(f) * \mathbb{F}\,(g)] \qquad = f\ g \qquad (3.59)$$

The first of these is derived as follows:

$$\mathbb{F}\,[f * g] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-2\pi ikx}\,f(x')g(x-x')dx'dx \qquad (3.60)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[e^{-2\pi ikx'}\,f(x')dx'\right]\left[e^{-2\pi ik(x-x')}\,g(x-x')dx\,\right] \qquad (3.61)$$

$$= \left[\int_{-\infty}^{\infty} e^{-2\pi ikx'}\,f(x')dx'\right]\left[\int_{-\infty}^{\infty} e^{-2\pi ikx''}\,g(x'')dx''\right] \qquad (3.62)$$

$$= \mathbb{F}\,[f\,]\mathbb{F}\,[\,g\,], \qquad (3.63)$$

Where $x'' = x - x'$.

There is also a somewhat surprising and extremely important relationship between the autocorrelation and the Fourier transform known as the Wiener-Khinchin theorem. Let

$\mathbb{F}_x [f(x)] (k) = F(k)$, and $\bar{f}$ denote the complex conjugate of f, then the Fourier transform of the absolute square of $F(k)$ is given by

$$\mathbb{F}_k [|F(k)|^2](x) = \int_{-\infty}^{\infty} \bar{f}(\tau)f(\tau + x)dt \qquad (3.64)$$

The Fourier transform of a derivative $f'(x)$ of a function $f(x)$ is simply related to the transform of the function $f(x)$ itself. Consider

$$\mathbb{F}_x [f'(x)](k) = \int_{-\infty}^{\infty} f'(x)e^{-2\pi ikx}\, dx. \qquad (3.65)$$

Now use integration by parts

$$\int v\, d\,u = [\,u\,v\,] - \int u\, d\,v \qquad (3.66)$$

With

$$d\,u = f'(x)\, dx \qquad (3.67)$$
$$v \quad = e^{-2\pi ikx} \qquad (3.68)$$

and

$$u \quad = f(x) \qquad (3.69)$$
$$d\,v = -2\pi ike^{-2\pi ikx}\, dx, \qquad (3.70)$$

then

$$\mathbb{F}_x [f'(x)](k) = [f(x)e^{-2\pi ikx}]_{-\infty}^{\infty} - \int_{-\infty}^{\infty} f(x)(-2\pi ike^{-2\pi ikx}\, dx) \qquad (3.71)$$

The first term consists of an oscillating function times $f(x)$. But if the function is bounded so that

$$\lim_{x \to \pm\infty} f(x) = 0 \qquad (3.72)$$

(as any physically significant signal must be), then the term vanishes, leaving

$$\mathbb{F}_x [f'(x)](k) = 2\pi ik \int_{-\infty}^{\infty} f(x)e^{-2\pi ikx}dx \qquad (3.73)$$

$$= 2\pi ik\mathbb{F}_x [f(x)](k). \qquad (3.74)$$

This process can be iterated for the n th derivative to yield

$$\mathbb{F}_x [f^{(n)}(x)](k) = (2\pi ik)^n\, \mathbb{F}_x [f(x)](k). \qquad (3.75)$$

The important modulation theorem of Fourier transforms allows $\mathbb{F}_x [\cos(2\pi k_o x) f(x)](k)$ to be expressed in terms of $\mathbb{F}_x[f(x)](k) = F(k)$ as follows,

$$\mathbb{F}_x[\cos(2\pi k_0 x)f(x)](k) = \int_{-\infty}^{\infty} f(x)\cos(2\pi k_0 x)e^{-2\pi ikx}\,dx \tag{3.76}$$

$$= \frac{1}{2}\int_{-\infty}^{\infty} f(x)e^{2\pi ik_0 x}e^{-2\pi ikx}\,dx + \frac{1}{2}\int_{-\infty}^{\infty} f(x)e^{-2\pi ik_0 x}e^{-2\pi ikx}\,dx \tag{3.77}$$

$$= \frac{1}{2}\int_{-\infty}^{\infty} f(x)e^{-2\pi i(k-k_0)x}\,dx + \frac{1}{2}\int_{-\infty}^{\infty} f(x)e^{-2\pi i(k+k_0)x}\,dx \tag{3.78}$$

$$= \frac{1}{2}[F(k-k_0) + F(k+k_0)]. \tag{3.79}$$

Since the derivative of the Fourier transform is given by

$$F'(k) = \frac{d}{dk}\mathbb{F}_x[f(x)](k) = \int_{-\infty}^{\infty} (-2\pi ix)f(x)e^{-2\pi ikx}\,dx, \tag{3.80}$$

it follows that

$$F'(0) = -2\pi i \int_{-\infty}^{\infty} x f(x)dx. \tag{3.81}$$

Iterating gives the general formula

$$\mu_n = \int_{-\infty}^{\infty} x^n f(x)dx \tag{3.82}$$

$$= \frac{F^{(n)}(0)}{(-2\pi i)^n} \tag{3.83}$$

The variance of a Fourier transform is

$$\sigma_f^2 = \langle (xf - \langle xf\rangle)^2\rangle, \tag{3.84}$$

and it is true that

$$\sigma_{f+g} = \sigma_f + \sigma_g. \tag{3.85}$$

If $f(x)$ has the Fourier transform $\mathbb{F}_x[f(x)](k) = F(k)$, then the Fourier transform has the shift property

$$\int_{-\infty}^{\infty} f(x-x_0)e^{-2\pi ikx}\,dx = \int_{-\infty}^{\infty} f(x-x_0)e^{-2\pi i(x-x_0)k}e^{-2\pi i(kx_0)}\,d(x-x_0) \tag{3.86}$$

41

$$= e^{-2\pi i k x_0} F(k), \tag{3.87}$$

So $f(x - x_0)$ has the Fourier transform

$$\mathbb{F}_x[f(x - x_0)](k) = e^{-2\pi i k x_0} F(k). \tag{3.88}$$

If $f(x)$ has a Fourier transform $\mathbb{F}_x[f(x)](k) = F(k)$, then the Fourier transform obeys a similarity theorem.

$$\int_{-\infty}^{\infty} f(ax)e^{-2\pi i k x}\, dx = \frac{1}{|a|} \int_{-\infty}^{\infty} f(ax)e^{-2\pi i (ax)(\frac{k}{a})}\, d(ax) = \frac{1}{|a|} F\left(\frac{k}{a}\right), \tag{3.89}$$

So $f(ax)$ has the Fourier transform

$$\mathbb{F}_x[f(ax)](k) = |a|^{-1} F\left(\frac{k}{a}\right). \tag{3.90}$$

The "equivalent width" of a Fourier transform is

$$w_e = \frac{\int_{-\infty}^{\infty} f(x)dx}{f(0)} \tag{3.91}$$

$$= \frac{F(0)}{\int_{-\infty}^{\infty} F(k)dk} \tag{3.92}$$

The "autocorrelation width" is

$$w_a = \frac{\int_{-\infty}^{\infty} f \circledast \bar{f}\, dx}{[f \circledast \bar{f}]_0} \tag{3.93}$$

$$= \frac{\int_{-\infty}^{\infty} f\, dx \int_{-\infty}^{\infty} \bar{f}\, dx}{\int_{-\infty}^{\infty} f \bar{f}\, dx}, \tag{3.94}$$

Where $f \circledast g$ denotes the cross-correlation of f and g and $\bar{f}$ is the complex conjugate.

Any operation on $f(x)$ which leaves its area unchanged leaves $F(0)$ unchanged, since

$$\int_{-\infty}^{\infty} f(x)\, dx = \mathbb{F}_x[f(x)](0) = F(0). \tag{3.95}$$

The following table summarized common Fourier transform pairs.

Table 3.2 Common Fourier transform pairs

Function	$f(x)$	$F(k) = \mathcal{F}_x[f(x)](k)$
Fourier transform--1	1	$\delta(k)$
Fourier transform--cosine	$\cos(2\pi k_0 x)$	$\frac{1}{2}[\delta(k-k_0)+\delta(k+k_0)]$
Fourier transform--delta function	$\delta(x-x_0)$	$e^{-2\pi i k x_0}$
Fourier transform--exponential function	$e^{-2\pi k_0 |x|}$	$\frac{1}{\pi}\frac{k_0}{k^2+k_0^2}$
Fourier transform--Gaussian	e^{-ax^2}	$\sqrt{\frac{\pi}{a}}\, e^{-\pi^2 k^2/a}$
Fourier transform--Heaviside step function	$H(x)$	$\frac{1}{2}\left[\delta(k)-\frac{i}{\pi k}\right]$
Fourier transform--inverse function	$-PV\,\frac{1}{\pi x}$	$i\,[1-2H(-k)]$
Fourier transform--Lorentzian function	$\frac{1}{\pi}\frac{\frac{1}{2}\Gamma}{(x-x_0)^2+\left(\frac{1}{2}\Gamma\right)^2}$	$e^{-2\pi i k x_0 -\Gamma\pi|k|}$
Fourier transform--ramp function	$R(x)$	$\pi i\,\delta'(2\pi k)-\frac{1}{4\pi^2 k^2}$
Fourier transform--sine	$\sin(2\pi k_0 x)$	$\frac{1}{2}i\,[\delta(k+k_0)-\delta(k-k_0)]$

In two dimensions, the Fourier transform becomes

$$F(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(k_x,\ k_y)e^{-2\pi i(k_x x+k_y y)}d\,k_x\, d\,k_y \tag{3.96}$$

$$F(k_x,k_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,\ y)e^{2\pi i(k_x x+k_y y)}d\,x\, d\,y \tag{3.97}$$

Similarly, the n-dimensional Fourier transform can be defined for $\mathbf{k}, \mathbf{x} \in R^n$ by

$$F(\mathbf{x}) = \int_{-\infty}^{\infty}\ldots(n)\ldots\int_{-\infty}^{\infty} f(\mathbf{k})\,e^{-2\pi i\,\mathbf{k}.\mathbf{x}}\,d^n\,\mathbf{k} \tag{3.98}$$

$$f(\mathbf{k}) = \int_{-\infty}^{\infty}\ldots(n)\ldots\int_{-\infty}^{\infty} F(\mathbf{x})\,e^{2\pi i\,\mathbf{k}.\mathbf{x}}\,d^n\,\mathbf{x}. \tag{3.99}$$

As a brief, let $f(x)$ be a given continuous signal in time domain. The continuous Fourier transform of $f(x)$ is defined by the equation:

$$F(k) = \int_{-\infty}^{\infty} f(x)e^{-2\pi ikx} dx \tag{3.100}$$

where $i = \sqrt{-1}$ and s is often called frequency variable. Given $F(k)$, we can go backwards and get $f(x)$ by using inverse continuous Fourier transform:

$$f(x) = \int_{-\infty}^{\infty} F(k)e^{2\pi ikx} dk \tag{3.101}$$

Equation (3.100) and Equation (3.101) are called Fourier transform pairs. $F(k)$ is the Fourier transform of $f(x)$ and that $f(x)$ is the inverse Fourier transform of $F(k)$. The only difference between the forward and inverse Fourier transform is the sign above e, which makes it easy to go back and forth between time domain and frequency domain.

3.4.3 Discrete Fourier Transform

The continuous Fourier transform is defined as

$$f(\nu) = \mathbb{F}_t[f(t)](\nu) \tag{3.102}$$

$$= \int_{-\infty}^{\infty} f(t)e^{-2\pi i\nu t} dt. \tag{3.103}$$

Now consider generalization to the case of a discrete function, $f(t) \rightarrow f(t_k)$ by letting $f_k = f(t_k)$, where $t_k = k\Delta$, with $k = 0, \dots, N\text{-}1$. Writing this out gives the discrete Fourier transform $F_n = \mathbb{F}_k[\{f_k\}_{k=0}^{N-1}](n)$ as

$$F_n = \sum_{k=0}^{N-1} f_k e^{-2\pi ink/N}. \tag{3.104}$$

The inverse transform $f_k = \mathbb{F}_n^{-1}[\{F_n\}_{n=0}^{N-1}](k)$ is then

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{2\pi ink/N}. \tag{3.105}$$

Discrete Fourier transforms are extremely useful because they reveal periodicities in input data as well as the relative strengths of any periodic components. There are a few subtleties in the

interpretation of discrete Fourier transforms, however, In general, the discrete Fourier transform of a *real* sequence of numbers will be a sequence of *complex* numbers of the same length. In particular, if f_k are real, then F_{N-n} and F_n are related by

$$F_{N-n} = \bar{F}_n \tag{3.106}$$

For $n = 0,1,\ldots, N\text{-}1$, where $\bar{z}$ denotes the complex conjugate. This means that the component F_0 is always real for real data.

As a result of the above relation, a periodic function will contain transformed peaks in not one, but two places. This happens because the periods of the input data become split into "positive" and "negative" frequency complex components.
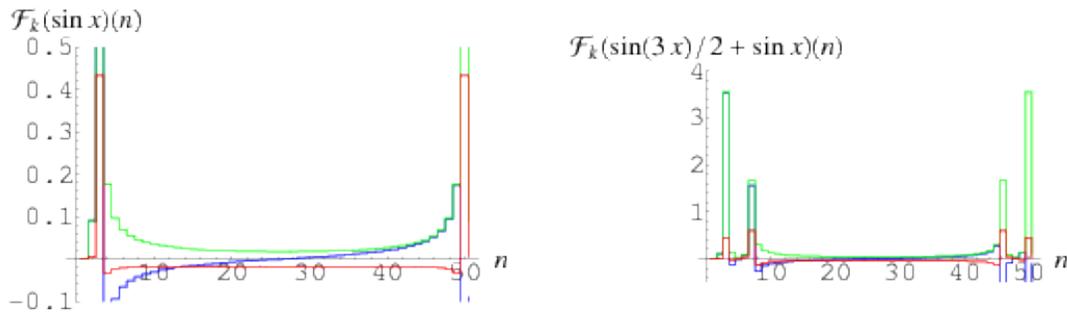
Figure 3.6 The real part, imaginary part, and complex part of the discrete Fourier transforms of the two indicated functions

The plots above show the real part (red), imaginary part (blue), and complex modulus (green) of the discrete Fourier transforms of the functions $f(x) = \sin x$ (left figure) and $f(x) = \sin x + \sin (3x)/2$ (right figure) sampled 50 times over two periods. In the left figure, the symmetrical spikes on the left and right side are the "positive" and "negative" frequency components of the single sine wave. Similarly, in the right figure, there are two pairs of spikes, with the larger green spikes corresponding to the lower-frequency stronger component $\sin x$ and the smaller green spikes corresponding to the higher-frequency weaker component. A suitably scaled plot of the complex modulus of a discrete Fourier transform is commonly known as a power spectrum. The discrete Fourier transform is a special case of the Z-transform.

The discrete Fourier transform can be computed efficiently using a fast Fourier transform.

In other words, The continuous Fourier transform is a continuous function of frequency and is not suitable for computation with a digital signal processing (DSP). Discrete Fourier transform (DFT) representation of the continuous time signal permits the computer analysis and is used extensively in signal processing applications. The analog signal which consists of an infinite number of contiguous points is sampled at regular intervals. The input to the DFT is a sequence of sampled values rather than a continuous function of time $f(t)$, so that

$$F_n = \sum_{k=0}^{N-1} f_k\, e^{-2\pi i n k/N} \qquad\qquad (3.107)$$

And,

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n\, e^{2\pi i n k/N} \qquad\qquad (3.108)$$

Equation (3.107) is called the DFT and the Equation (3.108) is called the inverse discrete Fourier transform (IDFT). f_k and F_n are the discrete sample values corresponding to $f(x)$ and $F(k)$. The N in the DFT pair denotes the number of elements in the f_k or F_n sequence.

The discrete Fourier transform allows calculating the Fourier transform on a computer, but it is not sufficient. The number of complex multiplications and additions required to implement Equation (3.107) and (3.108) is proportional to N^2. For every F_n, it needs to use all of $f(0),\ldots, f(N\text{-}1)$ and there are N F_n to calculate. For a large N, there computations can be prohibitively time consuming, even for a high speed computer.

3.4.4 Fast Fourier Transform

The fast Fourier transform (FFT) is a discrete Fourier transform algorithm which reduces the number of computations needed for N points from $2\,N^2$ to $2\,N\,lg\,N$, where lg is the base-2 logarithm. If the function to be transformed is not harmonically related to the sampling frequency, the response of an FFT looks like a sinc function (although the integrated power is still correct). Aliasing (also known as leakage) can be reduced by apodization using an apodization function. However, aliasing reduction is at the expense of broadening the spectral response.

FFTs were first discussed by Cooley and Tukey (1965), although Gauss had actually described the critical factorization step as early as 1805 (Bergland 1969, Strang 1993). A discrete Fourier transform can be computed using an FFT by means of the Danielson-Lanczos lemma if the

number of points N is a power of two. If the number of points N is not a power of two, a transform can be performed on sets of points corresponding to the prime factors of N which is slightly degraded in speed. An efficient real Fourier transform algorithm or a fast Hartley transform (Bracewell 1999) gives a further increase in speed by approximately a factor of two. Base-4 and base-8 fast Fourier transforms use optimized code, and can be 20-30% faster than base-2 fast Fourier transforms. Prime factorization is slow when the factors are large, but discrete Fourier transforms can be made fast for N=2, 3, 4, 5, 7, 8, 11, 13, and 16 using the Winograd transform algorithm (Press *et al.* 1992, pp. 412-413, Arndt).

Fast Fourier transform algorithms generally fall into two classes: decimation in time, and decimation in frequency. The Cooley-Tukey FFT algorithm first rearranges the input elements in bit-reversed order, then builds the output transform (decimation in time). The basic idea is to break up a transform of length N into two transforms of length *N/2* using the identity

$$\sum_{n=0}^{N-1} a_n \, e^{-2\pi i n k/N}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} a_{2n} \, e^{-2\pi i (2n)k/N}$$

$$+ \sum_{n=0}^{\frac{N}{2}-1} a_{2n+1} \, e^{-2\pi i (2n+1)k/N}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} a_n^{even} \, e^{-2\pi i n k/(\frac{N}{2})}$$

$$+ \, e^{-2\pi i k/N} \sum_{n=0}^{\frac{N}{2}-1} a_n^{odd} \, e^{-2\pi i n k/(\frac{N}{2})}, \tag{3.109}$$

sometimes called the Danielson-Lanczos lemma. The easiest way to visualize this procedure is perhaps via the Fourier matrix.

The Sande-Tukey algorithm (Stoer and Bulirsch 1980) first transforms, then rearranges the output values (decimation in frequency).

Chapter 4 - APPROACH AND PROCEDURE

The general equation for any structural system ambient vibration signal in time domain is:

$$\ddot{u}(t) = Ce^{-\xi\omega_n t}\sin(\omega_D t + \alpha) + Particular\ Solution \tag{4.1}$$

The first part of this equation $(Ce^{-\xi\omega_n t}\sin(\omega_D t + \alpha)$ represents the free vibration of the system embedded in the ambient vibration signal of the system, while the second part (Particular Solution) represents the ambient part of this signal, the vibration noise.

The primary purpose of this approach is to extract the free vibration of the structure from a random vibration response in the time domain (acceleration versus time) by averaging out the random component of the response and eliminating the particular solution. The result is the free vibration that includes all modes based on the sampling rate of time, which is then transferred to the frequency domain using a Fast Fourier Transform (FFT). Variations in the frequency response are function of the structural stiffness and member end-conditions. Such variations are used to identify the change is the structural dynamic properties, and ultimately detect damage.

4.1 Procedure Followed

Figure 4.1 shows a sample of an ambient vibration signal as recorded for a physical model experiment with magnification of the first part of that signal.
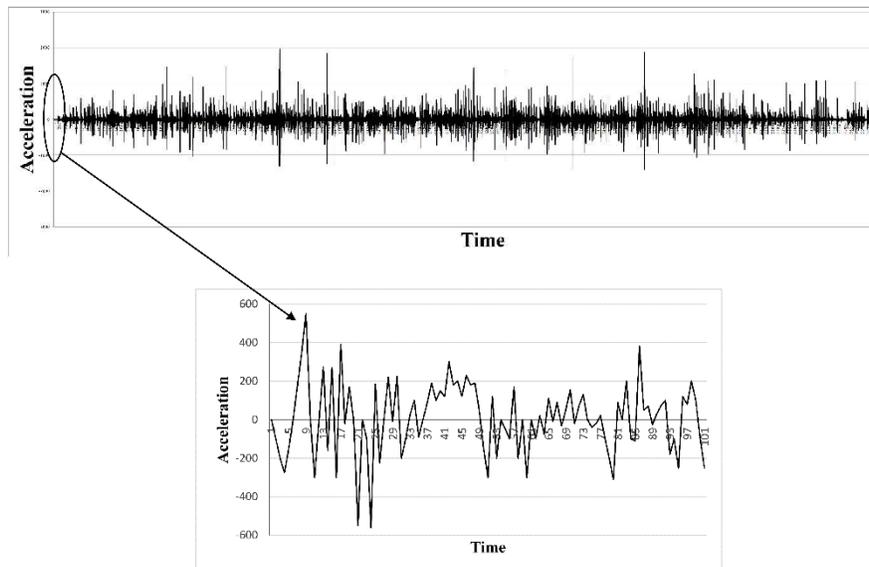
Figure 4.1 Sample of an ambient vibration signal as recorded for a physical model experiment with magnification of the first part

To simplify the formulation, the acceleration $\ddot{u}$ denoted as (u). The procedure required the following steps:

1) A trigger value for the initial acceleration (u_0) and the time intervals ($\varDelta T$) is carefully chosen so that the horizontal line will intersect as many points as possible on the signal curve.

2) A horizontal line parallel to the time axis is extended from the initial acceleration (u_0) point until it intersects the curve. It is important to determine whether the point of intersection is between two ascending or descending values.

3) From the point of intersection, the horizontal line is extended with a value equal to $\varDelta T_1$. (See Figure 4.2 that shows a case of an ascending point of intersection).

4) Wherever the length of $\varDelta T_1$ ends, a vertical line is extended (up or down) till it intersects the signal curve at a point. The ordinate of this point becomes the next acceleration value (u_1). Interpolation is usually necessary to obtain an accurate value between two ascending or descending points on the curve.
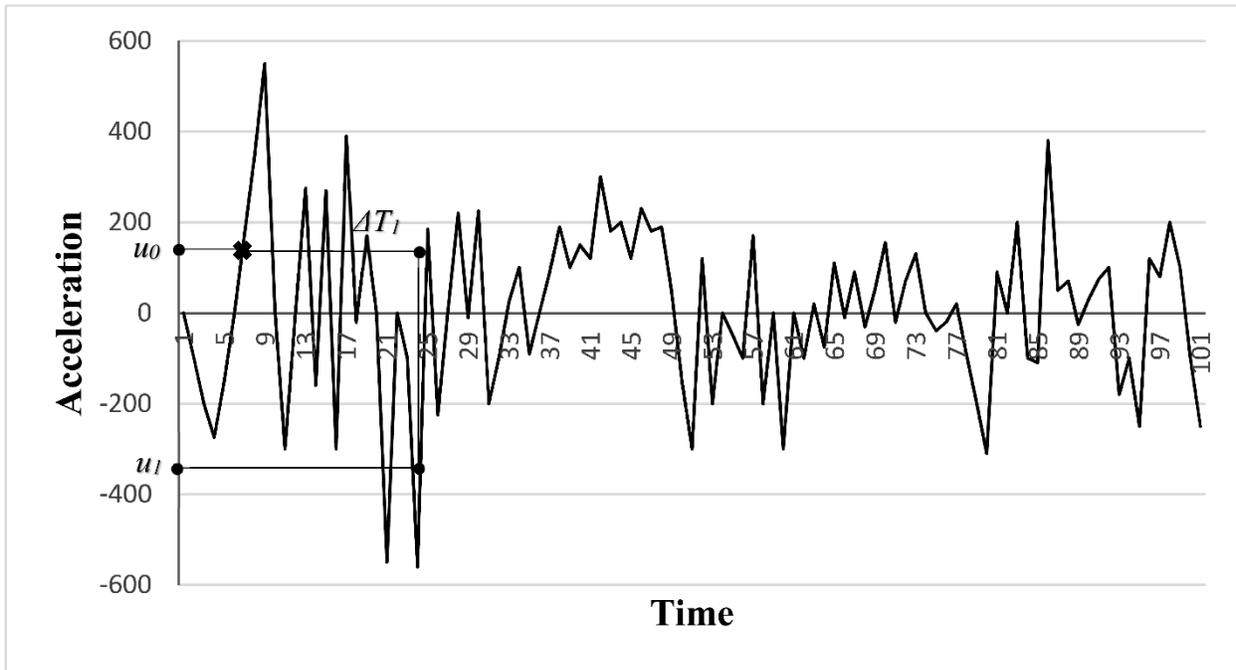
Figure 4.2 A graph showing how to choose a trigger value for the initial acceleration (u) and the time interval ($\varDelta T$) on the ambient vibration signal of Figure 4.1

49

5) From the point of intersection described in Step 3, a horizontal line parallel to the time axis is extended until it intersects the curve again (ascending if the first point was on the ascending portion, or descending otherwise).

6) From the new point of Step 5, the horizontal line is extended with a value of ΔT_1 once again, and a new acceleration value (u_2) is obtained as previously accomplished in Step 4. (See Figure 4.3).

7) Steps 5 and 6 are sequentially repeated till the end of the curve is reached and the final acceleration value (u_n) is recorded, where n is the last intersected point on the curve for this iteration. (See Figure 4.3).

8) Steps 3 to 7 are repeated with different values of (ΔT). These values will be denoted as ΔT_2, ΔT_3, …..…. , ΔT_m, where $\Delta T_2 = 2 \times \Delta T_1$, $\Delta T_3 = 3 \times \Delta T_1$, …. , $\Delta T_m = m \times \Delta T_1$, with ΔT_m being the value of ΔT in the last iteration.

Note: For the sake of computational effectiveness, the value of m is selected such that a sufficient portion on the ambient vibration curve is included (rather than the entire ambient vibration curve), as the free vibration curve is usually embeded in the first part of the ambient vibration curve, and the rest mostly represents the damping part.

Figure 4.3 Graph showing how to obtain different values for the acceleration by extending equal time interval (ΔT) untill the end of the ambient vibration signal

4.2 Free Vibration

The procedure described in the preceding section is used to generate a curve that should represent the free vibration of the system with time on the x-axis and acceleration on the y-axis, i.e., in time domain. This process is illustrated in Figure 4.4.
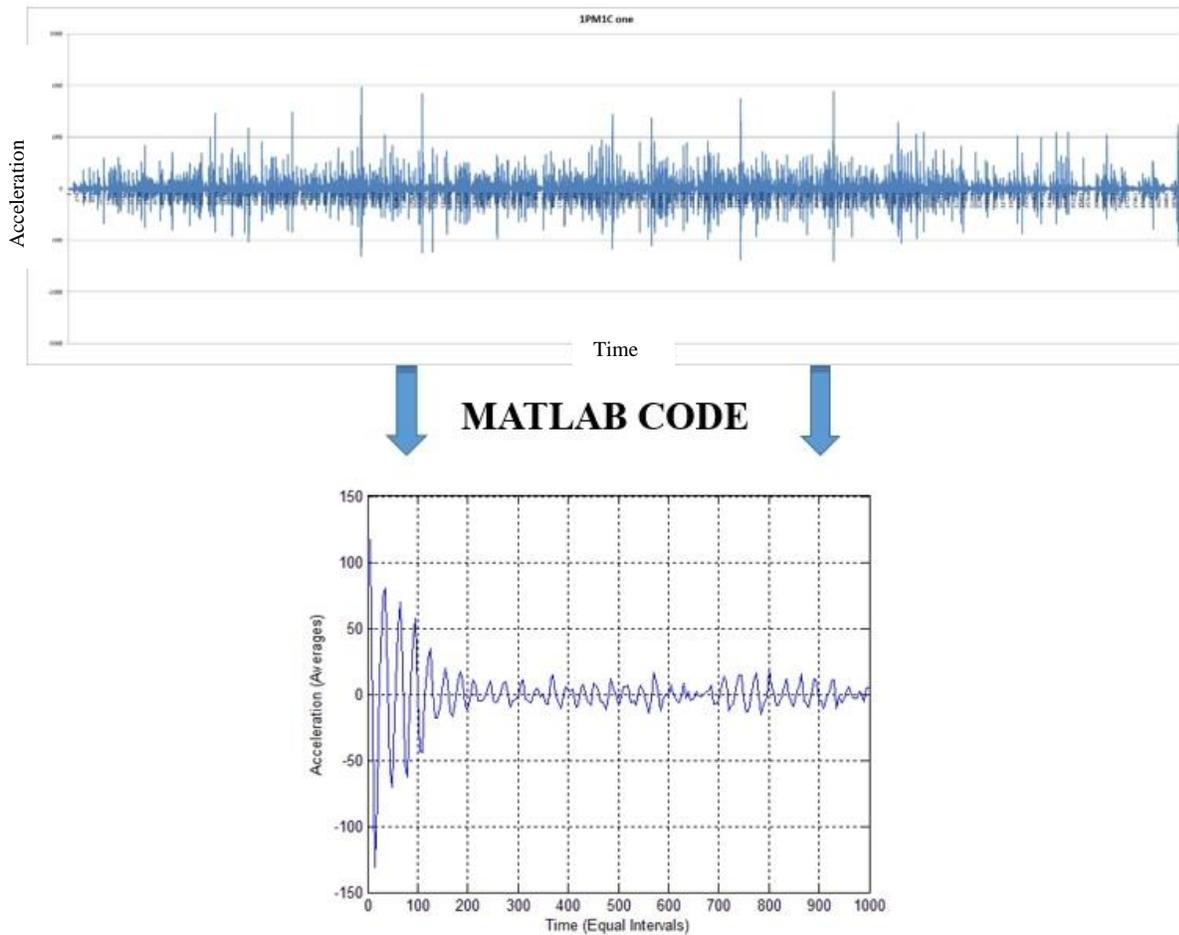
Figure 4.4 Extracting the free vibration of a system from a response of this system to random excitation

The first point on this curve is $(\Delta T_1 , U_1)$ where ΔT_1 is as previously defined, and U_1 is the average of all acceleration values resulting from the use of ΔT_1 (Steps 3 through 6 in the procedure described in Section 4.1). The second point on the free vibration curve is $(\Delta T_2 , U_2)$ where $\Delta T_2 = 2 \times \Delta T_1$, and U_2 is the average of all acceleration values resulting from the use of

$ΔT_2$. The third point on the free vibration curve is $(ΔT_3 , U_3)$ where $ΔT_3 = 3 × ΔT_1$ and U_3 is the average of all acceleration values resulting from the use of $ΔT_3$. This process is repeated m times. The last point on the curve is $(ΔT_m , U_m)$ where $ΔT_m = m × ΔT_1$, and U_m is a single acceleration value resulting from the use of $ΔT_m$.

In other words, the points on the free vibration curve are:

$$First\ Point\quad = (ΔT_1 , U_1 = \frac{u_1 + u_2 + u_3 + u_4 + \dots\dots\dots\dots\dots\dots + u_n}{n})$$

$$Second\ Point = (ΔT_2 , U_2 = \frac{u_1 + u_2 + u_3 + u_4 + \dots\dots\dots\dots\dots\dots + u_n}{n})$$

$$Third\ Point\quad = (ΔT_3 , U_3 = \frac{u_1 + u_2 + u_3 + u_4 + \dots\dots\dots\dots\dots\dots + u_n}{n})$$

.

.

.

$$m\ th\ \ Point\quad\quad = (ΔT_m , U_m = \frac{u_n}{n})$$

$$where,\ ΔT_2 = 2 × ΔT_1 , ΔT_3 = 3 × ΔT_1 , ΔT_4 = 4 × ΔT_1 , \dots\dots\dots\dots ΔT_m = m × ΔT_1$$

It should be noted that the value of n is not the same for all points, but it decreases each time a new point is computed till it reaches $n = 1$ for the last (m^{th}) point.

After extracting the free vibration (in the time domain) from the response of structure to random excitation for each system configuration, a Fast Fourier Transform (FFT) is applied to transform it to the frequency domain. This process is illustrated in Figure 4.5.
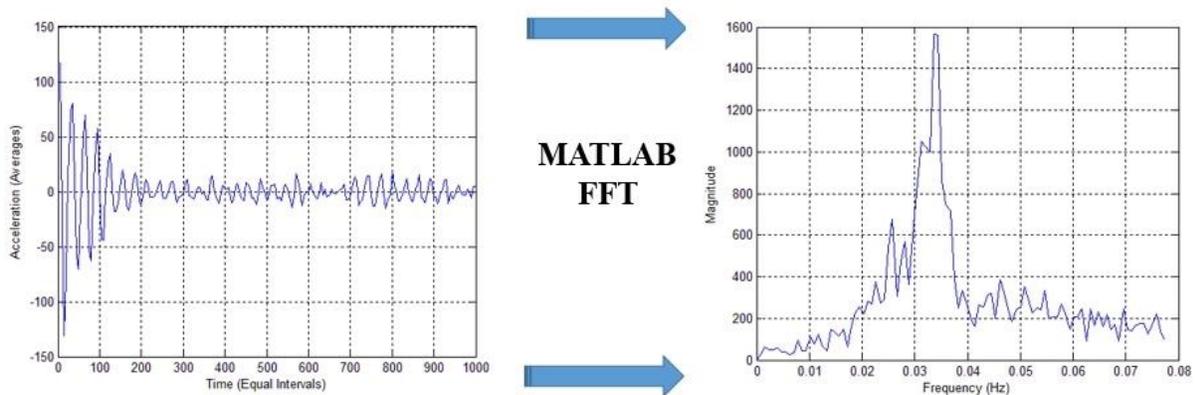
Figure 4.5 Getting the Frequency Domain from the Time Domain using Fast Fourier Transform (FFT)

4.3 Application

A physical steel model was constructed (discussed in details in Chapter 5) and different configurations were used. After applying a random vibration and recording the corresponding ambient vibration signals, the above approach is used to get the free vibration - in the time domain - for the individual cases, then FFT is applied to obtain its representation in the frequency domain for each case.

A numerical model is developed in ABAQUS (Abaqus/CAE 6.10-2, © Dassault Systems, 2010) to match each physical model configuration. The same approach is used on the ambient vibration signal resulting from each model to get its free vibration signal (in time domain), then FFT is applied to convert it into frequency domain.

By comparing the physical and numerical models frequency domain results, damage is detected based on variations of the frequency contents in the frequency domain.

4.4 Matlab Codes

In order to numerically implement the computational procedure presented above, two computer codes were developed. The first extracts the free vibration from the response of structure to random excitation, and the second transforms the signal from the time domain to the frequency domain.

4.4.1 Extraction of Free Vibration

In order to extract the free vibration from a response of structure to random excitation using this approach, a Matlab code (MATLAB R2013a, The MathWorks, Inc.) was implemented to be used in this approach. This code is listed in Appendix A.

4.4.2 Fast Fourier Transformation

In order to change the free vibration signal from the time domain to the frequency domain, a second Matlab code (MATLAB R2013a, The MathWorks, Inc.) was implemented. This code is listed in Appendix B.

Chapter 5 - SOFTWARE FOR NUMERICAL ANALYSIS

ABAQUS (Abaqus/CAE 6.10-2, © Dassault Systems, 2010) was used to model the physical model with different boundary condition configurations and respective loading conditions, and to analyze the corresponding numerical model to obtain the ambient vibration response for each case.

MATLAB (MATLAB R2013a, The MathWorks, Inc.) was used to numerically implement the computational procedure presented in Chapter 4. Two computer codes were developed: The first is to extract the free vibration signal from the ambient vibration signal, and the second is to transform the signal from the time domain to the frequency domain.

5.1 Abaqus/CAE 6.10-2 Analysis

The following is a step-by-step procedure (24 steps) to numerically simulate each physical model used in the experimental investigation. Accomplished tasks are shown in Table 5.1.

Table 5.1 Accomplished tasks and corresponding steps.

Tasks	*Steps #*
Creating model geometry	1 to 6
Defining material properties	7, 8
Instances assembly	9
Creating analysis steps	10
Identifying supports, loads and accelerometer positions	11 to 13
Boundary conditions modeling	14
Applying Loads	15
Selecting points where vibrations need to be recorded and boundary conditions need to be specified	16, 17
Creating the model mesh	18, 19
Submitting and solving the model	20 to 22
Output data and results	22 to 24

Below is the detailed description of the 24 steps referred to in Table 5.1.

1. Start Abaqus/CAE and click on "With Standard/Explicit Model", as shown in Figure 5.1.
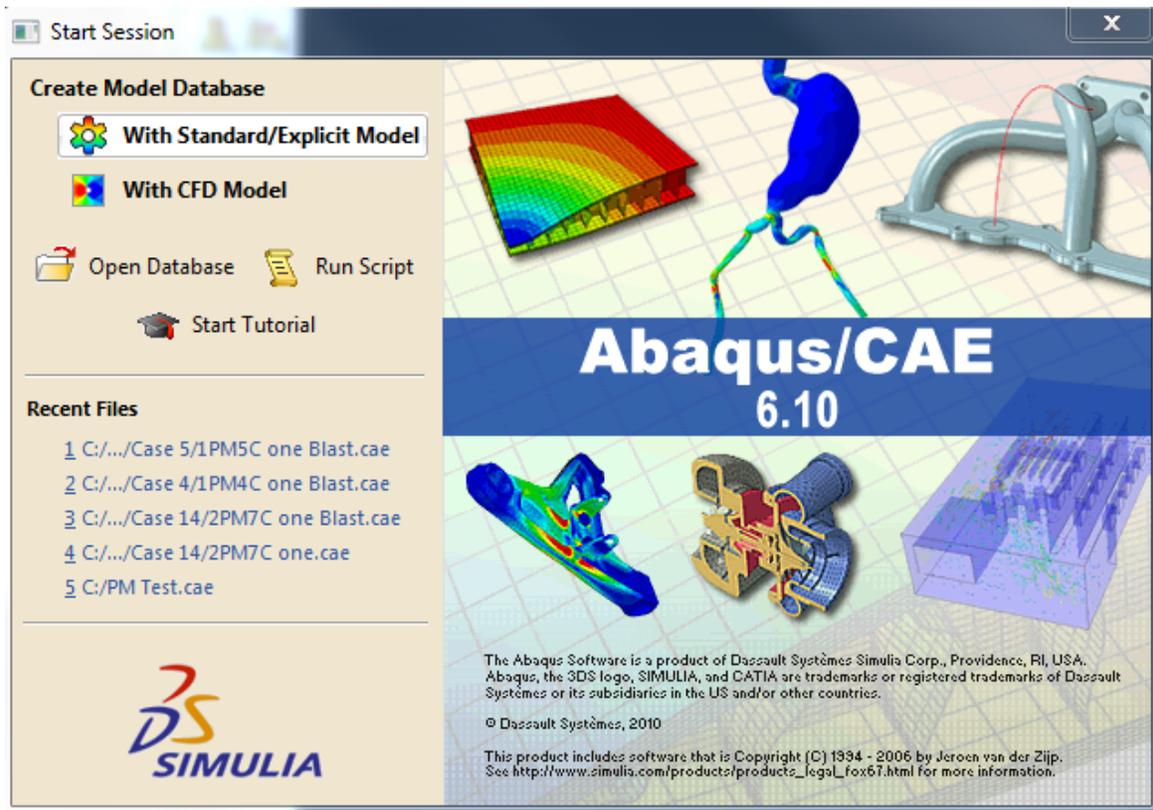
Figure 5.1 Abaqus/CAE 6.10 Interface

2. In the model tree double click on the "Parts" node or right click on "Parts" and select Create, as shown in Figure 5.2.
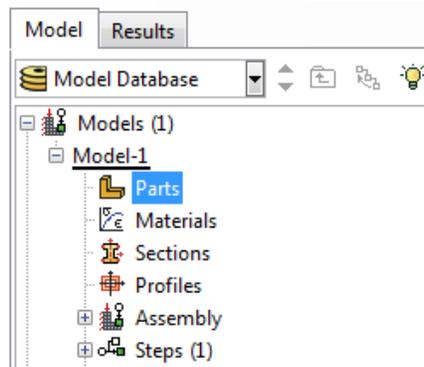
Figure 5.2 "Parts" node in Abaqus Model tree

3. As shown in Figure 5.3, in the "Create Part" dialog box name the part and:
   a. Select "3D" in "Modeling Space"
   b. Select "Deformable" in "Type"
   c. Select "Solid" in "Base Feature Shape"
   d. Select "Extrusion" in "Base Feature Type"
   e. Set Approximate size = 20
   f. Click "Continue…"
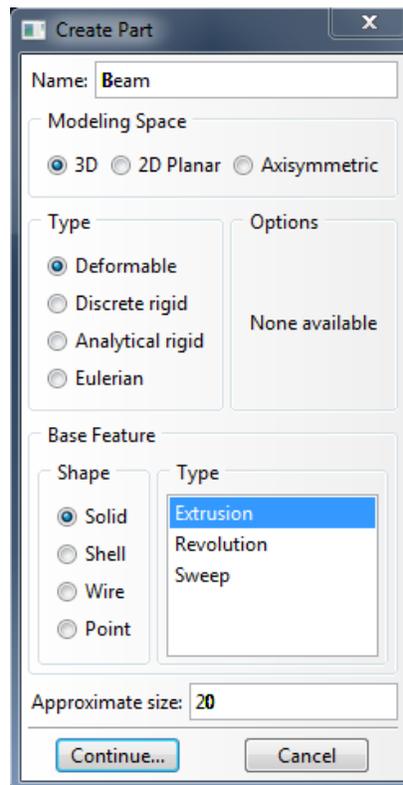
Figure 5.3 "Create Part" dialog box in Abaqus

56

4. As shown in Figure 5.4, click on "Create Lines: Rectangle (4 Lines)" icon, and then select a starting corner for the rectangle. Enter XY coordinates for the opposite corner of the rectangle using SI units, and then press Enter
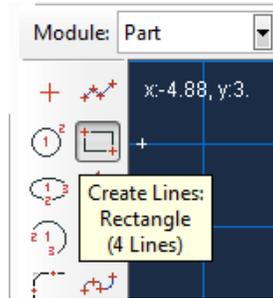
Figure 5.4 "Create Lines: Rectangle (4 Lines)" icon in Abaqus Module: Part

5. As shown in Figure 5..5, select "Offset Curves", select the entities to offset, click on "Done", enter the tube thickness value, and then press Enter then select "OK"
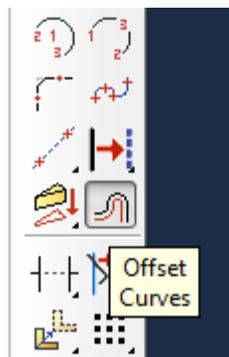
Figure 5.5 "Offset Curves" icon in Abaqus Module: Part

6. Click on "Done" then enter the Beam length value "End Condition, Depth", As shown in Figure 5.6.
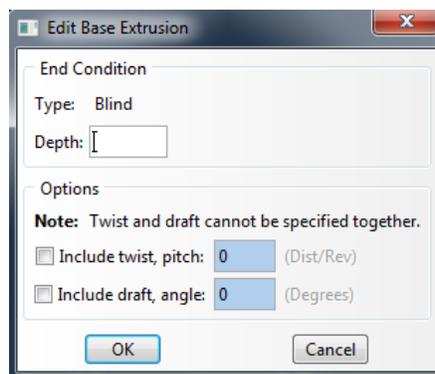
Figure 5.6 "Edit Base Extrusion" dialog box in Abaqus

7. Double click the "Materials" node in the model tree, as shown in Figure 5.7.
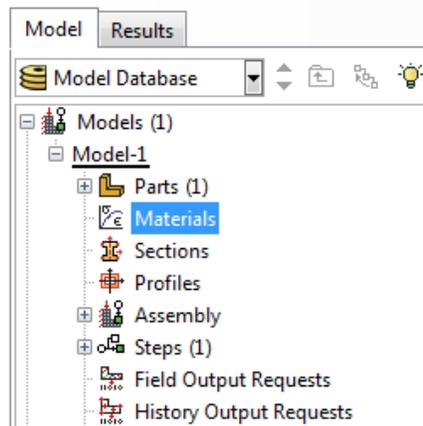
Figure 5.7 "Materials" node in Abaqus Model tree

   a. As shown in Figure 5.8, name the new material and give it a description
   b. Click on the "Mechanical" tab, and then select "Elasticity", and then "Elastic"
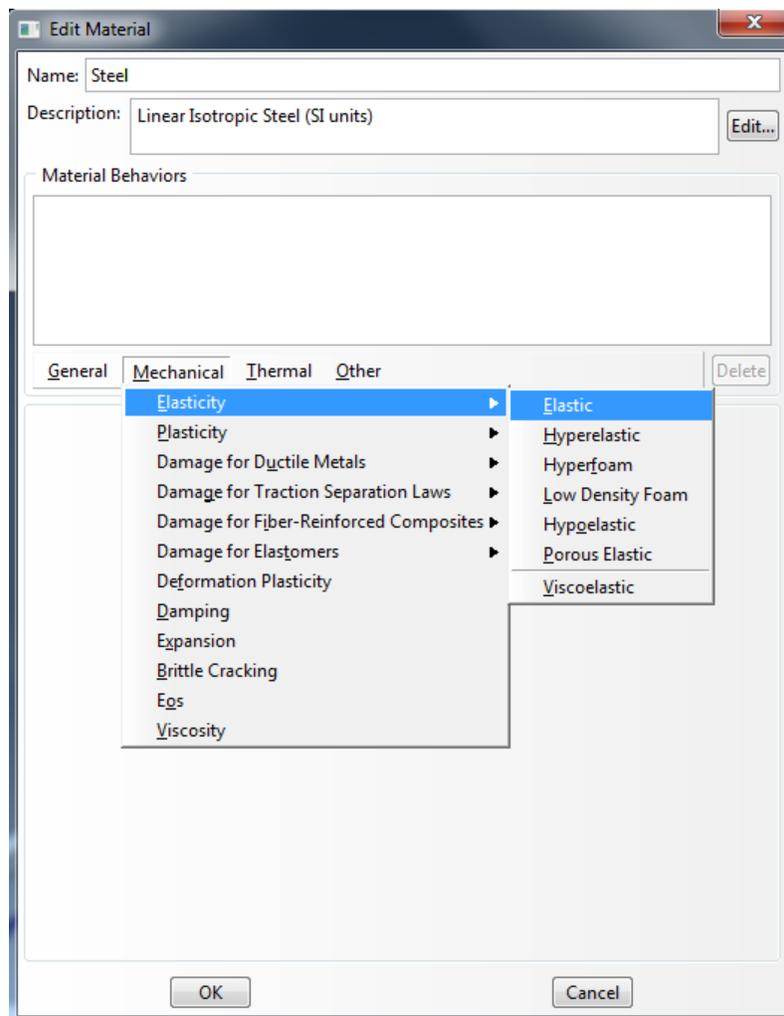
c.   Define Material Properties in SI units
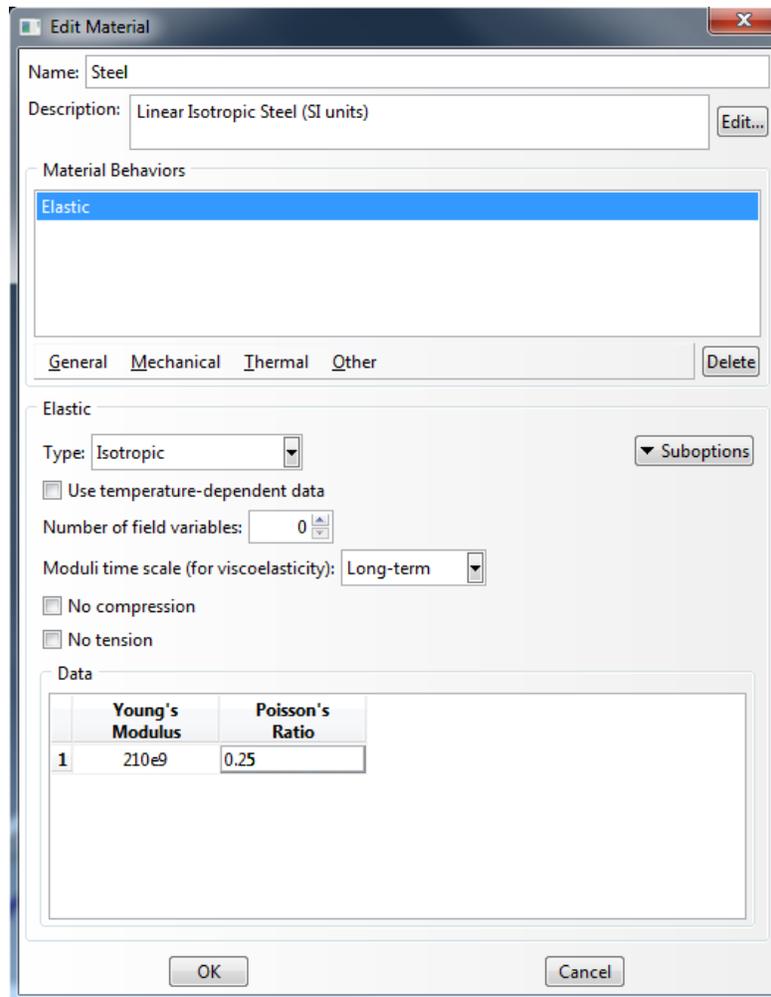
Figure 5.8 "Edit Material" dialog boxes in Abaqus

d.   Enter the values of Young's Modulus and Poisson's Ratio using SI units
e.   Click on "General" tab $\rightarrow$ Density
f.   Define the material density (7800 Kg/m^3)
g.   Click "OK"

8. Double click on the "Sections" node in the model tree, as shown in Figure 5.9.
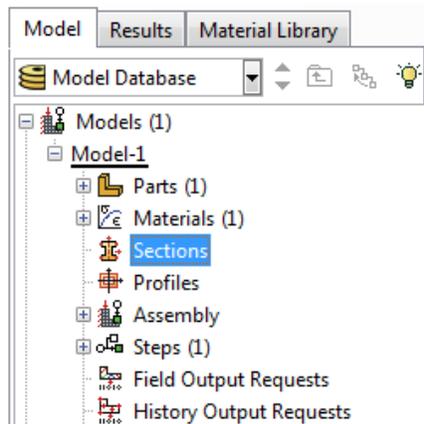
Figure 5.9 "Sections" node in Abaqus Model tree

a. As shown in Figure 5.10, name the section "BeamProperties" and select "Solid" for Category, and "Homogeneous" for Type, then click "Continue…"
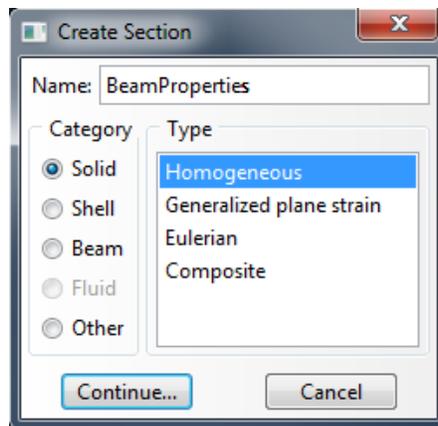
Figure 5.10 "Create Section" dialog box in Abaqus

b. Select "Steel" for Material, as shown in Figure 5.11.
c. Click "OK"
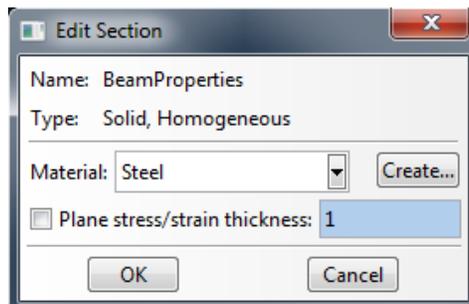
Figure 5.11 "Edit Section" dialog box in Abaqus

d. Click on "Assign Section", as shown in Figure 5.12.
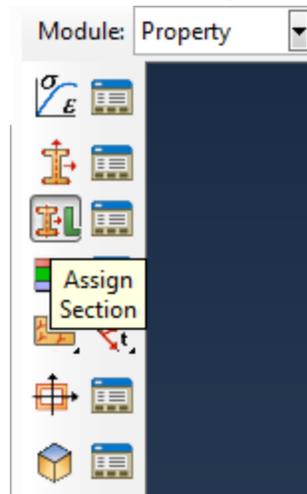
Figure 5.12 "Assign Section" icon in Abaqus Module: Property

e. Select the whole steel beam
f. Click "Done"
g. Select "BeamProperties" for Section, as shown in Figure 5.13.
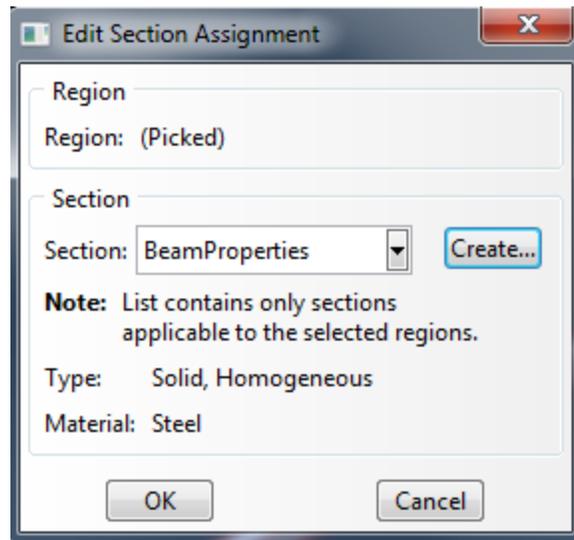
Figure 5.13 "Edit Section Assignment" dialog box in Abaqus

h. Click "OK"

9. Expand the "Assembly" node in the Model tree and then double click on "Instances", as shown in Figure 5.14.
    a. Select "Independent" for the instance type, as shown in Figure 5.15.
    b. Click "OK"
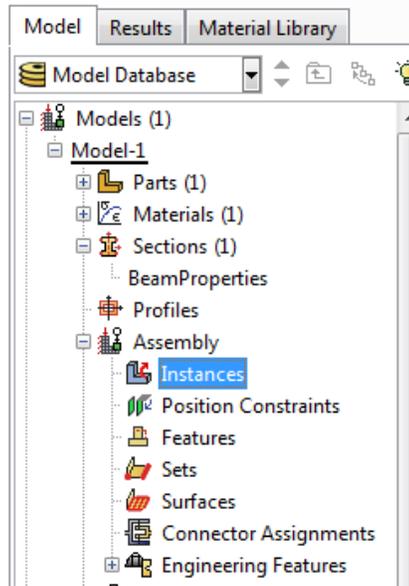
Figure 5.14 [Left] "Instance" node in Abaqus Model tree
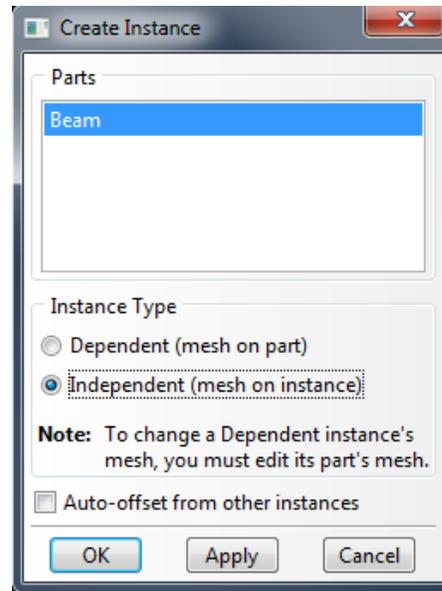
Figure 5.15 [Right] "Create Instance" dialog box in Abaqus

10. Double click on the "Steps" node in the Model tree, as shown in Figure 5.16.
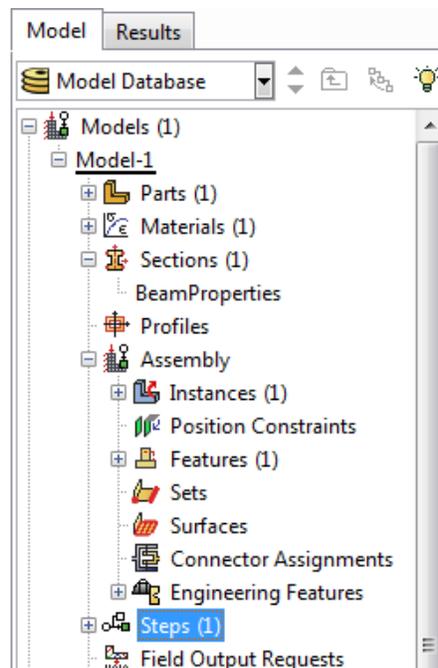
Figure 5.16 "Steps" node in Abaqus Model tree

a. Name the step, set the procedure type to "General" and select Dynamic, Explicit
b. Click "Continue…", as shown in Figure 5.17.
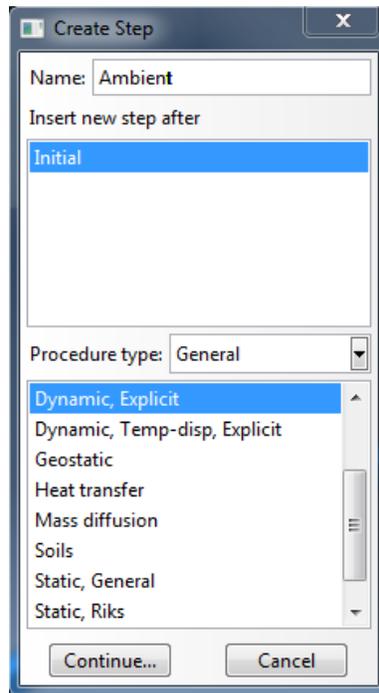
Figure 5.17 "Create Step" dialog box in Abaqus

c. As shown in Figure 5.18, give a step description
d. Choose a Time period in seconds (0.01 sec.)
e. Click "OK"
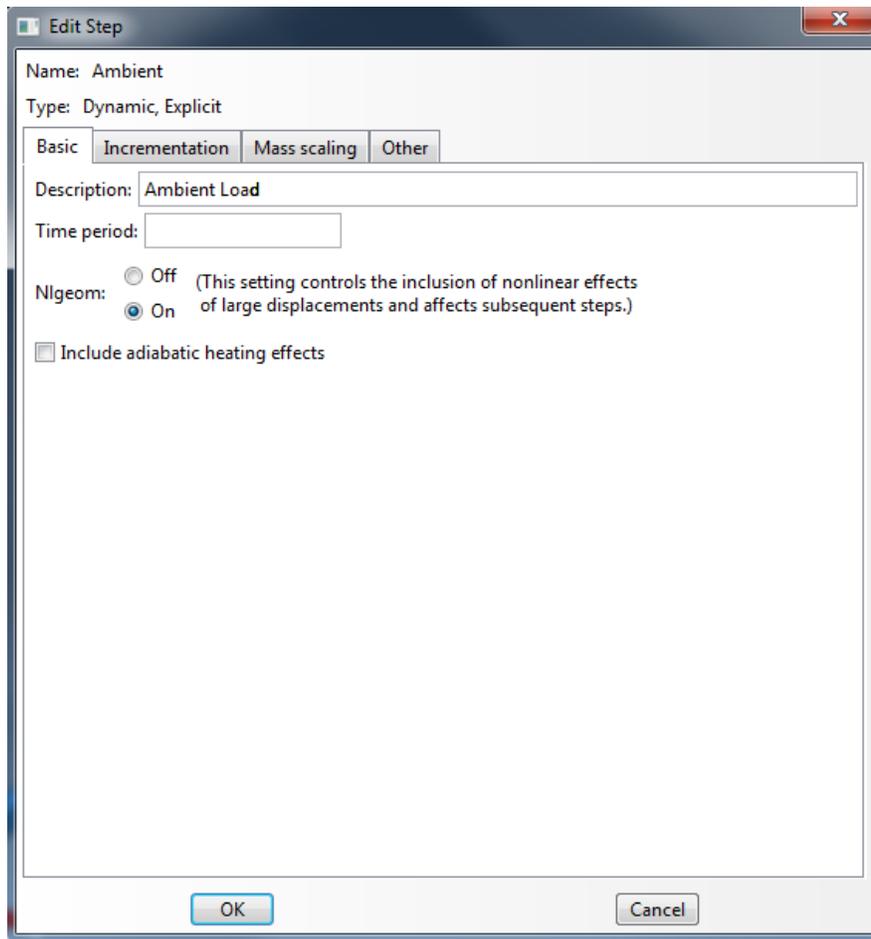
Figure 5.18 "Edit Step" dialog box in Abaqus

11. As shown in Figure 5.19, change the Module to "Assembly"
    a. Click on "Translate Instance"
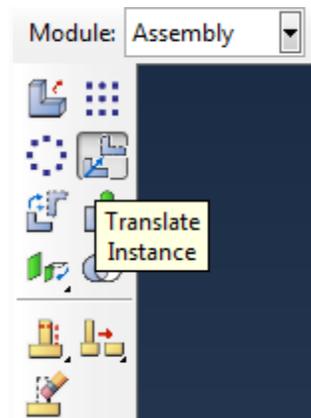
Figure 5.19 "Translate Instance" icon in Abaqus Module: Assembly tool bar

b. Select the instance to translate
c. Click "Done"
d. Select a starting point for the translation vector
e. Click on "Render Model: Wireframe" to show invisible nodes, as shown in Figure 5.20.
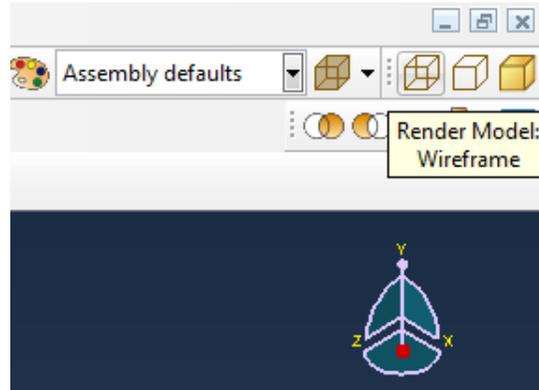
Figure 5.20 "Render Model: Wireframe" icon in Abaqus main tool bar

f. Select an end point for the translation vector, (the purpose is to translate the instance to the origin at a clear node so you can define the cutting planes easily in the following step)
g. Click "OK"

12. In the menu bar select:
a. Tools, Datum, Point, and then Enter coordinates, as shown in Figure 5.21.
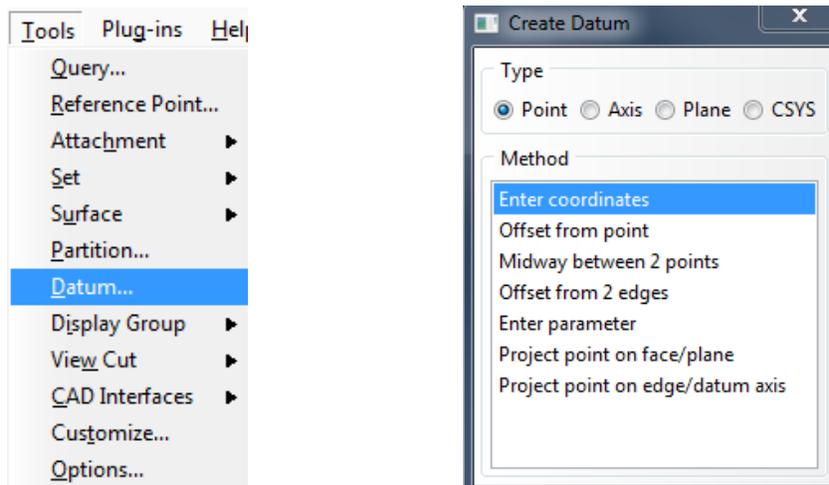
Figure 5.21 "Create Datum" path and dialog box in Abaqus

b. Enter XYZ coordinates for the three points that will represent each plane where the supports, accelerometer locations, loads … etc. are expected to be applied

13. As shown in Figure 5.22, change the Module to "Assembly"
    a. Click on "Partition Cell: Define Cutting Plane" icon
    b. Select three points for "How do you want to specify the plane?"
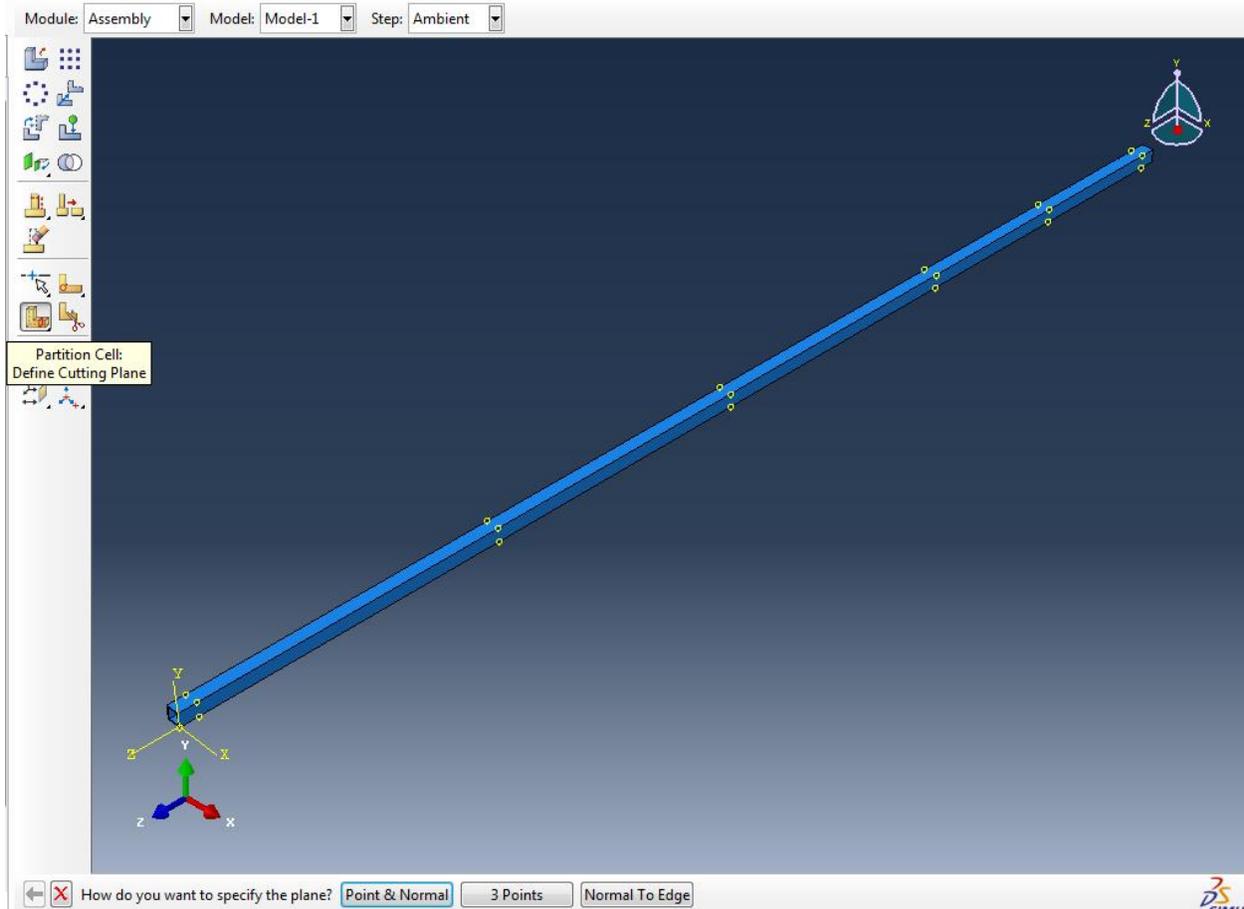
Figure 5.22 Created Portions on the numerical model at places where boundary conditions, loads and accelerometer will be modeled

c. Select three points for each plane
d. Click "Create Partition" for each plane
e. Click "Done"

14. Double click on the "BCs" node in the model tree, as shown in Figure 5.23.
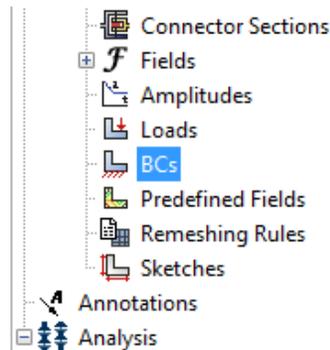
Figure 5.23 "Boundary Conditions" node in Abaqus Model tree

   a. As shown in Figure 5.24, name the boundary condition "Pinned" or "Roller"
   b. Select "Ambient" for Step
   c. Select "Mechanical" for Category
   d. Select "Displacement/Rotation" for Types for Selected Step
   e. Click "Continue…"
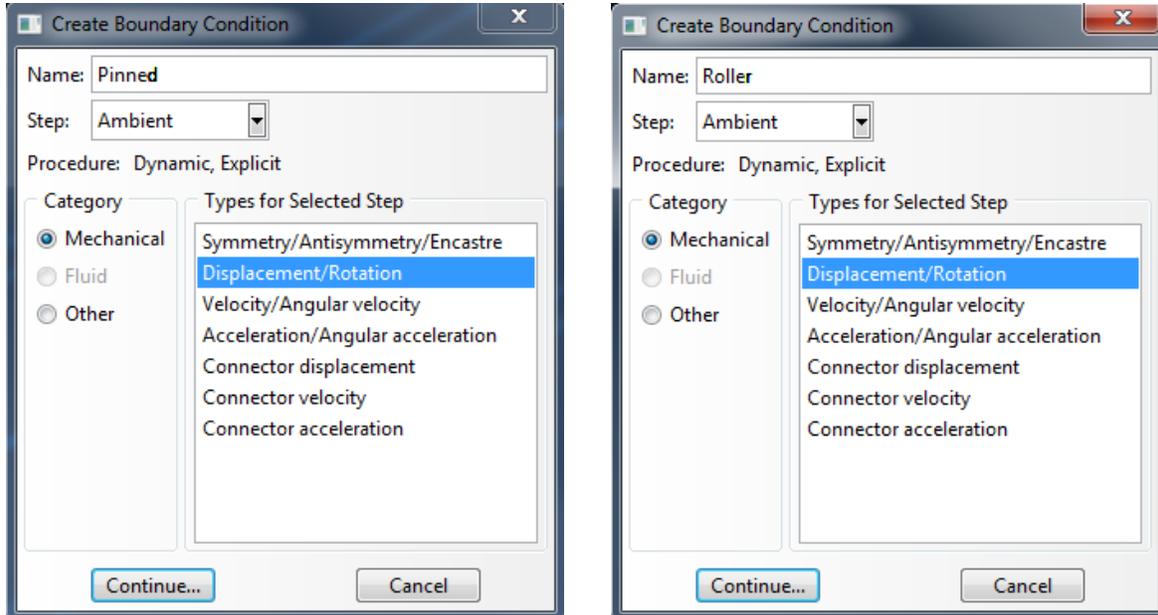
Figure 5.24 "Create Boundary Condition" dialog box in Abaqus

   f. As shown in Figure 5.25, check U1 , U2, and U3 and set them to 0 for "Pinned"
   g. Check U1 and U2 displacements and set them to 0 for "Roller"
   h. Click "OK"
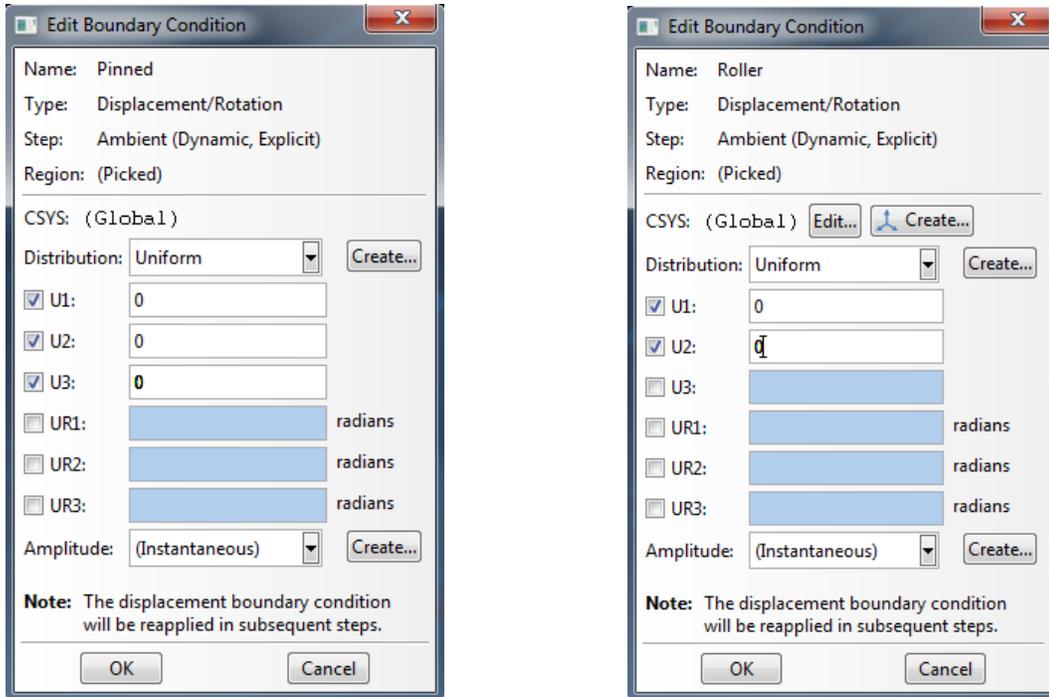
Figure 5.25 "Edit Boundary Condition" dialog box in Abaqus

Figure 5.26 Numerical Model with modeled boundary conditions and the created partitions

15. Double click on the "Loads" node in the model tree, as shown in Figure 5.27.
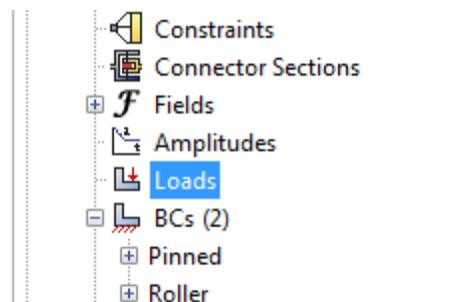
Figure 5.27 "Loads" node in the Abaqus Model tree

a. As shown in Figure 5.28, name the load
b. Select "Ambient" for Step
c. Select "Mechanical" for Category

d. Select "Pressure" for Types for Selected Step
e. Click "Continue…"
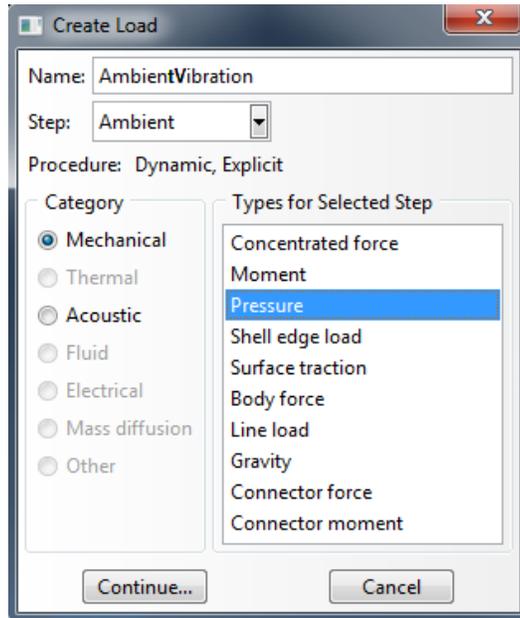
Figure 5.28 "Create Load" dialog box in Abaqus

f. As shown in Figure 5.29, select different instance surfaces to apply the load
g. Click "Done"
h. Select "Uniform" for Distribution
i. Select the Magnitude ($1.0E+006$ N/m^2)
j. Select the Amplitude
k. Click "OK"
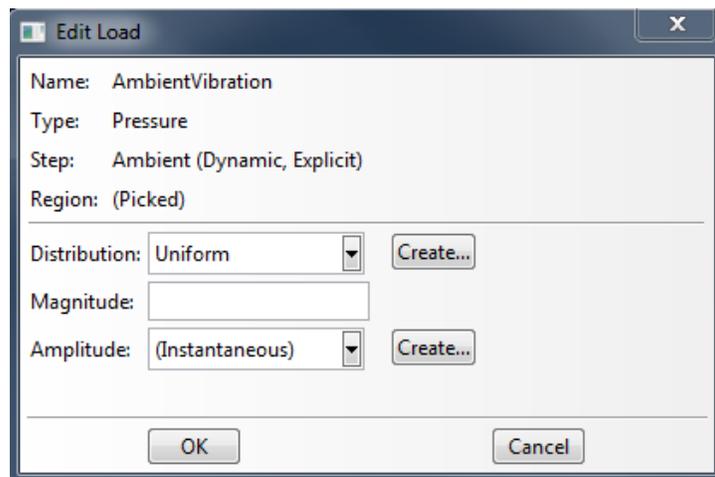
Figure 5.29 "Edit Load" dialog box in Abaqus

69

16. Expand the "Assembly" node in the model tree and double click on "Sets", as shown in Figure 5.30.
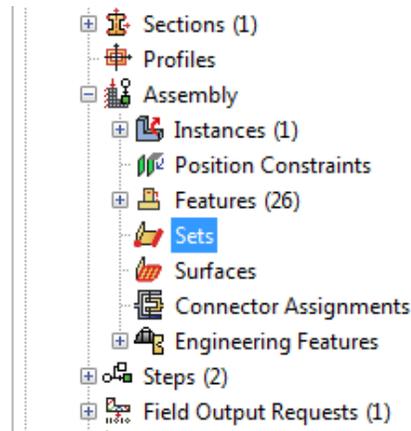
Figure 5.30 "Sets" node in Abaqus Model tree

a. As shown in Figure 5.31, name the Set
b. Click "Continue…"
c. Select a point at the measured location
d. Click "Done"
e. Repeat this step for each location

Figure 5.31 "Create Set" dialog box in Abaqus

17. Double click on the "History Output Requests" node in the model tree, as shown in Figure 5.32.
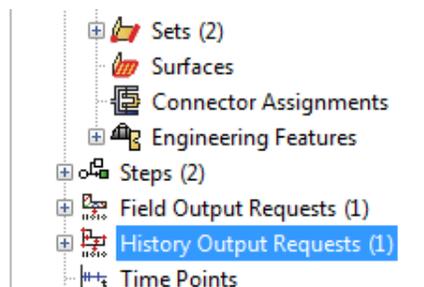
Figure 5.32 "History Output Requests" node in Abaqus Model tree

a. As shown in Figure 5.33, name the History
b. Select "Ambient" for Step
c. Click "Continue…"
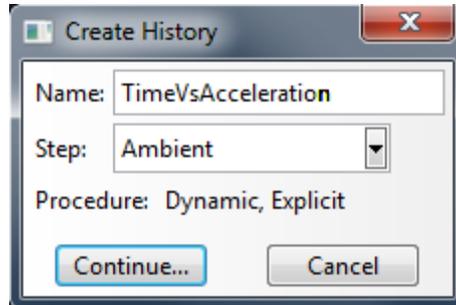
Figure 5.33 "Create History" dialog box in Abaqus

d. As shown in Figure 5.34, select "Set" for Domain, and then select one set from previous "Sets" step
e. Select "Every n time increment" for Frequency, then Select n=1
f. Select "Select from list below"
g. Expand "Displacement/Velocity/Acceleration"
h. Expand "A, Translational and rotational accelerations"
i. Check "A2"
j. Click "OK"
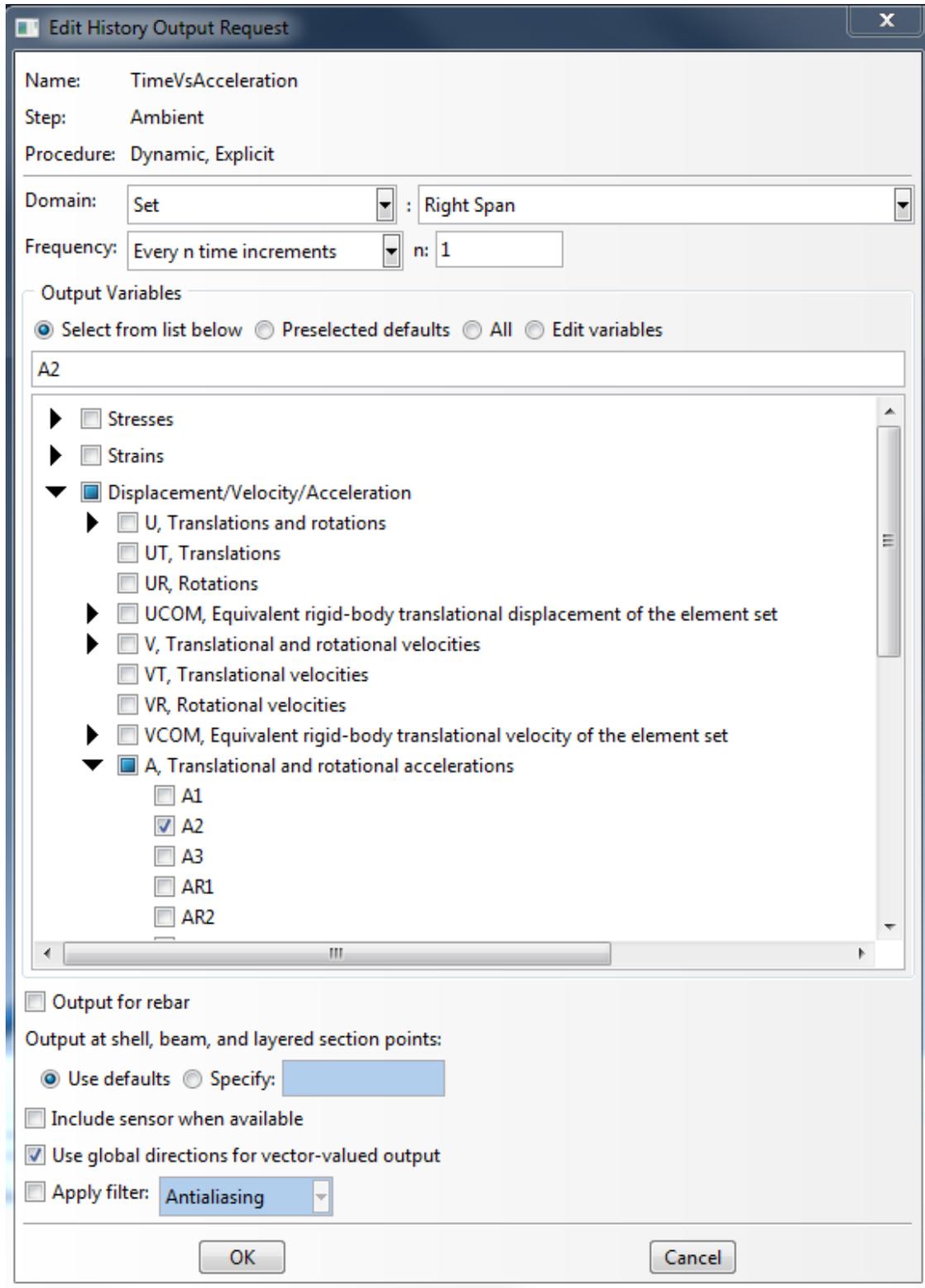
Figure 5.34 "Edit History Output Request" dialog box in Abaqus

18. Change the Module to "Mesh" and in the toolbox area click on the "Assign Element Type" icon, as shown in Figure 5.35.
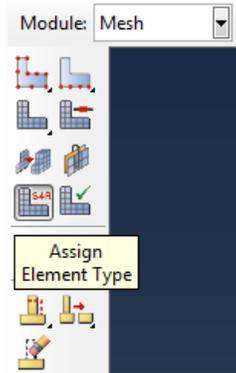
Figure 5.35 "Assign Element Type" icon in Abaqus Module: Mesh

a. Select the regions to be assigned element types
b. Click "Done"
c. Leave all default selections then click "OK", as shown in Figure 5.36.
d. Click "Done"
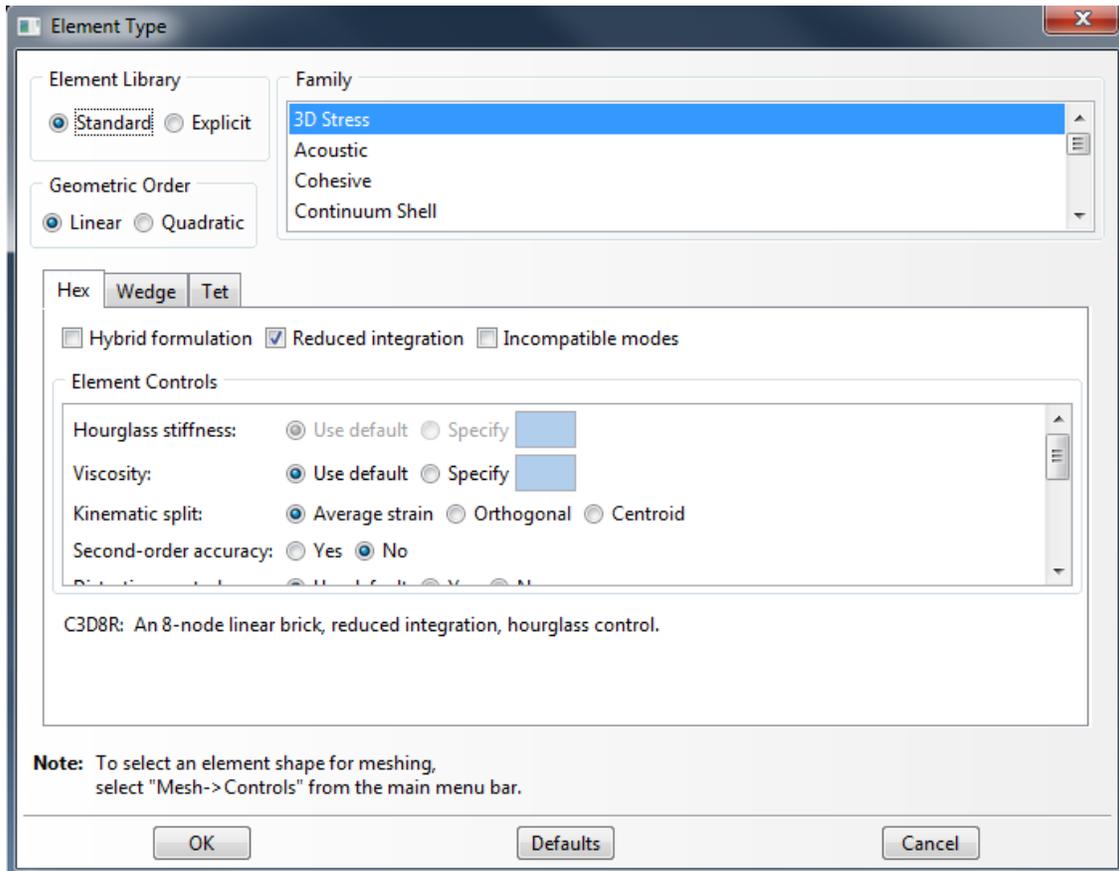
Figure 5.36 "Element Type" dialog box in Abaqus

e. Click on "Seed Part Instance" icon, as shown in Figure 5.37.

Figure 5.37 "Seed Part Instance" icon in Abaqus Module: Mesh tool bar

f. Choose an "Approximate global size", as shown in Figure 5.38.
g. Click "OK"
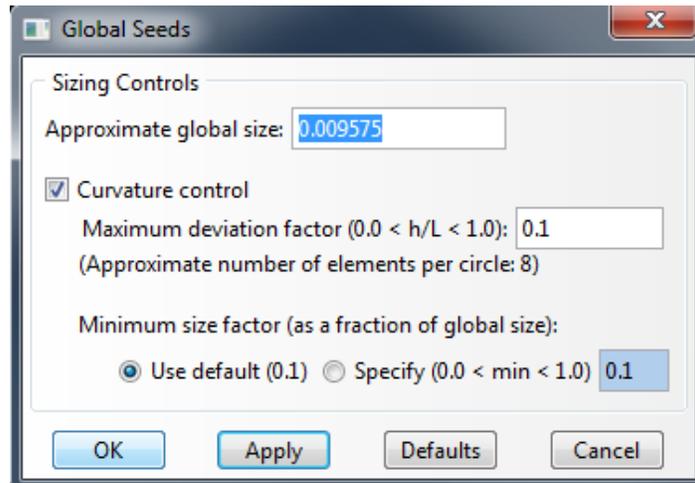
Figure 5.38 "Global Seeds" dialog box in Abaqus

h. Click on "Mesh Part Instance" icon, as shown in Figure 5.39.
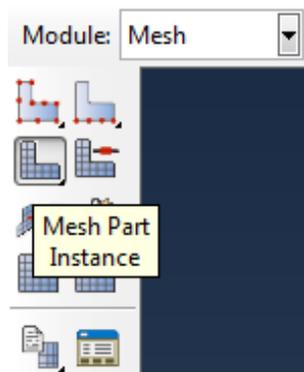
Figure 5.39 "Mesh Part Instance" icon in Abaqus Module: Mesh tool bar

i. Click on "Yes" for OK to mesh the part instance?, as shown in Figure 5.40.
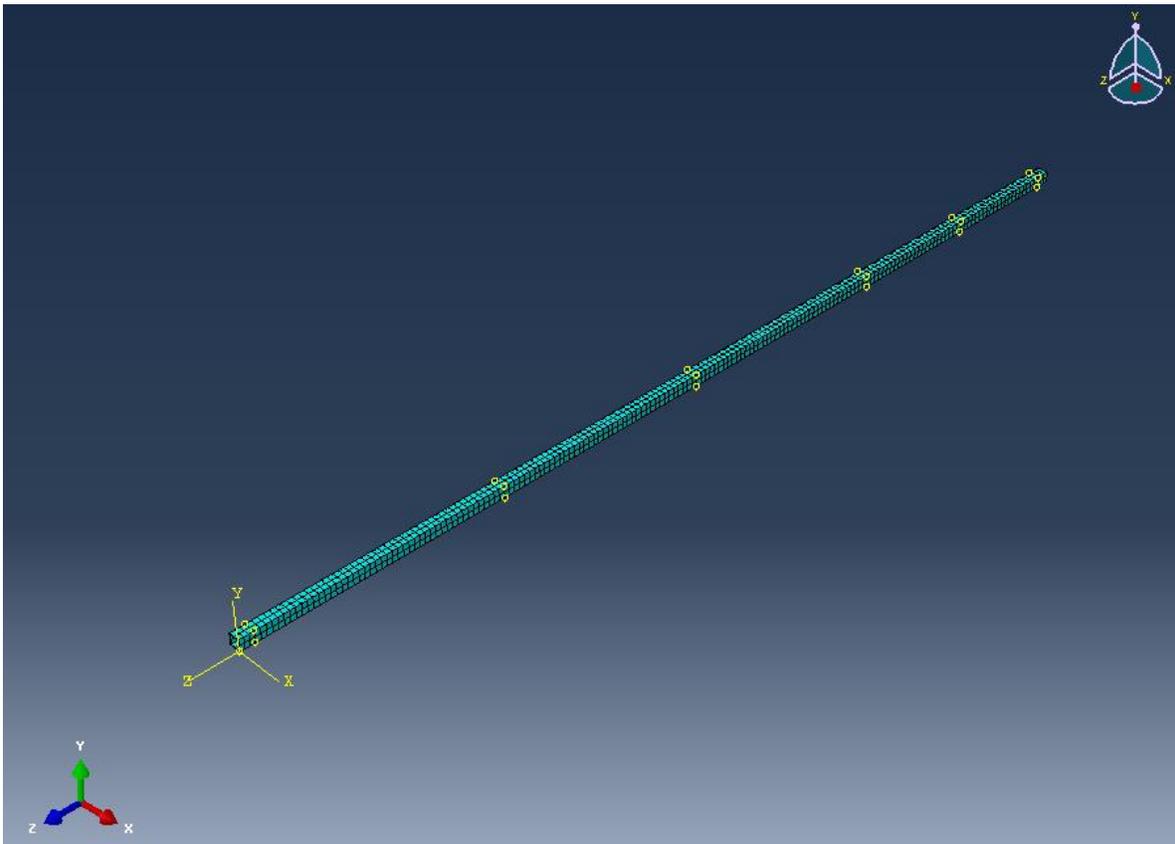
Figure 5.40 Meshing the part instance

19. As shown in Figure 5.41, in the menu bar select:
    a. View, and then "Assembly Display Options …"
    b. Check the box "Render beam profile"
    c. Click "OK"
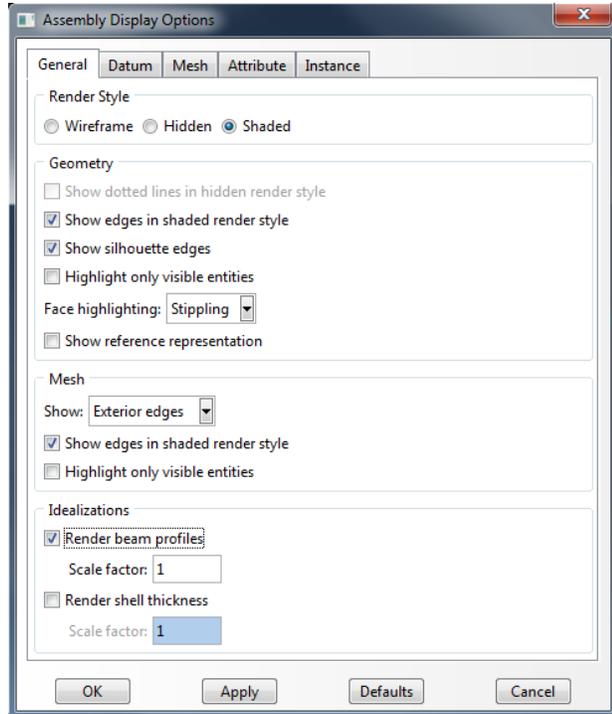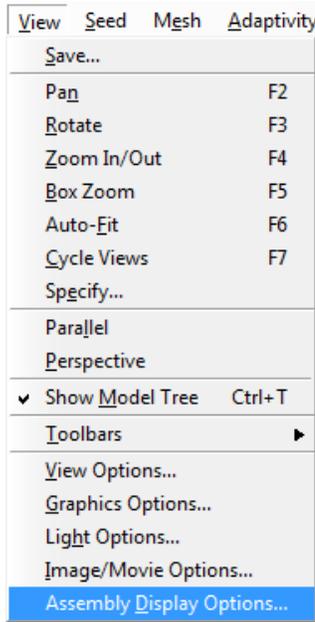
Figure 5.41 "Assembly Display Options…" path and dialog box in Abaqus

20. Double click on the "Job" node in the model tree, as shown in Figure 5.42.
    a. Name the job
    b. Click "Continue…", as shown in Figure 5.43.
    c. Give a description to the job
    d. Click "OK", as shown in Figure 5.44.
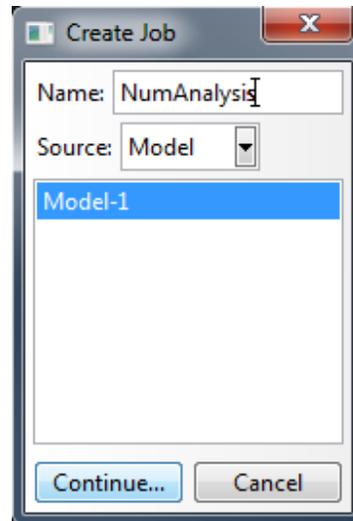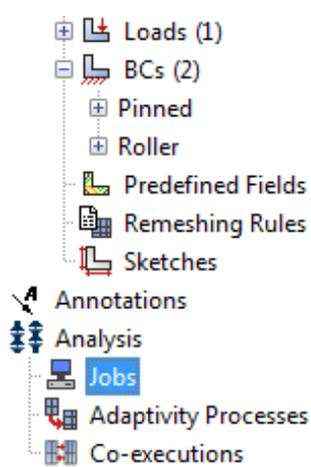
Figure 5.42 [Left] "Jobs" node in Abaqus Model tree

Figure 5.43 [Right] "Create Job" dialog box in Abaqus
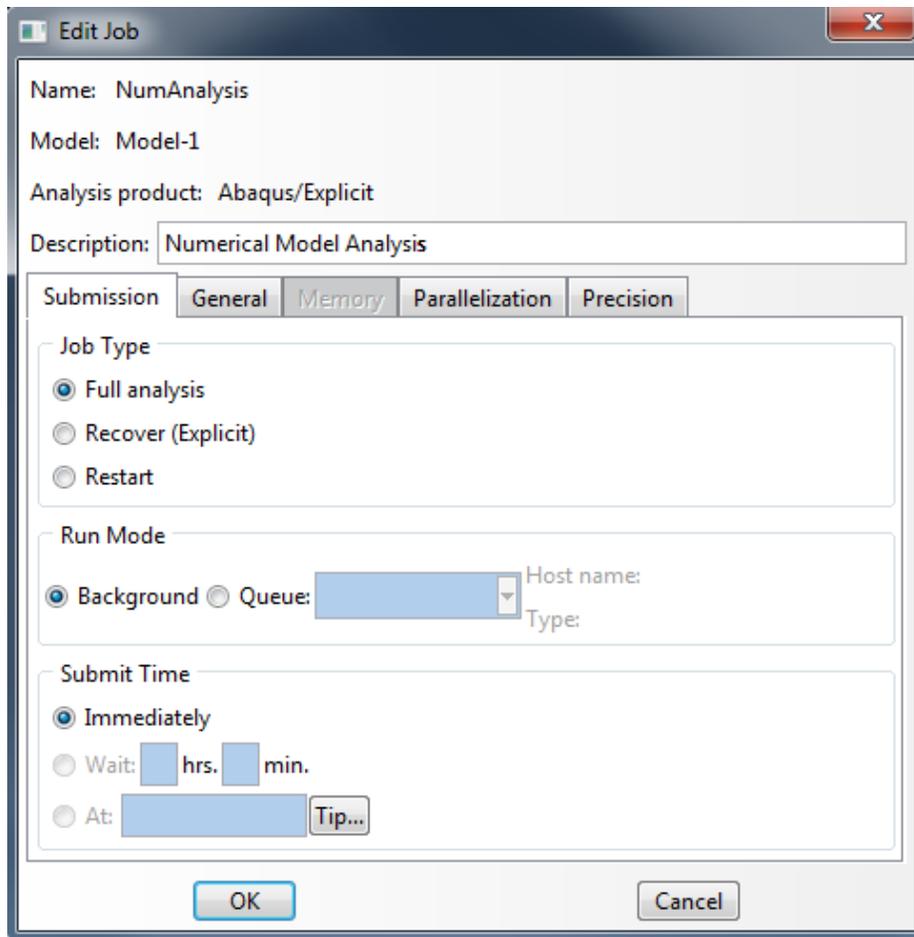
Figure 5.44 "Edit Job" dialog box in Abaqus

21. Expand the "Job" node in the model tree
    a. Right click on the job name and select "Submit", as shown in Figure 5.45.
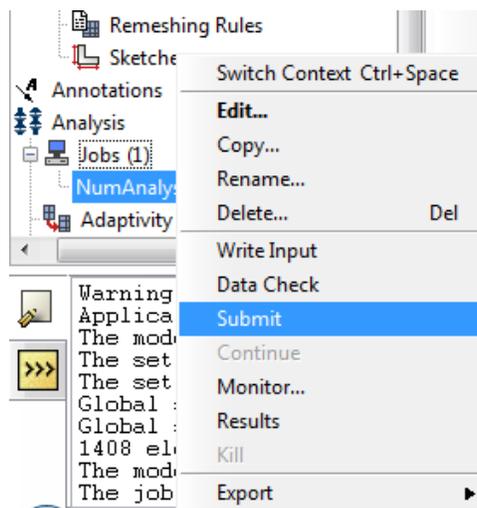
Figure 5.45 Submitting a job in Abaqus

b. Right click on the job name and select "Monitor", as shown in Figure 5.46.
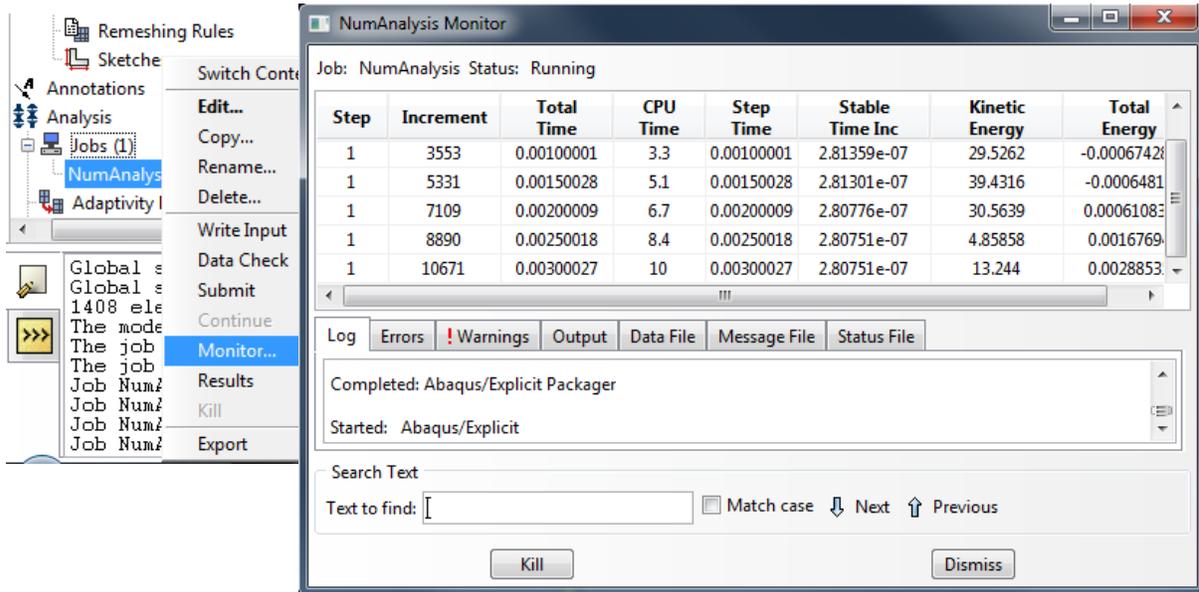
Figure 5.46 Monitoring a job analysis in Abaqus

c. Right click on the job name and select "Results", as shown in Figure 5.47.
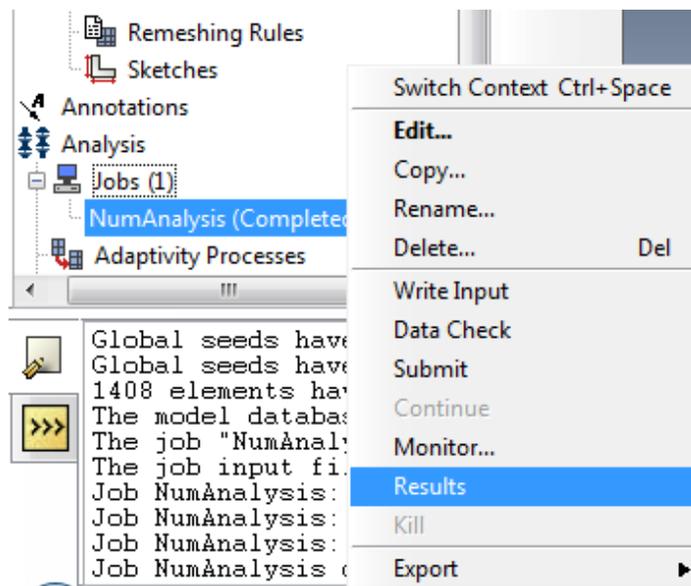
Figure 5.47 "Results" path in Abaqus

d. Click on "Plot Contours on Deformed Shape" to see the deformed shape cases due to the applied ambient load, boundary conditions and the material properties, as shown in Figure 5.48
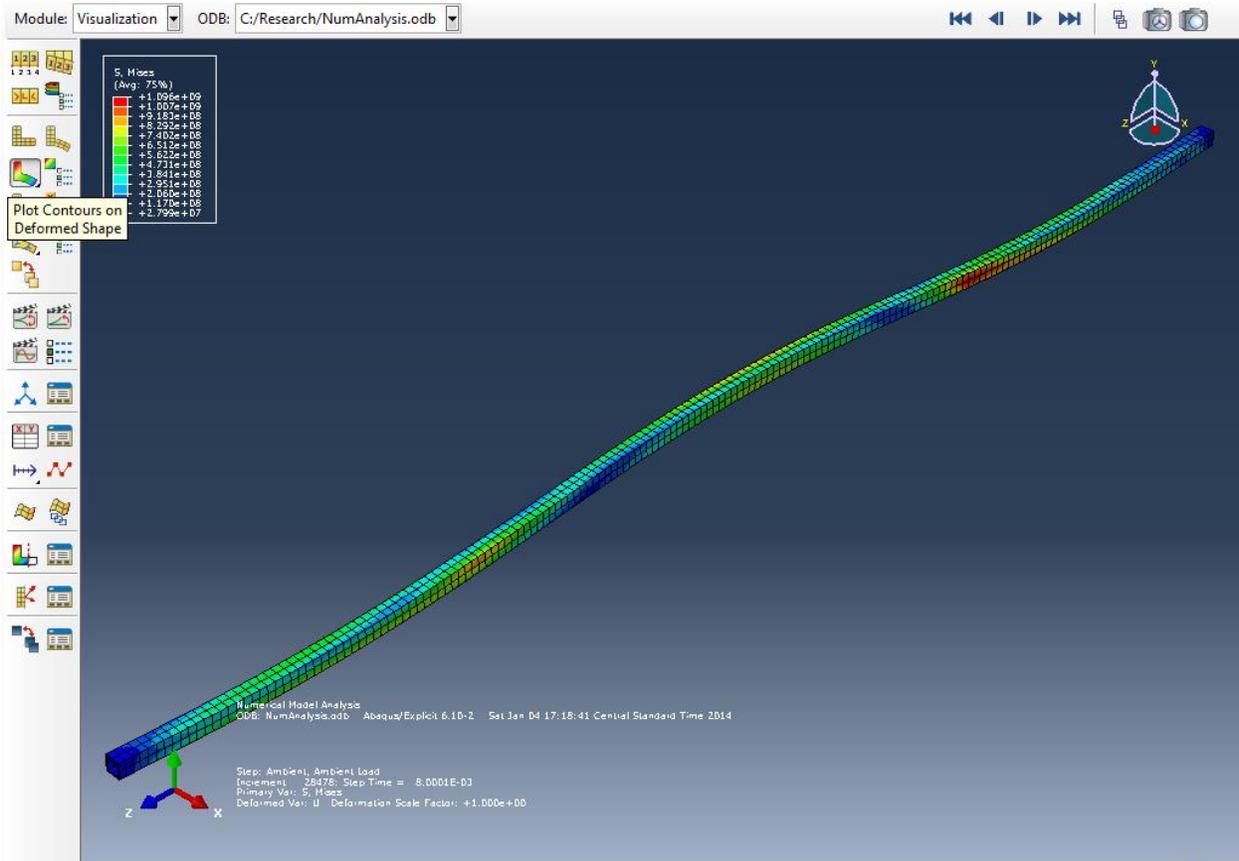
Figure 5.48 One case of the Beam Deformed Shape

22. Expand the "Output Databases" node in the results tree
    a. Expand the job name node
    b. Expand the "History Output" node
    c. Double click on "Spatial acceleration: A2 at Node 48 in NSET RIGHT SPAN", as shown in Figure 5.49.
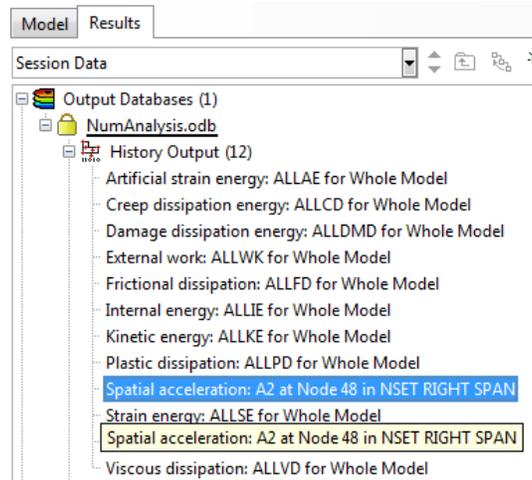
Figure 5.50 "Spatial acceleration: A2" at a specefic node in Abaqus Results tree

    d. See the graph showing the Time vs. Acceleration at the selected node, as shown in Figure 5.50.
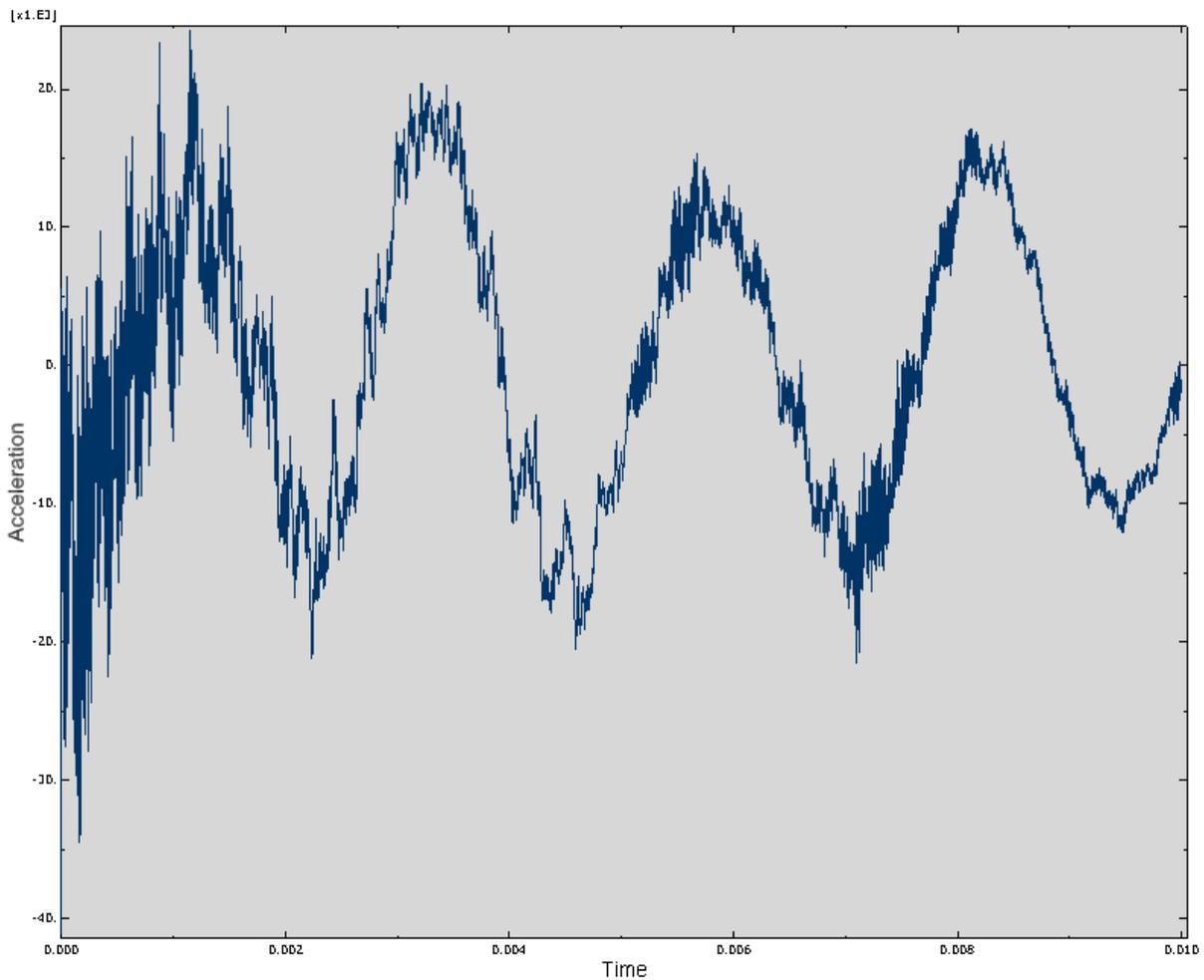
Figure 5.49 Time vs. Acceleration at a specific node (Time Domain)

23. As shown in Figure 5.51, click on "XY Data Manager" icon
    a. Click on "Edit" in the "XY Data Manager" window
    b. Export the output file to a spread sheet file
    c. The spread sheet will be used as an input file in Matlab
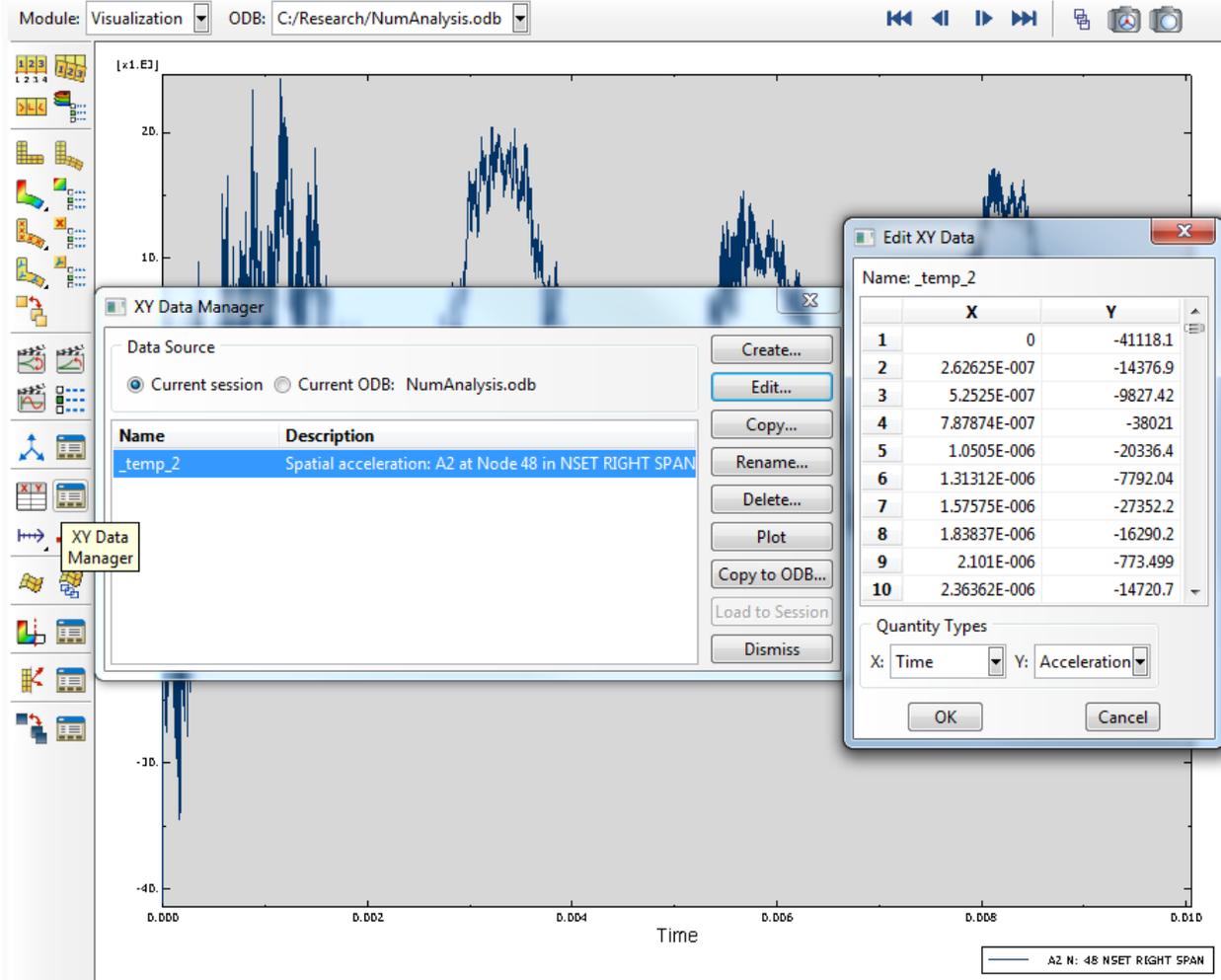
Figure 5.51 Path showing output file exportation to a spread sheet file to be used as an input file in Matlab

24. Change the boundary conditions and the applied ambient loads and obtain the results for each case.

Note: Applying ambient loads could be achieved in Abaqus/CAE in different pricedures other than what was used here.

5.2 MATLAB R2013a Analysis

The following is a step–by–step procedure (5 steps) to obtain the free vibration in frequency domain from a given structure ambient vibration response signal.

1. Start Matlab, as shown in Figure 5.52.
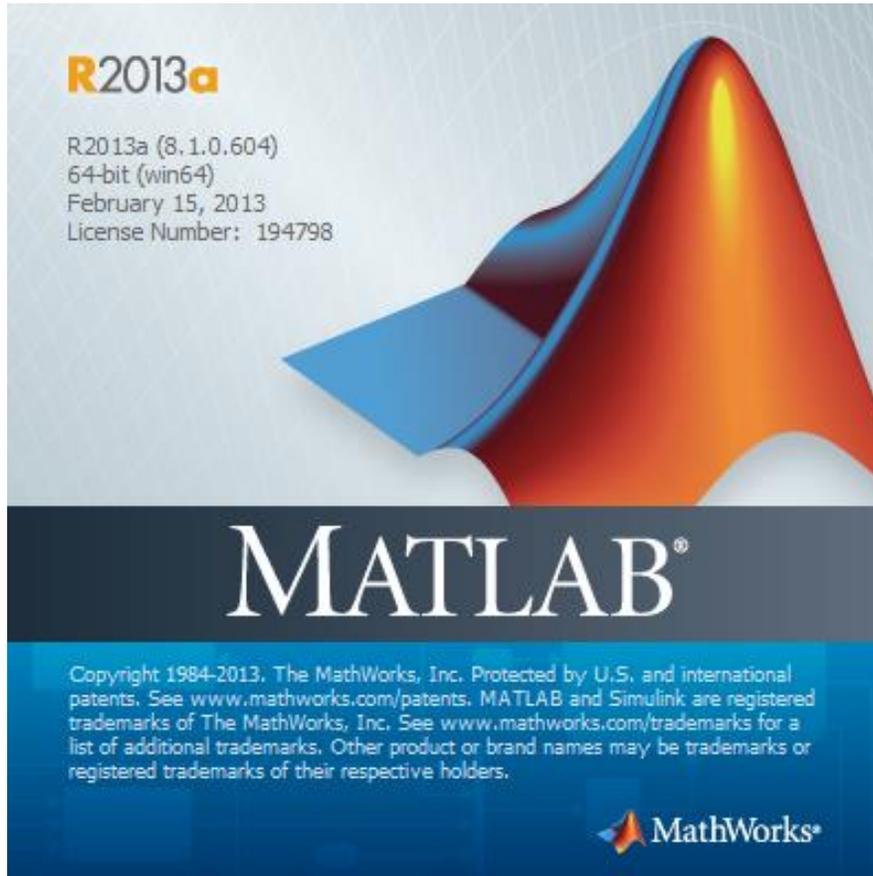
Figure 5.52 MATLAB R2013a Interface

   a. As shown in Figure 5.53, import the *Extracting Free Vibration Response from a Response of Structure to Random Excitation* Matlab Code to the "Current Folder" tree
   b. Import the *Fast Fourier Transformation* Matlab Code to the "Current Folder" tree
   c. Import the spread sheet data file for Time vs. Acceleration obtained from each Physical Model case and from each Numerical Model Abaqus case to the "Current Folder" tree
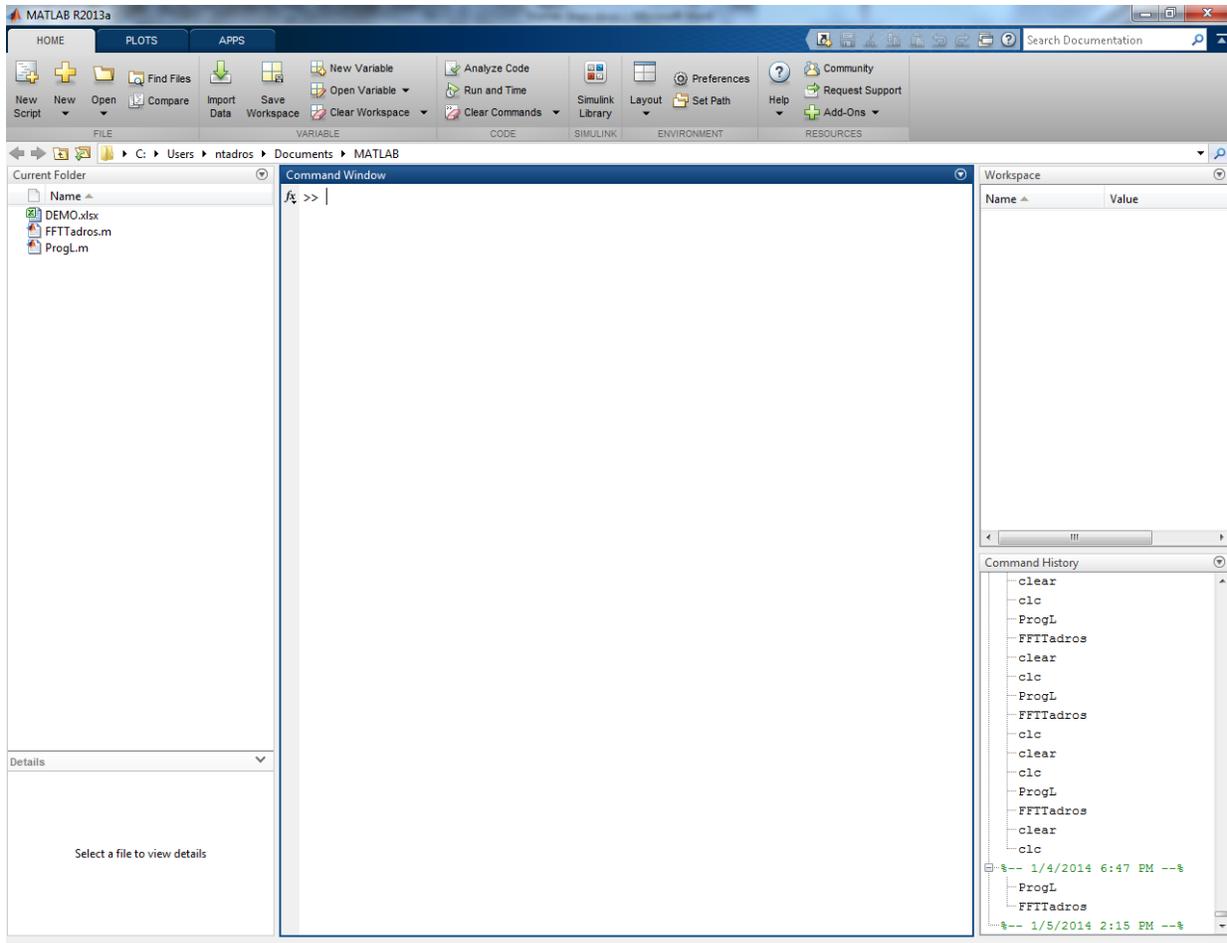
Figure 5.53 MATLAB R2013a general command window

2. Double click on the spread sheet data file in the "Current Folder" tree
   a. Select the two columns A & B, as shown in Figure 5.54 and its corresponding graph in Figure 5.55)
   b. Select "Matrix" in the "IMPORTED DATA" tab
   c. Click on "√ Import Selection" in the "IMPORT" tab
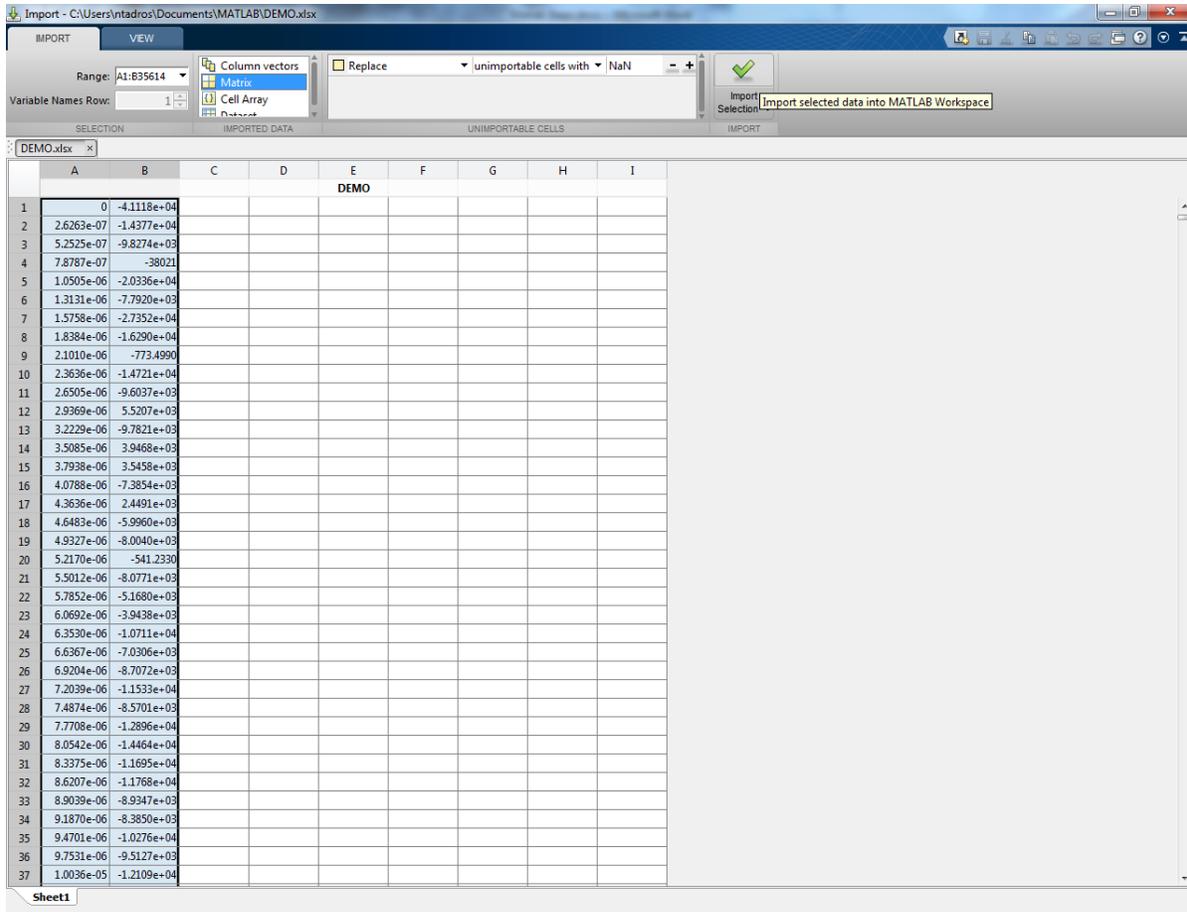
Figure 5.54 Importing spread sheet file to MATLAB environment
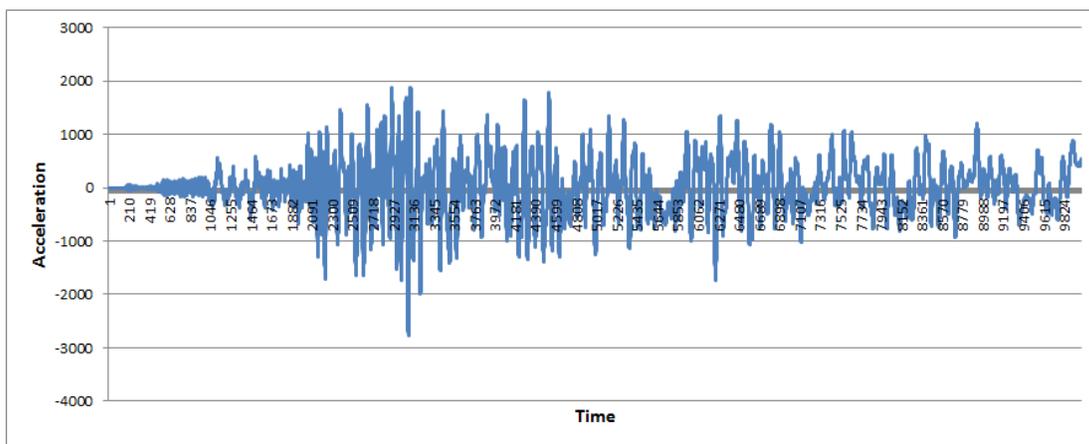
Figure 5.55 Typical graph showing Time vs. Acceleration for an ambient vibration

(Time Domain)

84

3. Rename the imported spread sheet file in the "Workspace" tree to match the input file name in the *Extracting Free Vibration Response from a Response of Structure to Random Excitation* Matlab Code

4. Run the *Extracting Free Vibration Response from a Response of Structure to Random Excitation* Matlab Code, as shown in Figure 5.56. Get the graph showing the free vibration for each case with Time on the horizontal axis vs. Acceleration on the vertical axis. See Figure 5.57 for an illustrative example
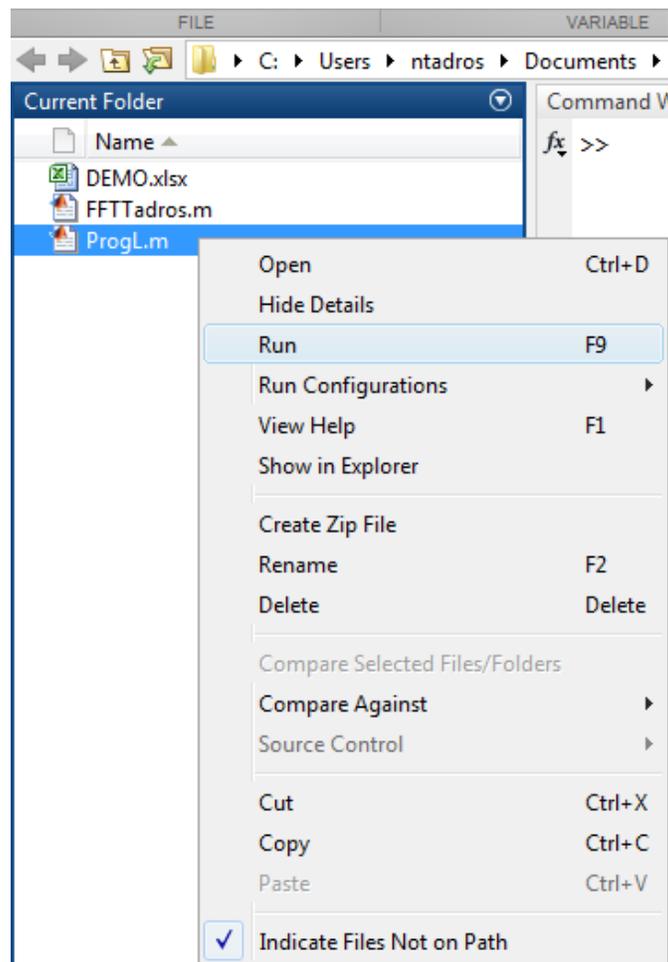
Figure 5.56 Running *Extracting Free Vibration Response from a Response of Structure to Random Excitation* **Matlab Code**

85
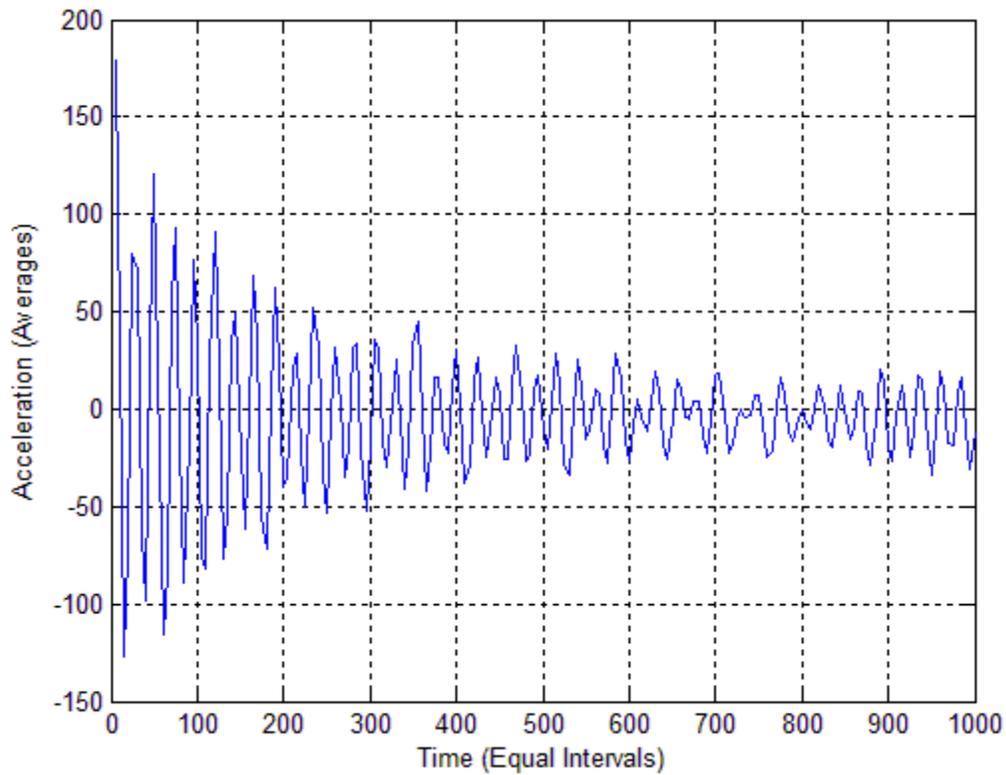
Figure 5.57 Free Vibration response extracted from response of structure to random excitation using the Matlab Code for a signal shown in Figure 5.55

5. Run the *Fast Fourier Transformation* Matlab Code to obtain the frequency domain for each case, as shown in Figure 5.58 and Figure 5.59.
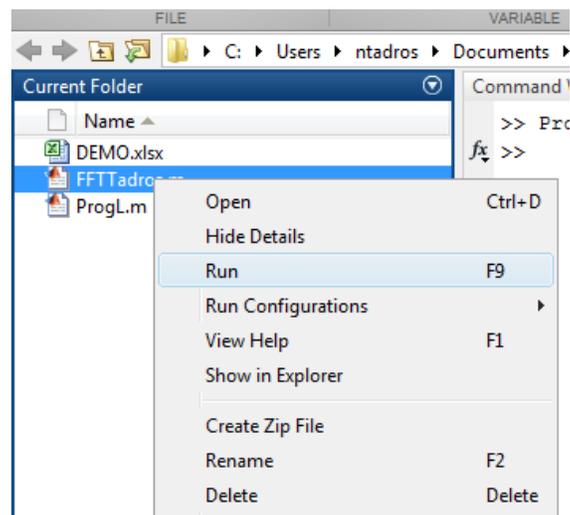
Figure 5.58 Running the *Fast Fourier Transformation* Matlab Code
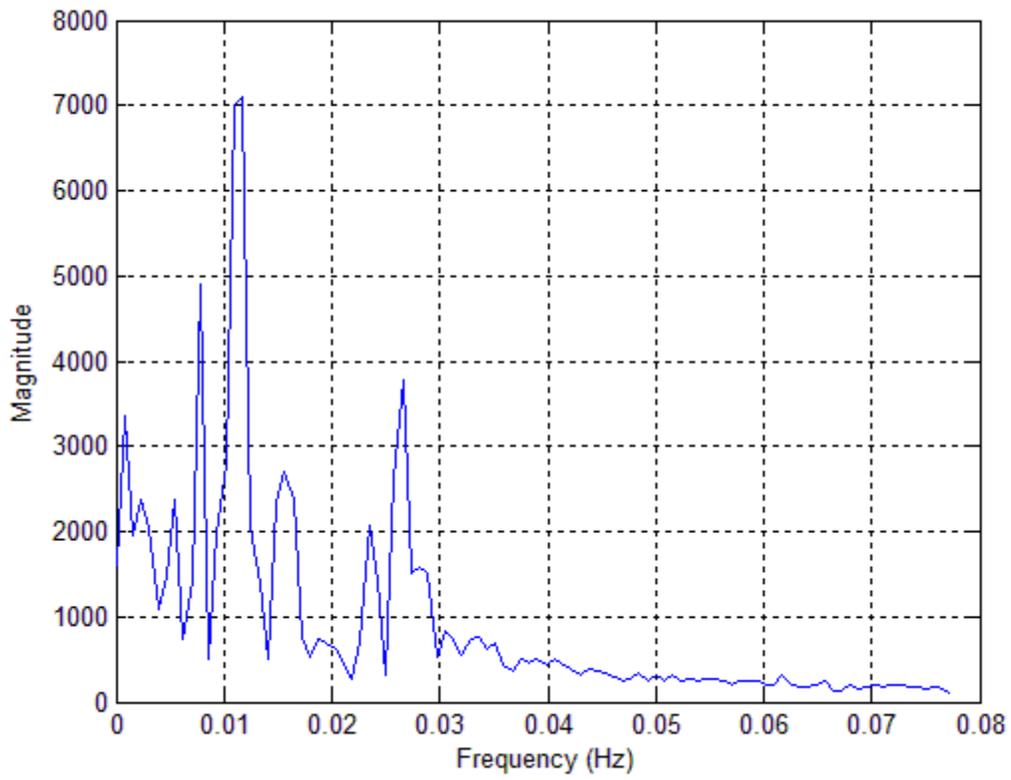
Figure 5.59 Free vibration response in frequency domain for the signal shown in Figure 5.57

Chapter 6 - EXPERIMENTS WITH PHYSICAL AND NUMERICAL MODELS

A physical model consisting of a 20 mm × 20 mm × 1670 mm long steel square tube with 1.65 mm average thickness and four supports (two hinges and two rollers) was used to validate the used approach described in Chapter 4. The beam was tested under difference supports conditions varying from a single- to three-span continuous configuration. Random excitation was applied to the beam, and the dynamic response was measured by a MicroStrain® accelerometer placed at various locations on the span.

A numerical model was constructed in ABAQUS (Abaqus/CAE 6.10-2, © Dassault Systems, 2010) and the dynamic response (ambient vibration) was obtained from the finite element model subjected to similar excitation as the physical model.

The dynamic response (acceleration as a function of time) resulting from the numerical model was correlated and compared to the one from the physical model. Matlab codes described in Chapter 5 were used for both responses (the first code was to extract the free vibration of the system from the ambient vibration, and the other code was used to transfer the free vibration from the time domain to the frequency domain). Comparisons were made between the different span/support configurations indicating a change in frequency and other dynamic properties of the structure.

6.1 Test Set Up

Figure 6.1 shows the general configuration of the physical model and MicroStrain® accelerometer, base station and laptop to record the random vibration signal for one of the test cases. In order to obtain several intentional stiffness changes, seven structure configurations were implemented with two accelerometer positions achieving a total of fourteen (14) cases.
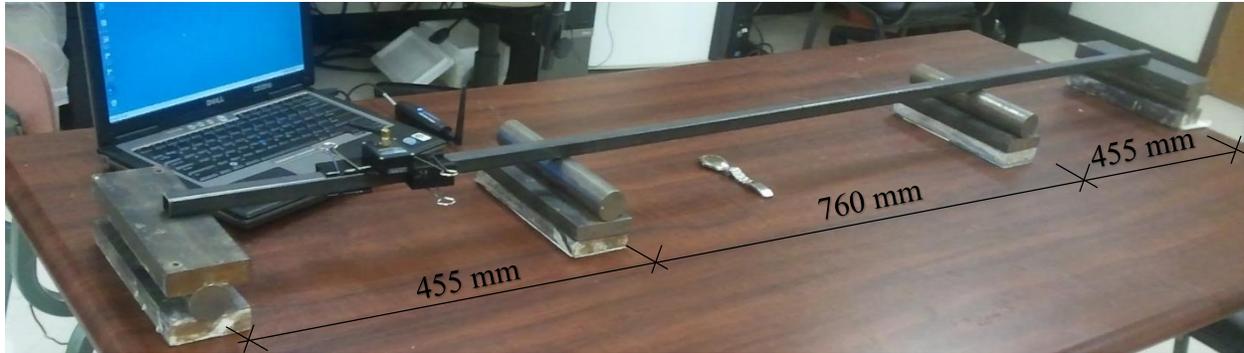
Figure 6.1 General configuration for the physical model and MicroStrain® accelerometer, base station and a laptop to record the random vibration signal

Figure 6.2 shows the numerical model in ABAQUS (Abaqus/CAE 6.10-2, © Dassault Systems, 2010) that corresponds to the physical model for the given case.

Figure 6.2 General configuration for the numerical model in ABAQUS/CAE 6.10 that corresponds to the physical model

For each of the fourteen cases, two different actual random loads were applied on the physical model, and a third simulated random load was applied to the corresponding numerical model, resulting in a total of forty two (42) test combinations.

A typical vibration signal obtained from one of the physical model experiments using MicroStrain® accelerometer is shown in Figure 6.3
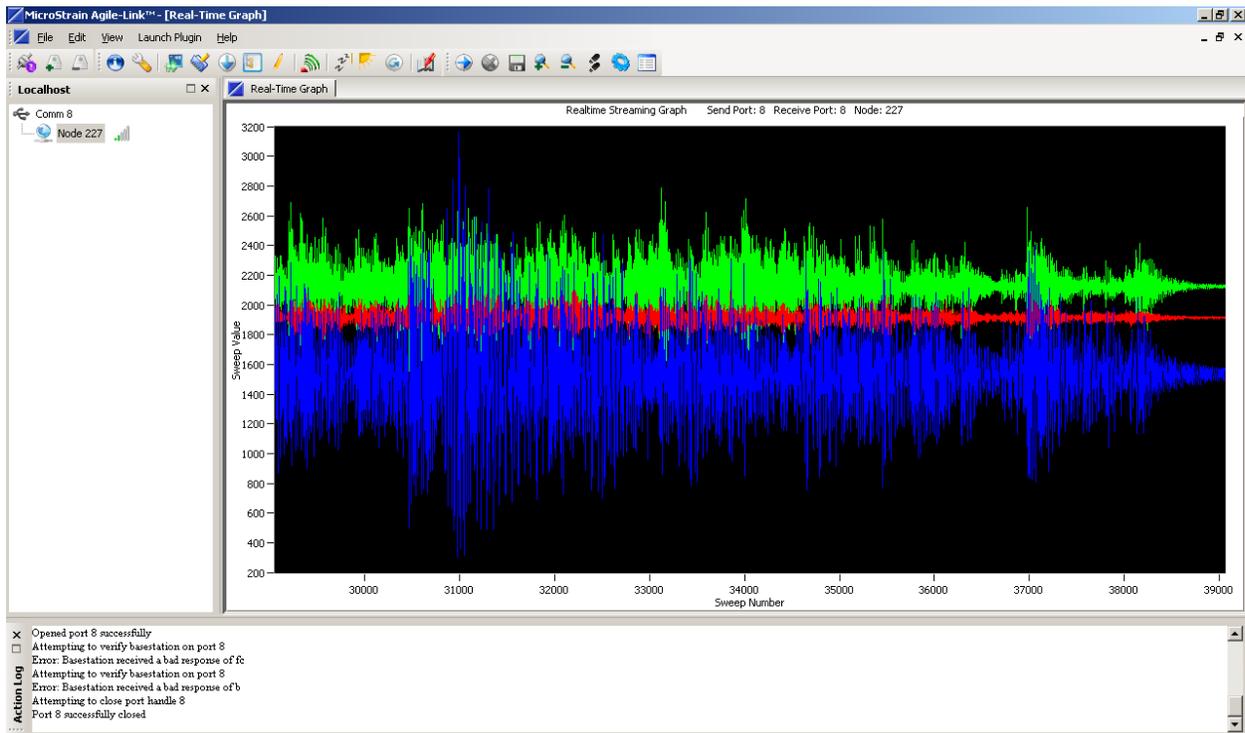
Figure 6.3 An ambient vibration signal from one of the physical model experiments

A typical vibration signal obtained from one of the numerical model experiments in Abaqus is shown in Figure 6.4
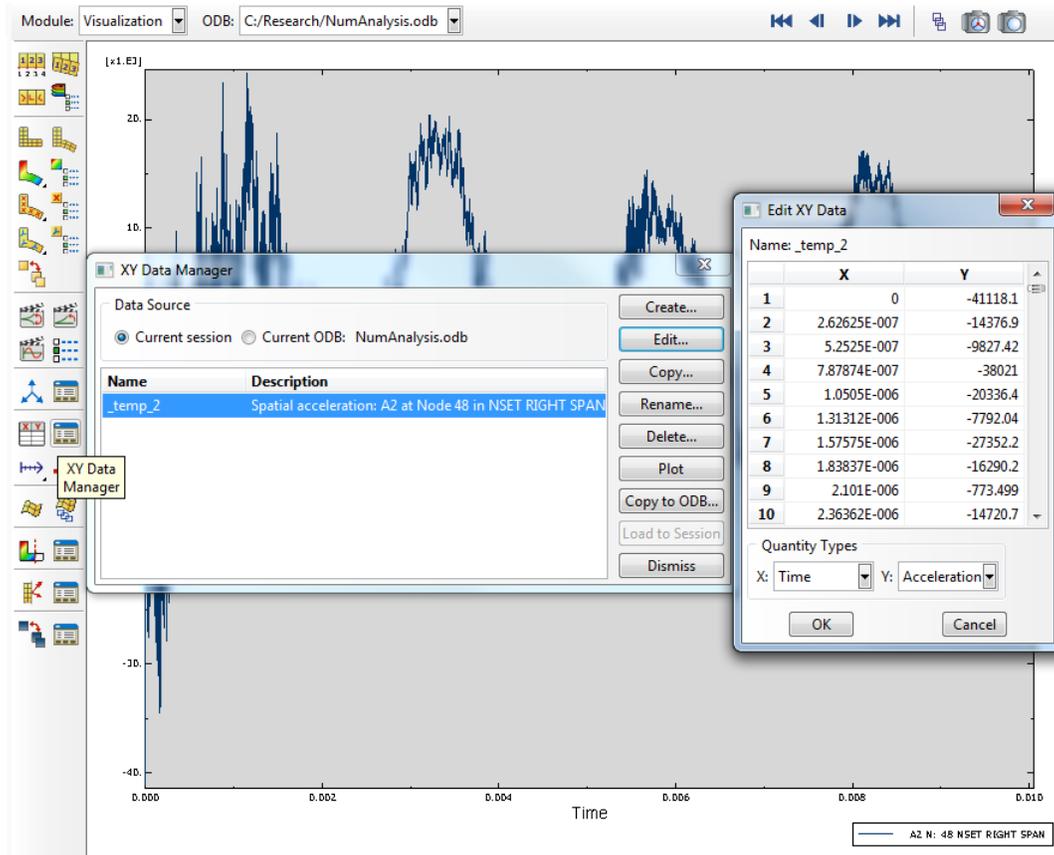
Figure 6.4 An ambient vibration signal from one of the numerical model experiments

A label with the general form of *xPMyC* was created for each case, where:

x refers to the accelerometer placement, 1 is when the accelerometer is placed in the outer span and 2 is for middle span. For consistency, the accelerometer was always positioned at the center of the span in which it is placed.

PM refers to the Physical Model.

y refers to the configuration arrangement, (seven configurations are explained below).

C refers to continuity of the beam.

Table 6.1 and Table 6.2 indicate schematic sketches for individual cases. Table 6.1 shows the seven configurations when the accelerometer was located on the outer span, and Table 6.2 shows these configurations when the accelerometer was located on the middle span.

Table 6.1 Case identification for the different configurations when the accelerometer was placed in the outer span

Case no.	Label	Configuration
Case 1	1PM1C	
Case 2	1PM2C	
Case 3	1PM3C	
Case 4	1PM4C	
Case 5	1PM5C	
Case 6	1PM6C	
Case 7	1PM7C	

Table 6.2 Case identification for the different configurations when the accelerometer was placed in the middle span

Case no.	Label	Configuration
Case 8	2PM1C	
Case 9	2PM2C	
Case 10	2PM3C	
Case 11	2PM4C	
Case 12	2PM5C	
Case 13	2PM6C	
Case 14	2PM7C	
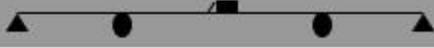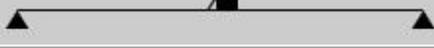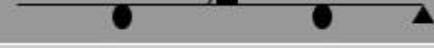
6.2 Test Results

Tables 6.3 to 6.16 summarize physical and numerical experiments, (as presented in Tables 6.1 and 6.2) and corresponding test results. These tables consist of two different actual random loads applied on the physical model (shown in the tables as "Physical Model 1ˢᵗ Trial" and "Physical Model 2ⁿᵈ Trial"), and a third simulated random load applied to the corresponding numerical model (shown in the tables as "Numerical Model"), resulting in a total of forty two (42) test combinations. The peak magnitude frequency (in Hz) for each trial is determined. This value is shown in the lower right corner of each trial test result table. These peak frequencies are used as the basis for comparison of the different test results.

Test configurations and results are presented in the following pages. Tables 6.3 to 6.9 are for cases 1 to 7, when the accelerometer was placed in the outer span. Tables 6.10 to 6.16 are for cases 8 to 14, when the accelerometer was placed in the middle span.

Table 6.3 Case 1 (1PM1C) Test Results

		Case 1 (1PM1C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0280 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0250 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0230 Hz

Table 6.4 Case 2 (1PM2C) Test Results

		Case 2 (1PM2C)	
Physical Model 1ˢᵗ Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0085 Hz
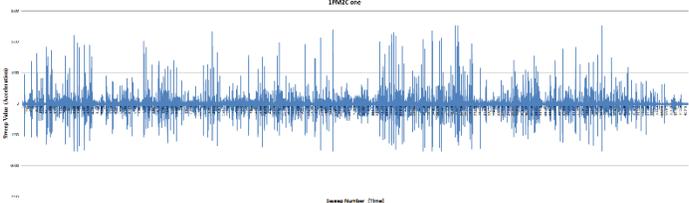
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0080 Hz
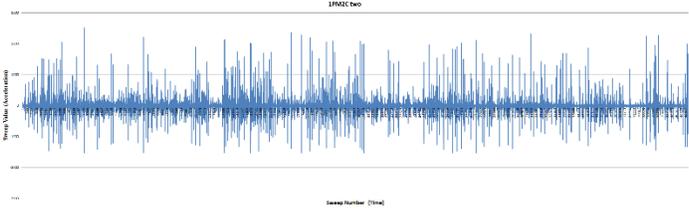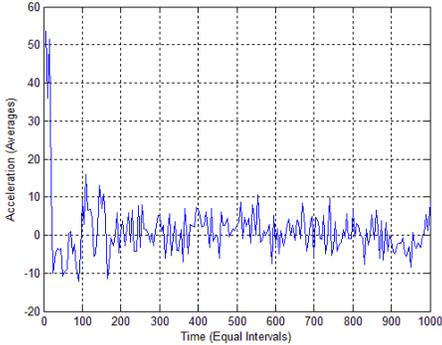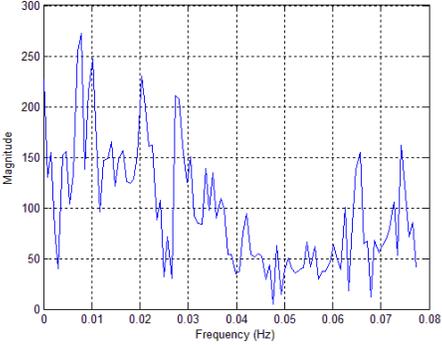
Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0040 Hz

Table 6.5 Case 3 (1PM3C) Test Results

		Case 3 (1PM3C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0765 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0690 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0750 Hz

Table 6.6 Case 4 (1PM4C) Test Results

		Case 4 (1PM4C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0330 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0330 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0420 Hz

Table 6.7 Case 5 (1PM5C) Test Results

		Case 5 (1PM5C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0100 Hz
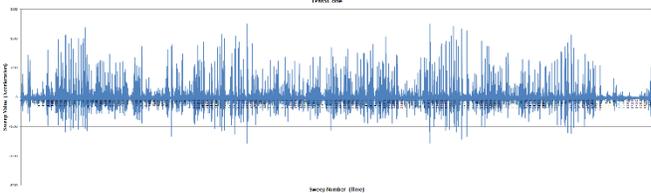
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0095 Hz

	Random Vibration		
Numerical Model	Free Vibration		
	Frequency Domain		0.0085 Hz

Table 6.8 Case 6 (1PM6C) Test Results

		Case 6 (1PM6C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0090 Hz
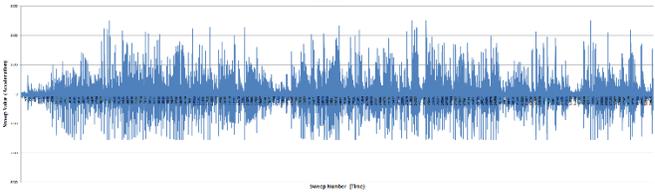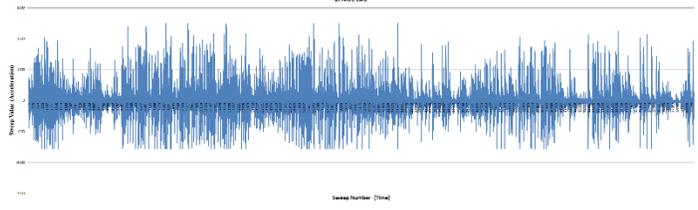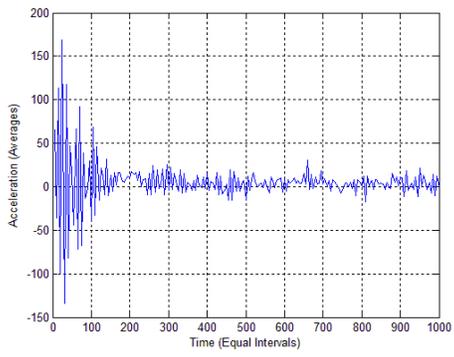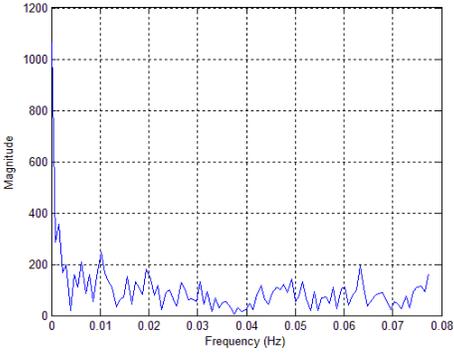
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0100 Hz
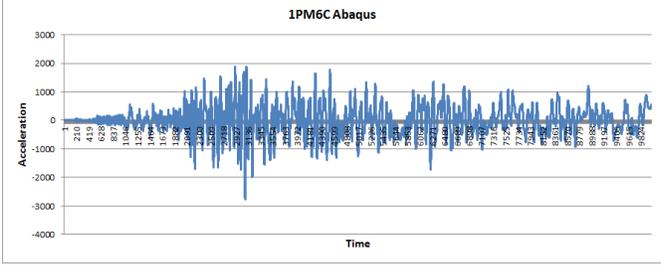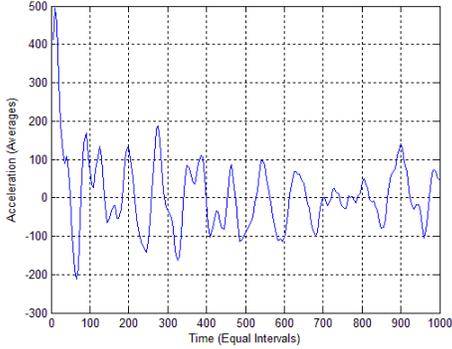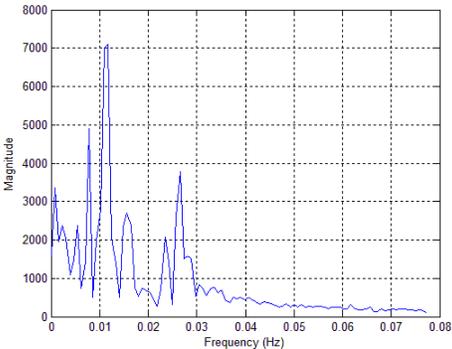
Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0105 Hz

Table 6.9 Case 7 (1PM7C) Test Results

		Case 7 (1PM7C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0675 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0680 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0690 Hz

Table 6.10 Case 8 (2PM1C) Test Results

		Case 8 (2PM1C)	
Physical Model 1ˢᵗ Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0310 Hz
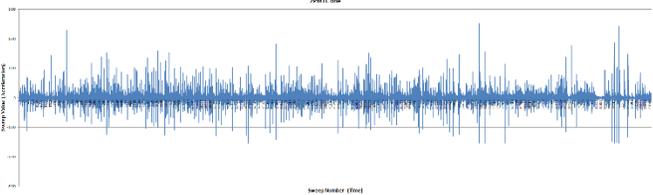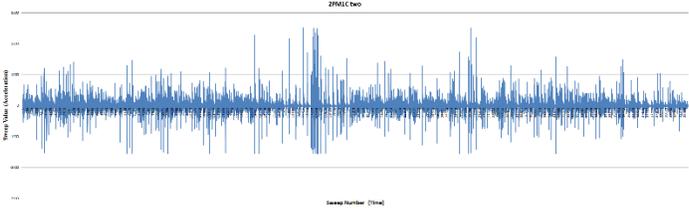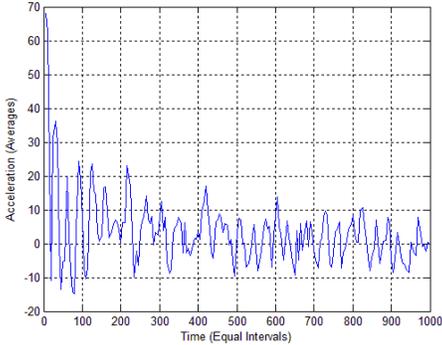
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0310 Hz
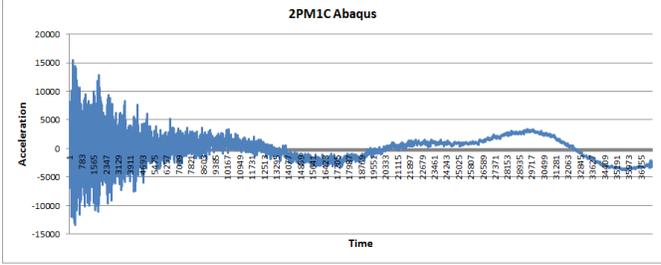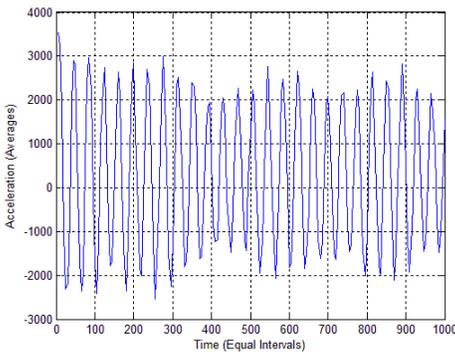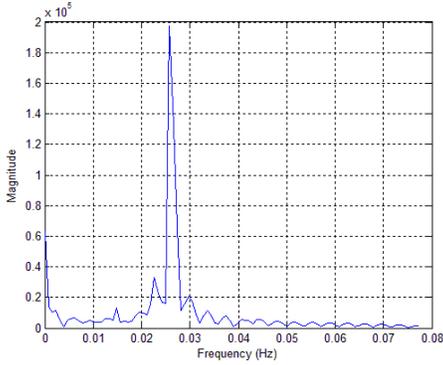
Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0270 Hz

Table 6.11 Case 9 (2PM2C) Test Results

Case 9 (2PM2C)			
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0600 Hz
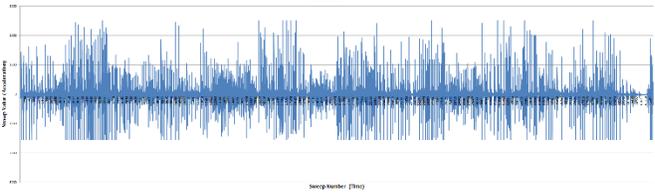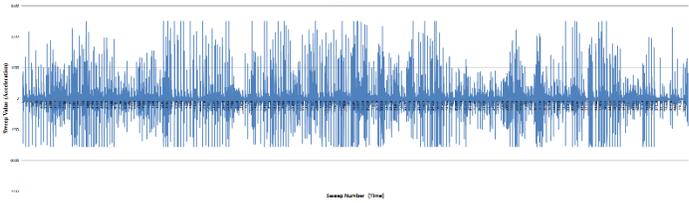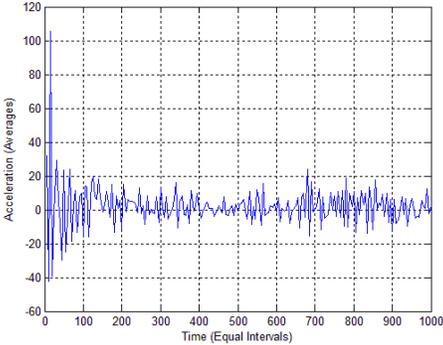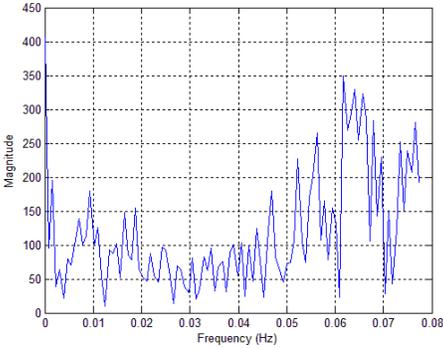
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0610 Hz
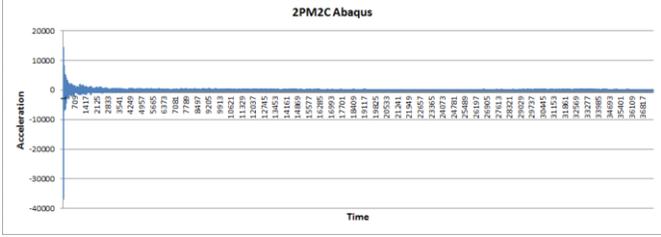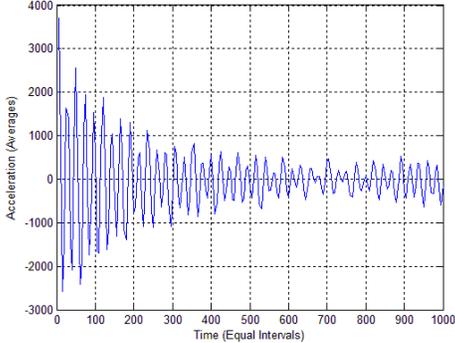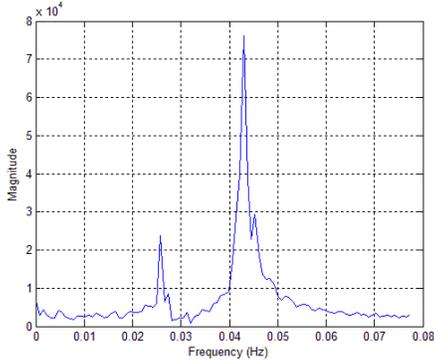
Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0440 Hz

Table 6.12 Case 10 (2PM3C) Test Results

Case 10 (2PM3C)			
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0760 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0770 Hz

	Random Vibration		
Numerical Model	Free Vibration		
	Frequency Domain		0.0440 Hz

Table 6.13 Case 11 (2PM4C) Test Results

Case 11 (2PM4C)			
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0330 Hz

Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0330 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0420 Hz

Table 6.14 Case 12 (2PM5C) Test Results

		Case 12 (2PM5C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0080 Hz

	Random Vibration		
Physical Model 2nd Trial	Free Vibration		
	Frequency Domain		0.0080 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0030 Hz

Table 6.15 Case 13 (2PM6C) Test Results

		Case 13 (2PM6C)	
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0050 Hz
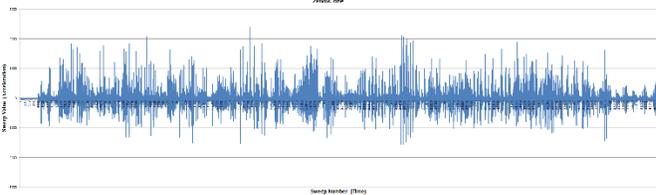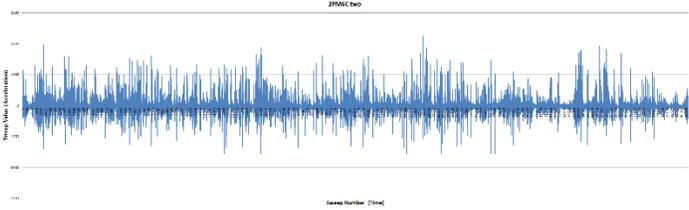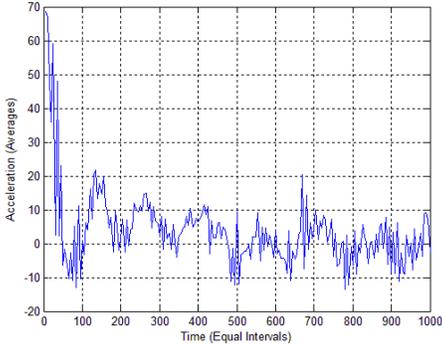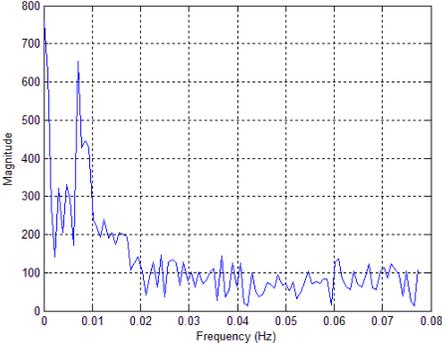
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0060 Hz

Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0030 Hz

Table 6.16 Case 14 (2PM7C) Test Results

Case 14 (2PM7C)			
Physical Model 1st Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0690 Hz
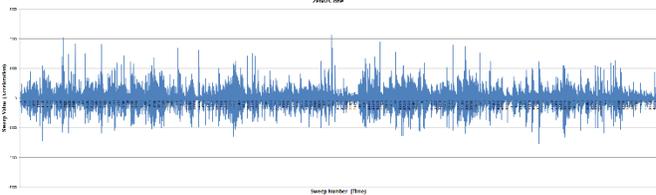
Physical Model 2nd Trial	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0690 Hz
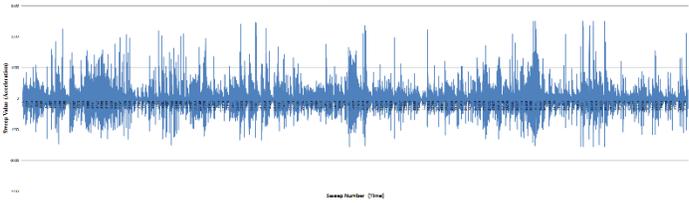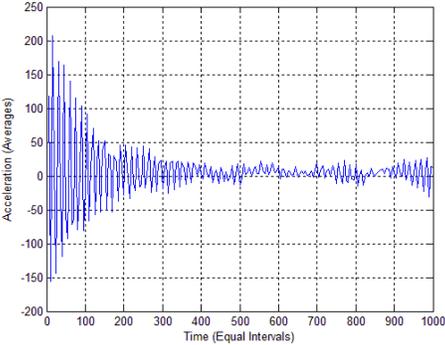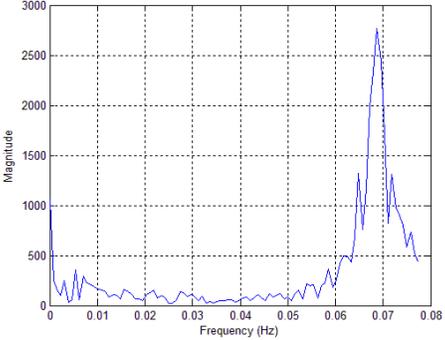
Numerical Model	Random Vibration		
	Free Vibration		
	Frequency Domain		0.0430 Hz

6.3 Results Summary

Table 6.17 and Figure 6.3 summarize all test results based on the peak frequency obtained for each case.

Table 6.17 All cases and trials results

Results Summary				
Case no.	Label	PM1	PM2	NM
Case 1	1PM1C	0.028	0.025	0.023
Case 2	1PM2C	0.0085	0.008	0.004
Case 3	1PM3C	0.0765	0.069	0.075
Case 4	1PM4C	0.033	0.033	0.042
Case 5	1PM5C	0.01	0.0095	0.0085
Case 6	1PM6C	0.009	0.01	0.0105
Case 7	1PM7C	0.0675	0.068	0.069
Case 8	2PM1C	0.031	0.031	0.027
Case 9	2PM2C	0.06	0.061	0.044
Case 10	2PM3C	0.076	0.077	0.044
Case 11	2PM4C	0.033	0.033	0.042
Case 12	2PM5C	0.008	0.008	0.003
Case 13	2PM6C	0.005	0.006	0.003
Case 14	2PM7C	0.069	0.069	0.043
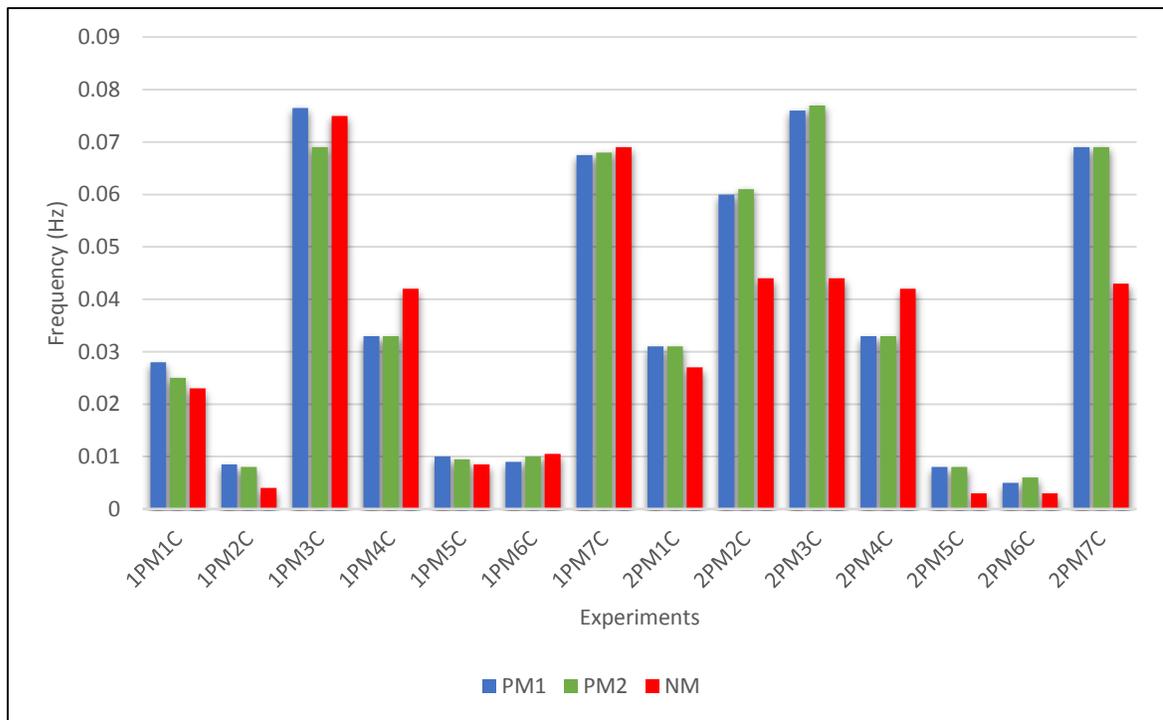
Figure 6.5 All cases and trials results

Chapter 7 - CONCLUSIONS AND SUGGESTED FUTURE RESEARCH

7.1 Conclusions

Based on the obtained test results, the peak magnitude frequency changes when the structure stiffness is altered by choosing different types of structure configurations, as demonstrated when two different actual random loads that were applied on the physical model, and a third simulated random load was applied to the corresponding numerical model. This resulted in a total of forty two test combinations.

For all the fourteen physical model cases (two trials for each case), the peak frequency had nearly the same value for the two trials. For a majority of the fourteen numerical model cases (one trial for each case), the peak frequency had a value close to that of the corresponding physical model case, with the exception of Cases 9, 10, and 14 for which these results were not so close.

In general, any difference in the peak frequency values between the physical and the corresponding numerical models could be due to the ideality of the numerical model compared to the physical model; the physical model had slight thickness variation along the tube cross sections, and the supports were not pure hinge or roller supports as in the numerical model. In addition, the values of material properties used in the numerical model could differ from the actual material properties of the physical model, such as the steel modulus of elasticity and the Poison's ratio.

Based on that, we conclude the following:

❖ The approach used is valid.

❖ Free vibration is embedded in any ambient vibration for the same system.

❖ The free vibration signal extracted from an ambient vibration signal by the approach used in this study is a correct representation of the system free vibration.

❖ Structure frequency and stiffness are dependent characteristics.

❖ Changes in the structure stiffness are reflected by a change in the dominant frequency.

❖ Ambient vibration can be a useful tool to detect structure damage.

7.2 Suggested Future Research

The following are suggested areas for future research:

❖ The physical model used in this research is simple. It had a small number of degrees of freedom and consisted of a steel tube with square cross section. The main goal was to verify the validity of the approach. Further research is needed on structures with a larger number of degrees of freedom and damage locations. This could be obtained by a more complex physical model with more degrees of freedom. In addition to changing the system span/support configurations, a possible further investigation could include saw cuts of various depths in the tube at different locations to simulate various levels of damage and loss of continuity.

❖ With only one accelerometer, it is not possible to identify the location of the symmetrical damage. For two cases in which the damage (removal of support) was at symmetrical locations from the accelerometer position – namely Cases 12 and 13 – the structure peak frequency was nearly identical for both cases. Further research is needed for structures with damage in symmetrical locations relative to the accelerometer position.

❖ In order to obtain accurate results for large physical models and full-size structures, several sensors should be placed at different locations. This will allow for better gathering and correlation of the collected data to obtain the corresponding signals. Further research is needed for the ideal accelerometer positions.

Bibliography

Abaqus/CAE 6.10-2, © Dassault Systems, 2010

Arfken, G. "Development of the Fourier Integral," "Fourier Transforms--Inversion Theorem," and "Fourier Transform of Derivatives." §15.2-15.4 in *Mathematical Methods for Physicists, 3rd ed.* Orlando, FL: Academic Press, pp. 794-810, 1985.

Arfken, G. "Discrete Orthogonality--Discrete Fourier Transform." §14.6 in *Mathematical Methods for Physicists, 3rd ed.* Orlando, FL: Academic Press, pp. 787-792, 1985.

Arfken, G. "Fourier Series." Ch. 14 in *Mathematical Methods for Physicists, 3rd ed.* Orlando, FL: Academic Press, pp. 760-793, 1985.

Arndt, J. "FFT Code and Related Stuff." *http://www.jjj.de/fxt/.*

Bergland, G. D. "A Guided Tour of the Fast Fourier Transform." *IEEE Spectrum* **6**, 41-52, July 1969.

Beyer, W. H. (Ed.). *CRC Standard Mathematical Tables, 28th ed.* Boca Raton, FL: CRC Press, 1987.

Biemans, C., Staszewski, W.J., Boller, C. and Tomlinon, G.R. (2001). "Crack detection in metallic structures using broadband excitation of acousto-ultrasonic." *Journal of Intelligent Material Systems and Structures,* Vol. 12, pp. 589-597.

Bracewell, R. *The Fourier Transform and Its Applications, 3rd ed.* New York: McGraw-Hill, 1999.

Brigham, E. O. *The Fast Fourier Transform and Applications.* Englewood Cliffs, NJ: Prentice Hall, 1988.

Broughton, S.A.; Bryan, K. *Discrete Fourier Analysis and Wavelets: Applications to Signal and Image Processing.* New York: Wiley. p. 72, 2008.

Brown, J. W. and Churchill, R. V. *Fourier Series and Boundary Value Problems, 5th ed.* New York: McGraw-Hill, 1993.

Byerly, W. E. *An Elementary Treatise on Fourier's Series, and Spherical, Cylindrical, and Ellipsoidal Harmonics, with Applications to Problems in Mathematical Physics.* New York: Dover, 1959.

Cooley, J. W. and Tukey, O. W. "An Algorithm for the Machine Calculation of Complex Fourier Series." *Math. Comput.* **19**, 297-301, 1965.

Chopra, Anil K., *Dynamics of Structures,* theory and applications to earthquake engineering, 2nd Ed., Prentice Hall, Upper Saddle River, N.J., 2000, Chapter 10.

Douka, E., Loutridis, S. and Trochidis, A. (2004). "Crack identification in plates using wavelet analysis." *Journal of Sound and Vibration,* Vol. 270, pp. 279-295.

Fertis, Demeter G. *Mechanical and Structural Vibrations.* New York: John Wiley & Sons, 1995.

Heyns, Stephan, "Structural Damage Assessment Through Operational Vibration Measurement", *5th Mini Conference on Vehicle System Dynamics, Identification and Anomalies.* Budapest, Hungary, November 1996.

Kar, C. and Mohanty, A.R. (2006). "Technical note: gearbox health monitoring through multiresolution fourier transformation of vibration and current signals." *Structural Health Monitoring,* Vol. 5, pp. 195-200.

Krantz, S. G. "Fourier Series." §15.1 in *Handbook of Complex Variables.* Boston, MA: Birkhäuser, pp. 195-202, 1999.

Krantz, S. G. "The Fourier Transform." §15.2 in *Handbook of Complex Variables.* Boston, MA: Birkhäuser, pp. 202-212, 1999.

Lee, Y. W., Cheatham, Jr., T. P., and Wiesner, J. B. *The Application of Correlation Functions in the Detection of Small Signals in Noise,* MIT Research Laboratory of Electronics, Technical Report No. 141, 1949.

Liao, G., Liu, S., Shi, T. and Zhang, G. (2004). "Gearbox condition monitoring using self-organizing feature maps." *Proceeding of the Institution of Mechanical Engineering. Part C, Journal of Mechanical Engineering Science,* Vol. 218, pp. 119-130.

Lighthill, M. J. *Introduction to Fourier Analysis and Generalised Functions.* Cambridge, England: Cambridge University Press, 1958.

Mathews, J. and Walker, R. L. *Mathematical Methods of Physics, 2nd ed.* Reading, MA: W. A. Benjamin/Addison-Wesley, 1970.

MATLAB R2013a, The MathWorks, Inc.

Melhem, H. and Kim, H. (2003). "Damage Detection in Concrete by Fourier and Wavelet Analysis." *Journal of Engineering Mechanics, ASCE,* pp. 571

Omenzetter, P. and Brownjohn, J.M.W. (2006). "Application of time series analysis for bridge monitoring." *Smart Material and Structures,* Vol. 15, pp. 129-138.

Omenzetter, P., Brownjohn, J.M.W. and Moyo, P. (2004). "Identification of unusual events in multi-channel bridge monitoring data." Mechanical Sysytems and Signal Processing, Vol. 18, pp. 409 – 430.

Ozturk, H. and Yesilyurt, I. (2008). "Early detection of pitting damage in gears using mesh frequency of scalogram." *Journal of Vibration and Control,* Vol. 14, pp. 469-484.

Poudel, U.P., Fu, G. and Ye, J. (2007). "Wavelet transformation of mode shape difference function for structural damage location identification." *Earthquake Engineering and Structural Dynamics,* Vol. 36, pp. 1089-1107.

Poudel, U.P., Fu, G. and Ye,J. (2005). "Structural damage detection using digital video imaging technique and Wavelet transformation." *Journal of Sound and Vibration,* Vol. 286, pp. 869-895.

Qiao, L., Esmaeily, A. and Melhem, H. (2008). "Structural Damage Detection using Signal Pattern-Recognition." *Key Engineering Material,* Vols. 400-402, pp. 465-470

Ramirez, R. W. *The FFT: Fundamentals and Concepts.* Englewood Cliffs, NJ: Prentice-Hall, 1985.

Ramirez, R. W. *The FFT: Fundamentals and Concepts.* Englewood Cliffs, NJ: Prentice-Hall, 1985.

Ramirez, Robert W. *The FFT: Fundamentals and Concepts.* Englewood Cliffs, NJ: Prentice-Hall, Inc., 1985.

Seibold, S. and Weinert, K. (1996). "A time domain method for the localization of cracks in rotors." *Journal of Sound and Vibration,* Vol. 195, pp. 57-73.

Staszewski, W.J. and Tomlinson, G.R. (1994). "Application of the wavelet transform to fault detection in a spur gear." *Mechanical Systems and Signals Processing,* Vol. 8, No. 3, pp.289-307.

Stoer, J. and Bulirsch, R. *Introduction to Numerical Analysis.* New York: Springer-Verlag, 1980.

Strang, G. "Wavelet Transforms Versus Fourier Transforms." *Bull. Amer. Math. Soc.* **28**, 288-305, 1993.

Trott, M. *The Mathematica GuideBook for Programming.* New York: Springer-Verlag, 2004. http://www.mathematicaguidebooks.org/.

Walker, J. S. *Fast Fourier Transform, 2nd ed.* Boca Raton, FL: CRC Press, 1996.

Walker, J. S. *Fast Fourier Transforms, 2nd ed.* Boca Raton, FL: CRC Press, 1996.

Wang, Q. and Deng, X.M. (1999). "Damage detection with spatial wavelets." International Journal of Solid and Structures, Vol. 36, pp. 3443-3468.

Wang, W.J. and McFadden. (1995). "Application of orthogonal wavelet to early gear damage detection." *Mechanical Systems and Signal Processing,* Vol. 9, No. 5, pp. 497-507.

Wang, W.J. and McFadden. (1996). "Application of wavelet to gearbox vibration signals for fault detection." *Journal of Sound and Vibration,* Vol. 192, No. 5, pp. 927-939.

Yan, Hao, H.N. and Yam, L.H. (2004). "Vibration-based construction and extraction of structural damage feature index." *International Journal of Solid and Structures,* Vol. 41, No. 2, pp. 6661-6676.

Zhang, Q.W. (2007). "Statistical damage identification for bridge using ambient vibration data." *Computer and Structures,* Vol. 85, pp. 476-485.

References

Askey, R. and Haimo, D. T. "Similarities between Fourier and Power Series." *Amer. Math. Monthly* **103**, 297-304, 1996.

Bathe, K. J., *Finite Element Procedure,* Prentice Hall, Englewood Cliff, N.J., 1996, Chapter 10 and 11.

Bell Laboratories. "Netlib FFTPack." *http://netlib.bell-labs.com/netlib/fftpack/.*

Blackman, R. B. and Tukey, J. W. *The Measurement of Power Spectra, From the Point of View of Communications Engineering.* New York: Dover, 1959.

Blahut, R. E. *Fast Algorithms for Digital Signal Processing.* New York: Addison-Wesley, 1984.

Carslaw, H. S. *Introduction to the Theory of Fourier's Series and Integrals, 3rd ed., rev. and enl.* New York: Dover, 1950.

Chang, C.C. and Chen, L.W. (2005). "Detection of the location and size of cracks in the multiple cracked beam by spatial wavelet based approach." *Mechanical Systems and Signal Processing,* Vol. 19, pp. 139-155.

Chu, E. and George, A. *Inside the FFT Black Box: Serial and Parallel Fast Fourier Transform Algorithms.* Boca Raton, FL: CRC Press, 2000.

Crandall, R.; Jones, E.; Klivington, J.; and Kramer, D. "Gigaelement FFTs on Apple G5 Clusters." 27 Aug 04.*http://images.apple.com/acg/pdf/20040827_GigaFFT.pdf.*

Crandall, S. H., and McCalley, R. B., Jr., "Matrix Methods of Analysis", Chapter 28 in *Shock and Vibration Handbook* (ed. C. M. Harris), McGraw-Hill, New York, 1988.

Davis, H. F. *Fourier Series and Orthogonal Functions.* New York: Dover, 1963.

Ding Y.L., and Li A.Q. (2007). "Structural health monitoring of long span suspension bridges using wavelet packet analysis." *Earthquake Engineering and Engineering Vibration,* Vol. 6, No.3, pp. 289-294.

Duhamel, P. and Vetterli, M. "Fast Fourier Transforms: A Tutorial Review." *Signal Processing* **19**, 259-299, 1990.

Dym, H. and McKean, H. P. *Fourier Series and Integrals.* New York: Academic Press, 1972.

Folland, G. B. *Fourier Analysis and Its Applications.* Pacific Grove, CA: Brooks/Cole, 1992.

Folland, G. B. *Real Analysis: Modern Techniques and their Applications, 2nd ed.* New York: Wiley, 1999.

Groemer, H. *Geometric Applications of Fourier Series and Spherical Harmonics.* New York: Cambridge University Press, 1996.

Humar, J. L., *Dynamics of Structures,* Prentice Hall, Englewood Cliffs, N.J., 1990, Chapter 11.

James, J. F. *A Student's Guide to Fourier Transforms with Applications in Physics and Engineering.* New York: Cambridge University Press, 1995.

Kammler, D. W. *A First Course in Fourier Analysis.* Upper Saddle River, NJ: Prentice Hall, 2000.

Körner, T. W. *Exercises for Fourier Analysis.* New York: Cambridge University Press, 1993.

Körner, T. W. *Fourier Analysis.* Cambridge, England: Cambridge University Press, 1988.

Körner, T. W. *Fourier Analysis.* Cambridge, England: Cambridge University Press, 1988.

Li, B., Goddu, G. and Chow, M.Y. (1998). "Detection of common motor bearing faults using frequency-domain vibration signals and a neural network based approach." *Proceedings of the 16th American Control Conference,* Philadelphia, PA, June, 1998.

Lipson, J. D. *Elements of Algebra and Algebraic Computing.* Reading, MA: Addison-Wesley, 1981.

Merriam, J.L., and Kraige L.G. *Engineering Mechanics – Dynamics 4th Ed.* New York: John Wiley & Sons, 1997.

Morrison, N. *Introduction to Fourier Analysis.* New York: Wiley, 1994.

Morrison, N. *Introduction to Fourier Analysis.* New York: Wiley, 1994.

Morse, P. M. and Feshbach, H. "Fourier Transforms." §4.8 in *Methods of Theoretical Physics, Part I.* New York: McGraw-Hill, pp. 453-471, 1953.

Nussbaumer, H. J. *Fast Fourier Transform and Convolution Algorithms, 2nd ed.* New York: Springer-Verlag, 1982.

Oberhettinger, F. *Fourier Transforms of Distributions and Their Inverses: A Collection of Tables.* New York: Academic Press, 1973.

Papoulis, A. *The Fourier Integral and its Applications.* New York: McGraw-Hill, 1962.

Papoulis, A. *The Fourier Integral and Its Applications.* New York: McGraw-Hill, 1962.

Parlett, B. N., *The Symmetric Eigenvalue Problem,* Prentice Hall, Englewood Cliffs, N.J., 1980.

Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. "Fast Fourier Transform." Ch. 12 in *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, pp. 490-529, 1992.

Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. "Fourier Transform of Discretely Sampled Data." §12.1 in *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, pp. 494-498, 1989.

Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, 1989.

Sansone, G. "Expansions in Fourier Series." Ch. 2 in *Orthogonal Functions, rev. English ed.* New York: Dover, pp. 39-168, 1991.

Sansone, G. "The Fourier Transform." §2.13 in *Orthogonal Functions, rev. English ed.* New York: Dover, pp. 158-168, 1991.

Sneddon, I. N. *Fourier Transforms.* New York: Dover, 1995.

Sogge, C. D. *Fourier Integrals in Classical Analysis.* New York: Cambridge University Press, 1993.

Spiegel, M. R. *Theory and Problems of Fourier Analysis with Applications to Boundary Value Problems.* New York: McGraw-Hill, 1974.

Stein, E. M. and Weiss, G. L. *Introduction to Fourier Analysis on Euclidean Spaces.* Princeton, NJ: Princeton University Press, 1971.

Strichartz, R. *Fourier Transforms and Distribution Theory.* Boca Raton, FL: CRC Press, 1993.

Titchmarsh, E. C. *Introduction to the Theory of Fourier Integrals, 3rd ed.* Oxford, England: Clarendon Press, 1948.

Tolstov, G. P. *Fourier Series.* New York: Dover, 1976.

Van Loan, C. *Computational Frameworks for the Fast Fourier Transform.* Philadelphia, PA: SIAM, 1992.

Weisstein, E. W. "Books about Fourier Transforms." http://www.ericweisstein.com/encyclopedias/books/FourierTransforms.html.

Weisstein, Eric W. "Discrete Fourier Transform." From *MathWorld*--A Wolfram Web Resource. http://mathworld.wolfram.com/DiscreteFourierTransform.html

Weisstein, Eric W. "Fast Fourier Transform." From *MathWorld*--A Wolfram Web Resource. http://mathworld.wolfram.com/FastFourierTransform.html

Weisstein, Eric W. "Fourier Series." From *MathWorld*--A Wolfram Web Resource. http://mathworld.wolfram.com/FourierSeries.html

Weisstein, Eric W. "Fourier Transform." From *MathWorld*--A Wolfram Web Resource. http://mathworld.wolfram.com/FourierTransform.html

Whittaker, E. T. and Robinson, G. "Practical Fourier Analysis." Ch. 10 in *The Calculus of Observations: A Treatise on Numerical Mathematics, 4th ed.* New York: Dover, pp. 260-284, 1967

Zadeh, L. A., *Journal of the Society for Industrial and Applied Mathematics 1: 35–51,* Theory of Filtering, 1953.

Appendix A - Extraction of Free Vibration Matlab Code

Extracting Free Vibration Response from a Response of Structure to Random Excitation

Matlab Code is listed below:

```matlab
%INITIAL DATA PROVIDED BY USER
realTlimit=1000; %in sec., the maximum period of the expected free vibration
frequency
theTimeStep=5; %in sec., real time step we choose to be added to previous
time
timeIncrement=1; %in sec., the time increment in the data file (time starts
from 0 and increases by this increment to the end
a=(1/10)*max(D(:,2));%the initial condition set as 1/10 of max
%END OF INITIAL DATA

mystep=theTimeStep/timeIncrement; %number of increments within the time step
we chose
Tlimit=realTlimit/timeIncrement;%number of increments within the time limit
TM=mystep;%this is each step (in number of increments)


myData(1:Tlimit/mystep,1:2)=0.0;

%size(myData)
%disp('beginning')

while TM<=Tlimit

    %disp('first while')

m=1;%this is number of row in the data, used to read the corresponding second
column that is value
n=0;%this is number of values found at the given TM for the given initial
condition a
data=0;%summation of values with TM and initial a
Flag=false;%checking if we have already set the AD flag
AD=false;%flag to know if ascending or descending at the inital point of a

while m+TM<length(D(:,1))
    %disp(' cycle second while')
if Flag==false
%disp('inside Falg faklse')
    if D(m,2)==a
        data=data+D(m+TM,2);
        n=n+1;
        %disp('Flag False')
            if D(m+1,2)>D(m,2)
                AD=true;
            else
                AD=false;
            end
```

```matlab
            Flag=true;
    elseif D(m+1,2)==a
            data=D(m+1+TM,2);
            n=n+1;
            %disp('Flag False')
                if D(m+1,2)>D(m,2)
                    AD=true;
                else
                    AD=false;
                end
                Flag=true;
    elseif D(m,2)<a && a<D(m+1,2)
                    ratio=(a-D(m,2))/(D(m+1,2)-D(m,2));
                    data=data+D(m+TM,2)+ratio*(D(m+1+TM,2)-D(m+TM,2));
                    n=n+1;
                        %if D(m+1,2)>D(m,2)
                            AD=true;
                        %else
                        %    AD=false;
                        %end
                        Flag=true;
    elseif D(m,2)>a && a>D(m+1,2)
                    ratio=(a-D(m+1,2))/(D(m+1,2)-D(m,2));
                    data=data+D(m+TM,2)+ratio*(D(m+1+TM,2)-D(m+TM,2));
                    n=n+1;
                        %if D(m+1,2)>D(m,2)
                        %    AD=true;
                        % else
                            AD=false;
                        % end
                         Flag=true;
    end

end
if Flag==true
    %disp('inside TRUE Flag')
        if AD==true
            if D(m+1)>D(m)
                if D(m,2)==a
                    data=data+D(m+TM,2);
                    n=n+1;
                    %disp('AD True')
                    elseif D(m+1,2)==a
                            data=data+D(m+1+TM,2);
                            n=n+1;
                            %disp('AD True')
                    elseif (D(m,2)<a) && (a<D(m+1,2))
                            ratio=(a-D(m,2))/(D(m+1,2)-D(m,2));
                            data=data+D(m+TM,2)+ratio*(D(m+1+TM,2)-
D(m+TM,2));
                            n=n+1;
                            %disp('AD True')
                end
            end
        elseif AD==false
                        if D(m+1)<D(m)
                            if D(m,2)==a
```

149

```matlab
                                                data=data+D(m+TM,2);
                                                n=n+1;
                                                %disp('AD False')
                                    elseif D(m+1,2)==a
                                                data=data+D(m+1+TM,2);
                                                n=n+1;
                                                %disp('AD False')

                            elseif D(m+1,2)<a && a<D(m,2)
                                                ratio=(a-D(m,2))/(D(m+1,2)-
D(m,2));

data=data+D(m+TM,2)+ratio*(D(m+1+TM,2)-D(m+TM,2));
                                                n=n+1;
                                                %disp('AD False')
                                    end
                                end
            end
end
 m=m+1;


end
%data
%n
%TM
%Tlimit
            datapoint=data/n;
            myData(TM/mystep,1)=TM*timeIncrement;
            myData(TM/mystep,2)=datapoint;
            TM=TM+mystep;
%TM
end
%size(myData)
%myData
  plot(myData(:,1),myData(:,2));


            %TM=nstep*step
xlabel('Time (Equal Intervals)')
ylabel('Acceleration (Averages)')
grid
```

150

Appendix B - FFT Matlab Code

Fast Fourier Transformation Matlab Code is listed below:

```matlab
% Fast Fourier Transform Spectrum.m
%====================
% Nader N Tadros
% Department of Civil Engineering
% Kansas State University
% -------------------------
% This MATLAB code trasfers time domain of a signal to frequency domain


%-------------------------------------------------------------------


% Here is the PROGRAM in MATLAB


disp(' *** KSU Civil Engineerin Department,Nader N Tadros,P.E. *** ')
disp('-------------------------------------------------------')
disp('This MATLAB code trasfers time domain of a signal to frequency domain
by using FFT')
disp('================================================')


myData = myData;                % The file name in Workspace
t=myData(:,1);
s=myData(:,2);
Ts=mean(diff(t));
fs=1/Ts;
L=length(s);
NFFT=2^nextpow2(L);        % Next power of 2 from length of y
M=L+1;
y=fft(s,NFFT);
mag=abs(y);
f=fs/2*linspace(0,1,NFFT/2+1);
r=real(y);
a=r.^2;
i=imag(y);
b=i.^2;
c=a+b;
Energy=sum(c(:))          % Energy of signal

figure;
plot(f(1:M/2),mag(1:M/2));
xlabel('Frequency (Hz)')
ylabel('Magnitude')
grid
fid=fopen('mag_result.txt','w');
fprintf(fid,'%6.3f\r\n',mag(1:M/2));
status=fclose(fid);
```