

**INVERSE PROBLEMS IN STRUCTURAL DAMAGE  
IDENTIFICATION, STRUCTURAL OPTIMIZATION, AND  
OPTICAL MEDICAL IMAGING USING ARTIFICIAL  
NEURAL NETWORKS**

Yong Yook Kim

Dissertation Submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
In partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Aerospace Engineering

Dr. Rakesh K. Kapania, Chair

Dr. Eric R. Johnson

Dr. Raymond H. Plaut

Dr. Joseph A. Schetz

Dr. Elaine P. Scott

January 23, 2004  
Blacksburg, Virginia

Keywords: inverse problem, neural networks, finite element method, optical  
medical imaging, damage identification, structural health monitoring,  
principal component analysis, orthogonal arrays, modal method.

Copyright 2004, Yong Yook Kim

# **Inverse Problems in Structural Damage Identification, Structural Optimization, and Optical Medical Imaging Using Artificial Neural Networks**

Yong Yook Kim

(ABSTRACT)

The objective of this work was to employ artificial neural networks (NN) to solve inverse problems in different engineering fields, overcoming various obstacles in applying NN to different problems and benefiting from the experience of solving different types of inverse problems. The inverse problems investigated are: 1) damage detection in structures, 2) detection of an anomaly in a light-diffusive medium, such as human tissue using optical imaging, 3) structural optimization of fiber optic sensor design. All of these problems require solving highly complex inverse problems and the treatments benefit from employing neural networks which have strength in generalization and pattern recognition. Moreover, the neural networks for the three problems are similar, and a method found suitable for solving one type of problem can be applied for solving other types of problems.

Solution of inverse problems using neural networks consists of two parts. The first is repeatedly solving the direct problem, obtaining the response of a system for known parameters and constructing the set of the solutions to be used as training sets for NN. The next step is training neural networks so that the trained neural networks can produce a set of parameters of interest for the response of the system. Mainly feed-forward backpropagation NN were used in this work.

One of the obstacles in applying artificial neural networks is the need for solving the direct problem repeatedly and generating a large enough number of training sets. To reduce the time required in solving the direct problems of structural dynamics and photon transport in opaque tissue, the finite element method was used. To solve transient problems, which include some of the problems addressed here, and are computationally intensive, the modal superposition and the modal acceleration methods were employed. The need for generating a large enough number of training sets required by NN was fulfilled by automatically generating the training sets using a script program in the MATLAB environment. This program automatically generated finite element models with different parameters, and the program also included scripts that combined the whole solution processes in different engineering packages for the direct problem and the inverse problem using neural networks.

Another obstacle in applying artificial neural networks in solving inverse problems is that the dimension and the size of the training sets required for the NN can be too large to use NN effectively with the available computational resources. To overcome this obstacle, Principal Component Analysis is used to reduce the dimension of the inputs for the NN without excessively impairing the integrity of the data. Orthogonal Arrays were also used to select a smaller number of training sets that can efficiently represent the given system.

## ACKNOWLEDGEMENTS

I am extremely grateful to my advisor Dr. Rakesh K. Kapania for giving me a valuable opportunity to work with him and to gain broad experience through many exciting projects and for his valuable support and encouraging advice throughout the program. I also want to express my gratefulness to Drs. Eric R. Johnson, Raymond H. Plaut, Joseph A. Schetz, and Elaine P. Scott for serving as committee members and giving me valuable input through the committee meetings or course work.

I have been so fortunate to have my parents, Bok Soo Kim and Myung Ja Jo, and my grandparents, Han Cheol Kim and Soo Nyu Lee. I am deeply grateful for their unchanging love and support.

Thanks are also due to many colleagues in our department throughout the program for their valuable company and encouragement, especially to Anand Natarajan, Vijay Goyal, and Vinay Goyal.

I also want to acknowledge the financial supports from Carilion Biomedical Institute, Roanoke, VA, Virginia Tech's Optical Sciences and Engineering Research Center, and Luna Innovations Inc., Blacksburg, VA. I also want to thank Dr. Owen Hughes for the financial support and considerations at the final stage of my program.

Finally, I want to express my deep thanks to my wife, Eun Young Jeong, for her precious love, encouragement, and sacrifice, and my son, Jin Woo, for his everlasting smile and good health.

# TABLES OF CONTENTS

(ABSTRACT).....	ii
ACKNOWLEDGEMENTS.....	iv
TABLES OF CONTENTS .....	v
LIST OF FIGURES .....	vii
LIST OF TABLES.....	xiii
Chapter 1 INTRODUCTION.....	1
1.1 Objectives .....	1
1.2 Three Inverse Problems.....	2
1.2.1 Damage Identification Problem .....	2
1.2.2 Optical Medical Imaging Problem.....	3
1.2.3 Structural Optimization Problem .....	5
1.3 Obstacles in Applying Neural Networks for Inverse Problems .....	6
1.4 Dissertation Overview .....	7
Chapter 2 LITERATURE REVIEW.....	8
2.1 Artificial Neural Networks .....	8
2.1.1 Multi-Layer Feed-Forward Backpropagation Neural Network .....	8
2.2 Damage Identification Using Artificial Neural Networks.....	12
2.3 Optical Imaging Using Near Infrared Light .....	19
2.4 Application of NNs in Engineering Optimization Problems.....	22
2.5 Principal Component Analysis .....	24
2.6 Orthogonal Arrays .....	27
Chapter 3 DAMAGE IDENTIFICATION .....	29
3.1 Damage Identification in a Beam .....	29
3.1.1 Direct Problem .....	29
3.1.2 Inverse Problems.....	30
3.2 Damage Identification in a Plate .....	37
3.2.1 Formulation of the Problem.....	37
3.2.2 Modal Method.....	38

3.2.3	Direct Problem .....	40
3.2.4	Inverse Problem .....	41
Chapter 4	OPTICAL IMAGING PROBLEM .....	46
4.1	Direct Problem.....	46
4.1.1	Development of Finite Element Code.....	48
4.1.2	Characteristics of the Solution of the Direct Problem .....	53
4.1.3	Validation of Finite Element Code .....	55
4.2	Inverse Problems .....	57
4.2.1	Identification of the Depth and the Orientation of an Anomaly .....	58
4.2.2	Identification of the Existence of an Anomaly .....	63
4.2.3	Identification of the Location and the Size .....	65
4.2.4	Identification of the Property of An Element .....	66
4.2.5	Identification with One-Sided Information.....	74
Chapter 5	STRUCTURAL OPTIMIZATION OF A FIBER OPTIC SENSOR.....	77
5.1	Direct Problem.....	77
5.1.1	Finite Element Modeling .....	77
5.1.2	Normal Mode Analysis.....	78
5.1.3	Transient Response Analysis Using Direct Method.....	80
5.1.4	Transient Analysis Using Modal Methods .....	86
5.2	Inverse Problem.....	91
5.2.1	Parametric Study Using NNs .....	91
5.2.2	Optimization Using Gradient Method and Relaxed Constraints .....	96
Chapter 6	SUMMARY AND CONCLUSIONS.....	101
6.1	Summary of Obstacles and Solutions.....	101
6.2	Conclusions .....	102
	BIBLIOGRAPHY.....	105
	APPENDIX.....	114
A.	Three-Dimensional Solid Finite Element Analysis of a Sting Balance.....	114
B.	Calculation of Photon Transport .....	126

## LIST OF FIGURES

Figure 1: Multil-layer feed-forward backpropagation neural network ( $p$ = inputs, $a$ = hidden neurons, $W$ = weights, $b$ = bias, $f$ = transfer function, $n$ = number of inputs, $r$ = number of hidden neurons, $s$ = number of output neurons, $t$ = outputs). .....	8
Figure 2: The graphs of a piecewise linear transfer function and a sigmoid transfer function. ....	9
Figure 3: Radial-basis neural network ( $p$ = inputs, $W$ = weights, $fr$ = radial basis function, $t$ = outputs). ....	11
Figure 4: Damaged beam model with three damage parameters ( $X$ = location, $L^D$ = extension of damage, $EI^D$ = magnitude of damage). ....	29
Figure 5: The simulation errors of a trained feed-forward NN in the prediction of the location of damage (39 testing points). ....	31
Figure 6: Distribution of testing and training points for a NN to identify two parameters. ....	32
Figure 7: The simulation errors of a trained NN at testing points in identifying two damage-parameters when the first three natural frequencies of the damaged beam are given (841 testing data points, $x$ : 29, $k$ : 29). ....	33
Figure 8: The simulation errors of a trained NN at testing points in identifying three parameters (931 testing points, $x$ : 19, $b$ : 7, $\beta$ : 7). ....	34
Figure 9: Prediction of the location of damage for different degrees of error in the measurement of the difference of frequency between damaged and undamaged beam ( $b = 0.05$ and $\beta = 0.3$ ). ....	35
Figure 10: Prediction of the location of damage for different values of the span and the degree of the damage ( $\varepsilon = 8\%$ ). ....	36
Figure 11: A cantilever plate for damage identification problem ( $E = 62.05$ GPa, $\nu = 0.35$ , $\rho = 2700$ kg/m <sup>3</sup> , thickness 5mm) .....	37
Figure 12: Procedure for obtaining dynamic response and training and testing NNs. ....	39
Figure 13: The time history of the force applied to the plate. ....	40
Figure 14: Location of the displacement reading and the location of the force applied. ....	41

Figure 15: Relative simulation error of a NN trained with FFT of the transient displacement vector at each testing points placed midway between training data points in predicting the location of the damage. (Input Matrix = 1600×200, Time required for training = 1 hour 30 minutes) ..... 42

Figure 16: Relative simulation Error of a NN trained with PCA of the transient displacement vector at each testing point placed midway between training data points in predicting the location of the damage. (Input Matrix = 44×200, Time required for training = 24 sec) ..... 43

Figure 17: Dividing 20×10 sections to 2×2×2×5×5 sections..... 44

Figure 18: Relative simulation error of a NN trained with a training set that was obtained from PCA of the transient displacement vector and the selection from OA at each testing points placed in the midway between training data points in predicting the location of the damage (Input Matrix = 44×100, Time required for training = 7 seconds)..... 45

Figure 19: Solution of the direct problem..... 46

Figure 20: An example mesh using triangular elements for the optical finite element code as applied to a circular domain. .... 49

Figure 21: A finite element mesh and the corresponding photon intensity contour over a rectangular shaped tissue. .... 49

Figure 22: The contour of photon intensity difference with and without the presence of an anomaly for two types of anomalies, (a) absorption and (b) scattering anomaly. .... 50

Figure 23: Finite element mesh and photon intensity contour over a circular shaped tissue. .... 50

Figure 24: The contours of photon intensity-difference with and without anomalies for two types of anomalies..... 51

Figure 25: Solution of photon diffusion equation using finite element method for the case when an absorption anomaly is present in a homogeneous medium (medium:  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu'_s = 2.0 \text{ mm}^{-1}$ , anomaly:  $\mu_a = 0.5 \text{ mm}^{-1}$ ). .... 52

Figure 26: Photon density difference with and without the presence of an absorption anomaly. (optical properties of tissue: medium:  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu_s = 2.0 \text{ mm}^{-1}$ ; anomaly  $\mu_a = 0.05 \text{ mm}^{-1}$ ) ..... 52

Figure 27: Finite element mesh and three anomaly locations for calculating PMTF and light intensity readings at detectors.....	53
Figure 28: Detector readings corresponding to a source placed at four locations. ....	54
Figure 29: Difference of PMTF between homogeneous and anomaly cases for 4 source locations. ....	54
Figure 30: PMTF calculation using MATLAB and TOAST (Matlab Result: 24 circles, Dirichlet boundary conditions, Collimated source condition; TOAST Result: 32 Circles, Robin boundary conditions, Collimated source condition).....	55
Figure 31: Convergence study of the finite element model for various anomaly locations. ....	56
Figure 32: Finite element mesh for the abnormality lined up in uni-radial direction.....	58
Figure 33: Simulation error compared testing data set. ....	59
Figure 34: Array of Light sources and detectors. ....	59
Figure 35: Error between target obtained from trained NN and training target. ....	60
Figure 36: Testing points and training points on the tissue. ....	61
Figure 37: Simulation error for the testing points with radial basis neural network. ....	61
Figure 38: Simulation error for the testing points with feed forward neural network.....	62
Figure 39: Examples of an anomaly in a non-homogeneous medium with different shapes of inside composition. (tissue properties: white area: medium, $\mu_a = 0.01 \text{ mm}^{-1}$ , $\mu'_s = 1.0 \text{ mm}^{-1}$ ; gray area: parenchyma, $\mu_a = 0.03 \text{ mm}^{-1}$ , $\mu'_s = 0.5 \text{ mm}^{-1}$ ; dark area: anomaly (tumor) $\mu_a = 0.5 \text{ mm}^{-1}$ , $\mu'_s = 3.0 \text{ mm}^{-1}$ ).	64
Figure 40: Probability of successfully predicting the existence of an anomaly in a non-homogeneous medium by a trained NN for different number of training sets, which were randomly generated.....	64
Figure 41: Distribution of training and testing set for a NN that predicts the location and the size of an anomaly. (Optical properties of tissue: medium $\mu_a = 0.025 \text{ mm}^{-1}$ , $\mu'_s = 2.0 \text{ mm}^{-1}$ , anomaly $\mu_a = 0.05 \text{ mm}^{-1}$ ).....	65
Figure 42: The simulation error of a NN in predicting the location and the size of an anomaly in FE-model with 384 elements. ....	66

Figure 43: The structure of a multi-layer feed-forward NN to identify the property of each element. Each output neuron corresponds to the parameters of each element. ....	67
Figure 44: Fine mesh for the direct problem solution and coarse mesh for the inverse problem solution showing two sizes of anomalies corresponding to each mesh used in the training. (fine mesh: 1536 elements ; coarse mesh: 384 elements; optical properties of tissue: medium: $\mu_a = 0.025 \text{ mm}^{-1}$ , $\mu_s = 2.0 \text{ mm}^{-1}$ ; anomaly $\mu_a = 0.5 \text{ mm}^{-1}$ ).....	68
Figure 45: Actual target and simulated output of trained NNs displaying the edges of an anomaly on fine and coarse finite element meshes in the testing set.....	69
Figure 46: The worst cases in the simulation results using different meshes for direct and inverse problems. ....	70
Figure 47: The learning process for various sets for the training inputs. ....	71
Figure 48: The simulation error in predicting the orientation of an anomaly (E: integrated intensity; PMTF: photon mean time of flight; PCA: principal component analysis). ....	72
Figure 49: Training process of the NNs to identify element properties .....	73
Figure 50: Simulation of element properties for 10 cases using a NN trained with PCA. Each case was tried separately. (Blue line: edge of testing absorption anomaly, Red line: edge of simulated anomaly).....	73
Figure 51: Contour plot of mean time of flight in a rectangular shaped tissue model with one active source.....	75
Figure 52: Identification of smaller anomalies (Radius of an anomaly = 2.5 mm) .....	75
Figure 53: Identification of larger anomalies (Radius of Anomaly = 5 mm) .....	76
Figure 54: Geometric drawing of the miniature fiber-optic pressure sensor and the light paths around the gap area.....	78
Figure 55: Finite Element Model of the miniature fiber-optic sensor. ....	78
Figure 56: The mode shapes of the sensor with sliding bottom boundary condition. ....	80
Figure 57: The mode shapes of the sensor with fixed bottom boundary condition.....	80
Figure 58: Pressure Time-history approximated from experimental data. ....	81
Figure 59: Calculated time history of the displacements at the tip of the fiber, obtained using NASTRAN .....	81

Figure 60: FFT of the response at the tip of the sensor .....	82
Figure 61: FFT of the output divided by FFT of the applied pressure .....	82
Figure 62: Time history of the Pressure Profile with negative phase.....	83
Figure 63: Time history of the gap change obtained with NASTRAN's direct method ..	84
Figure 64: FFT of Output divided by FFT of Input using Modal method.....	85
Figure 65: Piecewise linear approximation.....	88
Figure 66: Time history of the gap displacement using modal superposition method and NASTRAN modal method.....	89
Figure 67: Time history of the change in the length of gap obtained with the modal acceleration method and the NASTRAN direct method.....	90
Figure 68: Fixed geometric values in the sensor design around the grooved area .....	92
Figure 69: The procedure in parametric study on the design of the sensor. ....	93
Figure 70: Convergence of a simulation error in training of NNs for various dimensions and the response of the sensor .....	95
Figure 71: Simulation of the trained NN for unseen testing data set.....	95
Figure 72: Finite element model of the sensor in the first case. (Width, $W=3.48\text{mm}$ ; Length, $L=0.78\text{mm}$ ; Height, $H=0.83\text{mm}$ ).....	97
Figure 73: The finite element model of the sensor in the second case ( $W=3.74\text{mm}$ , $L=2.01\text{mm}$ , $H=2.23\text{mm}$ ).....	98
Figure 74: Finite element model of the sensor in the fourth case ( $W = 2.3 \text{ mm}$ , $L = 1.5$ $\text{mm}$ , $H = 2.02 \text{ mm}$ ).....	99
Figure 75: The finite element model in optimal dimensions in the fourth case ( $W = 3.76$ $\text{mm}$ , $L = 1.89 \text{ mm}$ , $H = 2.05 \text{ mm}$ ).....	100
Figure A. 1: Rendered drawing of the geometric model. ....	116
Figure A. 2: Wireframe drawing of the geometric model. ....	116
Figure A. 3: A solid finite element model with tetrahedral elements for the sting balance. .....	117
Figure A. 4: $x$ component strain contour with displacement when a lift force of $12.36 \text{ lb}$ and a drag force of $14.61 \text{ lb}$ are applied together showing relative displacement..	119

Figure A. 5:  $x$  component stress distribution in  $psi$  when a lift force of  $12.36 lb$  is applied at the tip..... 120

Figure A. 6: The contour diagram of  $\sigma_{xy}$  on the surface of the sensor when a rolling moment of  $10 lb-in.$  is applied around the tip of the balance ( $psi$ ). ..... 121

Figure A. 7:  $x$  component stress contour in  $psi$  when a pitching moment of  $10 lb-in.$  is applied at the tip showing relative displacement. .... 122

Figure A. 8:  $x$  component stress contour in  $psi$  when side force of  $10 lb$  is applied at the tip. .... 122

Figure A. 9:  $x$  component stress contour in  $psi$  when yawing moment of  $10 lb-in.$  is applied at the tip..... 123

Figure A. 10: The location of fiber optic sensor is chosen based on  $x$  component stress distribution (in  $psi$ ) when lift force of  $12.36 lb$  is applied at the tip. .... 123

Figure B. 1: A time varying intensity. ( $\langle t \rangle$  is mean time of the occurrence of the intensity) ..... 129

## LIST OF TABLES

Table 1: Example of an Orthogonal Array (7 factors, 2 levels, strength 2).....	28
Table 2: Part of an orthogonal array for 3 factors with 2 levels, 2 factors with 5 levels, and strength 2.....	45
Table 3: Simulation outputs for the training points from the trained neural network. ....	62
Table 4: Natural frequencies of the miniature fiber optic sensor for different boundary conditions.....	79
Table 5: The comparison of CPU time for calculating the transient response with different methods .....	90
Table 6: Optimal dimensions of a quarter model that fulfills relaxed constraints. (The values related to the constraints are in bold case and the values that violate a constraint are in italics and are underlined. The dimensions of the original design are W=2.5mm, L=2.5mm, and H=0.8mm).....	96
Table A. 1: Comparison of FEM and Experimental Stress Results.....	124

# Chapter 1

## INTRODUCTION

The transfer of a technology developed in one area to another area is very desirable. A soft computing method like artificial neural networks (NN), which are finding application in almost all branches of science and engineering, can be used in different engineering fields without changing much of the basic methodology. Here, the capability of solving inverse problems using Neural Networks has been studied for three different kinds of problems employing numerical simulations. The first one is a damage identification problem, the second is an optical imaging problem, and the third is the optimization of a fiber optic sensor.

### 1.1 Objectives

The objective in the damage identification problem is to identify damage parameters in a structure by measuring variations between the dynamic behavior of damaged and perfect systems. The objective in the optical imaging problem is to identify an anomaly in a light-diffusive medium, such as a human tissue, by studying the effect of an anomaly on the propagation of near-infrared light passing through the tissue. Both the problems have a common feature in that an attempt was made to find an anomaly in an otherwise perfect system using limited data such as natural frequencies and diffused measurements only at the boundary. Therefore, there exists a possibility of using similar approaches, such as an application of artificial neural networks, to solve both types of inverse problems. In addition, the similar approach can also be used in solving the optimization problem of a

fiber optic sensor, in which the role of neural networks is to identify the relationship between the change in design parameters and the structural response of the sensor.

The artificial neural networks are imitations of real neurons in the human brain in their structure, data processing and restoring method, and learning ability. In artificial neural networks and real neurons, knowledge is acquired by the network through a learning process and inter-neuron connection strengths known as synaptic weights are used to store the knowledge. This organization of neurons is what enables the brain to perform certain tasks, like pattern recognition and motor control, much faster than the fastest digital computer available even today. Neural networks can be considered as universal approximators for arbitrary functions that make probabilistic assumptions about data and the learning can be considered as methods for finding parameter values that appear to be closest to the available data [ 1 ] [ 2 ]. The neural networks, by having advantages in that they are capable of representing non-linear systems and are capable of providing instantaneous responses once properly trained, possess a strong possibility of solving the three inverse problems efficiently.

## **1.2 Three Inverse Problems**

### **1.2.1 Damage Identification Problem**

In the first inverse problem, that of damage identification, two structures were studied. The first was a one-dimensional beam and the second was two-dimensional plate. In the one-dimensional beam problem, a beam equation was solved to find the natural frequencies of a diffusively damaged beam and the obtained solutions were used to build the training set and the testing set for identifying a damaged zone using NNs. In the two-

dimensional plate problem, a locally damaged plate finite element model developed in the NASTRAN<sup>\*</sup>, a commercial finite element package, was utilized to build the training set and the testing set for a NN. The use of Principal Component Analysis (PCA) and Orthogonal Arrays (OAs) for NNs were also studied. The data used as inputs for NNs in identifying the location of the damage were the transient response data at a number of locations of the damaged plates. NNs were able to automatically map this two-dimensional damage location parameter space and give the location of the damage, given the transient response of the damaged beam, in an instant. The problem in utilizing the transient response data was that the dimension of the input was quite large, making the use of NN somewhat impractical because of the time needed in training NNs. Therefore, the use of PCA was considered to be a viable option to reduce the dimension of the inputs for NNs. On the other hand, the size of the training samples has to be such that they can cover most of the possible variations in the parameters. This requires a huge number of training sets to be selected and the direct problem has to be solved repeatedly. Using OAs, the training dataset to be selected and solved was greatly reduced. The experience in utilizing NNs with PCA in the damage identification problem was found to be very valuable in solving the optical imaging problem.

### **1.2.2 Optical Medical Imaging Problem**

Optical imaging means the methodology of using light in a narrow wavelength band in the near-infrared range (~700-1,000nm) to trans-illuminate a light diffusive medium such as a human tissue and to use the resulting measurements of intensity on the tissue boundary to map the internal optical properties. The main advantage of optical imaging is

---

<sup>\*</sup> NASTRAN is a registered trademark of MSC.Software Corp.

its capability of safely and portably measuring tissue functions to detect non-functioning cells, like cancerous cells. It is known that to be able to observe the functioning of cells, a continuous, non-invasive imaging method is required. Moreover, optical imaging has advantages over X-ray, CT-scans, and PET (Positron Emission Tomography) because of its portability and lower equipment cost as well [ 3 ]. The main idea of optical imaging is that light passes through the body in small amounts, carrying with it characteristics about tissues through which it has passed. Based on these characteristics, optical images can be obtained by solving the inverse problem of light propagation. However, diffusive and scattering characteristics of near-infrared light when propagating in a tissue lead to a highly non-linear inverse problem whose solution requires large amounts of computational time, even for relatively coarse measurements, if conventional methods are used [ 4 ][ 5 ].

The application of NN may help reduce the computational time to solve this inverse problem of optical imaging, thus making the use of optical imaging to determine the presence of an anomaly in a tissue a more viable approach. The process of solving this inverse problem consists of two parts. The first part entails solving the direct problem of the light diffusion equation to predict propagation of photons in a tissue. The second part, based on the solutions of the direct problem and the information observed in the detectors, entails obtaining an optical image by solving the inverse problem. A finite element program to solve the direct problem was developed in MATLAB<sup>†</sup> environment and has been integrated with NNs available in the Neural Network Toolbox in MATLAB to identify the location of an anomaly using measurements of near-infrared light which has

---

<sup>†</sup> MATLAB is a registered trademark of the Math Works, Inc.

propagated through a human tissue and has gone through a high degree of both scattering and absorption.

### **1.2.3 Structural Optimization Problem**

In optimizing the configuration of a fiber optic sensor, the dynamic analysis of the sensor subjected to a blast profile was performed and a parametric study using the dynamic response was performed. To briefly describe the construction of the sensor; it is composed of a silicon base and two optical fibers mounted in V-grooves in the base. When the air pressure between the two fibers is changed, the density of the air is changed and this results in a phase change of the light that passes through and is reflected from the gap between the two fibers. For the sensor to be able to achieve a desired accuracy, the change in the distance between the two fibers due to the pressure change should not interfere with the phase change due to the change in the density of the air between two fibers. Therefore, accurate dynamic analysis of the sensor is essential to ensure the accuracy of the measurement of the sensor. To this end, a normal mode analysis and a transient response analysis of the sensor were performed. A parametric study of the design of the sensor was required to minimize the displacement between the two fibers caused by a pressure change. In performing the parametric study, the need for a relationship between the design parameters and the response of the sensor was fulfilled by using an artificial neural network (NN). A NN was trained to map the relation between design parameters and the response of the sensor. The optimization Toolbox of MATLAB was used to perform the optimization in the NN space.

### **1.3 Obstacles in Applying Neural Networks for Inverse Problems**

One of the obstacles in applying artificial neural networks is the need for solving the direct problem repeatedly and generating enough number of training sets. To reduce the time required in solving the direct problems of structural dynamics and photon transport in opaque tissue, the finite element method was used. To solve transient problems in the dynamic analyses of structures, which are computationally intense, the modal superposition method and the modal acceleration method were employed. The need for generating enough training sets was fulfilled by automatically generating the training sets using a script program, which automatically generated the finite element model with different parameters and included scripts that could combine the whole solution processes in different engineering packages for the direct problem and the inverse problem using neural networks.

Furthermore, two of the common problems in applying NNs for solving inverse problems are that the dimension of the inputs for NNs can be too large for a quick training of the NN, and that the number of times that the direct problem needs to be solved to construct the training set can be too large to obtain meaningful solutions with the computing resources available. Principal Component Analysis can be used to reduce the dimension of the input needed to train a Neural Network and Orthogonal Arrays can be used to efficiently reduce the size of the training sets and obtain reasonable results. Therefore, the effects of using Principal Component Analysis (PCA) and Orthogonal Arrays (OAs) in applying NNs for solving the inverse problems have also been studied.

## 1.4 Dissertation Overview

In Chapter 2, a literature review on artificial neural networks, damage identification using NN, optical medical imaging, structural optimization using NN, principal component analysis, and orthogonal arrays is given. In Chapter 3, damage identification in beams and plates using neural networks based on the dynamic response is given. In Chapter 4, an explanation of optical medical imaging is given followed by the procedure used to solve the photon transport equation using the finite element method. Next, an explanation of the application of neural networks in identifying an anomaly using optical medical imaging is given. In Chapter 5, the dynamic analysis of a miniature fiber optic sensor using finite element method, modal superposition method, and modal acceleration method is given. Next, the application of neural networks to the design of a miniature fiber optic sensor is given. This chapter is followed by a chapter on conclusions.

In the Appendix, the steps and processes involved in the construction of a complicated shaped, three-dimensional solid finite element model is given, with the example of an analysis of 6-component sting balance used in wind tunnel testing. This example, for which experimental results were available, was solved to gain experience in developing three-dimensional finite element models of complex systems. The appendix also contains steps required in the formulation of the finite element analysis to solve the photon transport problem in human tissue.

# Chapter 2

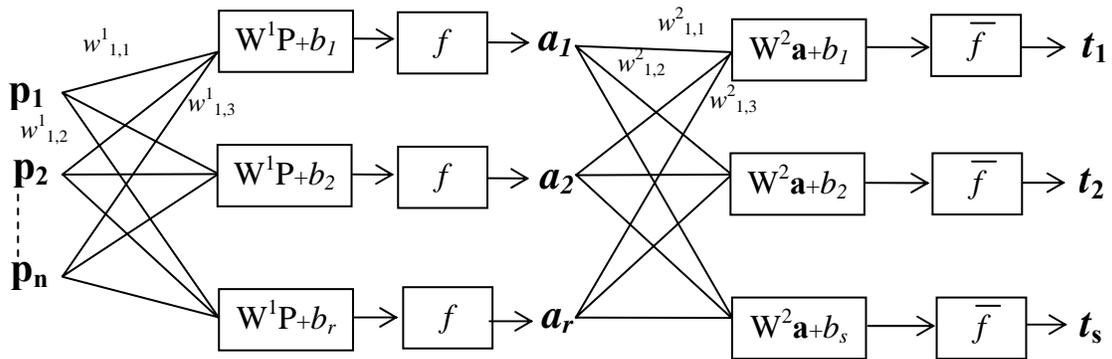
## LITERATURE REVIEW

### 2.1 Artificial Neural Networks

#### 2.1.1 Multi-Layer Feed-Forward Backpropagation Neural Network

The most commonly used neural network structure is multi-layer feed-forward backpropagation neural network as shown in Figure 1. The hidden neurons  $a$  are transferred sums of weighted inputs  $p$  with added bias and the outputs  $t$  of the network are transferred sums of weighted hidden neuron values  $a$  with added bias. A sigmoid function was used in hidden layers and a linear function was used in the output layer in our case. The relation of input and output is shown in Eq. (1) where  $w$  is weight and  $f$  is a transfer function [ 1 ] [ 2 ].

$$a_i = f\left(\sum_{j=1}^r w_{i,j}^1 \cdot p_j + b_i^1\right), \quad t_i = \bar{f}\left(\sum_{j=1}^s w_{i,j}^2 \cdot a_j + b_i^2\right) \quad (1)$$



**Figure 1:** Multil-layer feed-forward backpropagation neural network ( $p$  = inputs,  $a$  = hidden neurons,  $W$  = weights,  $b$  = bias,  $f$  = transfer function,  $n$  = number of inputs,  $r$  = number of hidden neurons,  $s$  = number of output neurons,  $t$  = outputs).

Two kinds of transfer functions are popularly used in multi-layer NNs. One is a piecewise linear function and the other is a logistic sigmoid transfer function shown in Figure 2. The piecewise linear transfer function can be expressed as

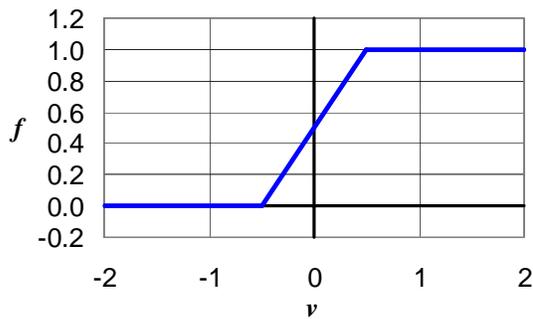
$$f(v) = \begin{cases} 1, & v \geq \frac{1}{2} \\ v, & \frac{1}{2} > v > -\frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \quad (2)$$

where  $v$  is an input to the transfer function.

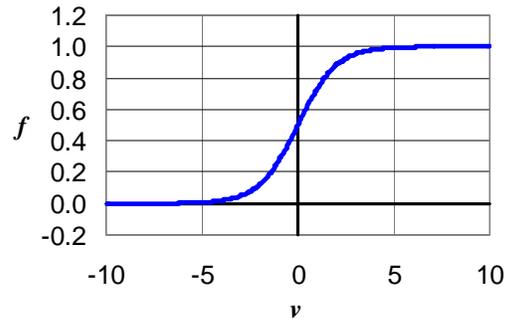
The logistic sigmoid transfer functions can be expressed as

$$f(v) = \frac{1}{1 + e^{-cv}} \quad (3)$$

where  $c$  is a slope parameter of the sigmoid function. The slope of the logistic function at the origin equals  $c/4$ .



(a) Piecewise linear transfer function



(b) Logistic sigmoid function

**Figure 2:** The graphs of a piecewise linear transfer function and a sigmoid transfer function.

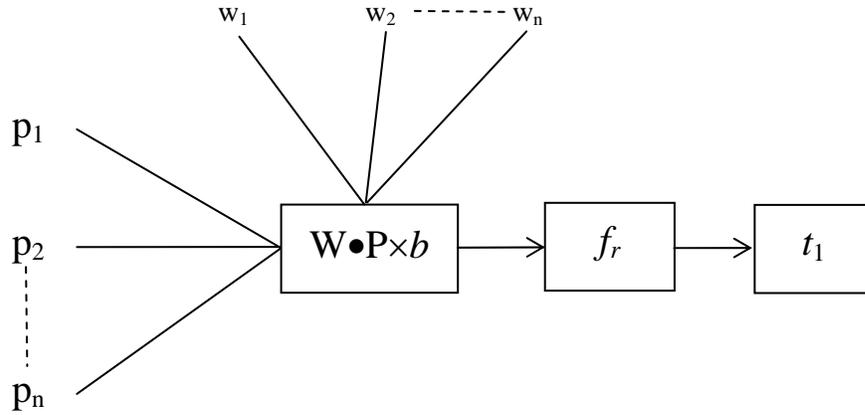
Once a collection of inputs and desired outputs for a NN is obtained, the training of the NN can be started. The process for the training of a multi-layer feed-forward backpropagation neural network can be outlined as follows.

In the first step, the weights and biases of the NN are randomly picked. Next, the weights and biases are adjusted by going through forward and backward calculation for each training cases. Here, the forward calculation is to calculate the output of the NN with the current values of the weights following forward direction of the NN and to calculate the error between the output of the NN and the desired output for the training case presented. The backward calculation is to calculate the local gradients of the network from the error between the output and the input and the slope of the transfer function and to adjust the weights of each layer in backward direction of the network. When all the training cases are presented, the whole process of the forward and the backward calculation for each training case is repeated by randomly varying the order of the presentation of the training cases. Here, one iteration is called one epoch. This process is repeated until the average error between the output of the NN and the desired output becomes less than or equal to the desired value.

Feed-forward networks with one or more hidden layers of sigmoid neurons and a linear output layer are often used. Multiple layers of neurons with nonlinear transfer functions allow the network to learn nonlinear and linear relationships between input and output. This structure of neural network can approximate any function with a finite number of discontinuities given that enough numbers of neurons are present in the hidden layers.

Another structure of neural network that is frequently used is a radial-basis function neural network, shown in Figure 3. A radial-basis function neural network has one input layer and only one hidden layer and one output layer. The relation between the input  $p_i$  and the output  $t_i$  is shown in (2) where  $b$  is bias,  $f_r$  is radial basis function and  $\| \cdot \|$  means dot product of two vectors.

$$t_i = f_r(\|w \cdot p\|b_i) \quad (4)$$



**Figure 3:** Radial-basis neural network ( $p$  = inputs,  $W$  = weights,  $f_r$  = radial basis function,  $t$  = outputs).

The output of the first layer neurons, each of which represents a basis function, is determined by the distance between the network input and the center of the basis function. A radial-basis function neural network uses a linear output layer and a radial-basis function neural network can be constructed very quickly by adding as many radial basis neurons as there are in the input layer. A radial-basis function neural network is more

suited to be used to find local minimum since it uses exponentially decaying localized nonlinearities [ 1 ].

## **2.2 Damage Identification Using Artificial Neural Networks**

There have been many efforts to use NNs in studying damage identification problems. Kudva, et al. [ 6 ] used back-propagation NN in identifying holes of various diameters in a plate stiffened with bays. Neural networks were used to identify the location and size of the holes based on the strain gage data. They demonstrated that a NN can be used to determine damage location and size in a typical structure. Wu, et al. [ 7 ] used back-propagation NN in identifying damage in a three story building. The NN simulated the level of damage in each of the members based on the Fourier transform of the acceleration data. They were able to predict damages in the first and the third floor but not on the second floor.

Leath, et al. [ 8 ] used multi-layer back-propagation NN in identifying damage in a four element cantilever beam based on the first two bending frequencies, and also developed a training algorithm that can create a NN that fits the data with a minimal number of neurons. The damage was modeled by reducing Young's modulus by up to 95%. They fixed the number of hidden nodes to be one less than the number of data points, and also found that exactly fitting the data can cause problems in most cases. Spillman, et al. [ 9 ] used feed-forward NN in identifying damage in a steel bridge using accelerometer and fiber optic modal sensor. Damage was modeled by loosening or removing a bolted plate in the bridge. The NN used the height and frequency of the first two modal peaks and the impact intensity and location as inputs. They were able to diagnose damage correctly in

58% of the testing cases. Worden, et al. [ 10 ] used back-propagation NN in a framework structure. Damage was modeled by removing one member from the framework. The NN identified the map from static strain data to a measure of the damage. They used three hidden layers and used a finite element method (FEM) to generate training data and an experimental model of the same geometry to test the trained NN. The trained NN was able to identify the location of the damage most of the time.

Kirkegaard and Rytter [ 11 ] used back-propagation NN in a steel lattice mast. Damage was simulated by bolted joints of reduced thickness. The NN simulated the degree of damage in stiffness based on the first five natural frequencies. They also used FEM to generate the training data and the training data had 0% to 100% reduction in diagonal cross sectional area. The trained NN was able to locate damage which had a magnitude of more than 50%. Manning, et al. [ 12 ] used back-propagation NN in a truss structure. Damage was modeled by a change in one of the members' cross-section. The NN simulated member cross-sectional area based on a pole-zero location extracted from the frequency response functions between the actuator and the two piezoceramic sensors on the same member and the stiffness information of the member. The networks were tested on three examples and could predict member area within 10% for most of the members. Povich and Lim [ 13 ] used a back-propagation NN in a planar truss with struts. Damage was modeled by removing struts from the structure. The NN simulated damage in each member based on the Fourier transform of the acceleration time history. The NN had 384 inputs and 60 outputs that corresponded to each strut. The NN was trained with 61 examples and was able to correctly identify a missing strut in 21 out of a total of 38 testing cases.

Tsou and Shen [ 14 ] used back-propagation NN in 3-DOF and 8-DOF spring-mass systems. Damage was modeled by reducing one of the spring constants by an amount that varied from 10% to 80%. The NN simulated the change in the spring stiffness based on the changes in the natural frequencies for the case of a 3-DOF system and using the residual modal force in an 8-DOF system. They employed a two-step approach in identifying the damage using NNs. The first NN identified which spring was damaged. Based on the output of the first NN, an appropriate NN was chosen to quantify the damage in the damaged spring. They were able to identify the extent of the damage with an accuracy of 5% in the interpolation case, but the error went up to 30% if the NN had to use extrapolation. Rhim and Lee [ 15 ] used back-propagation NN in identifying delamination in a composite cantilevered beam modeled using the finite element method. An auto-regressive system identification was performed on the transfer function of the beam from the force input to the displacement. Next, the NN was used to simulate an experimental damage scale based on the denominator of the transfer function. The network was trained with 10 training patterns and was able to correctly identify the damage in three test examples.

Barai and Pandey [ 16 ] used back-propagation NN to identify the changes in the stiffness in different elements in truss structures using nodal time histories. They used a finite element simulation of a truss structure representing a bridge with a moving point force to simulate a vehicle at constant velocity. They were able to predict the change in stiffness within an accuracy of 4%. Ceravolo, et al. [ 17 ] used back-propagation NN in a truss structure by a finite element method. Damage was modeled by removing elements from the structure. The NN simulated the location of damage using the first 10 natural

frequencies. They found out that a NN with two hidden layers performs better than a NN with only one hidden layer. Schwarz, et al. [ 18 ] used back-propagation NN in spring-mass systems. They used the NN to identify the changes in the spring constants that simulate damage based on the changes in the natural frequencies.

Ganguli and Chopra [ 19 ] used a comprehensive physics based model of a helicopter rotor in a forward flight. The rotor model employed finite elements in both space and time. They used two neural networks. The first was to classify the type of damage and the second was to characterize the level of damage from noise-contaminated simulated vibration and blade response test data. Back propagation NN with one hidden layer was used with adaptive learning. They were able to identify single and multiple faults by moisture absorption, damaged lag damper, and damaged pitch control system using a NN trained on simulated ideal and noise contaminated data. They found out that it is important to train a neural network using noise contaminated response data for accurate estimation. The NN, trained on noisy data, gave almost zero error for noise levels less than 10%.

Marwala [ 20 ] used a committee of NNs with three types of information, namely, the frequency response function, modal properties, and wavelet data, to identify four types of faults in a cylindrical shell. The committee of three networks used together resulted in a lower mean square error than the average mean square error given by any individual approach. He also found that the effectiveness of the method was enhanced when experimental data were used because the committee approach assumes that the three types of data used in the three approaches are uncorrelated. Palakal, et al. [ 22 ] used backpropagation NN with wavelet transform to predict material loss based on images of

corroded regions in the structure. They used wavelet transforms using the Harr filter for extracting parameters for segmenting the image. They obtained a large set of data required for the neural network training by perturbing images. The trained NN was able to predict the material loss quite well.

There have been a number of researchers who tried to use neural network types other than the feed-forward back-propagation neural network. Szewczyk and Hajela [ 23 ] used a counter-propagation NN, which is similar to building an adaptive table from the given data to find damage in a truss structure. The advantage of this NN is that the architecture is selected by the data. The trained NN identified the stiffness of the members based on the static deformation under a given load within an error of 30%. Klenke and Paez [ 24 ] used a probabilistic NN and a probabilistic pattern classifier in identifying the existence of damage. They used a Gaussian kernel-density estimator to approximate the probability density function to be used in their scheme. For a piece of new data, an estimated likelihood was computed for membership into two classes, damaged and undamaged. The greater of the two likelihoods was taken as the guess for class membership. Damage was modeled by changes in spring constants. In all the test cases, damage was clearly identified.

Luo and Hanagud [ 25 ] proposed a dynamic learning rate steepest descent (DSD) method in detection of delaminations and impact damage, using modal analysis and a finite element model. Real-time health monitoring techniques based on the modal analysis or finite element analysis are limited by time consuming signal processing. Their method, based on the simple steepest descent method, improved the learning convergence speed significantly so that they were able to show that a NN can be used in real-time signal

processing with modal data and finite element analysis. Doyle and Fernando [ 26 ] used a fast Fourier transform in the pre-processing stage, and back-propagation and learning vector quantization neural networks in detecting impact damage in a composite material using an optical fiber vibration sensor system. They achieved a lower error with fewer iterations using a learning vector quantization NN than with a back-propagation NN.

Li, et al. [ 27 ] used an adaptive resonance theory neural network embedded with fuzzy classifiers. The adaptive resonance theory based on neural networks is made up of two layers of cells, an input feature layer and an encoded category representation layer. When the pattern of an input matches one of the stored examples, only the matching cell resonates with the input and the outputs a value that is close to 1. The trained NN was able to correctly identify the condition of significant wear on the tool prior to the rapid wear stage based on the optical scattering image of a surface. Wu, et al. [ 28 ] used wavelet based NNs with a data-fusion method for the damage detection of anisotropic composite materials. They used a piezoelectric ceramic patch as an actuator and another as a sensor to obtain output acceleration signals. Using wavelet packets, they decomposed signals into eight different wave bands and were able to obtain different features in decomposed wave bands for a healthy structure and for structures with various levels of damages. The obtained features for different cases of damage were used to train an adaptive NN to recognize the characteristics of different damage, and the trained NN was able to distinguish between different levels of damage.

In our damage identification problem, a diffusively damaged Euler-Bernoulli beam model, described in Cerri and Vestroni, [ 29 ], was utilized to evaluate the use of NNs for identification purposes. In Cerri and Vestroni's work, a procedure based on modal

equations was used to solve the inverse problem. Differences in natural frequencies of the damaged beam and the perfect beam were utilized in determining three damage parameters. The process used in Cerri and Vestroni in finding the parameters was graphical and was difficult to implement for some cases, e.g., ill-conditioned cases which occur when the distinction between the two different cases is too small to identify with the regular method. On the other hand, NNs are able to automatically map this three-dimensional parameter space and give the values of damage parameters, given the natural frequencies of the damaged beam, in an instantaneous manner. Our work differs from previous works using NNs in that we are identifying the extent of the damage in addition to location and magnitude of the damage. In our case, the inputs for the NNs are the natural frequencies of a damaged beam and the targets to be identified using a NN are the location, the extent, and the magnitude of the damage. For this purpose, most suitable structure of NNs and training methods were studied. The experience in utilizing NNs in this damage identification problem was found to be very valuable in solving the optical imaging problem.

## 2.3 Optical Imaging Using Near Infrared Light

Optical imaging means the methodology of using light in a narrow wavelength band in the near-infrared (~700-1,000nm) range to transilluminate a light diffusive medium such as a human tissue, and to use the resulting measurements of intensity on the tissue boundary to reconstruct a map of the internal optical properties. The main advantage of optical imaging is its capability of safely and portably measuring tissue functions to detect non-functioning cells, such as cancerous cells. It is known that to be able to observe the functioning of cells, a continuous and non-invasive imaging method is required. Moreover, optical imaging has advantages over X-rays, CT-scans, and PET (Positron Emission Tomography) because of its portability and lower equipment cost as well [ 3 ]. The main idea of optical imaging is that light passes through the body in small amounts, carrying with it characteristics about tissues through which it has passed. Based on these characteristics, optical images can be obtained by solving inverse problem of light propagation. However, diffusive and scattering characteristics of near-infrared light when propagating in a tissue leads to a highly non-linear inverse problem whose solution requires large amounts of computational time, even for relatively coarse measurements, if conventional methods are used [ 4 ] [ 5 ] [ 30 ] [ 31 ] [ 40 ] [ 41 ] [ 42 ] [ 43 ] [ 44 ].

The application of NN's may help reduce this computational time, thus making the use of optical imaging to solve the underlying inverse problem a more viable approach. The process of solving this inverse problem consists of two parts. The first part entails solving the direct problem of the light diffusion equation to predict propagation of photons in a tissue. The second part, based on the solutions of the direct problem and the information

observed in the detectors, entails obtaining an optical image by solving the inverse problem.

There also have been a number of researchers who have used NNs in various medical imaging problems, but none have used NNs in optical medical imaging. Chiu and Yau [ 45 ] used neural networks for the reconstruction of x-ray computer tomographic (CT) images of a time-varying object that has motion artifacts. They introduced an imaging reconstruction method based on the previous knowledge of the projections and developed a novel training algorithm. Nakao, et al. [ 46 ] used a backpropagation algorithm to construct a CT image using 4 projections, compared the performance of this algorithm with that of the algebraic reconstruction technique (ART), and found that their algorithm is more effective than the ART. Kiliç and Korürek [ 47 ] used a finite element based neural network algorithm which calculates conductivity distribution from the electrical measurements on the medium boundary. Aizenberg, et al. [ 48 ] used a special kind of cellular neural network based on multiple valued threshold logic in the complex plane to detect edges and enhance resolution in a CT image.

Among the existing approaches to solve the direct problem of photon transport in tissue, the fastest method at present perhaps is the finite element method to solve the photon diffusion equation. In our research, we have used a finite element program and have integrated this program with NNs available in the Neural Network Toolbox in the MATLAB<sup>‡</sup> to identify the presence of an anomaly using measurements of near-infrared light which has propagated through a human tissue and has gone through a high degree of scattering and absorption. As far as the optical imaging is concerned, the inverse problem has been solved using traditional techniques like a gradient-based optimization scheme

---

<sup>‡</sup> MATLAB is a registered trademark of the Math Works, Inc.

and the perturbation approach [ 4] [ 5 ] [ 30 ] [ 31 ]. These conventional methods are slow and may get trapped in a local minimum. We have used NNs in this problem so as to be able to obtain both a set of outputs (location and size of an anomaly) to the given set of inputs (near-infrared light measured at a number of detectors and the light emitted by a number of sources) and to simultaneously achieve a higher resolution in an instantaneous manner.

## 2.4 Application of NNs in Engineering Optimization Problems

The capability of NNs has been utilized in various design optimization problem by many researchers. Mukherjee, et al. [ 32 ] combined NNs with rule-based expert system (RBES) approaches to overcome the shortcomings of RBES which are that RBES lacks the learning capability, the rules need to be stated explicitly. They used a mathematical optimization model to generate the training set and the testing set for NNs. Jayatheertha, et al. [ 33 ] applied a simulated annealing process in their approach in optimizing the configuration of a laminated composite plate. Simulated annealing is a statistical training method and resembles the annealing of a metal.

Ramasamy, et al. [ 34 ] used a feed-forward backpropagation neural network in an expert system and compared the optimization results with those of the genetic algorithm in designing a truss roof system. Rai et al. [ 35 ] started the design process with the initial condition from the reference design and used a sequence of response surfaces based on both neural networks and polynomial fits in traversing the design space to find an optimal dimension of an airfoil.

Nikolaidis et al. [ 36 ] used NNs and response surface polynomials to rapidly predict the performance characteristics of an automotive joint given the component dimensions. They found that the performance of NNs and the response surface polynomials are very similar to each other. Kaveh et al. [ 37 ] used feed-forward backpropagation NNs to predict the maximum deflection and the weight in the design of double layer grids for minimum weight. They used a data ordering method to reduce the nonlinearity of the data and to increase the speed of the training.

Greenman et al. [ 38 ] combined a bootstrap technique that estimates network generalization performance with stochastic and deterministic optimization techniques in adjusting the interconnection geometry of NNs to optimize a NN that models a multi-element airfoil with small training data sets.

Hadi [ 39 ] used NNs in optimizing a concrete beam for weight and cost with strength and serviceability constraints by storing many optimum designs and training a neural network for the stored design. They also used this approach in optimizing the design of fiber reinforced concrete beams.

The most important features that were utilized from NNs in most design processes listed above were the superior generalization quality that is capable of extracting significant information from a massive set of data, and the ability to cope with the ill-defined problems.

One of the main problems in applying NNs for engineering optimization problems is that the process of obtaining the solution for different design parameters is done manually and preparing data takes a great portion of the time. In this work, this problem has been solved by automating the whole optimization process including the direct problem and inverse problem.

## 2.5 Principal Component Analysis

Principal Component Analysis (PCA) is an algorithm of self-organized learning for NNs. PCA decomposes the data set to the principal components and reduces the dimension of the input. This method is most useful in cases when the dimension of the input is large but the data are highly correlated to each other and the size of the useful content is limited. The procedure in performing Principal Component Analysis (PCA) is as follows [ 49 ] [ 50 ].

Let  $[H]_{M \times N}$  be an output composed of  $M$  sampling points and  $N$  output sets. The first step in PCA is normalizing the matrix. The mean value and the standard variance at each point can be calculated as

$$\bar{H}_i = \frac{1}{N} \sum_{j=1}^N h_{ij}, \quad (5)$$

$$S_i^2 = \frac{1}{N} \sum_{j=1}^N (h_{ij} - \bar{H}_i)^2. \quad (6)$$

Then, the output matrix can be normalized for each point as follows.

$$\tilde{h}_{ij} = \frac{h_{ij} - \bar{H}_i}{S_i \sqrt{N}} \quad (7)$$

From the normalized output matrix, the correlation matrix is calculated as

$$[C]_{M \times M} = [\tilde{H}]_{M \times N} [\tilde{H}]_{N \times M}^T \quad (8)$$

The eigenvectors and eigenvalues of the correlation matrix are the principal components of the output matrix and can be calculated from  $[C]\{\Psi_i\} = \lambda_i\{\Psi_i\}$  where  $\lambda_i$  is the  $i^{\text{th}}$  eigenvalue and  $\Psi_i$  is the  $i^{\text{th}}$  eigenvector. The eigenvalues and eigenvectors are ordered in a way that the largest value of eigenvalue comes first.

The maximum variation from the mean is calculated as:

$$J_v = \sum_{i=1}^M \lambda_i \quad (9)$$

When we select the first  $P$  principal components to reconstruct the output matrix, the reconstruction error is

$$J_e = \sum_{i=1}^M \lambda_i - \sum_{i=1}^P \lambda_i = \sum_{i=P+1}^M \lambda_i \quad (10)$$

The relative reconstruction error in representing the system with  $P$  number of the principal components can be calculated as

$$E = \left( \sum_{i=P+1}^M \lambda_i \right) / \left( \sum_{i=1}^M \lambda_i \right) \quad (11)$$

The value of  $P$  can be determined so that the error is within a desired level.

The projection of the response variation matrix  $[\tilde{H}]_{M \times N}$  to the principal components is

$$[A]_{M \times N} = [\Psi]_{M \times M}^T [\tilde{H}]_{M \times N} \quad (12)$$

If we choose to select  $P$  number of the principal components, the projection of the response variation matrix is reduced to  $[A]_{P \times N}$ . This reduced response variation matrix is used as inputs for the NNs instead of the full output matrix.

The normalized output matrix restored back from the reduced response variation matrix can be calculated as

$$[\tilde{H}]_{M \times N} \approx [\Psi]_{M \times P} [A]_{P \times N} \quad (13)$$

The output matrix is restored back from the normalized output matrix by

$$h_{ijR}(\omega) = S_i \sqrt{N} \tilde{h}_{ij}(\omega) + \bar{H}_i. \quad (14)$$

## 2.6 Orthogonal Arrays

Orthogonal Arrays (OAs) can be defined as a simplified method of putting together an experiment. In Orthogonal Arrays, ‘Orthogonal’ means being balanced and not mixed and, in the context of experimental matrices, it means statistically independent [ 51 ]. The advantage of the orthogonal array is the benefit of efficiency and simplicity. Orthogonal arrays have been employed in number of different areas. One of their successful applications has been in quality engineering [ 52 ] [ 53 ].

As an example, Table 1 shows an OA with seven factors, two levels, and strength two. Here, ‘seven factors’ means that there are seven variables and ‘two levels’ means that there are two possible variations for the seven variables. ‘Strength two’ means that when we pick any two columns from the array, the possible combinations are 00, 01, 10, and 11. In Table 1, for any two columns picked, the number of the occurrence of the possible combination is always same. The maximum possible runs, which is called full factorial, can be calculated as  $2^7 = 128$ . Minimum runs selected by the orthogonal array is  $(\text{number of levels} - 1) \times (\text{number of factor}) + 1 = (2-1) \times 7 + 1 = 8$ .

For our use in NNs, OAs were selected from an internet database of orthogonal arrays for matching number of factors and levels.

[\(http://www.research.att.com/~njas/oadir/\)](http://www.research.att.com/~njas/oadir/).

**Table 1:** Example of an Orthogonal Array (7 factors, 2 levels, strength 2)

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	1	0	1	0	1	0	1
3	0	1	1	0	0	1	1
4	1	1	0	0	1	1	0
5	0	0	0	1	1	1	1
6	1	0	1	1	0	1	0
7	0	1	1	1	1	0	0
8	1	1	0	1	0	0	1

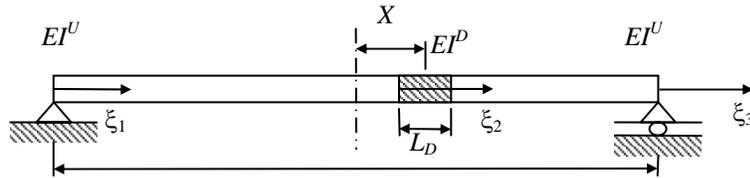
# Chapter 3

## DAMAGE IDENTIFICATION

### 3.1 Damage Identification in a Beam

#### 3.1.1 Direct Problem

As shown in Figure 4, the damage zone of the beam is represented as being at a distance  $X$  away from the midspan, having a length  $L_D$ , a density  $\rho$ , area  $A$  and a zone of reduced stiffness of  $EI^D$  in a beam that has initial bending stiffness  $EI$  and an overall span of  $L$ . These quantities are non-dimensionalized with respect to the total beam dimensions to form three non-dimensionalized damage parameters: [ 29 ] location  $x = X/(L/2)$ , damage length  $b = L^D/L$ , and damage magnitude  $\beta = (EI^U - EI^D)/EI^U$ .



**Figure 4:** Damaged beam model with three damage parameters ( $X = location$ ,  $L^D = extension of damage$ ,  $EI^D = magnitude of damage$ ).

The governing equation of a vibrating Euler-Bernoulli beam is given by

$$EI \frac{\partial^4 v(x,t)}{\partial x^4} + \rho A \frac{\partial^2 v(x,t)}{\partial t^2} = 0 \quad (15)$$

where  $v$  is the vertical displacement and  $t$  is time.

By segmenting the damaged beam with three zones,  $\xi_1$ ,  $\xi_2$ , and  $\xi_3$ , each segment can be expressed as (14) where  $\lambda_i^4 = \omega^2(\rho A / (EI)_i)L^4$ ,  $\xi_i = x_i / L$ . Utilizing the solution in the form shown in (15) and satisfying the boundary conditions at the beam ends and at each of the segment boundaries, the characteristic equation for the problem is obtained [ 29 ]. The solution of the direct problem was obtained by numerically solving this characteristic equation for each beam segment:

$$\frac{d^4 V_i}{d\xi_i^4} - \lambda_i^4 V_i = 0, \quad i = 1, 2, 3 \quad (16)$$

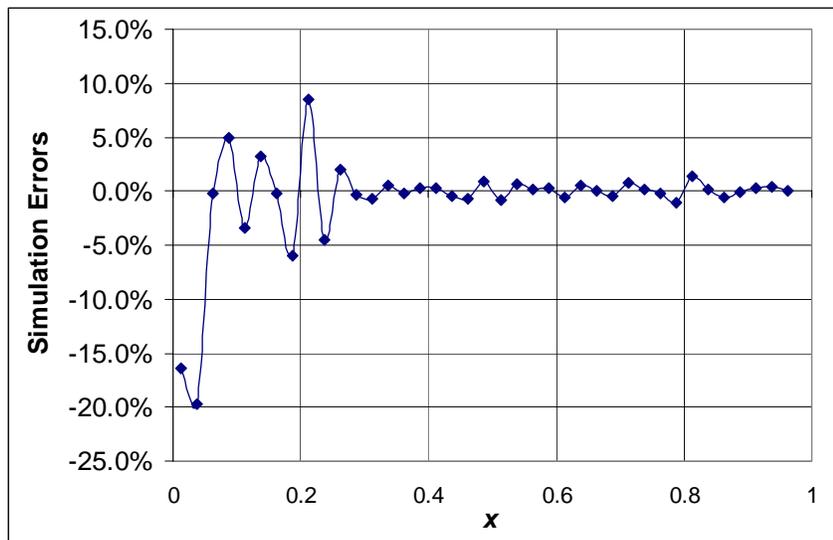
$$V_i(\xi_i) = A_{i1} \sinh \lambda_i \xi_i + A_{i2} \cosh \lambda_i \xi_i + A_{i3} \sin \lambda_i \xi_i + A_{i4} \cos \lambda_i \xi_i \quad (17)$$

MATLAB was used to solve the characteristic equation. To develop a NN to identify damage parameters, the Neural Network Toolbox in MATLAB has been used. A neural network was trained for the first three to five natural frequencies, which themselves are obtained by solving the direct problem, as inputs and the damage parameters as the training targets. The trained NN was able to provide damage parameters as output for a new set of inputs (natural frequencies). These cases where the NNs have to identify one, two, and three damage parameters have been studied separately.

### 3.1.2 Inverse Problems

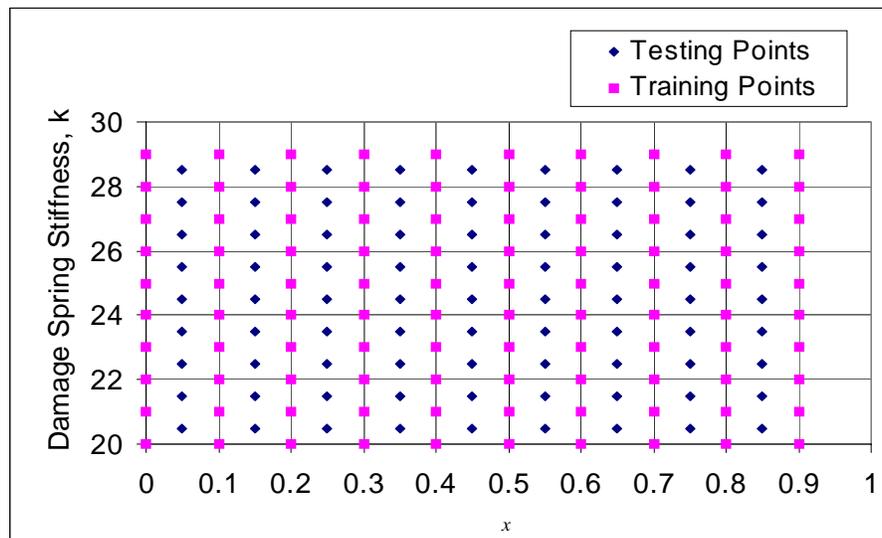
In the first case, only one damage parameter was identified from the three known natural frequencies. A total of the 3 inputs, the three natural frequencies, and as output, the damage location, were considered. In the first step, the characteristic equation was solved to get natural frequencies for various values of possible locations of damage and these natural frequencies coupled with the target, the location of the damage, were used to train a NN.

The artificial NN used in this analysis was a multi-layer feed-forward NN with two hidden layers. Transfer functions found to be most efficient for the NN were the sigmoid transfer function for the two hidden layers and a linear transfer function for the output layer. Among many backpropagation training algorithms available, the algorithm which was found to be most versatile and efficient was the resilient back-propagation method. This algorithm eliminates harmful effects of magnitudes of partial derivatives required in the steepest descent method and enables a faster training of a multi-layer NN that uses Sigmoid transfer functions requiring minimal storage [ 54 ]. Figure 5 shows the simulation errors, which are the errors between the desired value and the output of the trained NN, in the case when the NN was trained with 40 data points. In this case, the errors were relatively small compared to the case when the NN was trained with only 20 data points.



**Figure 5:** The simulation errors of a trained feed-forward NN in the prediction of the location of damage (39 testing points).

Next, the two parameter case was studied. Here, the diffused damage is represented by a concentrated damage in the form of a reduced beam bending rigidity. Figure 6 shows an example of training and testing points. The training points are shown as squares and the testing points, which lie midway between training points, are shown as diamonds.

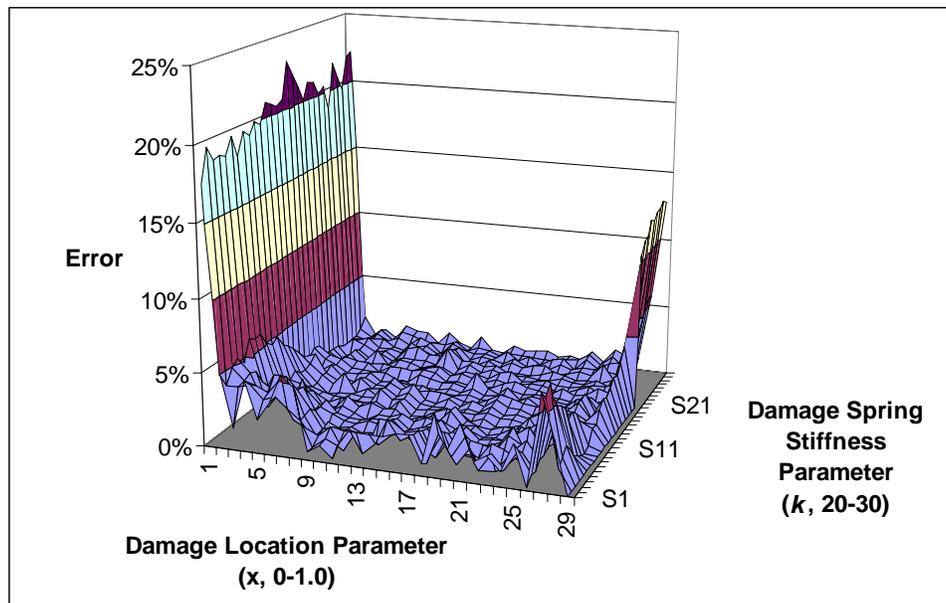


**Figure 6:** Distribution of testing and training points for a NN to identify two parameters.

Selecting the location  $x$  and the stiffness  $k$  values for 30 points each resulted in 900 training data points. To train NNs for 900 points, a radial basis NN was used instead of a multi-layer feed-forward NN, because a radial basis NN was better suited for the case in which a large number of testing data points are available.

A radial basis NN has one input layer, one hidden layer and one output layer and has a radial basis function as the transfer function. The difference between a feed-forward NN and a radial basis NN is that in the case of a feed-forward NN, a NN with a random parameter is created and the NN so created is trained with the training data set, but in the

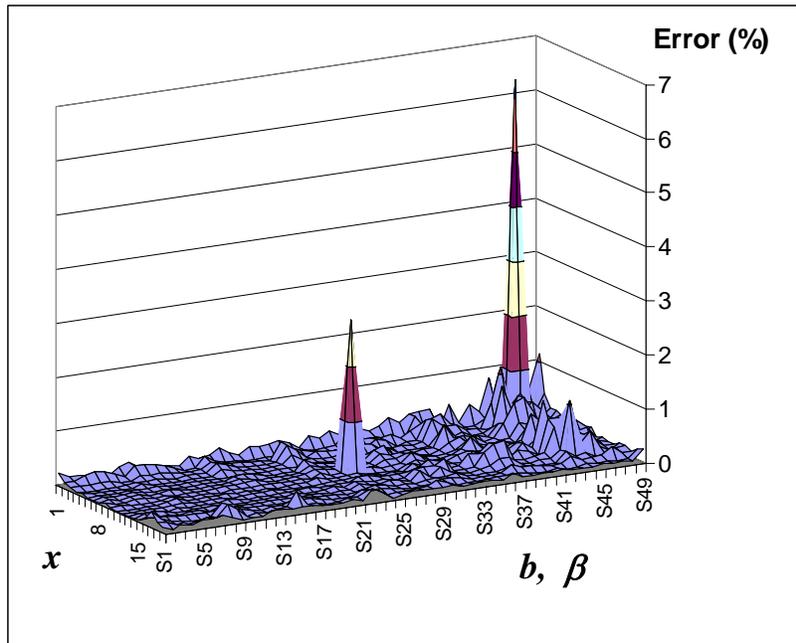
case of a radial basis neural network, a radial basis NN is created with a training data set that has near zero error vectors. When a multi-layer feed-forward NN was tried, it took too much time to train that NN. Therefore, the radial basis NN can be more suitable for a case with a large number of training data points. The ‘newrb’ function in MATLAB creates a radial basis NN by adding neurons to the network until the sum-squared error falls below an error goal or a maximum number of neurons have been reached. Figure 7 shows the case when the NN, which is trained with 900 data sets, was tested using the “test” data points.



**Figure 7:** The simulation errors of a trained NN at testing points in identifying two damage-parameters when the first three natural frequencies of the damaged beam are given (841 testing data points,  $x: 29, k: 29$ ).

Moreover, a NN was trained to identify three parameters. The training was done for 20 locations of  $x$ , 8 values of span size  $b$  between 0.05 and 0.2, and 8 different values of the

damage-magnitude parameter  $\beta$  between 0.1 and 0.5. In this case, a multi-layer feed-forward NN was used with resilient back-propagation training. The trained NN was tested for a new set of data which lies between the training points. The trained NN showed a relative error of only 1% for the testing points as shown in Figure 8.

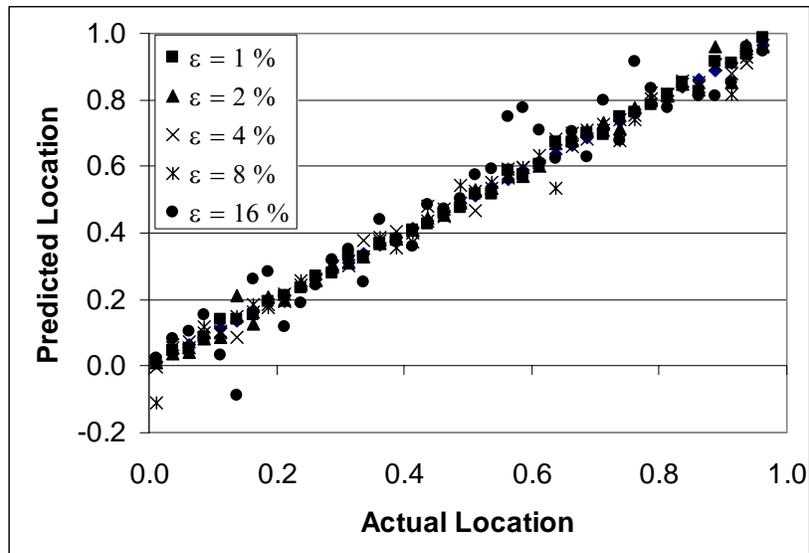


**Figure 8:** The simulation errors of a trained NN at testing points in identifying three parameters (931 testing points,  $x$ : 19,  $b$ : 7,  $\beta$ : 7).

### 3.1.2.1 Application to Noisy Data

The effect of measurement error in the frequency inputs was studied. The noise was simulated by introducing errors in the measurement of the difference between the natural frequencies of the damaged and the undamaged beam for the first five modes and these error infested data were used as inputs for the trained NNs to predict the location of

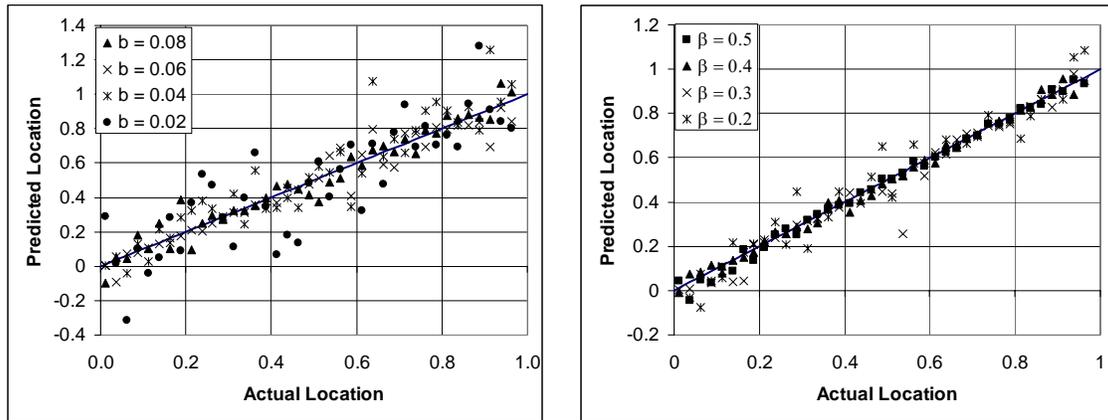
damage. Figure 9 shows the predicted location of damage versus the actual location for different degrees of errors in the measurements of the difference between the natural frequency of the damaged beam and that of the undamaged beam. The straight line with a slope of unity shows the points with no errors in predicting the damage location. From Figure 9, we can see that the error in the measurement should be less than 8% for NNs to correctly predict the location of damage.



**Figure 9:** Prediction of the location of damage for different degrees of error in the measurement of the difference of frequency between damaged and undamaged beam ( $b = 0.05$  and  $\beta = 0.3$ ).

Next, we investigated how well the damage in a beam can be predicted using error infested frequency data for different values of the span and the degree of damage. Figure 10 (a) shows the prediction of the location of damage for different values of the span of damage, b. Figure 10 (b) shows the prediction of the location for different values of the

degree of damage,  $\beta$ . The straight lines with a slope of unity here also show the points with no errors in predicting the damage location. From these figures, we can see that a trained NN can predict the location of the damage even though errors are present in the input, but the span of the damage should be larger than 8 % of the whole length of the beam and the reduction in the bending stiffness should be larger than 20 % of the bending stiffness value of an undamaged beam for the present neural network based approach to identify the damage.



(a) For different values of  $b$  (when  $\beta = 0.3$ ).

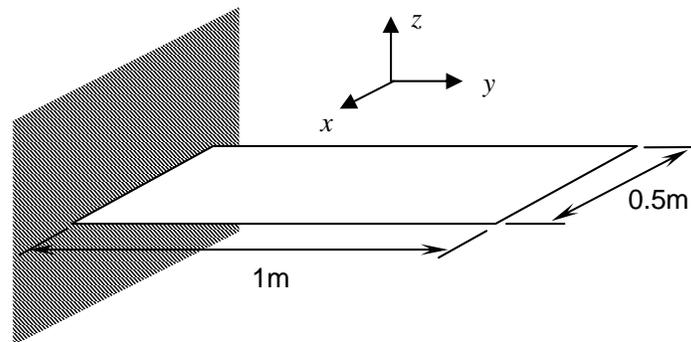
(b) For different values of  $\beta$  (when  $b = 0.08$ ).

**Figure 10:** Prediction of the location of damage for different values of the span and the degree of the damage ( $\epsilon = 8\%$ ).

## 3.2 Damage Identification in a Plate

### 3.2.1 Formulation of the Problem

Damage was identified in a cantilevered plate shown in Figure 11. NASTRAN was used to build a FE model and obtain dynamic response. The damage was represented as elements with reduced Young's modulus. To identify the location of damage, two properties of the dynamic behavior of the damaged plate were utilized. One was the transient displacement response and the other was the frequency response function. To calculate the transient displacement response, NASTRAN was used to calculate the modal stiffness and mode shapes of the plates repeatedly, each with a damaged zone at different locations. Next, the modal superposition method was used to calculate transient response based upon the transient input, the modal stiffness matrix, and the mode shapes of the plate for each case. PCA was used to reduce the dimension of the transient response before being used as input to NNs. To calculate the frequency response function, the fast Fourier transform (FFT) was applied to the transient response obtained using the mode superposition method.



**Figure 11:** A cantilever plate for damage identification problem ( $E = 62.05$  GPa,  $\nu = 0.35$ ,  $\rho = 2700$  kg/m<sup>3</sup>, thickness 5mm)

One of the problems in applying NNs to the damage identification problem was that we have to perform FE analysis for each damage location and feed the output file to the NN's training and testing. Therefore, a code in the MATLAB environment was developed that can automatically generate the input file for NASTRAN for different locations of damage, execute the NASTRAN analysis for each case, read the modal data from the NASTRAN output file, and perform the modal superposition method and FFT. Next, the same code was used to train NNs and test the trained NNs. This procedure is illustrated in Figure 12. This automatic code made it possible to generate training sets and testing sets, as many as needed, and to use the set directly in the NN analysis.

### 3.2.2 Modal Method

To solve the dynamic equation shown in equation (16), the reduced dynamic equation shown in (17) was used and this reduced dynamic equation was solved independently using piecewise linear approximation to obtain the transient displacement response at various locations.

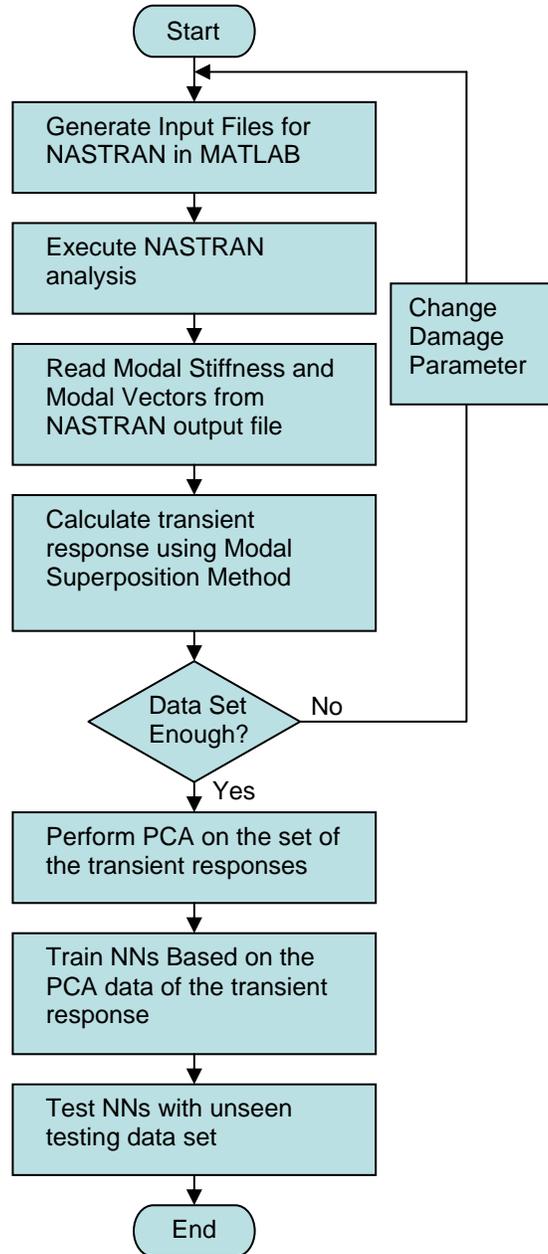
$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p}(t) \quad (18)$$

$$\bar{\mathbf{M}}\ddot{\boldsymbol{\eta}} + \bar{\mathbf{K}}\boldsymbol{\eta} = \bar{\mathbf{p}}(t) \quad (19)$$

$$\begin{aligned} \mathbf{u} &= \Phi \boldsymbol{\eta} \\ \bar{\mathbf{M}} &= \Phi' \mathbf{M} \Phi \\ \bar{\mathbf{K}} &= \Phi' \mathbf{K} \Phi \\ \bar{\mathbf{p}} &= \Phi' \mathbf{p} \end{aligned}$$

where  $\mathbf{M}$  is the mass matrix,  $\mathbf{K}$  is the stiffness matrix,  $\mathbf{p}(t)$  is the forced input vector,  $\mathbf{u}$  is the displacement vector,  $\bar{\mathbf{M}}$  is the modal mass matrix,  $\bar{\mathbf{K}}$  is the modal stiffness matrix,  $\bar{\mathbf{p}}$

is the modal force vector,  $\eta$  is the modal displacement vector, and  $\phi$  is the eigenvector matrix.

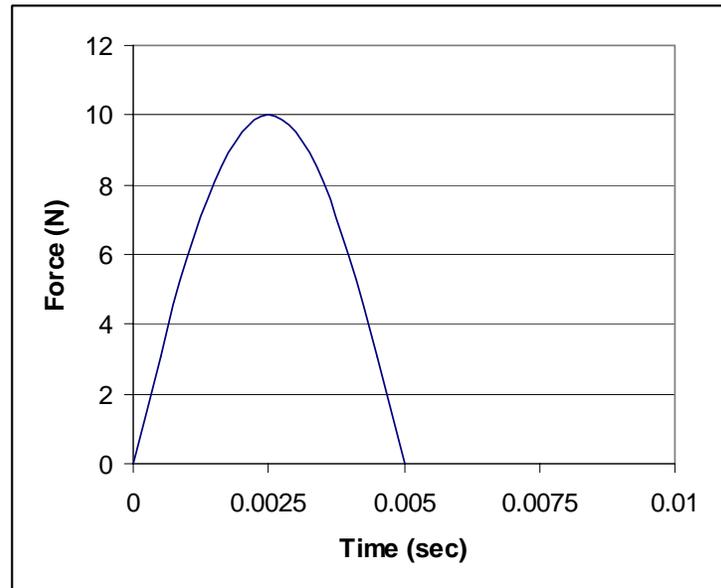


**Figure 12:** Procedure for obtaining dynamic response and training and testing NNs.

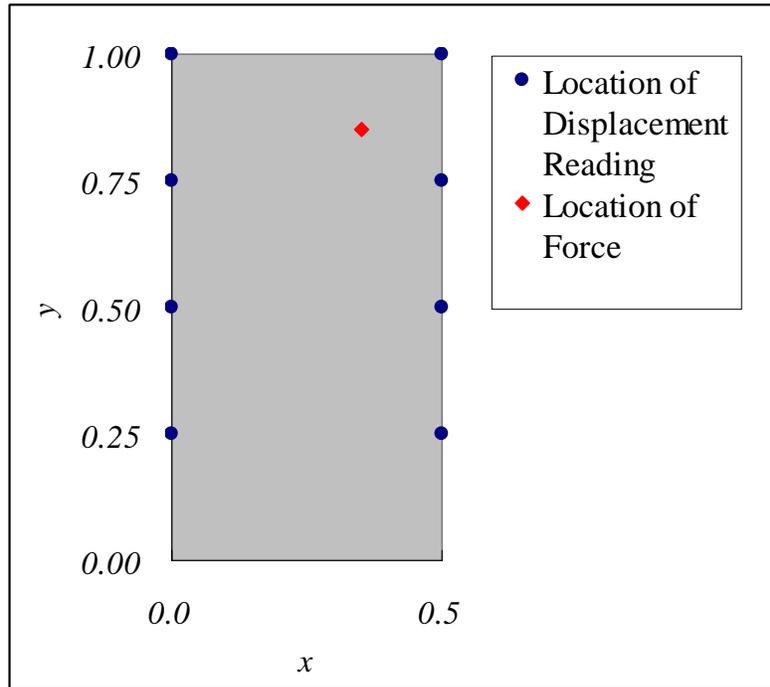
### 3.2.3 Direct Problem

The finite element model constructed for the damage identification has  $40 \times 80 = 3200$  elements. The training set was selected for 10 locations in the  $x$  direction and 20 locations in the  $y$  direction. This resulted in 200 training sets. The testing set was selected at the midpoint between the training points. This resulted in  $9 \times 19 = 171$  testing sets. For each set selected, the modal stiffness and the modal displacement vector were calculated using NASTRAN.

The transient out-of-plane response of the plate was calculated against the excitation by the half sine wave shown in Figure 13. The transient response was calculated at 8 locations using modal superposition method with 10 modes for each case. Figure 14 shows the location where the half sine wave is applied and the locations where the transient displacement vectors were calculated.



**Figure 13:** The time history of the force applied to the plate.



**Figure 14:** Location of the displacement reading and the location of the force applied.

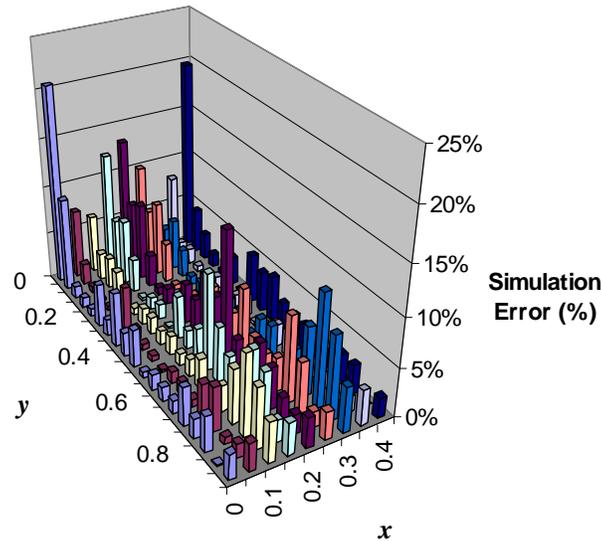
The transient out-of-plane displacement vectors at 8 locations were selected for the duration of 0.25 second.

### 3.2.4 Inverse Problem

#### 3.2.4.1 FFT of Transient Displacement as Input

The FFT of the transient out-of-plane displacement vector was used as the training inputs for NNs. The input vector was constructed from the first 200 frequency points that include the highest mode considered in the modal superposition method for each location of displacement calculation. The size of the training input matrix was 1600 by 200. The NN used was multi-layer feed-forward back-propagation NN. The resilient

backpropagation training scheme was used. The target was the  $x$  and  $y$  coordinates of the center of the damaged zone. Figure 15 shows the relative error of the trained NN in simulating the damage location with the FFT of the transient displacement vector. It took about 1 hour and 30 minutes to train the NN.

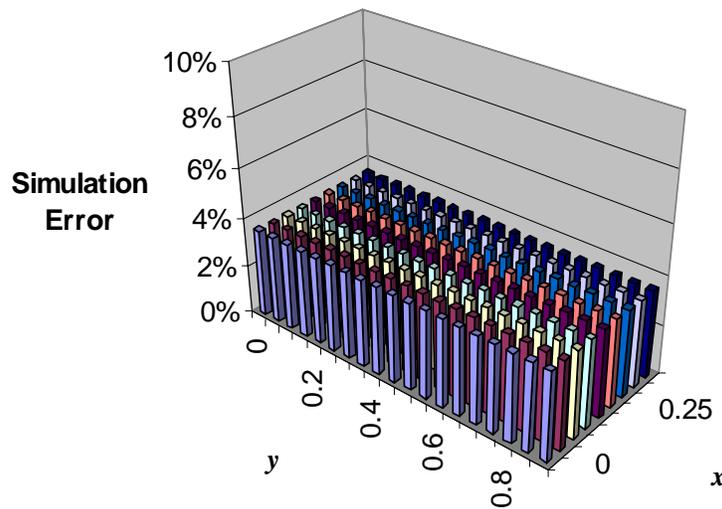


**Figure 15:** Relative simulation error of a NN trained with FFT of the transient displacement vector at each testing points placed midway between training data points in predicting the location of the damage. (Input Matrix =  $1600 \times 200$ , Time required for training = 1 hour 30 minutes)

### 3.2.4.2 PCA of Transient Displacement as Input

Next, PCA was applied to the transient displacement vector and reduced the dimension from 4000 to 44 when the relative error was set to be 0.1%. A NN was trained with the reduced set of principal components of the displacement vector. The training target was also the  $x$  and  $y$  coordinates of the center of the damaged zone. The size of the training

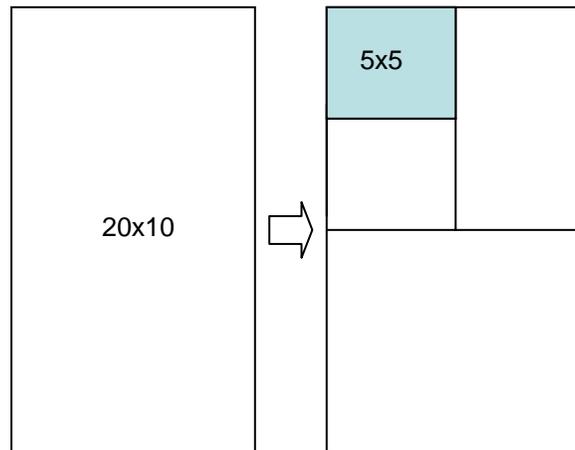
input matrix was composed of reduced principal components with 44 dimensions and a total of 200 cases. Therefore, the training input matrix was 44 by 200. The target was composed of two values for the location coordinates, and each coordinate values corresponded to each training input. Therefore, the training target matrix was 2 by 200. The training took 24 seconds on a Sun Blade 1000 workstation to reach the simulation error goal. Figure 16 shows the simulation errors of trained NNs relative to the length of the plate. Here, we can see that the simulation error is not more than 4 % and not less than 2%. The value of the average mean error in the case without PCA was close to that of the case where PCA of the displacement vector was used as input. However, there were more variations in the simulation error and there were certain points with high value of error as can be seen in Figure 15.



**Figure 16:** Relative simulation Error of a NN trained with PCA of the transient displacement vector at each testing point placed midway between training data points in predicting the location of the damage. (Input Matrix = 44×200, Time required for training = 24 sec)

### 3.2.4.3 Use of Orthogonal Arrays

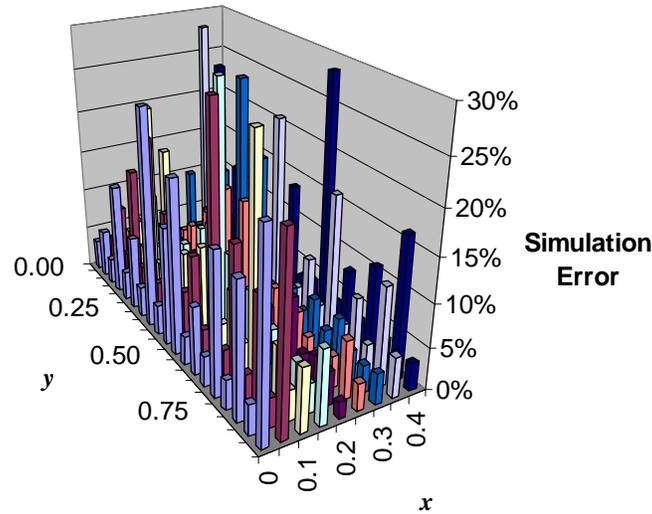
To select a smaller number of training sets, OAs were used. The original training space is 10 by 20 and composed of 200 cases. To apply an OA of known construction to this problem, the space composed of 10 by 20 design points was divided into a different parameter space; 10 x 20 was divided into 2 x 2 x 2 x 5 x 5 sections as shown in Figure 17 and an orthogonal array was used in selecting the minimum set. Table 2 shows part of an orthogonal array selected. The number of the set selected with this OA was half the number of the whole set. Therefore, the size of the training input matrix became 44 by 100 from 44 by 200. The training took only 7 seconds while the NN with PCA took 24 seconds and NN with FFT took one and one-half hour. Figure 18 shows the simulation errors to be between 2 % to 29%.



**Figure 17:** Dividing 20x10 sections to 2x2x2x5x5 sections.

**Table 2:** Part of an orthogonal array for 3 factors with 2 levels, 2 factors with 5 levels, and strength 2

	1	2	3	4	5
1	0	0	0	0	4
2	0	1	1	0	4
3	1	0	1	0	4
4	1	1	0	0	4
5	0	0	0	1	2
6	0	1	1	1	2
7	1	0	1	1	2
8	1	1	0	1	2



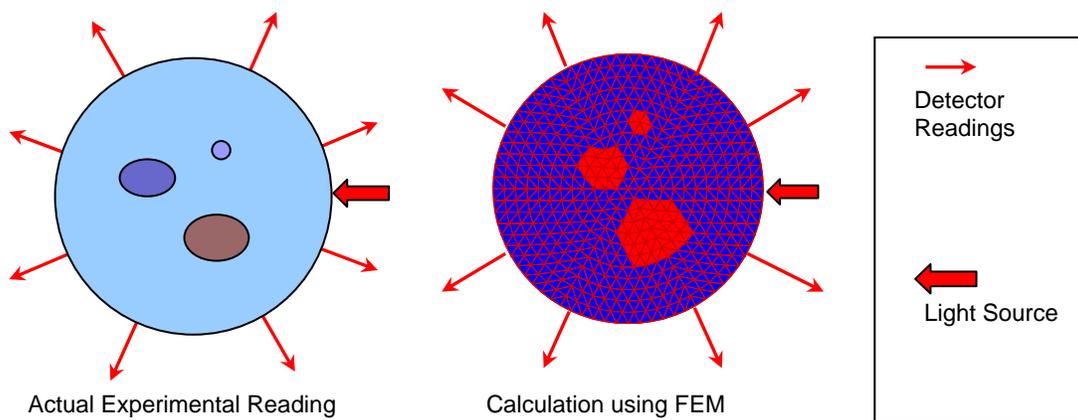
**Figure 18:** Relative simulation error of a NN trained with a training set that was obtained from PCA of the transient displacement vector and the selection from OA at each testing points placed in the midway between training data points in predicting the location of the damage (Input Matrix =  $44 \times 100$ , Time required for training = 7 seconds).

# Chapter 4

## OPTICAL IMAGING PROBLEM

### 4.1 Direct Problem

The first step in the optical imaging problem is solving the direct problem, which is, for given location and strength of a light source, calculating the intensity of the light or photon time of flight measurements at detector locations. This can be done using the finite element method that solves the photon diffusion equation. The photon diffusion equation is known to be a very good approximation of photon transport in human tissue [ 40 ]. A diagram depicting the direct problem solution is shown in Figure 19.



**Figure 19:** Solution of the direct problem.

The photon diffusion equation, which is the approximate solution of the photon transport equation, is shown below. This is valid for the scattering dominant case with isotropic source condition [ 40 ].

$$\frac{1}{c_n} \frac{\partial}{\partial t} \Phi(\mathbf{r}, t) - \nabla \cdot [D(\mathbf{r}) \nabla \Phi(\mathbf{r}, t)] + \mu_a(\mathbf{r}) \Phi(\mathbf{r}, t) = S(\mathbf{r}, t) \quad (20)$$

$\Phi(\mathbf{r}, t)$ : Photon intensity at position  $\mathbf{r}$  and time  $t$ .

$D(\mathbf{r})$ : Diffusion constant.  $F(t)$ : Diffusion intensity.

$S(\mathbf{r}, t)$ : Source strength.  $\mu_a(\mathbf{r})$ : Absorption coefficient.

$c_n$  : Speed of light in the medium

The solution of the light diffusion equation can be obtained by the finite element method using the Galerkin approach as follows:

$$[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)] \Phi + \mathbf{B} \frac{\partial \Phi}{\partial t} = \mathbf{Q} \quad (21)$$

$$\begin{aligned} K_{ij} &= \int_{\Omega} \kappa(\mathbf{r}) \nabla \psi_j(\mathbf{r}) \cdot \nabla \psi_i(\mathbf{r}) d\Omega, \\ C_{ij} &= \int_{\Omega} \mu_a(\mathbf{r}) c \psi_j(\mathbf{r}) \psi_i(\mathbf{r}) d\Omega, \\ B_{ij} &= \int_{\Omega} \psi_j(\mathbf{r}) \psi_i(\mathbf{r}) d\Omega \\ Q_j(t) &= \int_{\Omega} \psi_j(\mathbf{r}) q_0(\mathbf{r}, t) d\Omega, \\ \Phi &= [\Phi_1(t), \Phi_2(t), \dots, \Phi_D(t)]^T \\ \kappa(\mathbf{r}) &= \frac{c}{3[\mu_a(\mathbf{r}) + \mu'_s(\mathbf{r})]}, \end{aligned}$$

where  $\mathbf{K}$ ,  $\mathbf{C}$ , and  $\mathbf{B}$  are the system matrices,  $\mathbf{Q}$  represents source values,  $\psi_i$  are the element interpolation functions,  $\kappa$  is the diffusion coefficient,  $\Omega$  is the tissue domain,  $\mu_a$  is the absorption coefficient,  $\mu'_s$  is the reduced scattering coefficient, and  $c$  is the speed of light.

For the time independent case, expression (19) is reduced as follows:

$$[\mathbf{K}(\kappa) + C(\mu_a, c)]\Phi = \mathbf{Q} \quad (22)$$

### 4.1.1 Development of Finite Element Code

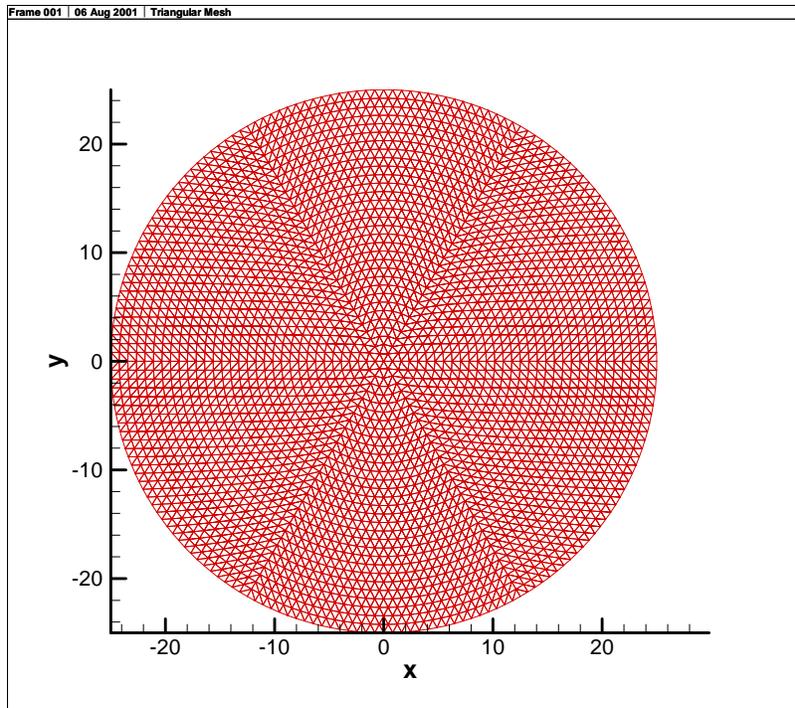
The triangular element is chosen to discretize the media because it is more convenient to construct a circular or an oval shape using triangular elements than using rectangular elements. A linear interpolation function is chosen as the interpolation function over each element. The developed code can easily generate a mesh with any level of detail as shown in Figure 20, which shows the mesh generated by the MATLAB code and visualized using Tecplot<sup>§</sup>.

The developed code can generate a finite element model for a circular shape when the radius of the circle and the number of elements across the radial direction is specified. In evaluating element matrices, exact integral formulae were used for integration over a triangle [ 55 ]

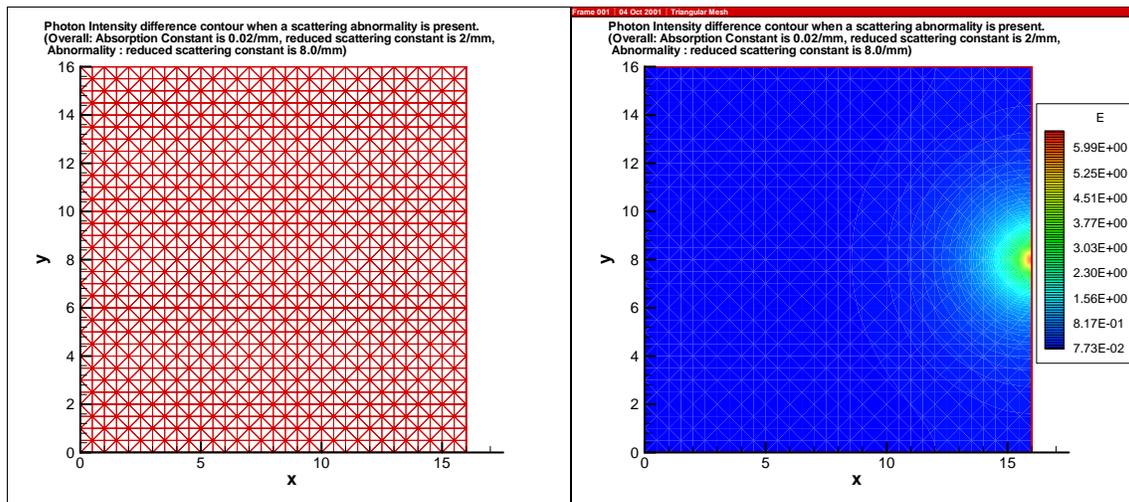
The code can also automatically generate a finite element mesh for rectangular shaped domains. Figure 21 shows the photon intensity contour throughout a rectangular shaped model when a light source is present on the right middle end. Figure 22 shows the photon intensity difference contour with and without an anomaly. Figure 23 and Figure 24 are for the case when the tissue shapes are circular.

---

<sup>§</sup> Tecplot is a registered trademark of the Amtec Engineering, Inc.

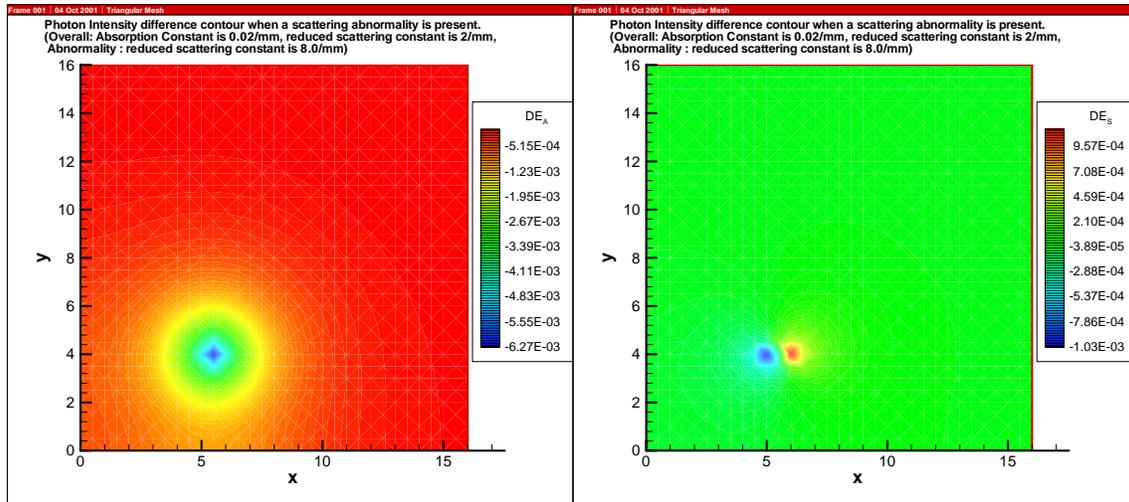


**Figure 20:** An example mesh using triangular elements for the optical finite element code as applied to a circular domain.



(a) Rectangular Finite element mesh (b) Photon intensity contour with one source.

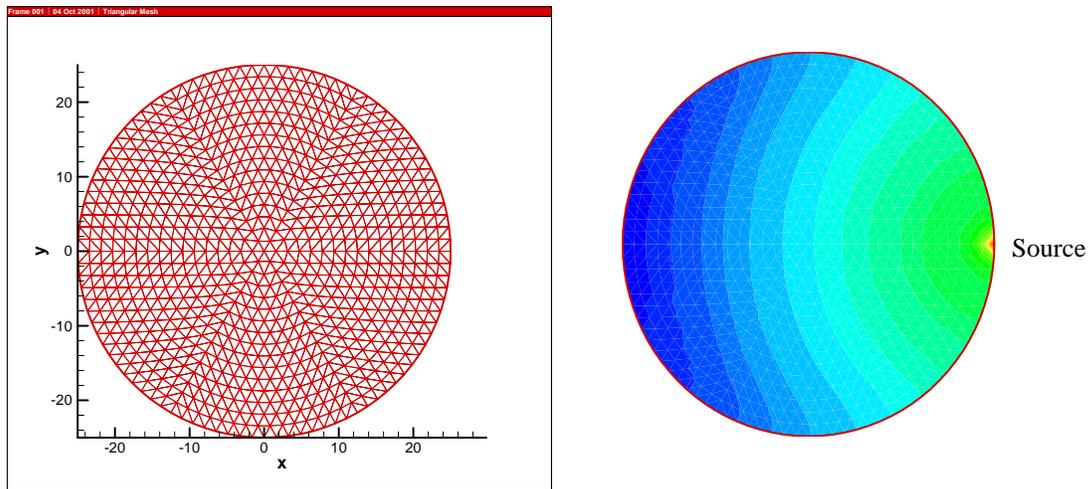
**Figure 21:** A finite element mesh and the corresponding photon intensity contour over a rectangular shaped tissue.



(a) Absorption anomaly

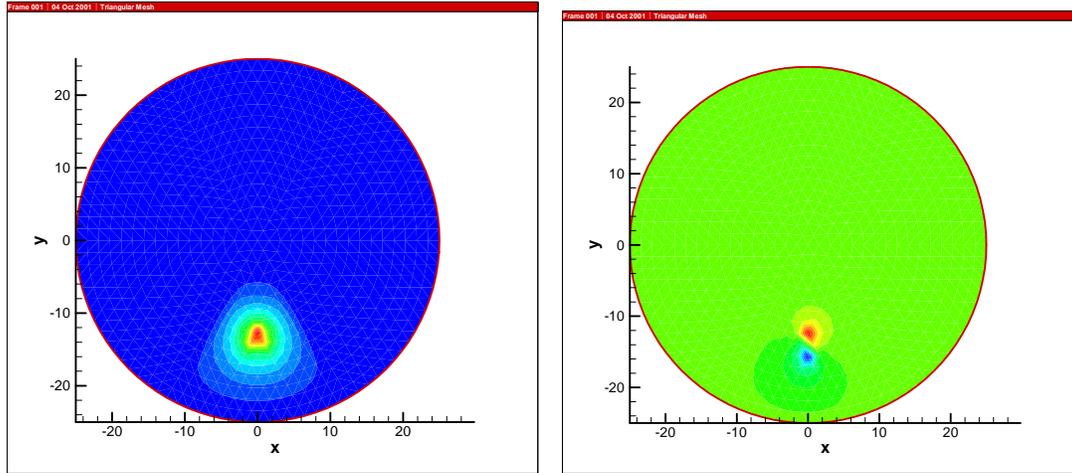
(b) Scattering anomaly

**Figure 22:** The contour of photon intensity difference with and without the presence of an anomaly for two types of anomalies, (a) absorption and (b) scattering anomaly.



(a) A finite element mesh for a circular domain. (b) Photon intensity contour.

**Figure 23:** Finite element mesh and photon intensity contour over a circular shaped tissue.



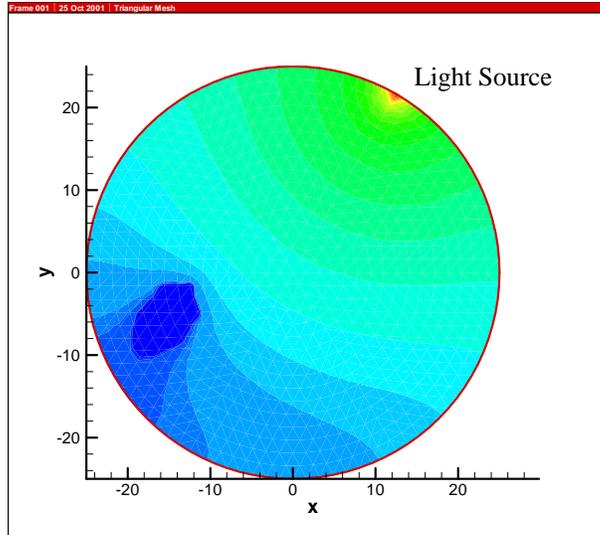
(a) An absorption anomaly

(b) A scattering anomaly

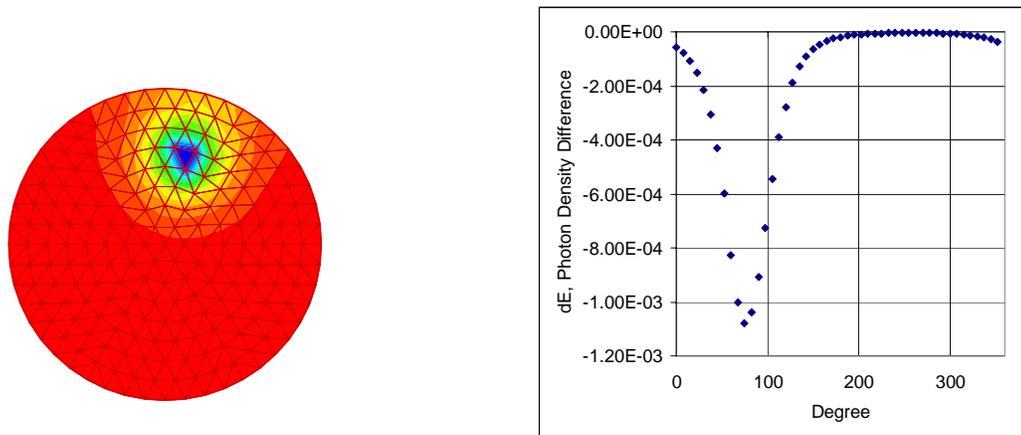
**Figure 24:** The contours of photon intensity-difference with and without anomalies for two types of anomalies.

Figure 25 shows a photon density contour when a light source is present at the top right corner. The contour clearly shows the disturbance of the contour when an anomaly is present in the medium. Figure 26 shows, along the boundary, the difference in the photon density when an anomaly is present as compared to when it is not. Based on this information, NNs can be trained for two different cases.

To be able to solve the inverse problem using NN, enough training cases should be generated to train a NN to correctly solve the inverse problem.



**Figure 25:** Solution of photon diffusion equation using finite element method for the case when an absorption anomaly is present in a homogeneous medium (medium:  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu_s' = 2.0 \text{ mm}^{-1}$ , anomaly:  $\mu_a = 0.5 \text{ mm}^{-1}$ ).

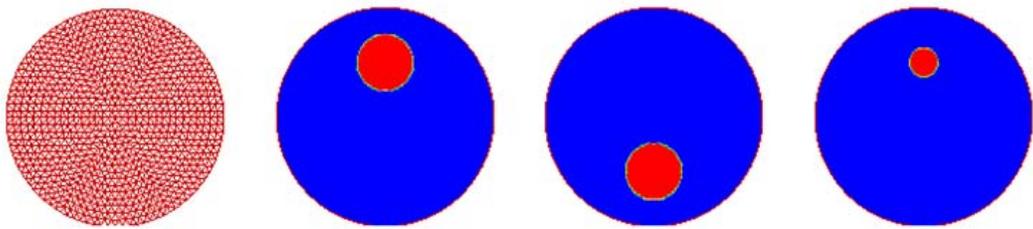


(a) Photon density difference contour. (b) Photon density difference along the edge.

**Figure 26:** Photon density difference with and without the presence of an absorption anomaly. (optical properties of tissue: medium:  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu_s = 2.0 \text{ mm}^{-1}$ ; anomaly  $\mu_a = 0.05 \text{ mm}^{-1}$ )

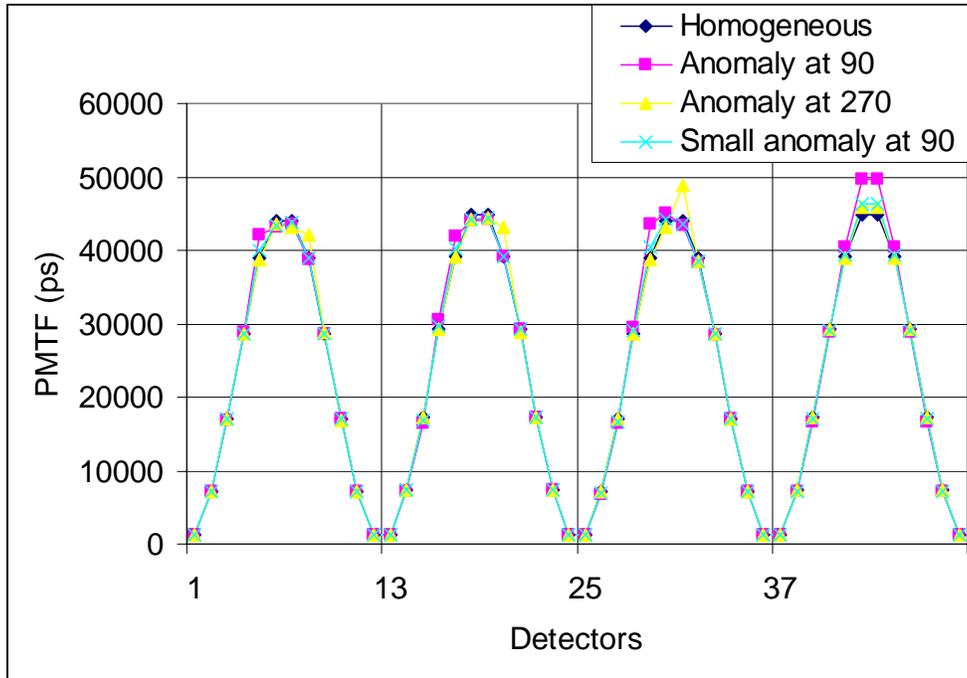
### 4.1.2 Characteristics of the Solution of the Direct Problem

The results on the photon mean time of flight (PMTF) calculation at the boundary of the tissue were compared for different anomaly locations and sizes. Figure 27 shows the case of a homogeneous tissue and the results from three trial cases in the presence of an anomaly in the tissue. Figure 28 shows the PMTF calculation for each case. Here, we can observe some distinctive patterns in each case at the peak of each detector reading graph. The pattern can be clearly seen when we see the difference between the homogeneous case and each of the anomaly cases in Figure 29. These distinctive patterns can be used in solving the inverse problems in NN. The locations of these peaks in the detector reading graphs are obtained by the detectors located at the opposite end of the light source. Because this distinctive pattern happens at the detectors located at the opposite end of the source, it is clear that the calculation of PMTF needs a higher resolution of FE model.

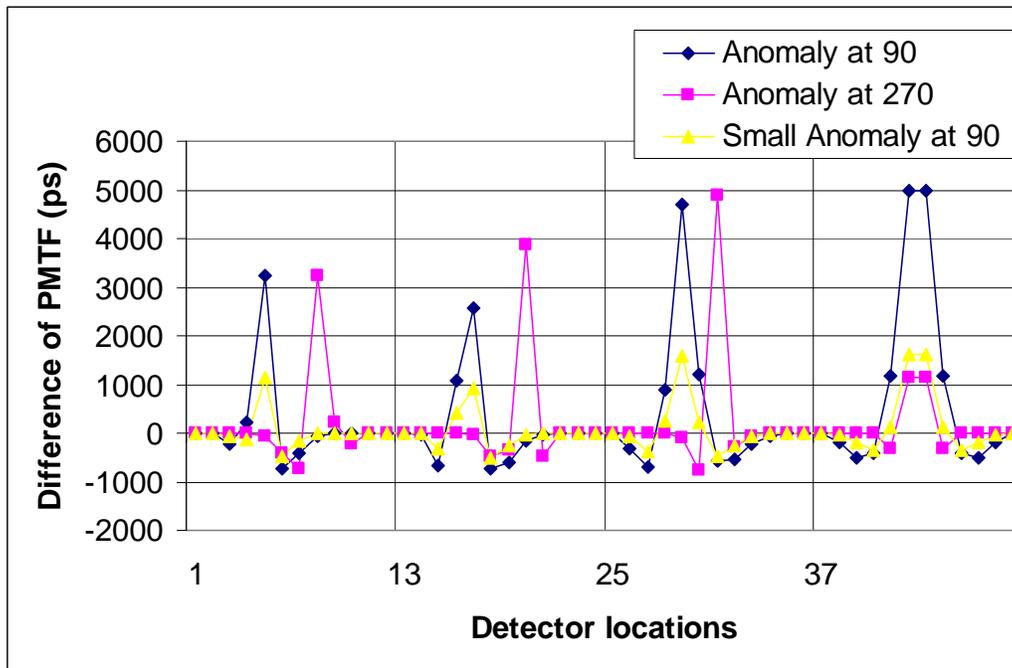


(a) FE mesh      (b) anomaly at  $90^\circ$     (c) anomaly at  $270^\circ$     (d) small anomaly at  $90^\circ$

**Figure 27:** Finite element mesh and three anomaly locations for calculating PMTF and light intensity readings at detectors.



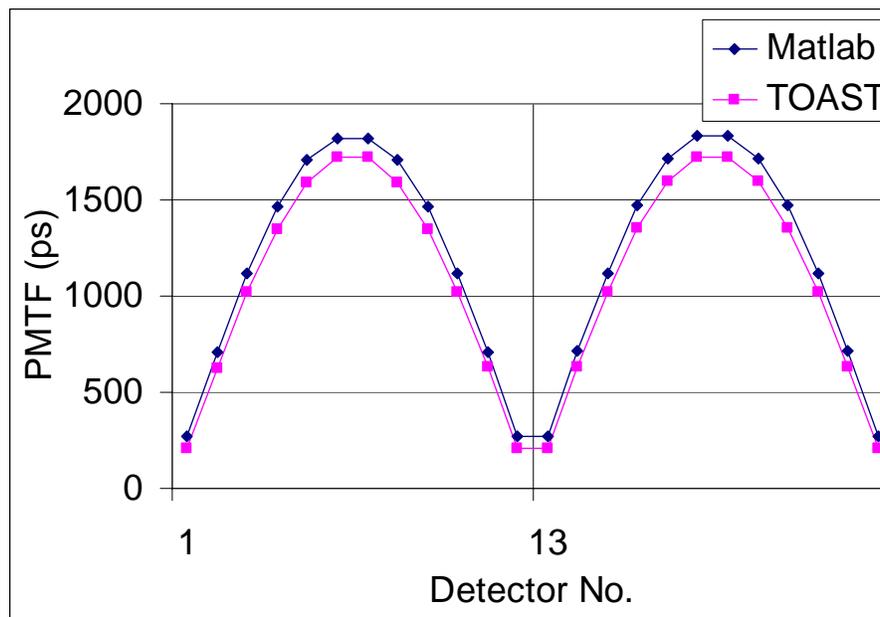
**Figure 28:** Detector readings corresponding to a source placed at four locations.



**Figure 29:** Difference of PMTF between homogeneous and anomaly cases for 4 source locations.

### 4.1.3 Validation of Finite Element Code

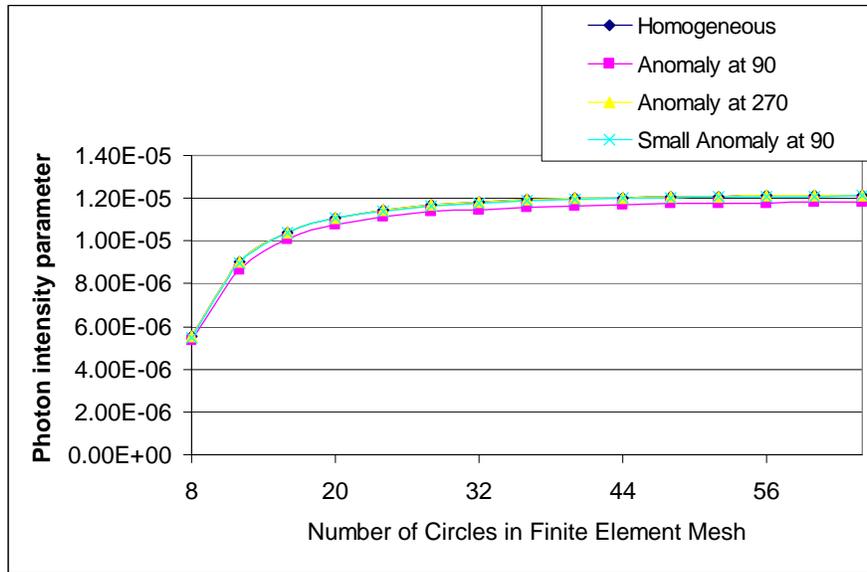
Figure 30 shows the comparison of the PMTF calculation using both the code developed here and TOAST, an optical imaging program developed by the Biomedical Research Group at London College. The difference between the two results is due to the fact that different boundary conditions were used in each method. The simpler Dirichlet condition was used in our code and the Robin condition was used in TOAST.



**Figure 30:** PMTF calculation using MATLAB and TOAST (Matlab Result: 24 circles, Dirichlet boundary conditions, Collimated source condition; TOAST Result: 32 Circles, Robin boundary conditions, Collimated source condition)

Next, convergence studies were performed for the finite element model. When the number of elements was increased, the detector readings converged to a certain value as

seen in Figure 31. The studies were done up to a case with 24,576 elements, with 64 concentric circles in the mesh.



**Figure 31:** Convergence study of the finite element model for various anomaly locations.

## 4.2 Inverse Problems

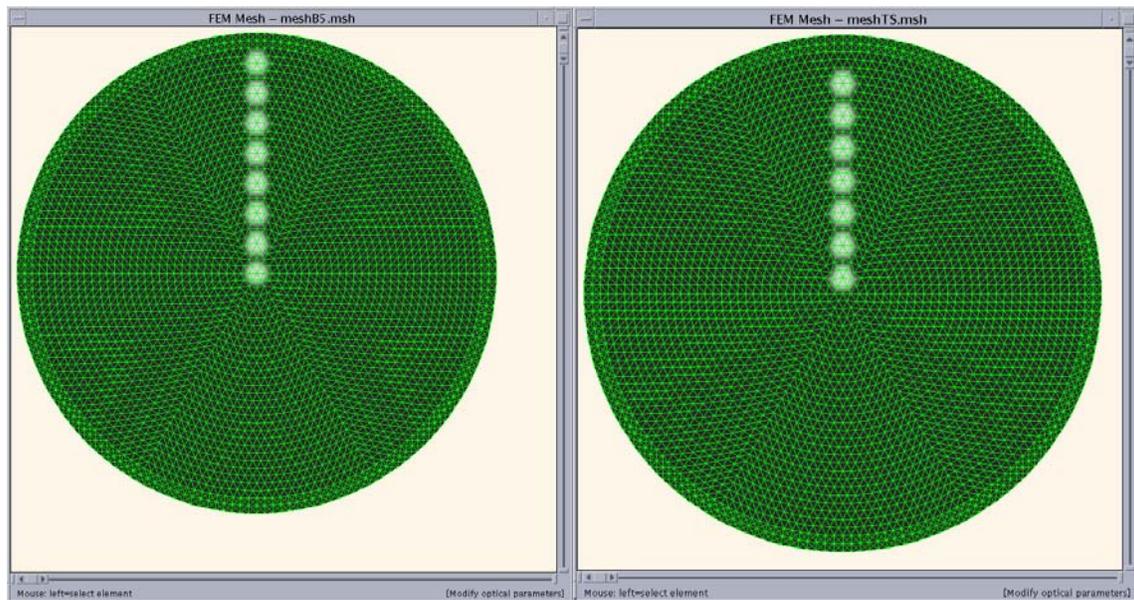
Various methods and complexity were utilized in using NNs in identifying an anomaly in a circular shaped tissue. A simple problem that only identifies one parameter was solved first, and the complexity was increased gradually to the case where the number of outputs was the same as the number of elements in a finite element model. Throughout the solution, a multi-layer feed-forward neural network with two sigmoid hidden layers and linear output layers and resilient back propagation training scheme were used except in one case that identifies both the orientation and depth of an anomaly. Here, the tissue is assumed to have a large degree of scattering coefficient, which is the valid assumption to use the photon diffusion equation to approximate photon transport in tissue, and the anomaly is assumed to have different values of the scattering coefficient and larger values of the absorption coefficient. The exact values of the coefficients are different from problem to problem and are noted in the figure captions.

Preliminary results were obtained using an optical finite element program available on the Internet (Biomedical Optics Research Group's internet homepage, <http://www.medphys.ucl.ac.uk/research/borg/index.htm>, 2001.) but subsequently a finite element code was developed to solve the direct problems. Because utilizing an existing finite element code had limitations in regards to the code's integration with the MATLAB Neural Network Toolbox, such as in generating enough training cases, we have developed our own finite element code for solving the direct problem in two dimensions as part of solving the inverse problem. The code was developed in the

MATLAB environment for an easy and efficient integration with the MATLAB Neural Network Toolbox.

#### 4.2.1 Identification of the Depth and the Orientation of an Anomaly

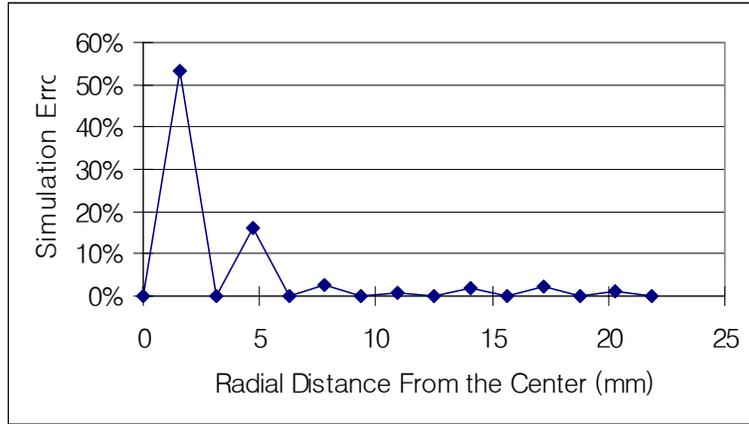
In the first case, the neural network was only trained in one radial direction for different locations. Figure 32 (a) shows the training data set that was used to generate training data set for NN. A NN was trained with 8 data sets from FEM to locate an abnormality in radial directions. Figure 33 shows the simulation error when the NN was tested with testing data generated from 7 data points between training data points. Here we can see that simulation errors near the center are higher.



(a) 8 Training Data Points

(b) 7 Testing Data Points

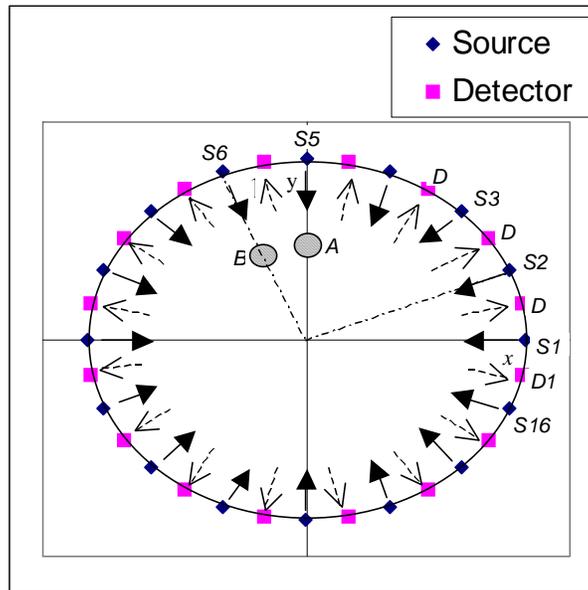
**Figure 32:** Finite element mesh for the abnormality lined up in uni-radial direction.



**Figure 33:** Simulation error compared testing data set.

The results from the finite element method were obtained only for cases when the abnormality is lined up along the positive y-axis as shown in Figure 32.

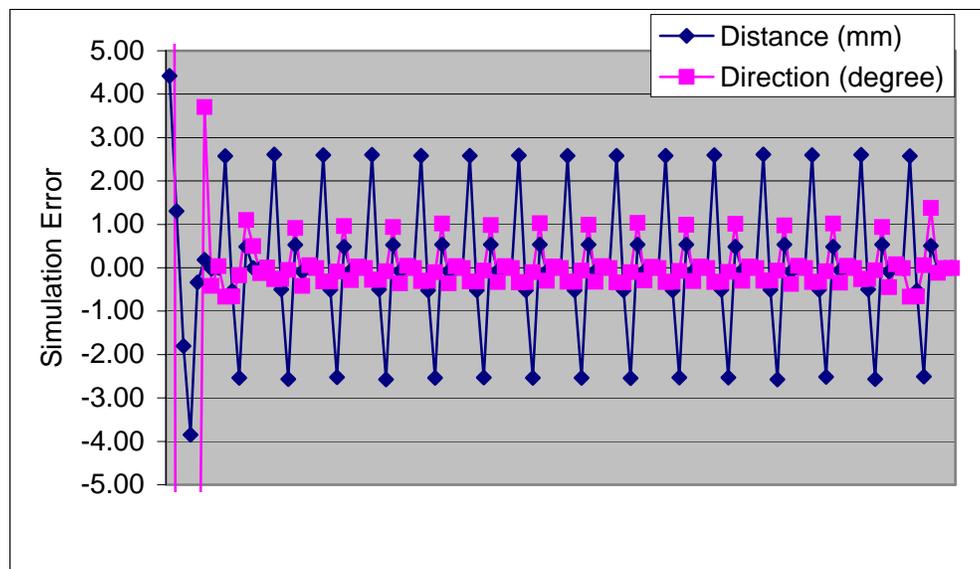
For locations other than on the positive y-axis, axi-symmetry of the cross section was utilized to generate training data. As can be seen in Figure 34, the detected intensities at D1 and D2, when the abnormality is at B and the light source is at S1, are the same as the intensities at D16 and D1, when the abnormality of the same size is at A and the light source is at S16. In this way, the training data set can be easily generated for all the directions of sources using data for the cases where the abnormality is on the positive y-axis.



**Figure 34:** Array of Light sources and detectors.

The total number of data sets for each location was  $1+16\times 7=113$ . Each set contained  $16\times 16=256$  training data and 2 target data. The target data are the direction and the distance from the center of the abnormality.

Radial basis neural network was used because the amount of data was large. The training was completed with success in simulating the direction of the abnormality but with some errors in predicting radial distance in the case when the abnormalities were near the center, as shown in Figure 35.

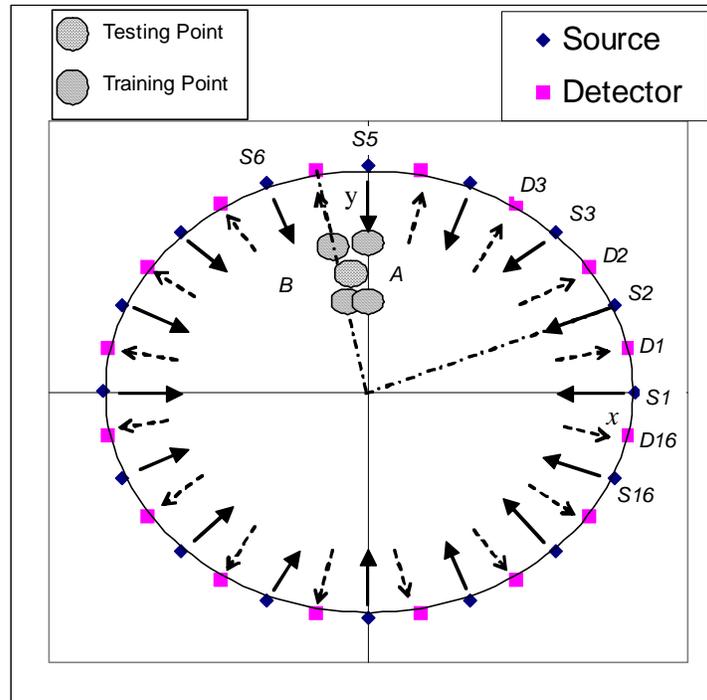


**Figure 35:** Error between target obtained from trained NN and training target.

In the further analysis of optical imaging with NN, it has been found that radial-basis NN is less efficient than backpropagation feed forward NN in converging to the training data. Although a feed forward neural network takes a lot of time to train with a large amount of data, feed-forward neural network seems to be a more viable structure to solve optical imaging problems.

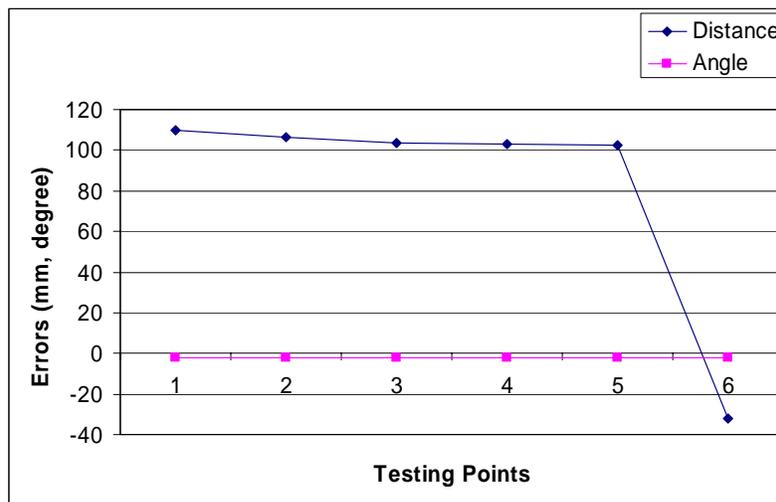
Figure 36 shows one example of training and testing data sets on one segment. In this study, the number of training points along the radial direction was 7 and the number of testing points was 6. There are two targets in this neural network. They are the distance

from the center and the direction of the abnormality. These two values yield the two dimensional location of the abnormality.



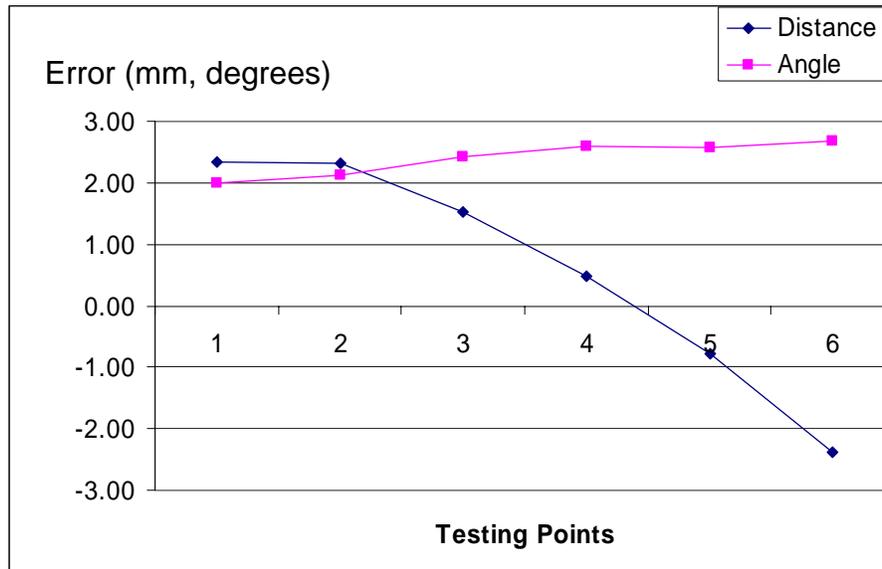
**Figure 36:** Testing points and training points on the tissue.

Figure 37 shows the simulation error in the case of a radial-basis neural network. The simulation error for distance from the center was too high to give any useful information.



**Figure 37:** Simulation error for the testing points with radial basis neural network.

Figure 38 shows the simulation error for the testing points with feed forward neural network. We can see that the simulation error for the distance from the center in the case of feed forward NNs is greatly reduced. Table 3 shows the simulation of training data for the trained neural network. In this Table, the simulated distance and the angle from a trained feed-forward NN is the same as, or very close to, the training data. However, we can see that the simulation output in the case of a radial-basis neural network for the distance failed to converge to the training data. The reason that a radial basis neural network (RBF) was less efficient than a back-propagation feed-forward multi-layer neural network (MLP) seems to be that the result of MLP is a global approximation and that of RBF is a local approximation [ 1 ]. Therefore, MLP can be more efficient when there are many local optimals. So, the method of using RBF to find the direction of abnormality and MLP to find the distance from the center of the abnormality seem to lead to the correct solution. Therefore, we need to use a feed-forward neural network to result in a viable solution even though it would take a much longer time to train the network.



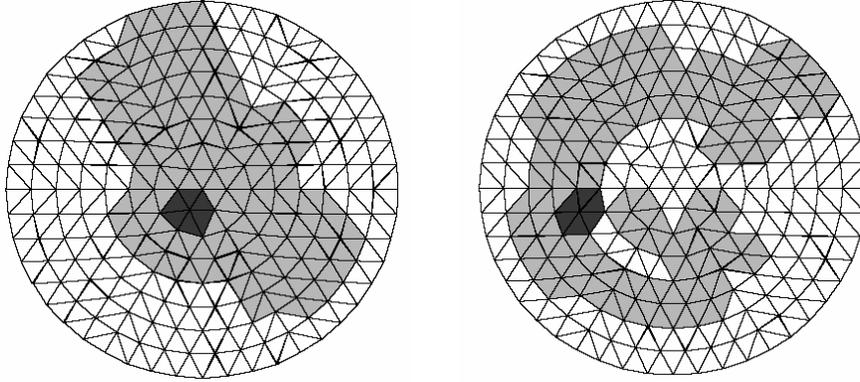
**Figure 38:** Simulation error for the testing points with feed forward neural network

**Table 3:** Simulation outputs for the training points from the trained neural network.

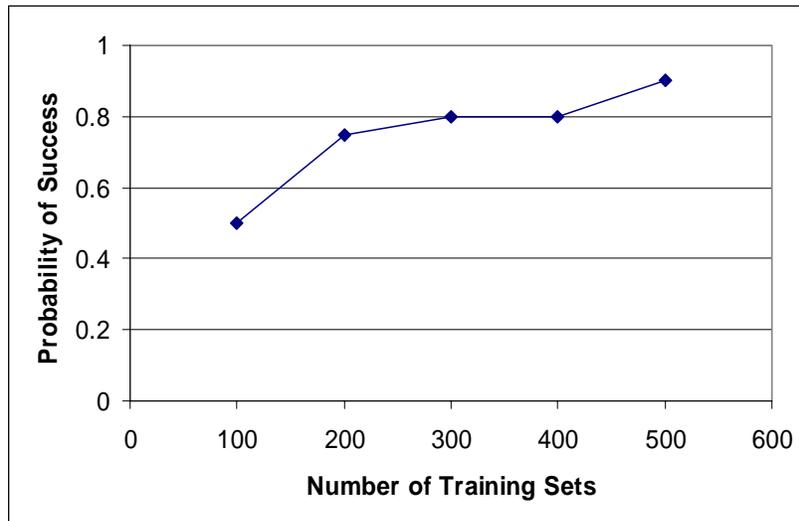
Distance	Radial	7.65	7.66	7.78	8.80	14.67	19.07	21.85
	FF	3.13	6.25	9.38	12.50	15.62	18.75	21.87
Angle	Radial	101.25	101.25	101.25	101.25	101.25	101.25	101.25
	FF	101.25	101.25	101.24	101.26	101.24	101.25	101.25

### **4.2.2 Identification of the Existence of an Anomaly**

In human tissue, the rough shape of the cross section of a tissue would be similar for different people but there would be variety of variations in its size and detailed shape. So, we wanted to find out whether a trained NN would be able to tell the existence of an anomaly in a non-homogeneous medium with known absorption and scattering properties but having an unknown distribution of its constituents such as fat and parenchyma. Figure 39 provides examples of a non-homogeneous medium with anomalies. The white-colored medium is the fat and the light gray-colored area is the parenchyma. The dark-colored region is the anomaly. The NN was trained with randomly selected parenchyma locations and shapes and both with and without anomalies. After training, the probability of a correct prediction was found to be only 50% with a NN trained with 100 training sets. However, this probability increased to 70% for a NN trained with 200 samples, and to 90% for a NN trained with 500 samples. Figure 40 shows the increase in the probability of correctly predicting the presence of an anomaly as the training set size increases. From this, we can see that it is possible for a trained NN to tell whether a region having an abnormal property exists in a tissue of known absorption and scattering properties but with an unknown distribution of fat and parenchyma.



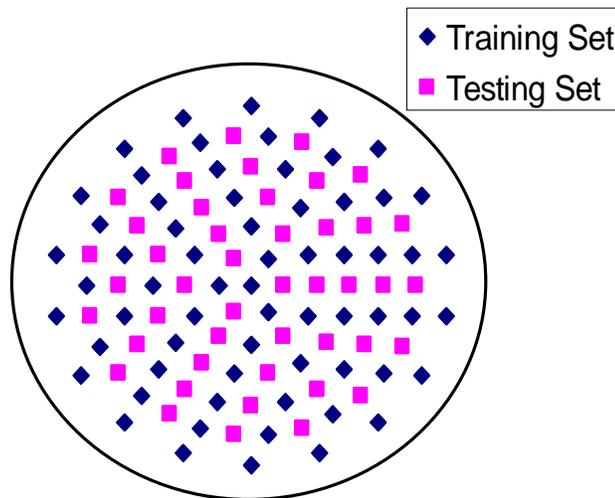
**Figure 39:** Examples of an anomaly in a non-homogeneous medium with different shapes of inside composition. (tissue properties: white area: medium,  $\mu_a = 0.01 \text{ mm}^{-1}$ ,  $\mu'_s = 1.0 \text{ mm}^{-1}$ ; gray area: parenchyma,  $\mu_a = 0.03 \text{ mm}^{-1}$ ,  $\mu'_s = 0.5 \text{ mm}^{-1}$ ; dark area: anomaly (tumor)  $\mu_a = 0.5 \text{ mm}^{-1}$ ,  $\mu'_s = 3.0 \text{ mm}^{-1}$ )



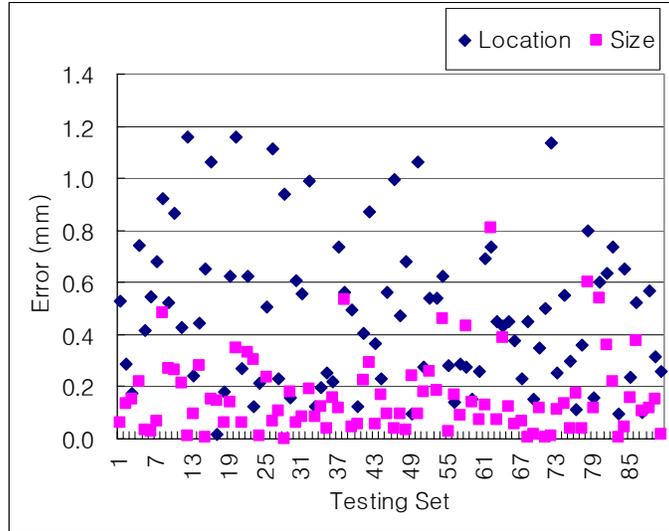
**Figure 40:** Probability of successfully predicting the existence of an anomaly in a non-homogeneous medium by a trained NN for different number of training sets, which were randomly generated.

### 4.2.3 Identification of the Location and the Size

In the next case, a NN was trained to identify three anomaly parameters, which are the location and the size of an anomaly. They are two coordinate values of the center location of the anomaly and the radius of the anomaly. The training was done based on training sets distributed evenly over the cross section. Figure 41 shows the locations of testing and training points. The training succeeded in converging to a simulation error of  $10^{-5}$ . Figure 42 shows that the simulation error displaying the deviation in location prediction was mostly less than 1 mm in a sample with an overall diameter of 50 mm. From these results, we can expect that higher accuracy can be achieved by using a higher number of training sets and a finer mesh.



**Figure 41:** Distribution of training and testing set for a NN that predicts the location and the size of an anomaly. (Optical properties of tissue: medium  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu'_s = 2.0 \text{ mm}^{-1}$ , anomaly  $\mu_a = 0.05 \text{ mm}^{-1}$ )



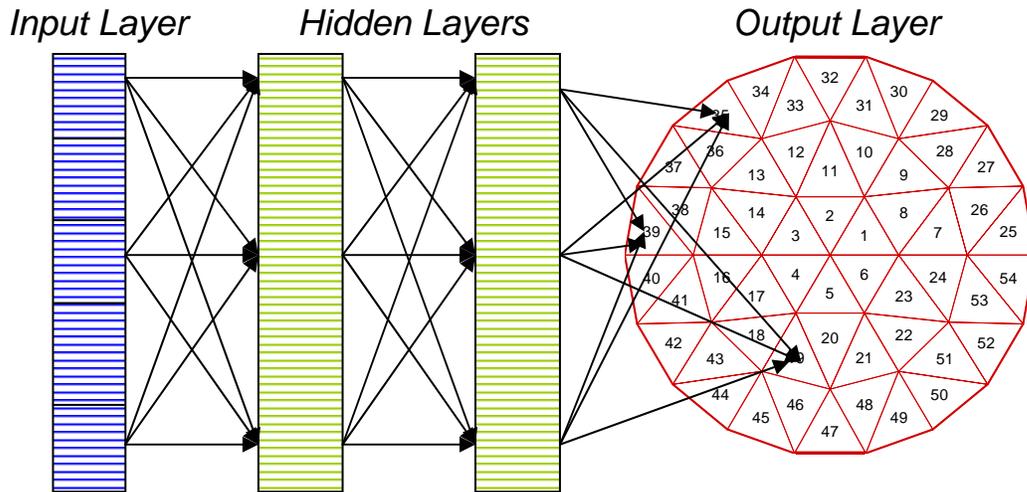
**Figure 42:** The simulation error of a NN in predicting the location and the size of an anomaly in FE-model with 384 elements.

#### 4.2.4 Identification of the Property of An Element

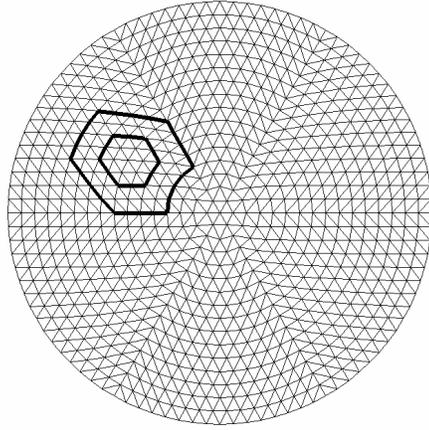
Next, we trained a NN in a different way. We assigned each output neuron to represent the absorption coefficient of each element in the finite element model as shown in Figure 43. In this way, drawing contours based on the values of the simulation output of neurons corresponding to each finite element can reveal the intrinsic tomography of the section.

We used two finite element meshes having different resolutions for solving the direct and inverse problems, respectively, for restudying the aforementioned case. Recall, in that case, each output neuron represented the absorption coefficient of an element in the finite element mesh. In Figure 44, the two meshes are shown for the case with the presence of

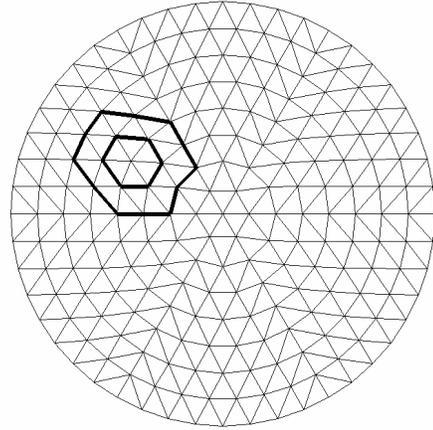
an anomaly in the tissue. The fine mesh, with the anomaly present, is used in solving the direct problem to simulate various detector readings. At the same time, the absorption coefficients of each element in the corresponding coarse mesh with the same shape of anomaly are used as the training targets for the NN's output layer. The training was done for every possible location in the coarse mesh with two different sizes of the anomaly as shown in Figure 44. In this way, we have more accurate results for the direct problem and a smaller mapping space for the NN. The training now took 3000 epochs. This is much shorter than the case when only a coarse mesh was used for obtaining both the data for simulating the measurements and for training the neural network. When only a coarse mesh was used, it took more than 6000 epochs for training.



**Figure 43:** The structure of a multi-layer feed-forward NN to identify the property of each element. Each output neuron corresponds to the parameters of each element.



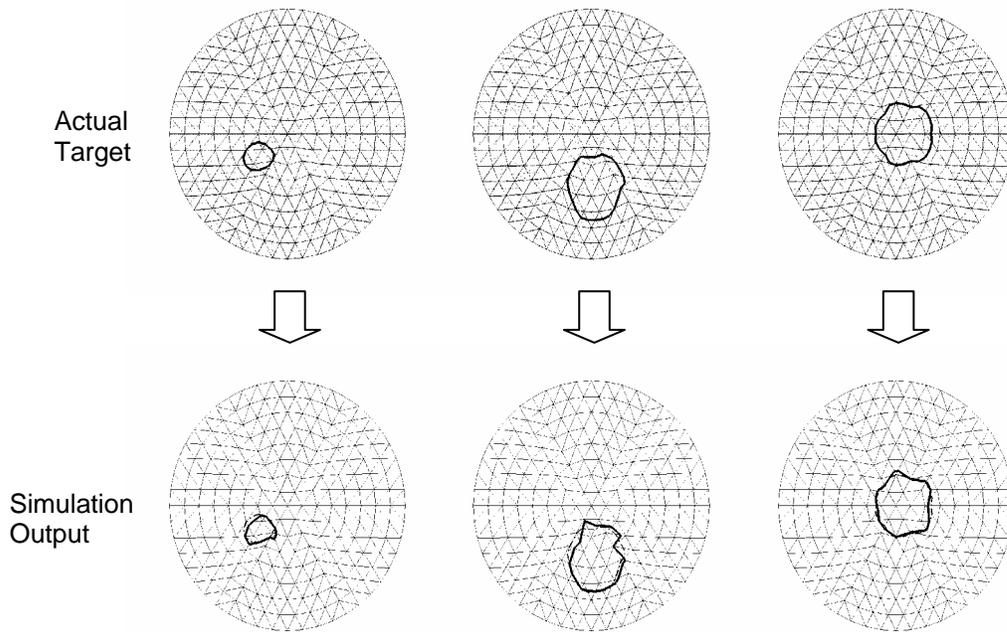
(a) Fine mesh for the direct problem



(b) Coarse mesh for the inverse problem

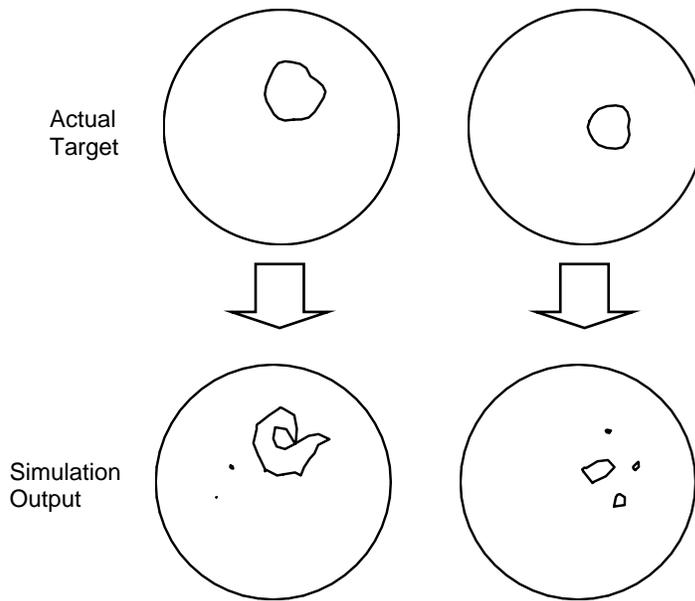
**Figure 44:** Fine mesh for the direct problem solution and coarse mesh for the inverse problem solution showing two sizes of anomalies corresponding to each mesh used in the training. (fine mesh: 1536 elements ; coarse mesh: 384 elements; optical properties of tissue: medium:  $\mu_a = 0.025 \text{ mm}^{-1}$ ,  $\mu_s = 2.0 \text{ mm}^{-1}$ ; anomaly  $\mu_a = 0.5 \text{ mm}^{-1}$ )

The trained NN was tested with new data obtained using the fine mesh and with an anomaly whose edges do not coincide with those of the anomaly that can be represented in a coarse mesh. In four out of five testing cases, the NN revealed quite similar anomaly shapes. Figure 45 shows the close prediction given by a trained NN. Although Figure 46 shows less accurate results, we can still observe that the NN gives good information about the anomaly. In this figure, both the actual target in the testing set and the simulation result by the trained NNs to inputs corresponding to the actual target are shown on the fine and coarse meshes. Here, we can see that although the trained NN has only seen the anomaly whose edges coincide with those of a coarse FE-mesh, the trained NN can successfully predict the shape of an anomaly whose edges do not coincide with the edge of the coarse FE-mesh.



**Figure 45:** Actual target and simulated output of trained NNs displaying the edges of an anomaly on fine and coarse finite element meshes in the testing set.

In the optical imaging problem, the orientation of an anomaly in a circular shaped tissue was identified based on integrated intensity, and photon mean time of flight was based on the detector readings placed around the tissue as depicted in Figure 19. The properties of photons which have passed through the tissue can be calculated using photon transportation theory. A good approximation to this theory is the photon diffusion equation. The finite element method can be readily used to solve this photon diffusion equation to obtain the property of light which went through the tissue. Two kinds of photon properties were calculated in this study. One is the time-independent integrated light intensity and the other is the photon mean time of the flight. The methods to solve these two properties followed Arridge et al.'s work [ 40 ] [ 41 ] and are explained in the appendix.

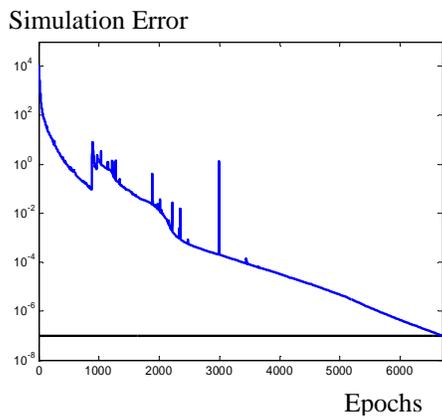


**Figure 46:** The worst cases in the simulation results using different meshes for direct and inverse problems.

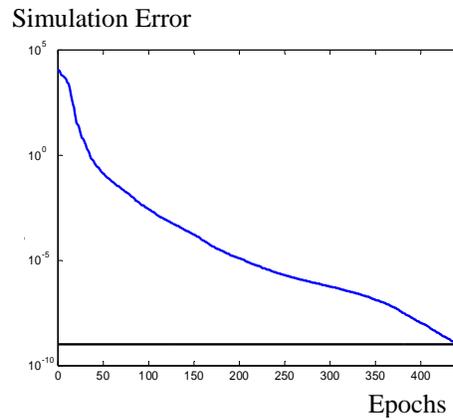
There are various kinds of source and boundary conditions in tissue optics. The Dirichlet boundary condition assumes a perfect absorbing medium surrounding the domain. The Robin boundary condition assumes a non-scattering medium surrounding the domain. The Dirichlet boundary condition was assumed in our analysis because of its ease in application.

For source conditions, two kinds of conditions are mainly used in tissue optics. One is a collimated source condition which is represented by an isotropic point source located at a depth. The other is a diffused source condition which assumes that an inward directed diffuse photon current is distributed over the illuminated boundary segment. The collimated source condition was assumed in our analysis.

In the numerical experiment, the dimension of the original training set for the model with 20 sources and 20 detectors was 400. When the contribution limit set was 0.075% for photon mean time of flight (PMTF) inputs, the dimension was reduced from 400 to 80 after principal component analysis. In the case of integrated intensity (E), the dimension reduced to 82 from 400 when the contribution limit was set to 0.005%. As shown in Figure 47, principal component analysis saves training time and achieves a smoother training process. Figure 48 shows the simulation error for different combinations of inputs. Among the various cases, we can see that a combined input of PMTF and E with principal component analysis gave the most accurate results.

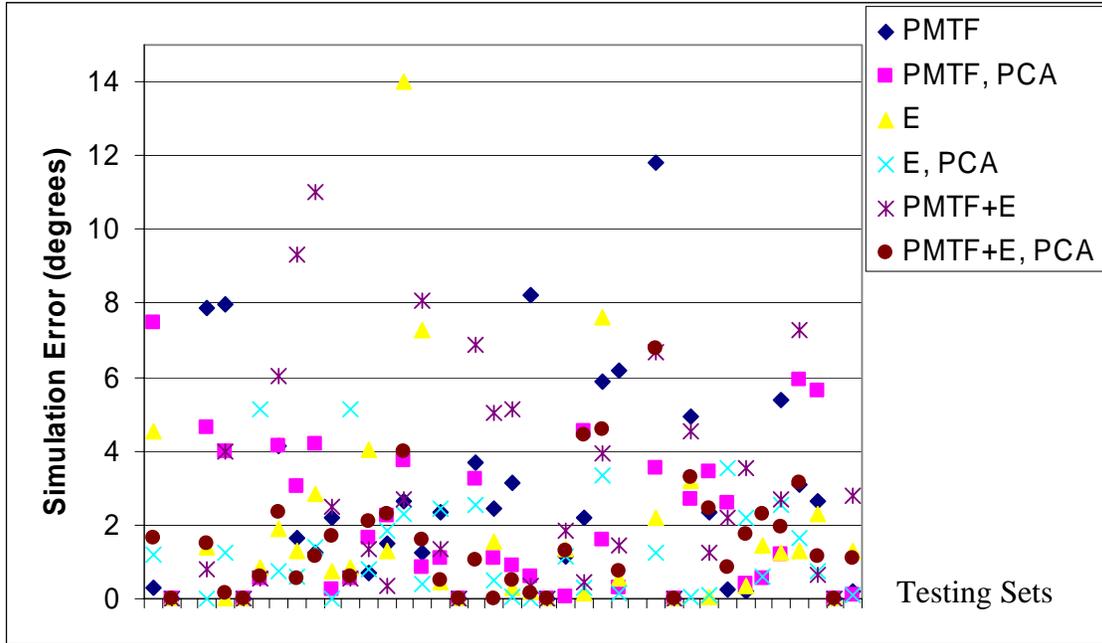


(a) PMTF, 6697 Epochs.



(b) PMTF with PCA, 441 epochs.

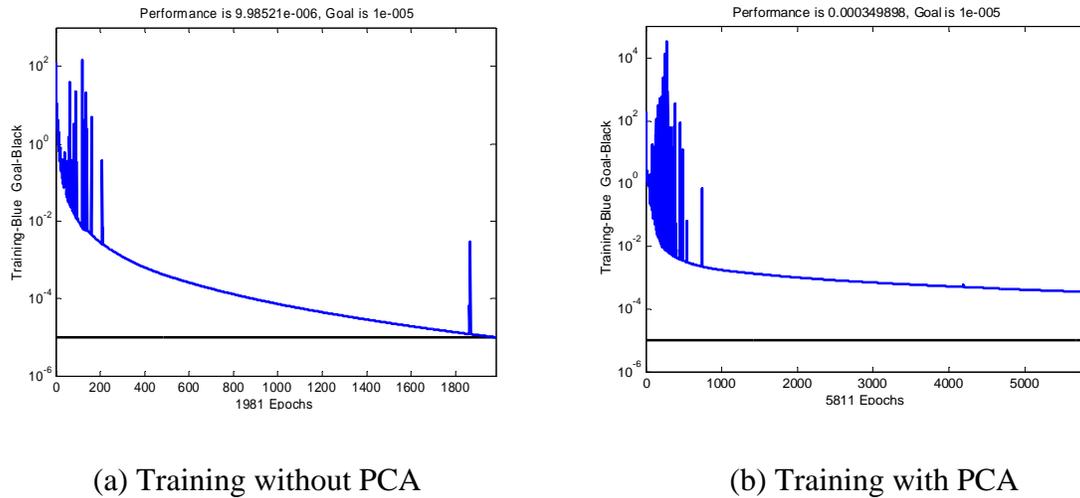
**Figure 47:** The learning process for various sets for the training inputs.



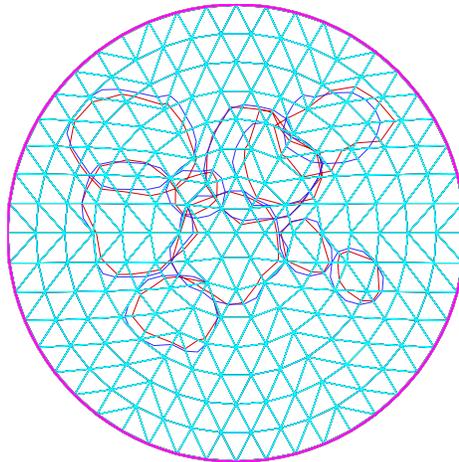
**Figure 48:** The simulation error in predicting the orientation of an anomaly (E: integrated intensity; PMTF: photon mean time of flight; PCA: principal component analysis).

Next, Principal Component Analysis (PCA) was used in reducing the size of the training set for the element property simulation NN, whose output neurons simulate the absorption constants of each element in the finite element model. The PCA reduced the size of training input from a  $144 \times 274$  data set to a  $72 \times 274$  data set, thereby reducing the time to train the NN by a factor of four and increasing the probability of success in the prediction of the edge of an anomaly using integrated photon intensity measurements. Figure 49 (a) shows the convergence of the simulation error in the learning process without principal component analysis on the training input. It didn't reduce the error to less than  $10^{-4}$  even after 5,811 epochs. Figure 49 (b) shows the convergence of the simulation error in the learning process with principal component analysis. Here, we can see that it took only 1,981 epochs to reduce the error below  $10^{-5}$ . Figure 50 shows the

simulated edges of anomalies by NN in red along with the corresponding edges of testing targets in blue. Here, we can see that in 10 out of 10 testing cases, the NN trained with PCA closely identified the edge of the anomaly. In the case of the NN trained without PCA, the prediction was successful in 9 out of 10 testing sets.



**Figure 49:** Training process of the NNs to identify element properties

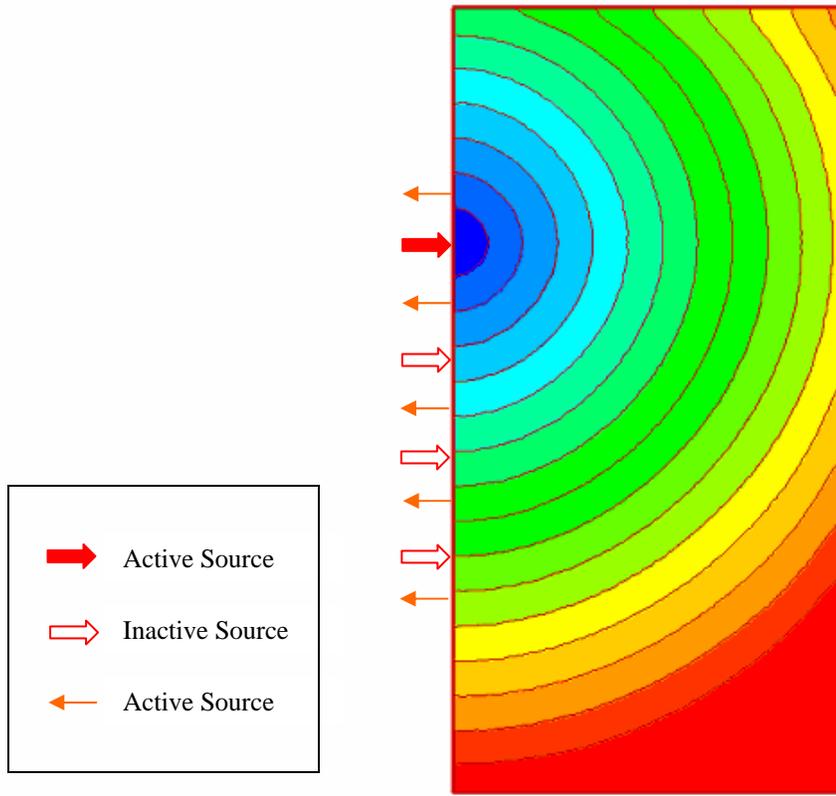


**Figure 50:** Simulation of element properties for 10 cases using a NN trained with PCA. Each case was tried separately. (Blue line: edge of testing absorption anomaly, Red line: edge of simulated anomaly)

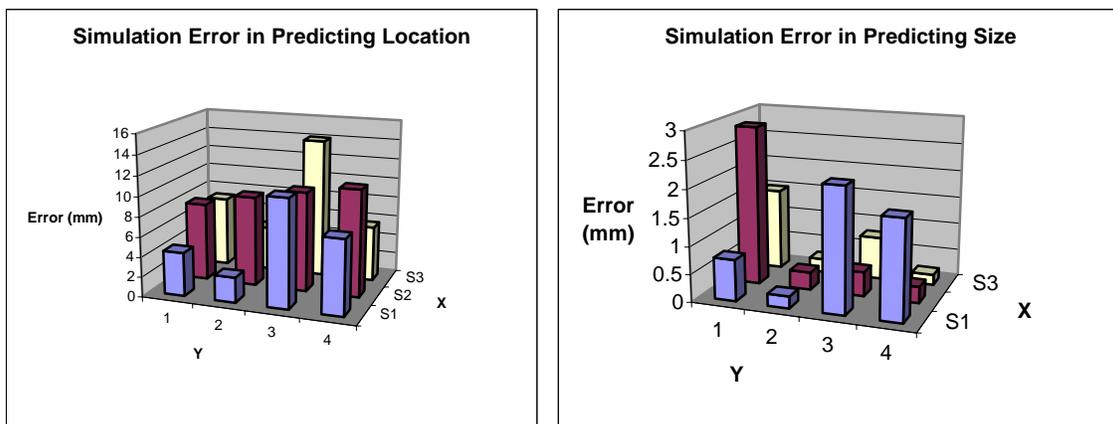
#### **4.2.5 Identification with One-Sided Information**

In the next case, the case when the light sources and the detectors were arranged on only one side of a tissue sample was studied. For this purpose, a rectangular shaped mesh was created and the training and testing sets for the NN were constructed by using the finite element method. From the solution of the direct problem, it was found out that the anomaly should be less than 25mm in depth for the detectors to have reasonable readings. Figure 51 shows the arrangement of the light sources and the detectors, along with contours of the light intensity when one light source is active.

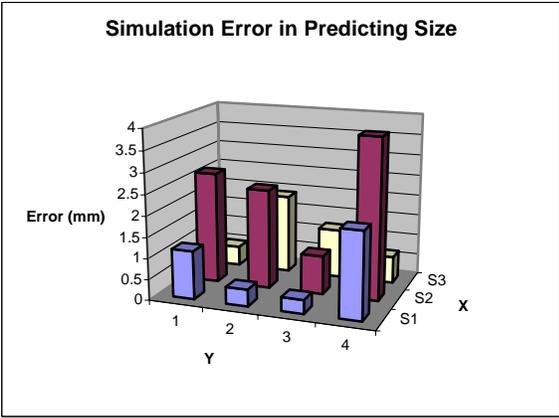
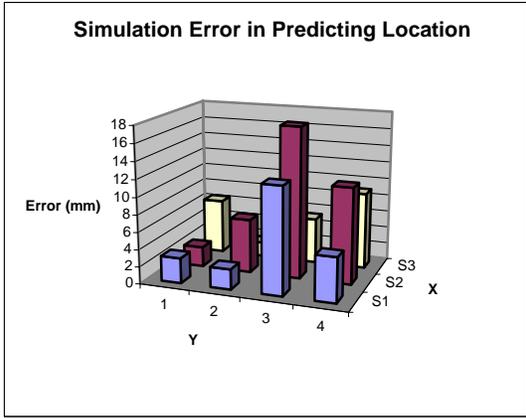
Feed-forward backpropagation NN were used in this case as well. Figure 52 shows the simulation error of a trained NN in identifying a smaller anomaly, and Figure 53 shows the simulation error of a trained NN in identifying a larger anomaly. Both figures show somewhat reasonable results for their size and errors, and we can say that one-sided arrangement imaging can be possible using NNs.



**Figure 51:** Contour plot of mean time of flight in a rectangular shaped tissue model with one active source.



**Figure 52:** Identification of smaller anomalies (Radius of an anomaly = 2.5 mm)



**Figure 53:** Identification of larger anomalies (Radius of Anomaly = 5 mm)

## **Chapter 5**

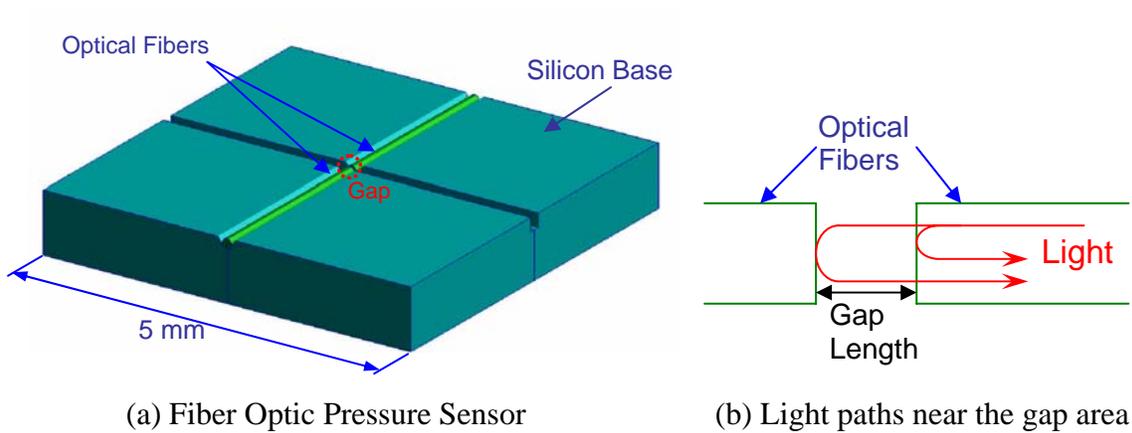
# **STRUCTURAL OPTIMIZATION OF A FIBER OPTIC SENSOR**

### **5.1 Direct Problem**

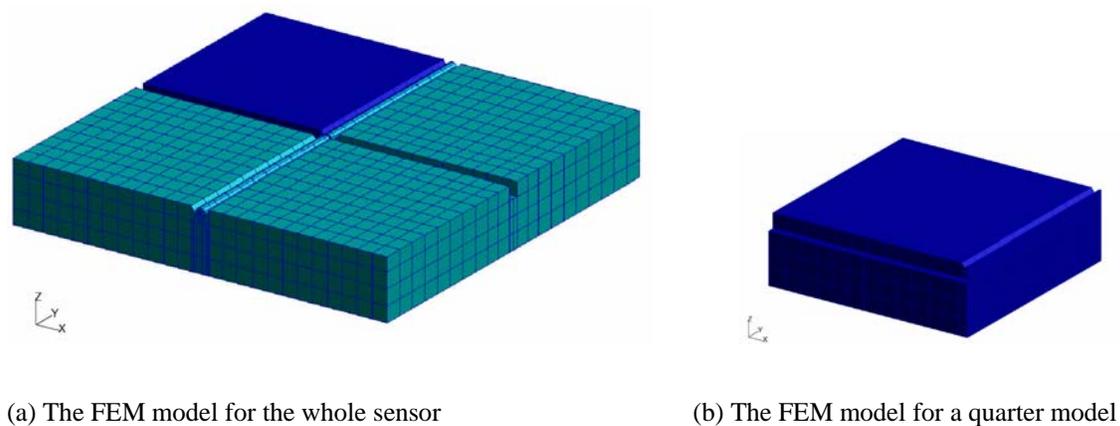
#### **5.1.1 Finite Element Modeling**

The miniature fiber-optic sensor analyzed here is a sensor which is used to accurately measure pressure change in air by utilizing the phase change in an optical fiber in the sensor. Figure 54 (a) shows the geometric drawing of the sensor and Figure 54 (b) shows the detailed drawing around the gap area of the sensor displaying two light paths in an optical fiber which reflect from two surfaces. The change in the phase difference between two light paths occurs as a result of the change in the air density in the gap area due to pressure changes in surroundings. This change is measured to accurately determine a pressure change. However, the structural vibration can also result in the change in the phase difference. Therefore, structural analysis and optimization is essential to minimize the interference between structural vibration and air density change.

A three-dimensional finite element model was used to analyze and optimize the miniature fiber optic sensor. This model represents one quarter of the sensor that represents the whole model by employing two planes of symmetry. Figure 55 shows the full finite element model and one quarter of the finite element model. The quarter model is represented using a hexagonal finite element that has 20 nodes. The model has 4,593 nodes and 838 elements, and has 13,677 degrees of freedom.



**Figure 54:** Geometric drawing of the miniature fiber-optic pressure sensor and the light paths around the gap area.



**Figure 55:** Finite Element Model of the miniature fiber-optic sensor.

### 5.1.2 Normal Mode Analysis

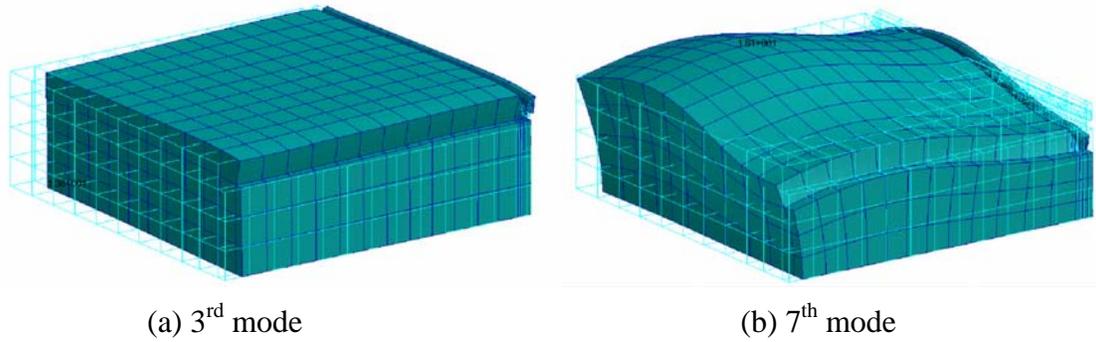
A modal analysis was performed to obtain the natural frequencies and the modal displacement vector of the sensor. The calculation was done using the MSC/NASTRAN normal mode solver. In studying the normal modes of the sensor, two kinds of boundary conditions were considered. One was a sliding bottom boundary condition and the other

was a fixed bottom boundary condition. The natural frequencies for the two cases are shown in Table 4.

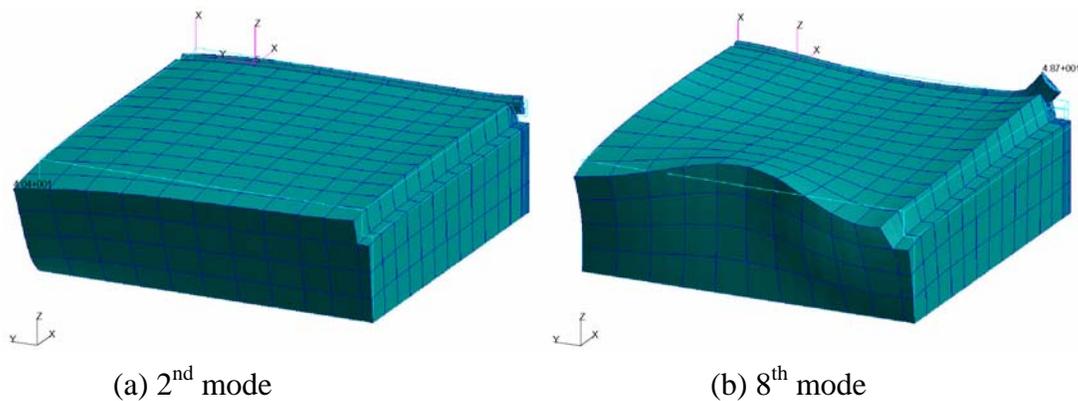
**Table 4:** Natural frequencies of the miniature fiber optic sensor for different boundary conditions

Mode No.	Natural Frequency (KHz)	
	Sliding Bottom Case	Fixed Bottom Case
1	647.05	1648.5
2	722.96	1728.2
3	818.86	1802.4
4	1404.3	1964.4
5	1588.4	2243.9
6	1929.3	2411.2
7	2019.2	2437.8

The mode shapes for each case were also studied to find the modes that can affect the displacements of the fiber at the gap. For the sliding boundary condition, the 3<sup>rd</sup> mode and the 7<sup>th</sup> mode were found to be most influential on the displacement at the gap area. Figure 56 shows these two modes. For the fixed bottom condition, the 2<sup>nd</sup> and the 8<sup>th</sup> modes were found to be most influential on the response of the sensor, as shown in Figure 57.



**Figure 56:** The mode shapes of the sensor with sliding bottom boundary condition.

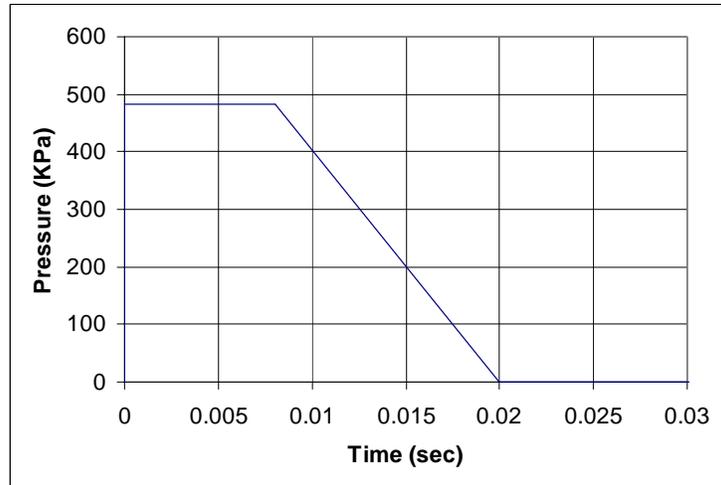


**Figure 57:** The mode shapes of the sensor with fixed bottom boundary condition.

### 5.1.3 Transient Response Analysis Using Direct Method

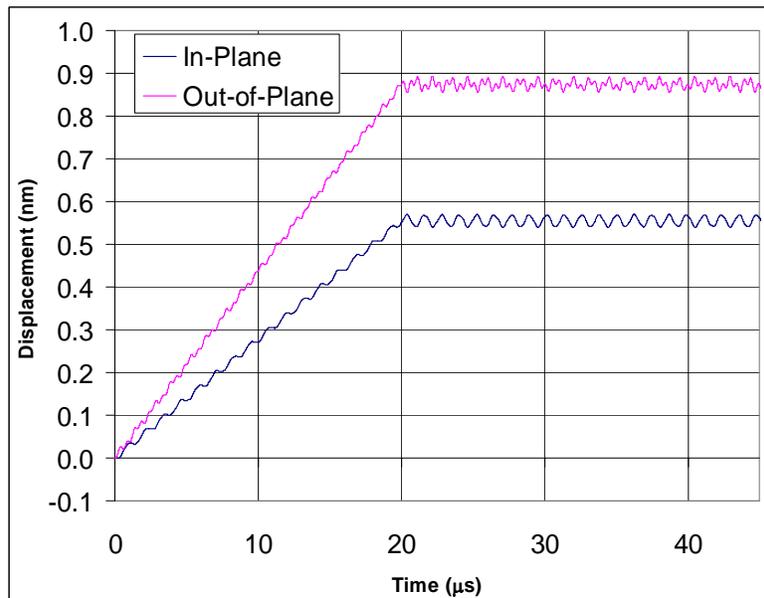
#### 5.1.3.1 Experimental Blast Profile

The transient response analysis was performed using a transient blast input that was obtained from an experiment. The approximate experimental blast profile, given to us by Luna, consists of a pressure rise to 70 psi in 20 microseconds, and the trailing edge is approximated by a line from  $t = 0.008$  sec and  $p = 70$  psi to  $t = 0.02$  sec and  $p = 0$  psi. This blast profile is shown in Figure 58.



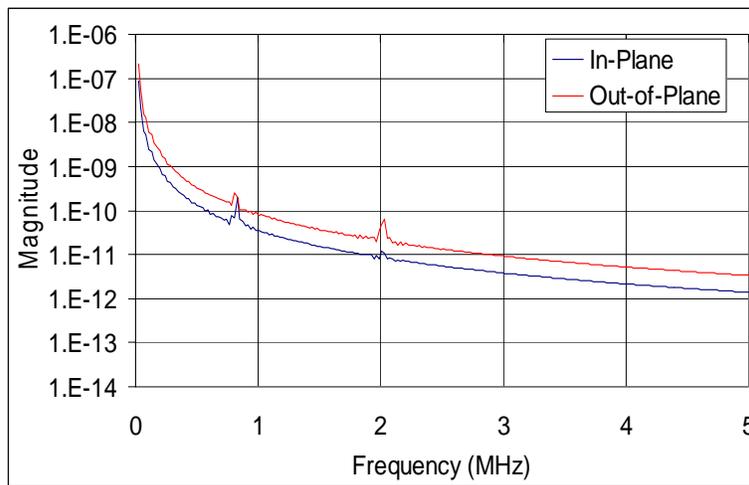
**Figure 58:** Pressure Time-history approximated from experimental data.

The transient response was calculated using NASTRAN's direct time-integration method with a sliding bottom boundary condition. Figure 59 shows the time history of the displacement of the gap. The analysis was done only for a short duration of time that includes the pressure rise because of the limitation in computational hardware.

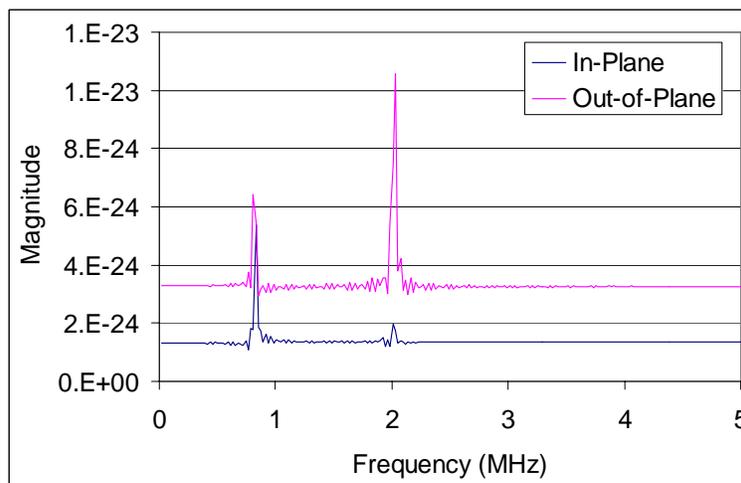


**Figure 59:** Calculated time history of the displacements at the tip of the fiber, obtained using NASTRAN

The Fast Fourier Transform (FFT) was calculated for the time history of the gap displacement. Figure 60 shows the FFT of the gap displacement. The small peaks are at 0.808 MHz and 2.13 MHz which correspond to the 3<sup>rd</sup> and 7<sup>th</sup> modes, respectively. Figure 61, which shows the transfer function FFT of the displacement output divided by the FFT of the pressure input, clearly shows the peaks that correspond to the 3<sup>rd</sup> and 7<sup>th</sup> modes.



**Figure 60:** FFT of the response at the tip of the sensor

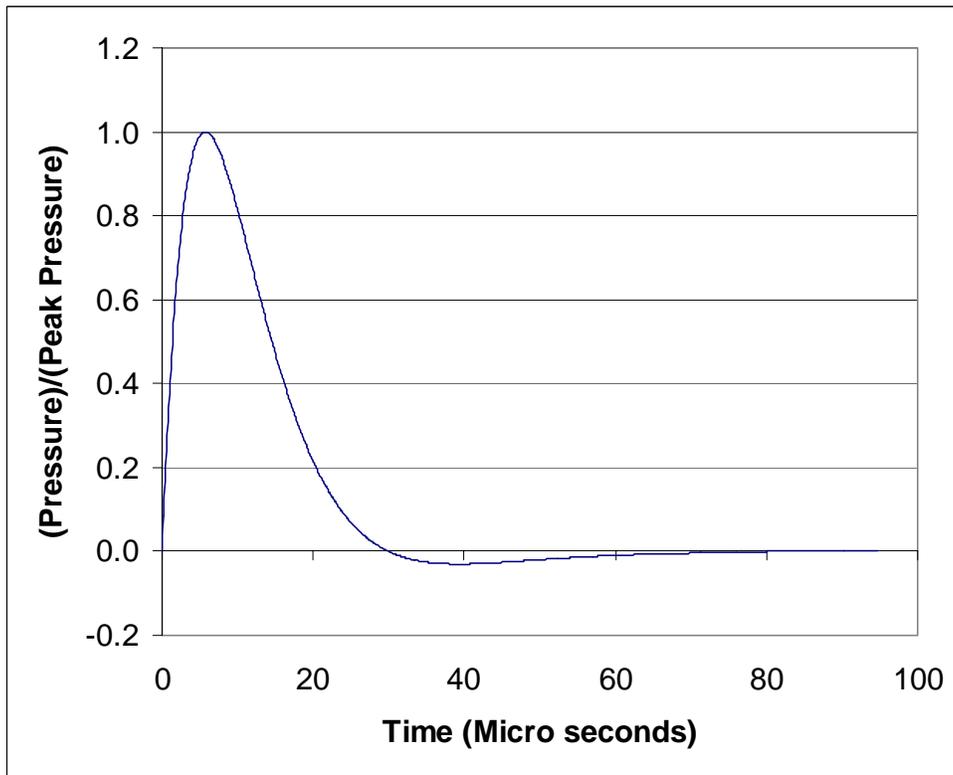


**Figure 61:** FFT of the output divided by FFT of the applied pressure

### 5.1.3.2 Functional Form of a Blast Profile

The blast profiles are the result of pressure waves that are created by the sudden and violent release of energy in an explosive charge, which causes a sharp rise in the pressure of the surrounding medium. The general blast profile shown in Figure 62 can be represented by a functional form. One proposed functional form was developed by Baker [2] and is given by

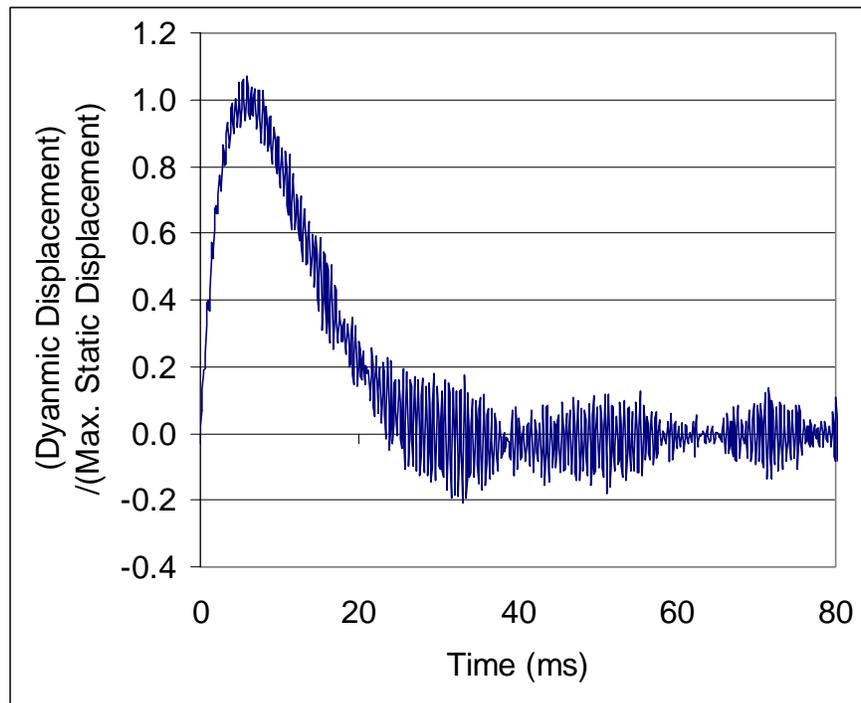
$$p(t) = p_0 - p_s \left( \frac{t}{T^-} \right) \left( 1 - \frac{t}{T^-} \right) e^{-\frac{4t}{T^-}} \quad (23)$$



**Figure 62:** Time history of the Pressure Profile with negative phase.

where  $t$  denotes time,  $p_0$  ambient pressure,  $p_s^-$  peak negative phase, and  $T^-$  the duration of the negative pressure phase. This functional form was used as the blast profile in most of the cases in this research, because this includes both the positive and negative phases of the pressure profile.

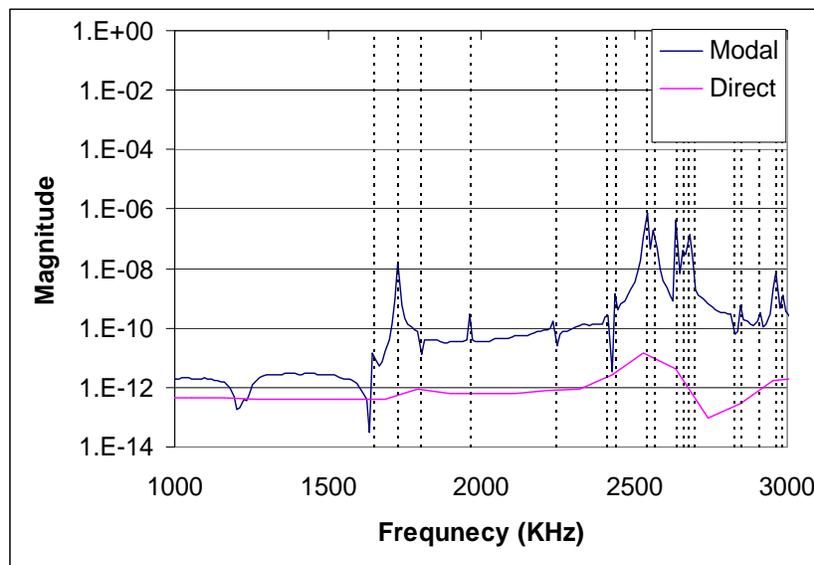
The modal analysis capability in NASTRAN was used to approximate the transient response. For the fixed bottom boundary condition, we retained fifty modes. The time step for integrating the modal equations was selected to be one-tenth of the time for a wave to traverse the thickness of the base. We compared the modal method to the direct method of integration. Figure 63 shows the time history of the gap change obtained with the direct method.



**Figure 63:** Time history of the gap change obtained with NASTRAN's direct method

Next, an FFT analysis was performed on the time history of the gap displacement of the sensor. Figure 64 shows the FFT of the output divided by the FFT of the input. Here, the highest peak for the modal case was found to be at the 2<sup>nd</sup> and the 8<sup>th</sup> modes. In Figure 64, we can find that the results from the direct method correspond to the main signature of the modal method results.

The transient response calculation using the direct method in NASTRAN took a relatively long period of time, and required a large amount of disk space. In the current case, which uses the three-dimensional FEM model and 1000 time steps, the calculation took about one-half hours and required about 1GB of disk space. Therefore, we needed to have a method that could speed up the calculation and require much less disk space to perform a parametric study, because a parametric study requires repeated calculations of the transient response for different design parameters. The solution for this was the modal method and is explained in the next section.



**Figure 64:** FFT of Output divided by FFT of Input using Modal method

## 5.1.4 Transient Analysis Using Modal Methods

### 5.1.4.1 Modal Superposition Method

Any forced motion of a linear Multi-Degree-Of-Freedom (MDOF) system may be expressed in terms of forced modal vibrations. The first modal method to be used is the modal superposition method. The dynamic equation for a case of Multi-Degree-of-Freedom (MDOF) Systems can be expressed as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p}(t), \quad (24)$$

where  $\mathbf{M}$  is the mass matrix,  $\mathbf{K}$  is the stiffness matrix,  $\mathbf{u}$  is the displacement vector, and  $\mathbf{p}$  is the forced input. Here, the effect of damping of the structure was not considered.

The first step in a modal superposition method is to obtain natural frequencies and normal modes of the system. The natural frequencies and modes satisfy

$$(\mathbf{K} - \omega_r^2 \mathbf{M})\phi_r = \mathbf{0} \quad (25)$$

The modes can be collected to form the so-called modal matrix

$$\Phi = [\phi_1 \ \phi_2 \ \dots \ \phi_N] \quad (26)$$

The coordinate transformation that relates the displacement vector and the modal displacement vector is

$$\mathbf{u}(t) = \Phi \boldsymbol{\eta}(t) = \sum_{r=1}^N \phi_r \eta_r(t) \quad (27)$$

The reduced dynamic equation can be expressed in terms of the modal displacement vector  $\boldsymbol{\eta}$  as

$$\bar{\mathbf{M}}\ddot{\boldsymbol{\eta}} + \bar{\mathbf{K}}\boldsymbol{\eta} = \bar{\mathbf{p}}(t), \quad (28)$$

where the modal mass matrix is calculated as  $\bar{\mathbf{M}} = \Phi' \mathbf{M} \Phi$ , the modal stiffness matrix  $\bar{\mathbf{K}} = \Phi' \mathbf{K} \Phi$ , and the modal force vector  $\bar{\mathbf{p}} = \Phi' \mathbf{p}$ . The components of the modal displacement vector  $\boldsymbol{\eta}$  can be obtained by solving the equations independently since the modal mass and stiffness matrices have diagonal form.

#### 5.1.4.2 Modal Acceleration Method

The modal acceleration method can account for the effects of higher modes in the modal analysis with a small number of modes [ 56 ]. Therefore, the modal acceleration method has many advantages over the modal superposition method including the improved convergence properties and more accurate results with a smaller number of modes. The dynamic equation for MDOF system shown in Eq. (26) can be expressed as :

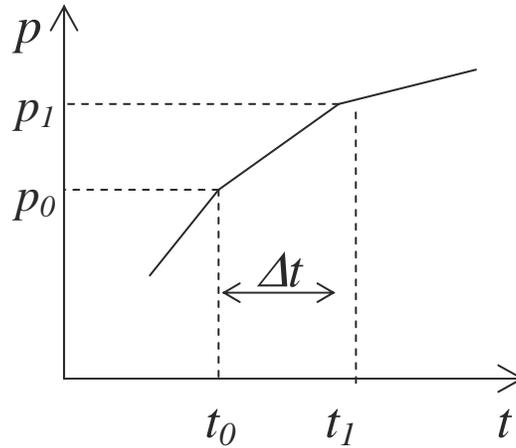
$$\mathbf{u} = \mathbf{K}^{-1}(\mathbf{p} - \mathbf{M}\ddot{\mathbf{u}}) \quad (29)$$

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{p} - \mathbf{K}^{-1} \sum_{r=1}^N \mathbf{M}\Phi_r \ddot{\eta}_r \quad (30)$$

The solution of Eq. (28) can be obtained by solving for the modal acceleration vector  $\ddot{\boldsymbol{\eta}}$  from Eq. (26).

#### 5.1.4.3 Transient Analysis Using Modal Methods

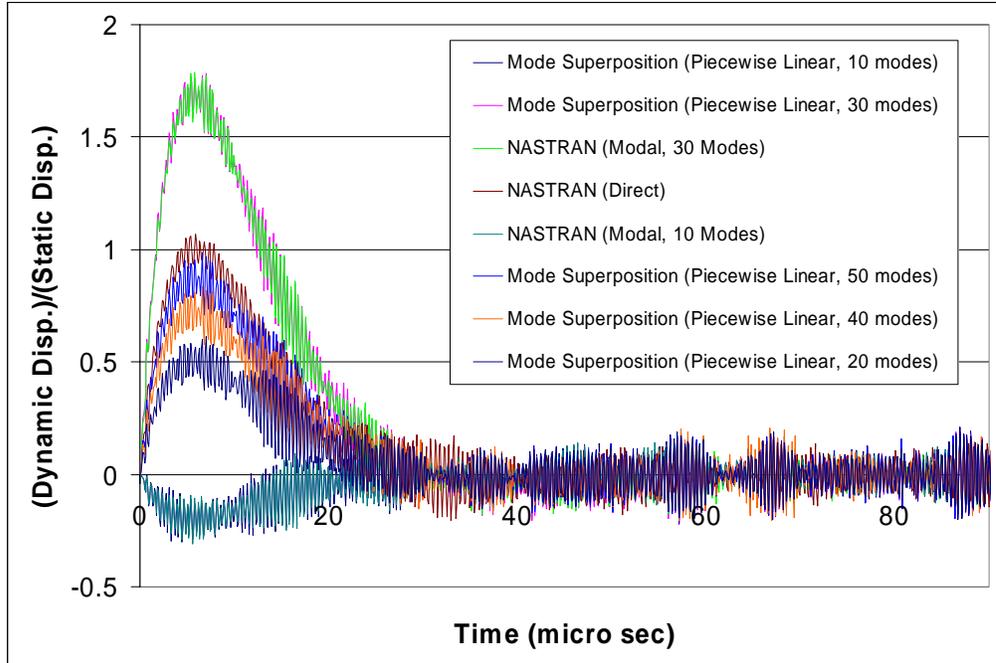
A piecewise linear approximation of applied force, as shown in Figure 65, was utilized in solving the independent equations (26) or (28). The exact solution of the equation below was used in piecewise fashion to obtain  $\boldsymbol{\eta}$  and  $\ddot{\boldsymbol{\eta}}$ .



**Figure 65:** Piecewise linear approximation

$$\begin{aligned}
 \bar{\mathbf{M}}\ddot{\eta} + \bar{\mathbf{K}}\eta &= at + b \\
 \eta(t_0) &= \eta_0 \\
 \dot{\eta}(t_0) &= \dot{\eta}_0
 \end{aligned}
 \tag{31}$$

The use of the modal method to approximate the transient response permitted a much larger number of time steps within the available computational resources. The transient response was calculated using the modal superposition method in MATLAB based on normal modes extracted from NASTRAN. The results of the modal superposition method were close to those of the NASTRAN modal method when the same numbers of modes were used as can be seen in Figure 66. As the number of modes increased, the results from modal superposition were close to those of NASTRAN's direct method.

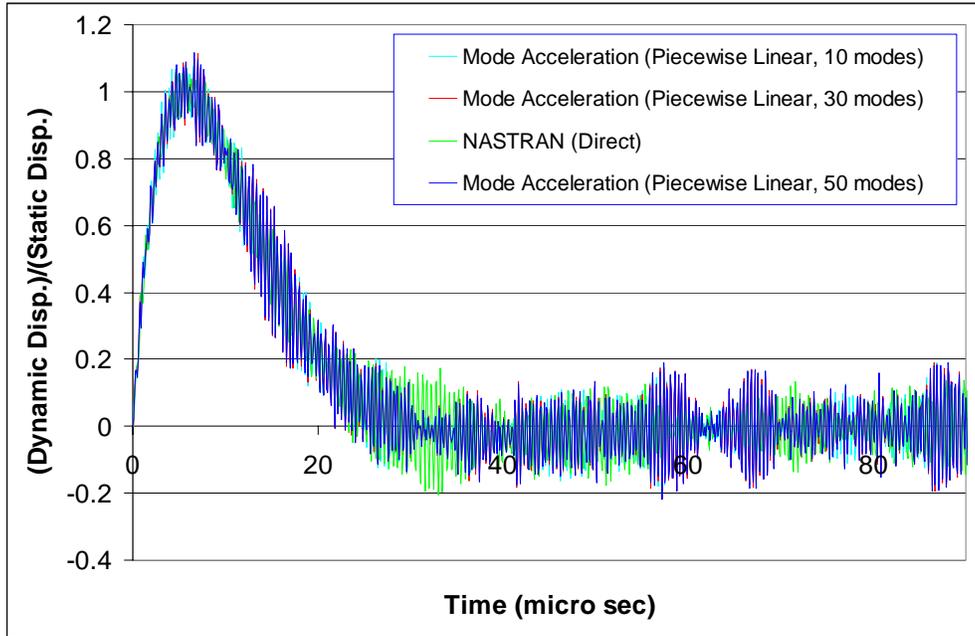


**Figure 66:** Time history of the gap displacement using modal superposition method and NASTRAN modal method.

Next, the transient response was calculated using the modal acceleration method. As can be seen in Figure 67, the results of the modal acceleration method were close to those of the NASTRAN direct method even when a small number of modes were used.

The advantage of the modal acceleration method is that it saves considerable CPU time compared to the CPU time required by the direct method used in NASTRAN.

Table 5 shows the comparison of CPU time once normal mode and static displacement calculations are prepared for the modal methods using a Sun Blade 1000 Workstation. Also, the advantage of the modal acceleration method over the modal superposition method is found to be that the latter requires a lesser number of modes to be extracted, thus saving time.



**Figure 67:** Time history of the change in the length of gap obtained with the modal acceleration method and the NASTRAN direct method.

**Table 5:** The comparison of CPU time for calculating the transient response with different methods

Method	NASTRAN Direct	NASTRAN Modal (50 modes)	Mode Superposition (50 modes)	Mode Acceleration (10 modes)
Time	6 m 20 s	1 m 02 s	39 sec	10 sec
Relative	38	6	4	1

## 5.2 Inverse Problem

### 5.2.1 Parametric Study Using NNs

The objective in this design study is to minimize the size and the weight of the sensor which can sustain a 50,000 psi pressure wave. The design constraints are that the fundamental frequency in the axial direction should be greater than 1MHz and the structural displacement at the gap should be less than 1nm. Also, the design has to prevent silicon failure and prevent fiber rebound failure. The fixed values in this design are the fiber diameter (124mm), gap length (100mm), V-groove etch angle ( $54.7^\circ$ ), and bond length (larger than 2 mm). Figure 68 shows the fixed values in the design.

Therefore the design parameters are to be the size of the silicon base (height, width, and depth (H, W, D), the size of the rectangular shaped groove (width and depth (Wg, Dg)), the depth of the V-groove (Dv), and the exposed length of the fiber (Lf).

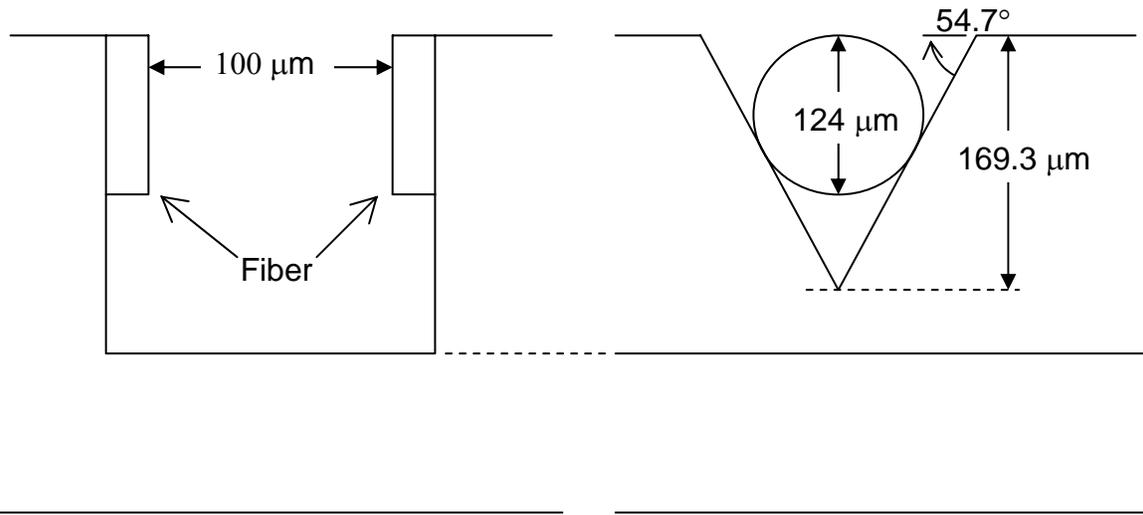
From the design constraints, the related geometrical constraints were found to be

$$H > Dg > Dv > 169.3 \text{ mm}, |Wg - Lf| > 100 \text{ mm}$$

In performing a parametric design study on the sensor, the MATLAB code developed in the damage identification problem that can connect the input and output of NASTRAN to the MATLAB environment was used.

The steps followed in this parametric study are as follows.

The first step in performing the parametric study on the design of the sensor was to obtain the response of the sensor for different dimensions of the sensor. A MATLAB code was used to automatically generate input files for NASTRAN with different dimensions of the sensor.

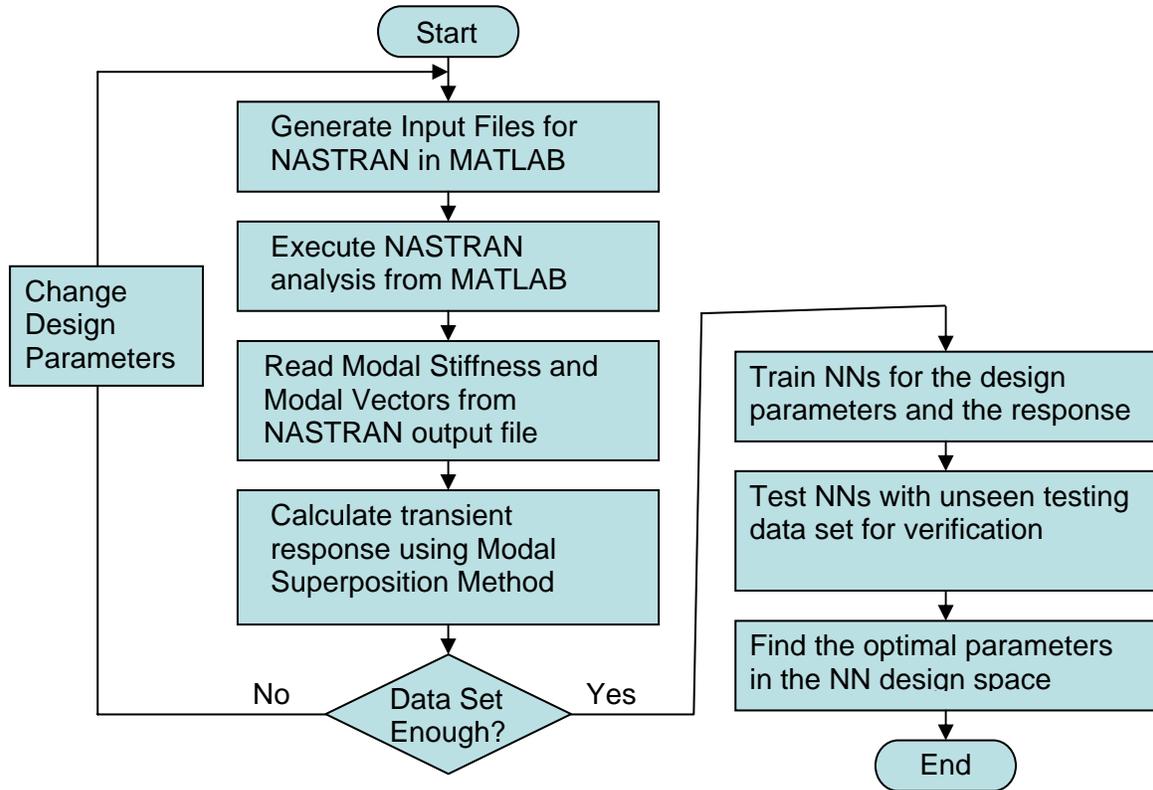


**Figure 68:** Fixed geometric values in the sensor design around the grooved area

Next, a NN was trained for the set of the dimension of the sensor and the calculated response of the sensor such that for a given dimension of the sensor, the NN can output the response of the sensor. The trained NN was tested with an unseen set of data for verification.

In the next step, a gradient based optimization was performed on the input and output mapping of the trained NN. The objective function was minimizing the overall volume of the sensor. The constraints were the minimum of 1MHz natural frequency and the maximum of 1nm displacement at the gap. The calculations of the constraints were quickly performed using the trained NN. The constrained nonlinear multivariable optimization function in the optimization toolbox of MATLAB was used to find the optimal dimensions of the sensor.

The overall step is shown as a flow chart in Figure 69. This whole process involving NASTRAN analysis was performed automatically in the MATLAB environment.



**Figure 69:** The procedure in parametric study on the design of the sensor.

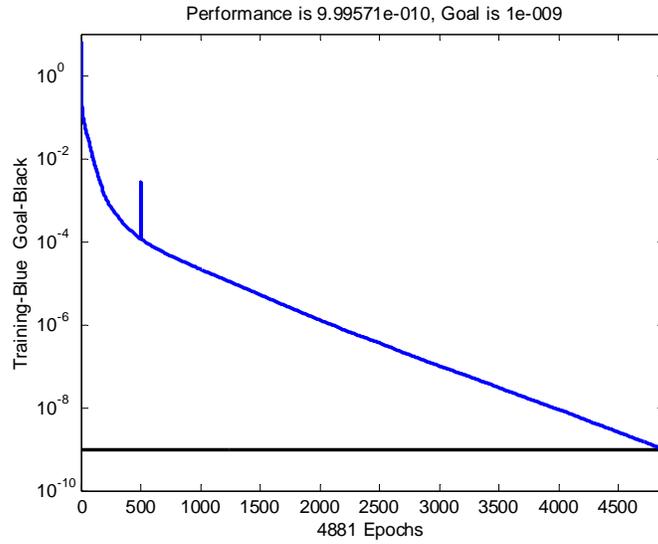
The responses of the sensor calculated for each set of dimensions were the first natural frequency and the peak displacement at the gap when 50,000 psi peak pressure was applied at the top of the sensor in the form shown in Figure 62. Multi-Layer Feed-forward back-propagation NNs were used with a resilient backpropagation learning scheme.

First, the response of the sensor was obtained varying the width between 2.1mm and 2.9mm, length between 2.4mm and 2.6mm, and height between 0.4mm and 1.2mm by

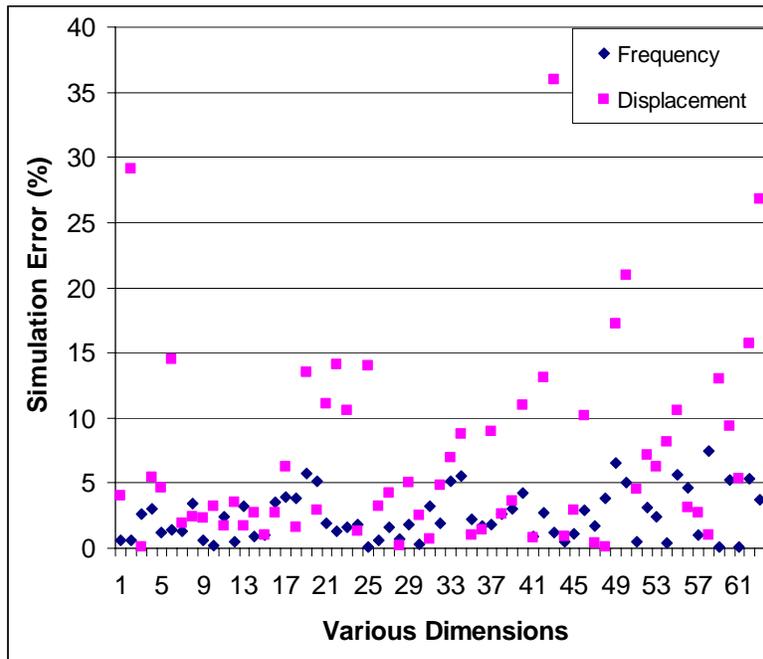
0.1mm step. Figure 70 shows the simulation error in the training process of the NN and Figure 71 shows the simulation error for testing sets of data. The optimization that fulfills exact constraints, which are 1MHz natural frequency and 1nm of displacement, did not converge. The optimization that fulfilled loose constraints, 1MHz natural frequency and 100nm displacement at the gap, resulted in the dimensions 2.82mm (W), 2.41mm (L), and 1.15mm (H).

The response of the sensor was obtained for a broader range of dimensions. The width was varied between 2.5mm and 3.5mm, length between 2.3mm and 3.5mm, and height between 0.8mm and 2.0mm. The optimization, which fulfilled loose constraints 1MHz frequency and a gap displacement of 100nm, resulted in the width 2.69mm, length 2.3mm, and height 1.15mm. A verification run of MATLAB and NASTRAN resulted in 1.18MHz natural frequency and 142nm of displacement at the gap.

The optimization using trained NN, which fulfilled the constraints 1MHz first natural frequency and 1nm displacement gap, resulted in the width 2.5 mm, length 2.3 mm, and height 1.0 mm.



**Figure 70:** Convergence of a simulation error in training of NNs for various dimensions and the response of the sensor



**Figure 71:** Simulation of the trained NN for unseen testing data set

## 5.2.2 Optimization Using Gradient Method and Relaxed Constraints

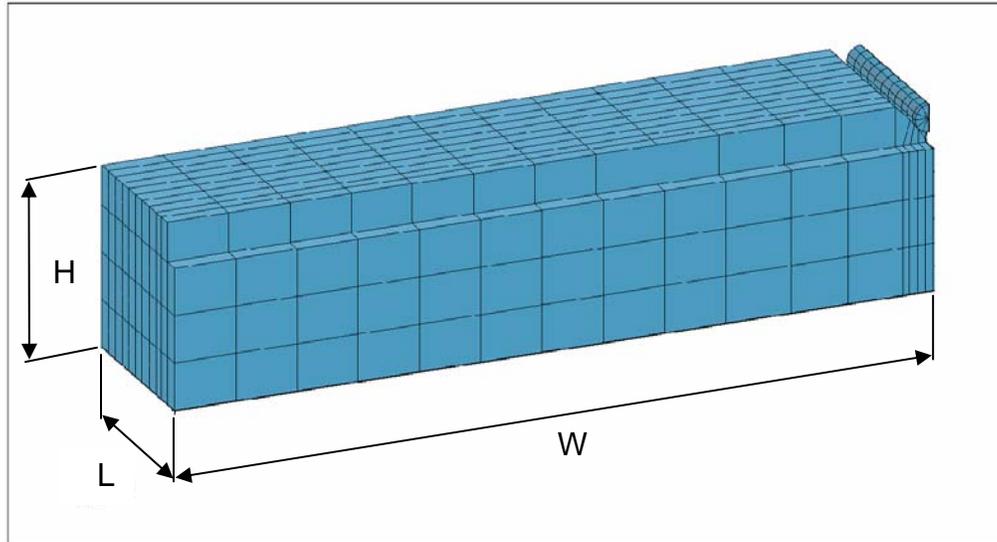
Next, relaxed constraints were used to find optimal dimensions using only a gradient-based optimization method. Four sets of optimal dimensions, which violate one or more constraints to some extent, were found. The result is summarized in Table 6. The details and diagrams of the four sets of optimal dimensions are as follows. Since the optimization was done without the constraints on stress concentration, further study is needed that can satisfy all the constraints.

**Table 6:** Optimal dimensions of a quarter model that fulfills relaxed constraints. (The values related to the constraints are in bold case and the values that violate a constraint are in italics and are underlined. The dimensions of the original design are W=2.5mm, L=2.5mm, and H=0.8mm).

	First Case	Second Case	Third Case	Fourth Case
Width (W, mm)	3.48	3.74	2.30	3.76
<b>Length (L, mm)</b>	<b><u>0.78</u></b>	<b>2.01</b>	<b><u>1.50</u></b>	<b><u>1.89</u></b>
Height (H, mm)	0.83	2.23	2.02	2.05
<b>Gap Displacement (nm)</b>	<b>0.75</b>	<b>0.58</b>	<b>0.67</b>	<b><u>2.00</u></b>
<b>Natural Frequency (MHz)</b>	<b>1.63</b>	<b><u>0.66</u></b>	<b><u>0.84</u></b>	<b><u>0.70</u></b>
Upper bound of Maximum Principal Stress (GPa)	1.01	1.09	1.32	1.06
Lower bound of Maximum Principal Stress (GPa)	-2.53	-2.79	-2.81	-2.77

***First Case:***

Optimal dimensions, which do not comply with the bond length constraint of 2mm, were found. The optimal dimensions found were width = 3.48 mm, length = 0.78 mm, and height = 0.83 mm as shown in Figure 72. (original dimensions 2.5mm x 2.5mm x 0.8mm)

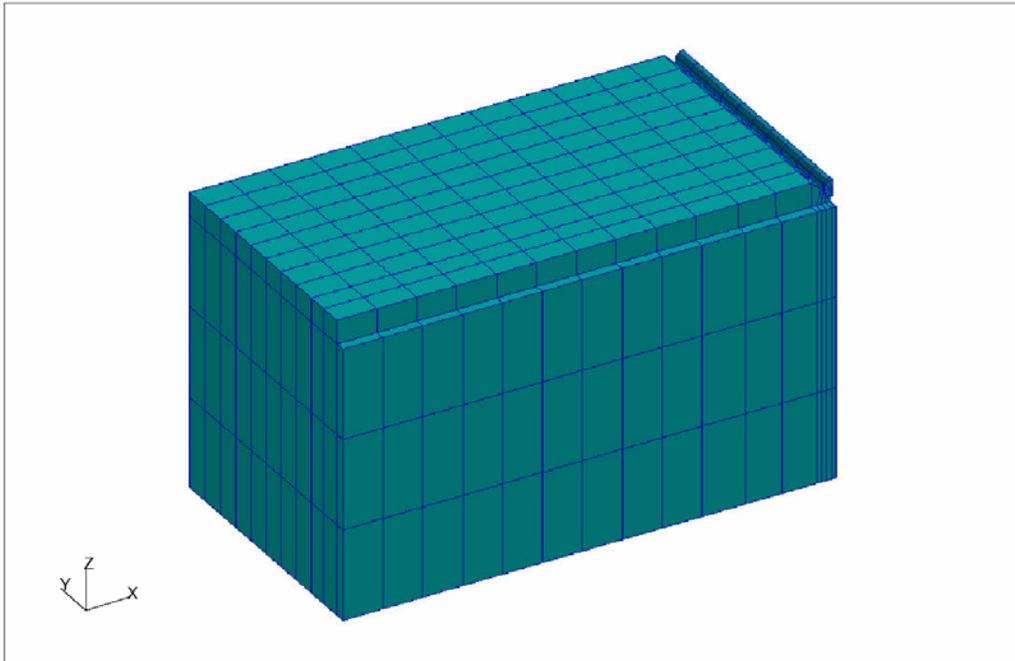


**Figure 72:** Finite element model of the sensor in the first case. (Width,  $W=3.48\text{mm}$ ; Length,  $L=0.78\text{mm}$ ; Height,  $H=0.83\text{mm}$ ).

The maximum static displacement at 50,000psi is 0.75 nm and the natural frequency is 1.63 MHz for the optimal dimensions found. From Figure 2, we found that the silicone base would not fail less than 50,000 psi of pressure but the stress in the fiber and the bond need to be examined further.

### *Second Case*

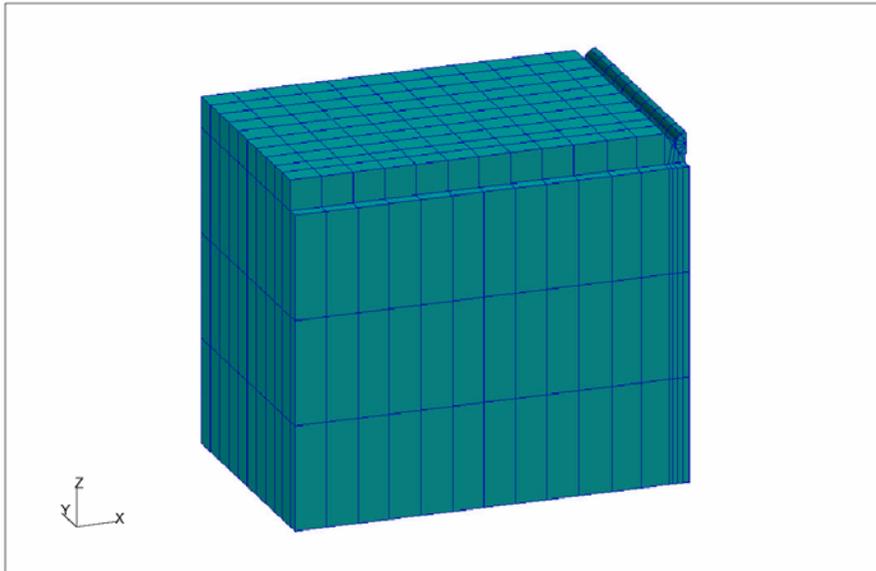
In the second case, the natural frequency was allowed to be below 1MHz but more than 0.5 MHz. The optimal dimensions found in this case were width=3.74 mm, length=2.01mm, height=2.23mm as shown in Figure 73. The gap displacement was 0.58 nm and the natural frequency was 0.66 MHz.



**Figure 73:** The finite element model of the sensor in the second case (W=3.74mm, L=2.01mm, H=2.23mm).

### *Third Case*

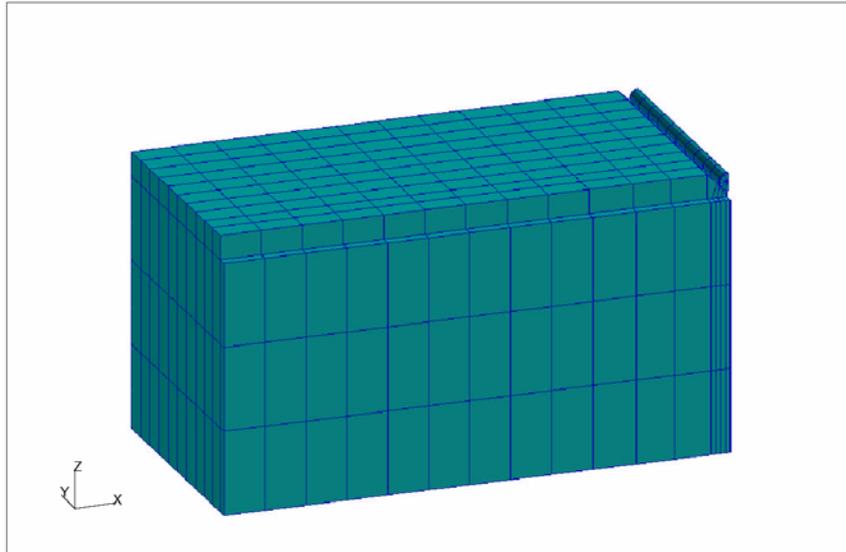
In the third case, the natural frequency was allowed to be more than 0.8 MHz and the bond length was allowed to be more than 1.5mm. The optimal dimensions found were width = 2.30 mm, length = 1.50 mm, and height = 2.02 mm as shown in Figure 74. In this case, the gap displacement was 0.67 nm and the natural frequency was 0.840547.



**Figure 74:** Finite element model of the sensor in the fourth case ( $W = 2.3$  mm,  $L = 1.5$  mm,  $H = 2.02$  mm).

#### **Fourth Case**

In the fourth case, the gap displacement was allowed to be more than 1 nm but less than 2 nm, the natural frequency was allowed to be less than 1MHz but more than 0.7 MHz, and the bond length was allowed to be less than 2 mm but more than 1.8 mm. The optimal dimensions found in this case were width = 3.76 mm, length = 1.89 mm, and height = 2.05 mm as shown in Figure 75. The gap displacement was 2 nm and the natural frequency was 0.7 MHz.



**Figure 75:** The finite element model in optimal dimensions in the fourth case ( $W = 3.76$  mm,  $L = 1.89$  mm,  $H = 2.05$  mm)

# Chapter 6

## SUMMARY AND CONCLUSIONS

### 6.1 Summary of Obstacles and Solutions

There were many obstacles encountered in application of NN to the three inverse problems studied in this dissertation. For those obstacles, many solutions were also found.

To summarize the obstacles encountered and the solutions arrived:

The problems which required time marching and needed long computational time caused repeated solving of the direct problem implausible. In the damage identification problem and in the structural optimization problem, modal methods were used to reduce the calculation time for the dynamic transient response. In the optical imaging problem, the use of mean time reduced the time in calculation of time of arrival.

When large dimensions of correlated data were present in input, the training of NN can take a lot of time. This could be solved by use of Principal Component Analysis (PCA). The use of PCA was essential in optical imaging problem which used the photon property values at different detector locations for each source locations. The experience on the use of PCA in optical imaging problem enabled the use of complex data in damage identification problem like the use of transient response of structures.

When a vast number of direct problem was required to be solved to generate large enough training sets in damage identification problem, orthogonal arrays can be used to help selecting the sets to be used in training of the NN.

When different computational environments which required manual labor and delayed computational process, MATLAB script was used to integrate a commercial FEM package to NN solution process in MATLAB in damage identification problem and this was further extended in structural optimization problem. In optical imaging problem, a FEM code in MATLAB was developed to integrate the whole solution process including use of NN.

## **6.2 Conclusions**

The first step in solving inverse problems of damage detection and optical imaging is finding the best information that will most efficiently identify the parameters that we are seeking. In the damage identification problem, natural frequencies of a beam and transient response of a plate provided sufficient information. In the optical imaging problem, the intensity of light and the photon mean time of flight observed at various locations were utilized to both locate and determine the size of an anomaly. In the optimization problem, the first natural frequency and the maximum deflection in transient analysis of a fiber optic sensor were the constraints in solving the problem.

The second step lies in finding the best structure and training method to employ NN to solve the specific inverse problem. In all three problems, Multilayer Feed-Forward Backpropagation NN which has 2 hidden layers that uses sigmoid transfer functions for hidden layers and a linear transfer function for output layer and employs a resilient backpropagation training scheme was found to be most efficient, it was found that using

radial basis NNs in solving some aspects of the problem can have benefits. Further, the damage identification problem in two-dimensions would need similar steps as in the optical imaging problem. In both the problems, for NN to predict accurate values, the generation of a large enough number of training sets and long learning processes are needed.

In all three problems, the finite element method was used as a highly flexible and efficient tool to generate enough training sets and testing sets for NN. Hence, FEA provides the essential calculation speed and desired accuracy.

In applying NN for solving inverse problems related to optical imaging, one of the important features that the NN can supply seems to be their capability of classifying the existence of an anomaly in a tissue and of providing information related to both the geometry and the optical properties of a constituent in a tissue. This would also be the case for the damage identification problem wherein a well-trained neural network can provide both the location and the extent of damage.

Principal component analysis (PCA) and orthogonal arrays (OA) were used to reduce the time and effort required in solving inverse problems. In the damage identification problem in a plate, the use of PCA reduced the training time significantly and resulted in a NN which gave outputs with a consistent level of error but not lower than certain value. The reason seems to be that certain characteristics of the data are lost using PCA, another characteristic, however, strengthens the consistency of the data when using PCA. The use of PCA in the optical imaging problem gave output without a lower limit for the simulation error. This seems to be due to the fact that the characteristics of the input data

are different in the two problems. Also, the use of OA showed some prospect for reducing the time and effort required in applying NN to solve the damage identification problem.

The code developed in the MATLAB environment that can automatically connect NASTRAN analysis to MATLAB analysis made it possible to generate enough training data for NN in the damage identification problem. Moreover, this code has made it possible to use a highly complex NASTRAN structural models in an analysis or a design process that involves NN and other analyses like the modal superposition method, and Fast Fourier Transform analysis in the MATLAB environment.

A miniature fiber optic sensor was analyzed for its reliability against high peak blast pressure. Moreover, a parametric study was performed to obtain optimal dimensions for the sensor. The normal mode solution made sure the design of the sensor was within the limit of the natural frequency. However, the parametric study utilizing NASTRAN normal mode analysis for the displacement constraints and neural networks found an optimal solution without the need for solving every possible direct problem existing in the design space.

This work provides an example of applying a technology such as identifying damage using neural networks and structural optimization developed in the field of structural engineering to a problem in a totally different field, which is biomedical engineering, and applying the experience gained in the second field back to the structural engineering field. Eventually, we can have benefits in solving inverse problems in both the fields.

## BIBLIOGRAPHY

- [ 1 ] Haykin, S. S., *Neural Networks: Fundamental Foundation, 2<sup>nd</sup> Ed*, Prentice Hall, Upper Saddle River, 1998.
- [ 2 ] Hagan, M. T., Demuth, H. B., and Beale, M., *Neural Network Design*, PWS Publishing Company, Boston, MA, 1996.
- [ 3 ] Müller, G. J., *Medical Optical Tomography: Functional Imaging and Monitoring*, SPIE Optical Engineering Press, Washington, D.C., 1993, pp. 5-12.
- [ 4 ] Arridge, S. R., "A gradient-based optimization scheme for optical tomography", *Optics Express*, Vol. 2, No. 6, 1997, pp. 213-226.
- [ 5 ] Arridge, S.R. and Schweiger, M., "Image reconstruction in optical tomography", *Philosophical Transactions: Biological Sciences*, Vol. 352, No. 1354, 1997, pp. 716-726.
- [ 6 ] Kudva, J. N., Munir, N., and Tan, P. W., "Damage detection in smart structures using neural networks and finite-element analyses," *Smart Materials and Structures*, Vol. 1, No. 2, 1992, pp. 108-112.
- [ 7 ] Wu, X. J., Ghaboussi, J., and Garrett, J. H., "Use of neural networks in detection of structural damage," *Computers and Structures*, Vol. 42, No. 4, 1992, pp. 649-659.

- [ 8 ] Leath, W. J. and Zimmerman, D. C., “Analysis of neural network supervised training with application to structural damage detection,” *Proceedings of the 9th Virginia Polytechnic Institute and State University Symposium on Dynamics and Control of Large Structures*, VPI&SU, Blacksburg, VA, 1993, pp. 583-594.
- [ 9 ] Spillman, W., Huston, D., Fuhr, P., and Lord, J., “Neural network damage detection in a bridge element,” *Proceedings of SPIE Vol. 1918, Smart Structures and Materials 1993: Smart Sensing, Processing, and Instrumentation*, Richard O. Claus; Ed, SPIE - The International Society for Optical Engineering, Bellingham, WA, 1993, pp 288-295.
- [ 10 ] Worden, K., and Tomlinson, G. R., “Damage location and quantification using neural networks,” from *Engineering Integrity Assessment*, Edwards, J. M., Kerr, J., Stanley, P., Eds., Chameleon Press Ltd, London, 1993, pp. 11-33.
- [ 11 ] Kirkegaard, P. and Rytter, A., “Use of neural networks for damage assessment in a steel mast,” *Proceedings of the 12th International Modal Analysis Conference*, SPIE - The International Society for Optical Engineering, Bellingham, WA, 1994, pp. 1128–1134.
- [ 12 ] Manning, R., “Damage detection in adaptive structures using neural networks,” *Proceedings of 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1994, pp. 160–172.

- [ 13 ] Povich, C., and Lim, T., “An Artificial Neural Network Approach to Structural Damage Detection Using Frequency Response Functions,” *Proceedings of 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1994, pp. 151–159.
- [ 14 ] Tsou, P. and Shen, M. H. H., “Structural damage detection and identification using neural networks,” *AIAA Journal*, Vol. 32, No. 1, 1994, pp. 176–183.
- [ 15 ] Rhim, J. and Lee, S., “A neural network approach for damage detection and identification of structures,” *Proceedings of 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1994, pp. 173–180.
- [ 16 ] Barai, S.V. and Pandey, P.C., “Vibration signature analysis using artificial neural networks,” *Journal of Computing in Civil Engineering*, Vol. 9, No. 4, 1995, pp. 259–265.
- [ 17 ] Ceravolo, R. and De Stefano, A., “Damage location in structures through a connectivistic use of FEM modal analyses,” *Modal Analysis: The International Journal of Analytical and Experimental Modal Analysis*, Vol. 10, No. 3, 1995, pp. 176.
- [ 18 ] Schwarz, B.J., McHargue, P. L. and Richardson, M. H., “Using SDM to train neural networks for solving modal sensitivity problems,” *Proceedings of the 14th International Modal Analysis Conference*, Society of Experimental Mechanics, Bethel, CT, 1996, pp. 1285–1291.

- [ 19 ] Ganguli, R., and Chopra, I., "Helicopter rotor system fault detection using physics-based model and neural networks," *AIAA Journal*, VOL. 36, No. 6, 1998, pp. 1078-1086.
- [ 20 ] Marwala, T., "Damage identification using committee of neural networks," *Journal of Engineering Mechanics*, Vol. 126, No. 1, Jan 2000, pp. 43-50.
- [ 21 ] Marwala, T., "Probabilistic fault identification using a committee of neural networks and vibration data," *Journal of Aircraft*, Vol. 38, No. 1, 2001, pp. 138-146
- [ 22 ] Palakal, M. J., Pidaparti, R. M. V., and Rebbapragada, S., "Intelligent computational methods for corrosion damage assessment," *AIAA Journal*, Vol. 39, No. 10, 2001, pp. 1936-1943.
- [ 23 ] Szewczyk, P.Z. and Hajela, P., "Damage detection in structures based on feature-sensitive neural networks," *ASCE Journal of Computing in Civil Engineering*, Vol. 8, No. 2, 1994, pp. 163-178.
- [ 24 ] Klenke, S.E. and Paez, T. L., "Damage identification with probabilistic neural networks," *Proceedings of the 14th International Modal Analysis Conference*, Society of Experimental Mechanics, Bethel, CT, 1996, pp. 99-104.
- [ 25 ] Luo, H. and Hanagud, S., "Dynamic learning rate neural network training and composite structural damage detection," *AIAA Journal*, Vol. 35, No. 9, 1997, pp 1522-1527.

- [ 26 ] Doyle, C and Fernando, G., “Detecting impact damage in a composite material with an optical fiber vibration sensor system,” *Smart Materials and Structures*, Vol. 7, No. 4, 1998, pp 543-549.
- [ 27 ] Li, X. Q., Wong, Y. S. and Nee, A. Y. C., “Intelligent tool wear identification based on optical scattering image and hybrid artificial intelligence techniques,” *Proceedings of the Institution of Mechanical Engineers. Part B, Journal of Engineering Manufacture*, Vol. 213, No. B2, 1999, pp 191-196.
- [ 28 ] Wu, Y. J., Shi, X. Z., and Zhuang, T. G., “Fusion of wavelet packets and neural network in detection of composites,” *AIAA Journal*, Vol. 38, No. 6, 2000, pp. 1063-1069.
- [ 29 ] Cerri, M.N. and Vestroni, F., “Detection of damage in beams subjected to diffused cracking,” *Journal of Sound and Vibration*, Vol. 234, No. 2, 2000, pp. 259 – 276.
- [ 30 ] Barbour, R. L., Graber, H. L., Chang, J., Barbour, S. L., Koo, P. C., and Arronson, R., “MRI-guided optical tomography: prospects and computation for a new imaging method,” *IEEE Computational Science and Engineering*, Vol. 2, No. 4, 1995, pp. 63-77
- [ 31 ] Chang, J., Graber, H. L., Barbour, R. L., and Aronson, R., “Recovery of optical cross-section perturbations in dense-scattering media by transport-theory-based imaging operators and steady state simulated data,” *Applied Optics*, Vol. 35, No. 20, 1996, pp. 3963-3978.

- [ 32 ] Mukherjee, A. and Deshpande, J. M., “Application of artificial neural networks in structural design expert systems,” *Computers and Structures*, Vol. 54, No. 3, 1995, pp. 367-375.
- [ 33 ] Jayatheertha, C., Webber, J. P. H., and Morton, S. K., “Application of artificial neural networks for the optimum design of a laminated plate,” *Computers and Structures*, Vol. 59, No. 5, 1996, pp. 831-845.
- [ 34 ] Ramasamy, J. V. and Rajasekaran, S., “Artificial neural network and genetic algorithm for the design optimization of industrial roofs,” *Computers and Structures*, Vol. 58, No. 4, 1996, pp. 747-755.
- [ 35 ] Rai, M. M. and Madavan N. K., “Aerodynamic design using neural networks,” *AIAA Journal*, Vol. 38, No.1, 2000, pp. 173-182.
- [ 36 ] Nikolaidis, E., Long, L., and Ling, Q., “Neural networks and response surface polynomials for design of vehicle joints,” *Computers and Structures*, Vol. 75, 2000, pp. 593-607.
- [ 37 ] Kaveh, A. and Servati, H., “Design of double layer grids using backpropagation neural networks,” *Computers and Structures*, Vol. 79, 2001, pp. 1561-1568.
- [ 38 ] Greenman, R. M., Stepniewski, S. W., Jorgensen, C. C., and Roth, K. R., “Designing compact feedforward neural models with small training data sets,” *Journal of Aircraft*, Vol. 39, No. 3, 2002, pp. 452-459.

- [ 39 ] Hadi, M. N. S., "Neural networks applications in concrete structures," *Computers and Structures*, Vol. 81, 2003, pp. 373-381.
- [ 40 ] Arridge, S. R., Schweiger, M., Hiraoka, M., and Delpy, D. T., "A finite element approach for modeling photon transport in tissue," *Medical Physics*, Vol. 20, No. 2, 1993, pp. 299-309.
- [ 41 ] Arridge, S. R. and Schweiger, M., "Direct calculation of the moments of the distribution of photon time of flight in tissue with a finite-element method," *Applied Optics*, Vol. 34, No. 15, 1995, pp. 2683-2687.
- [ 42 ] Schweiger, M., and Arridge, S. R., "The finite-element method for the propagation of light in scattering media: Frequency domain case," *Medical Physics*, Vol. 24, No. 6, 1997, pp. 895-902.
- [ 43 ] Wang, L., and Jacques, S. L., *Monte Carlo Modeling of Light Transport in Multilayered Tissues in Standard C*, University of Texas M. D. Anderson Cancer Center, Houston, TX, 1992.
- [ 44 ] Okada, E., Schweiger, M., Arridge, S. R., Firbank, M., and Delpy, D. T., "Experimental validation of Monte Carlo and finite-element methods for the estimation of the optical path length in inhomogeneous tissue," *Applied Optics*, Vol. 35, No. 19, 1996, pp. 3362-3371.
- [ 45 ] Chiu, Y. H. and Yau, S. F., "An artificial neural network for tomographic reconstruction of time varying object," *Proceedings of the International*

- Symposium on Speech, Image Processing and Neural Networks*, Institute of Electrical and Electronics Engineers, New York, 1994, pp. 409-412.
- [ 46 ] Nakao, Z., Ali, F. E. A. F. ,and Chen, Y. W., “CT image reconstruction by back-propagation,” *Proceedings of the First International Conference on Knowledge-Based Intelligent Electronic Systems*, Institute of Electrical and Electronics Engineers, New York, 1997, pp. 323-326.
- [ 47 ] Kiliç, B., and Korürek, M., “A finite element method based neural network technique for image reconstruction in electrical impedance imaging,” *Proceedings of the 2<sup>nd</sup> International Biomedical Engineering Days*, Institute of Electrical and Electronics Engineers, New York, 1998, pp. 100-102.
- [ 48 ] Aizenberg, I., Aizenberg, N., Hiltner, J., Moraga, C., and Meyer zu Bexten, E., “Cellular neural networks and computational intelligence in medical image processing,” *Image and Vision Computing*, Vol. 19, No. 4, 2001, pp. 177-183.
- [ 49 ] Diamantaras, K. I. and Kung, S. Y., *Principal Component Neural Networks*, 1996, Wiley Inter-science, New York.
- [ 50 ] Zang, C. and Imergun, M., “Structural damage detection using artificial neural networks and measured FRF data reduced via principal component projection,” *Journal of Sound and Vibration*, Vol. 242, No. 5, 2001, pp. 813-827.
- [ 51 ] Peace, G. S. *Taguchi Methods, A Hands On Approach*, 1993, Addison-Wesley, Reading, MS.

- [ 52 ] Hedayat, A. S., Sloane, N. J. A., and Stufken, J., *Orthogonal Arrays*, 1999,  
Springer, New York.
- [ 53 ] Chang, C. C., T. Y. P. Chang, and Y. G. Xu, "Selection of training samples for  
model updating using neural networks," *Journal of Sound and Vibration*, Vol. 249,  
No. 5, 2002, pp. 867-883.
- [ 54 ] Demuth, H. and Beale, M., *Neural Network Toolbox for Use with MATLAB*, The  
Math Works Inc., 2000, pp. 5.16-5.17.
- [ 55 ] Reddy, J. N., *An Introduction to the Finite Element Method*, 2nd Edition,  
McGraw-Hill, New York, 1993, pp. 304-312.
- [ 56 ] Craig, R. R., *Structural Dynamics: An Introduction to Computer Methods*, Wiley,  
New York, 1981.
- [ 57 ] Dharmalingam, A., Kapania, R. K., and Johnson, E. R., *VPI-AOE: Three–  
dimensional Finite Element Analyses of a Miniature Fiber-Optics Sensor*,  
Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute  
and State University, Blacksburg, VA, July 2002.
- [ 58 ] Baker, W. E., *Explosions in Air*, University of Texas Press, Austin, 1973.

## APPENDIX

### A. Three-Dimensional Solid Finite Element Analysis of a Sting Balance

#### A.1 Introduction

The process involved in constructing a solid finite element model for a complex geometry requires careful attention in many steps. Here, the steps and the processes involved in the construction of a complex shaped three-dimensional solid finite element model and FE analysis has been demonstrated using a complex shaped 6-component internal sting balance.

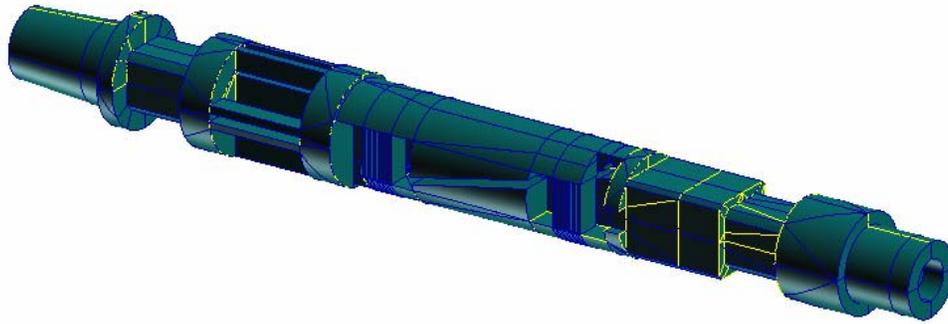
Internal sting balances are used for the measurement of 6 components of aerodynamic loads associated with wind tunnel testing. They usually fit into a model and are arranged to send data out through electrical wiring. A sting balance is a carefully designed and precisely machined cantilever beam that has complicated cross-sections and is generally equipped with electrical resistance strain-gage sensors. The balance used in this research project is a six-component moment balance designed and developed by Arnold Engineering Development Center and is designated as Balance ID 8106 (4.00-Y-36-081) [ A.1 ]. Figure A. 1 shows a rendered CAD drawing of the 6-component sting balance, and Figure A. 2 shows a wireframe of the developed geometric model. One can see that the shape of the model is complicated both inside and on the outside.

The main goal of the research project was comparing the performance of the existing strain gages on the balance and new fiber optic strain sensors. Therefore, the fiber optic sensors had to be placed on the pre-existing sting balance avoiding the location of already

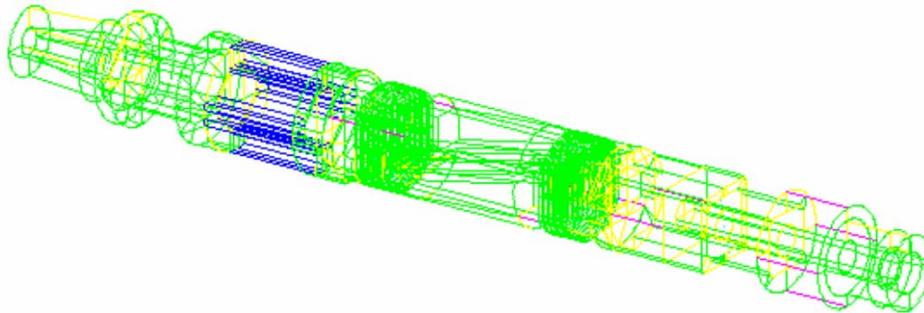
installed strain gages and at the same time giving nearly the same order of magnitude of strain as the already existing strain gages. A finite element analysis of the balance was required to help determine suitable locations for the placement of the fiber optic strain sensors. The FEM results were confirmed by comparing the predictions to measured values of the strain gage located with static loads applied.

## **A.2 Finite Element Model of the Sting Balance**

To fully represent the complicated geometry of the balance, tetrahedral solid finite elements were chosen. The main advantage of using tetrahedral solid finite element is that meshing can be automatically done in MSC/PATRAN, a pre and post-processor for NASTRAN, as long as the surface geometric model, which fully covers the geometry of the complex solid model, can be developed. Therefore, a surface geometric model, shown in Figure A. 1 and Figure A. 2, which represents the detailed geometry of the balance was developed via MSC/PATRAN, and the solid that is represented inside of this surface model is set. This solid is called a body representative solid (B-Rep Solid) in PATRAN [ A.2 ]. Then, the inside of the B-Rep. solid model is filled with a tetrahedral finite element mesh, and the solid finite element model that fully represents the geometry of the balance is constructed. Figure A. 3 shows the mesh drawing of the developed tetrahedral finite element model of the balance. The constructed geometric model, which was used to generate tetrahedral elements, can be meshed again to generate a finer meshing if the user wishes to reach a more accurate result.



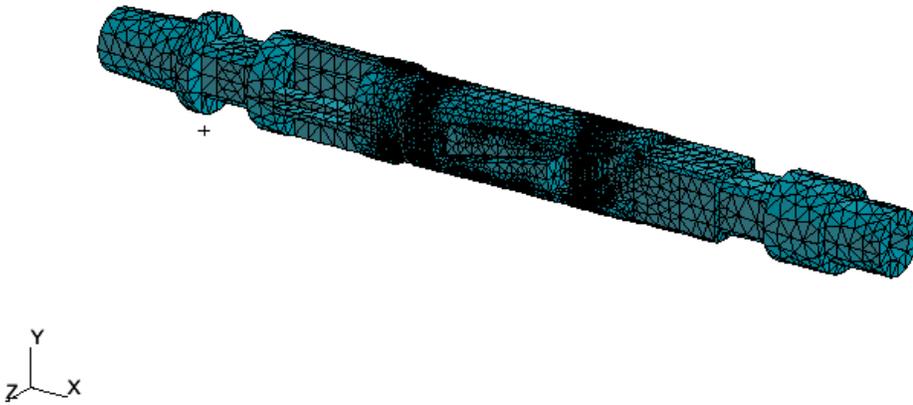
**Figure A. 1:** Rendered drawing of the geometric model.



**Figure A. 2:** Wireframe drawing of the geometric model.

The other solid finite element meshing method available in MSC/PATRAN is isometric meshing. This method meshes the solid model with isometric hexahedral elements resulting in more uniform elements and a finite element model with a lesser number of elements and which reaches the desired result more easily when the geometry is a simple symmetric solid model. Although the tetrahedral mesh needed a relatively high number of

elements for correct representation of the solid, the results obtained with such a mesh can also be considered to be quite accurate with less effort in the construction of the model. Moreover, a finite element mesh that uses hexahedral elements needs all the solids to be constructed as 5 or 6-sided solids which share common faces to the adjacent solid. Constructing this kind of geometric model for the balance, which has various complex cross-sections, would have taken far longer. Although an isometric solid mesh can be more accurate with a lesser number of elements, construction of this kind of a solid model would have taken a large amount of manual labor without any guarantee of success in the short term.



**Figure A. 3:** A solid finite element model with tetrahedral elements for the sting balance.

The details of the construction of the geometric model using tetrahedral meshing are as follows. The solid model constructed is composed of 3 lumps of B-rep solids and a number of simple solids. The whole geometric model could not be developed as a one-lump body because a body representative (B-rep) solid in PATRAN cannot contain a hole.

Also, to enable PATRAN to recognize these 3 lumps of B-rep solids as one continuous body, each of the B-rep solids was made to have common surfaces at the contacting faces. Much time would have had to be spent in matching the faces of solids if various solid parts of the body did not combine to construct a bigger lump of the B-rep solid. Also, when a tetrahedral mesh is generated, curvature error was set to be less than 25%. Without curvature error definition, PATRAN resulted in a finite element model that somewhat misrepresented the load. A finite element model without the setting of a curvature error limit resulted in unsymmetrical displacement under a symmetric load. For example, when the balance was symmetric in one plane, there was a relative error of 6% in the FEM model without the curvature error limit setting, compared to a model constructed with the curvature error limit set.

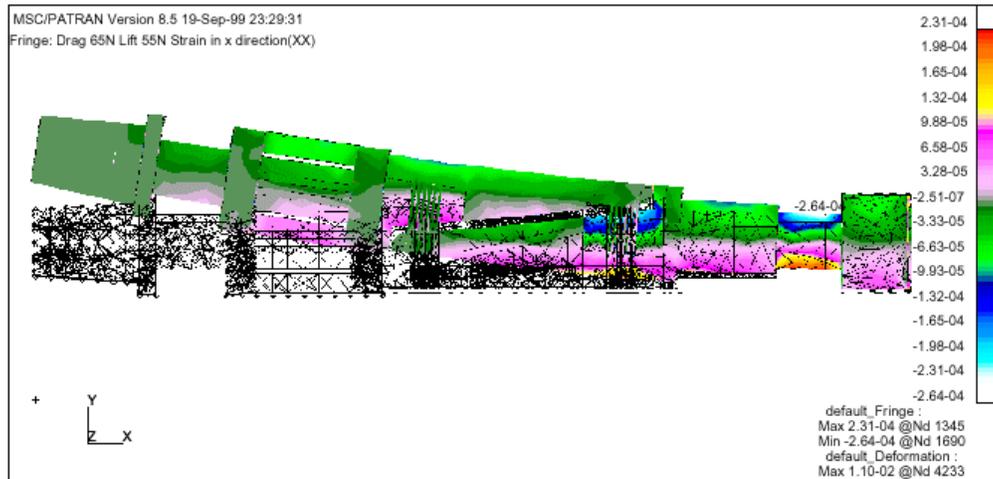
Next, the specific material properties of the balance were put into the finite element model. The material is stainless steel 17/4. The developed finite element model can give detailed distribution diagrams of the stress tensor, the strain tensor, and the displacements in any direction. Numerical results can be obtained at points and areas where nodal points and elements are present. Also, the mesh of the finite element can be refined to give better results.

To calculate strains and stresses, simulated lift, drag, and side forces were applied to the balance at the end of the center hole at the tip. Rolling, pitching, and yaw moments were

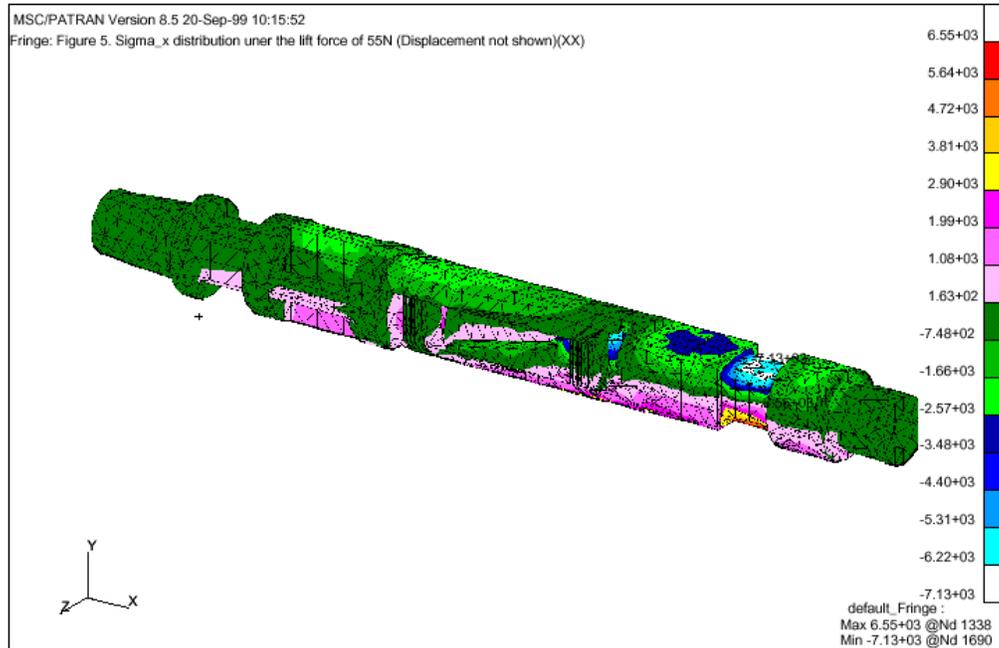
applied as two forces applied tangentially around the tip, because the tetrahedral element is not efficient in getting results when a point moment is applied at one point.

### A.3 Results

To demonstrate the capability of the model, stress contour diagrams of the balance were obtained for various loading cases. Figure A. 4 shows the stress distribution in the  $x$ -direction in  $psi$  when a drag force of  $14.61 lb$  and a lift force of  $12.36 lb$  are applied together at the tip. These results were obtained with the linear static solver of NASTRAN [ A.3 ]. Here, we can clearly see where the highest stress is produced. Figure A. 5 shows the stress contours in  $psi$  when a lift force of  $12.36 lb$  alone is applied at the tip.

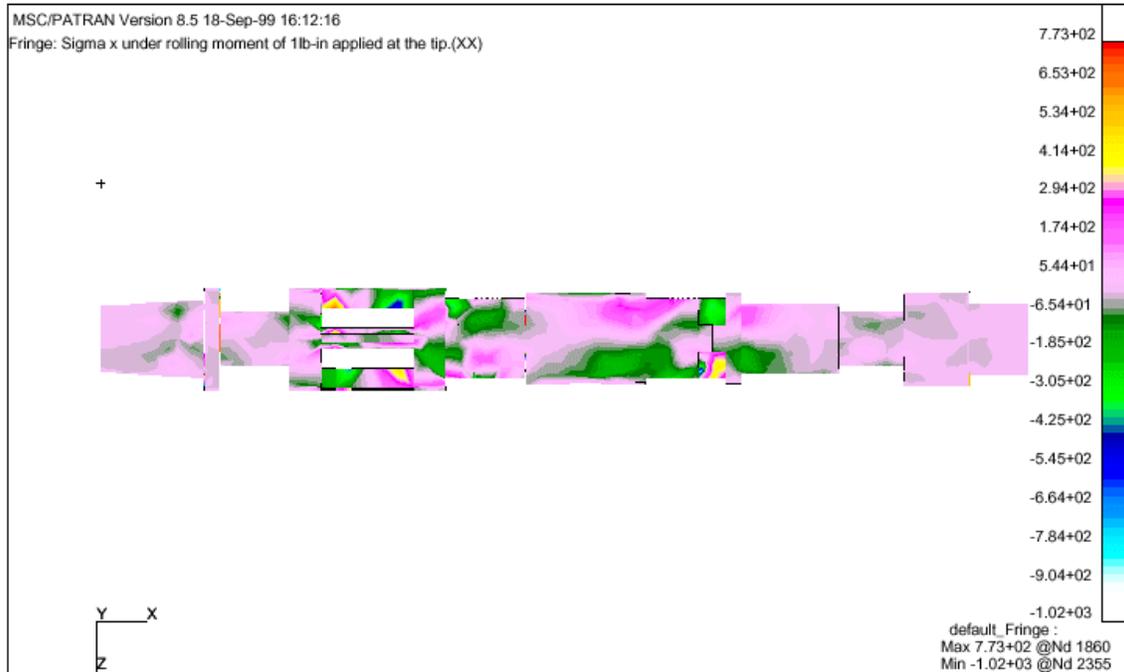


**Figure A. 4:**  $x$  component strain contour with displacement when a lift force of  $12.36 lb$  and a drag force of  $14.61 lb$  are applied together showing relative displacement.



**Figure A. 5:**  $x$  component stress distribution in  $psi$  when a lift force of  $12.36\ lb$  is applied at the tip.

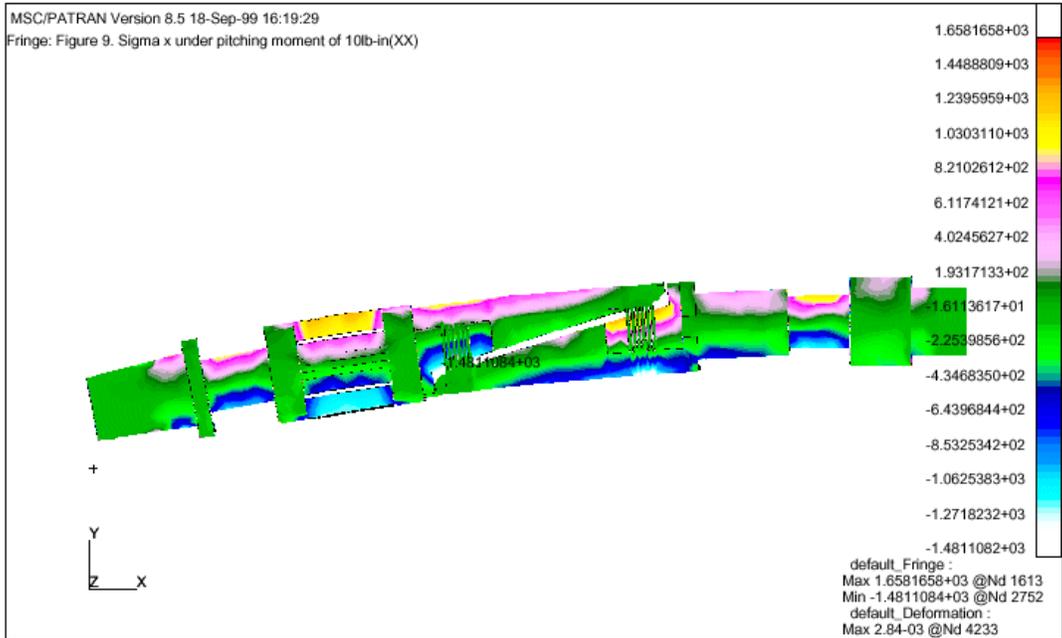
Figure A. 6 shows the contour of  $\sigma_{xy}$  on the surface of the sensor when a rolling moment of  $10\ lb\text{-in.}$  is applied. In the diagram, the highest stresses are observed near the base and the end of the balance. From this, the best placement of the sensor to observe the rolling moment seems to be putting the sensor on top of the place where the highest value of  $\sigma_{xy}$  exists. Figure A. 7 gives the  $\sigma_x$  distribution and the displacement profile when a pitching moment of  $10\ lb\text{-in.}$  is applied at the tip. Figure A. 8 shows the  $x$  component stress contour when a side force of  $10\ lb$  is applied at the tip, and Figure A. 9 shows the same under a yawing moment of  $10\ lb\text{-in.}$  applied at the tip.



**Figure A. 6:** The contour diagram of  $\sigma_{xy}$  on the surface of the sensor when a rolling moment of 10 *lb-in.* is applied around the tip of the balance (*psi*).

So far, the finite element model has been tested with all 6 components of forces and moments and all have displayed different stress concentrations for each case. From the contours, it is easy to identify the area which is affected most by each load.

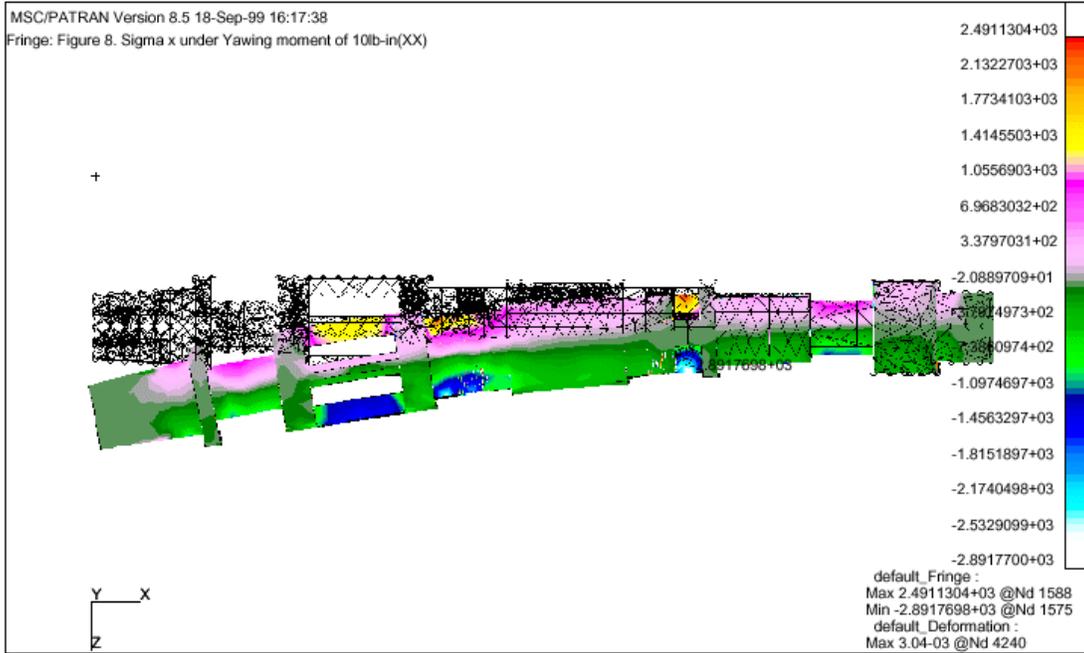
The locations determined for placement of the fiber optic sensors can be seen in Figure A. 10. From this model, we can deduce that the forward and aft foil strain gages will experience approximately 750 *psi* and 70,000 *psi*, respectively, while the chosen fiber optic locations will provide approximately 2500 *psi* in the forward section and 3500 *psi* in the aft section of the balance.



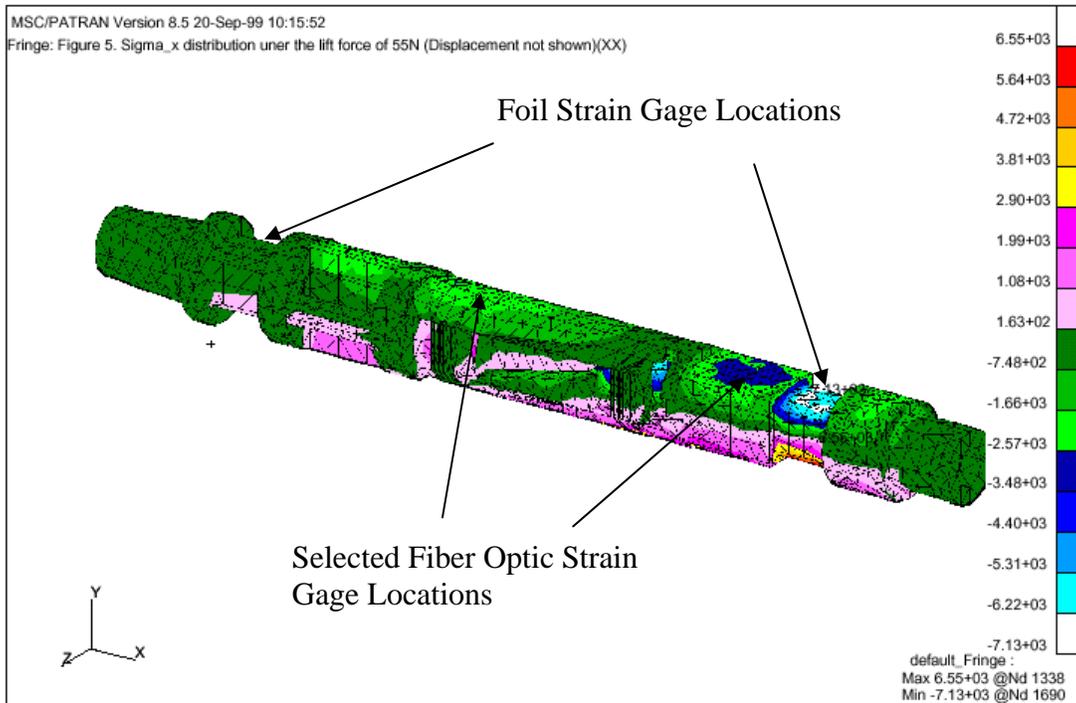
**Figure A. 7:**  $x$  component stress contour in  $psi$  when a pitching moment of  $10\text{ lb-in.}$  is applied at the tip showing relative displacement.



**Figure A. 8:**  $x$  component stress contour in  $psi$  when side force of  $10\text{ lb}$  is applied at the tip.



**Figure A. 9:**  $x$  component stress contour in  $psi$  when yawing moment of  $10\text{ lb-in.}$  is applied at the tip.



**Figure A. 10:** The location of fiber optic sensor is chosen based on  $x$  component stress distribution (in  $psi$ ) when lift force of  $12.36\text{ lb}$  is applied at the tip.

To check the validity of the finite element model, the stresses observed by the strain gages and the stresses predicted through solving the finite element model were compared to each other. For this purpose, a static calibration was performed on the balance using a calibration rig along with the calibration matrix from the manufacturer of the balance.

Table 1 shows the comparison of the experimental results with the FEM results. The experimental result was obtained by Edwards [A.1]. The results show that the FEM corresponds well to the actual stress encountered by the strain gage sensors. Results show that the error in the FEM was less than 6% in the forward and rear balance locations.

**Table A. 1:** Comparison of FEM and Experimental Stress Results.

$\sigma_{xx}$	Experimental Results	FEM Results	Error
Front Location	-710 <i>psi</i>	-748 <i>psi</i>	5.4 %
Rear Location	-6,944 <i>psi</i>	-7,130 <i>psi</i>	2.7 %

#### A.4 Remarks

The finite element model was able to successfully model the complicated strain distribution over a complex shaped sting balance. Also, the strain contours over the

complex surface helped determine the best locations of gages for the calculation of force and moments over the body. Any complicated solid model can be meshed by constructing the surface model first.

## **A.5 References**

[A.1] Edwards, Alex T. *Comparison of Strain Gage and Fiber Optic Sensors on a Sting Balance in a Supersonic Wind Tunnel*, MS Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2000.

[A.2] MSC, *MSC/PATRAN V7 User's Guide*, MSC, Los Angeles.

[A.3] Lee, J. M. *MSC/NASTRAN Linear Static Analysis User's Guide*, MSC, Los Angeles, CA, 1997.

## B. Calculation of Photon Transport

Here, the detailed procedure and steps involved in the formulation for finite element analysis for solving photon transport from the approximation of the photon diffusion equation is shown.

The photon diffusion equation that approximates the photon transport in human tissue can be expressed as below:

$$\frac{1}{c_n} \frac{\partial}{\partial t} \Phi(\mathbf{r}, t) - \nabla \cdot [D(\mathbf{r}) \nabla \Phi(\mathbf{r}, t)] + \mu_a(\mathbf{r}) \Phi(\mathbf{r}, t) = S(\mathbf{r}, t) \quad (\text{B.1})$$

$D(\mathbf{r})$ : Diffusion constant.       $F(t)$ : Diffusion intensity.

$S(\mathbf{r}, t)$ : Source strength.       $\mu_a(\mathbf{r})$ : Absorption coefficient.

$\mathbf{r}$  : Position.       $t$ : time.

$c_n$  : Speed of light in the medium

The solution of this light diffusion equation is obtained by the finite element method using the Galerkin approach:

$$[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)] \Phi + \mathbf{B} \frac{\partial \Phi}{\partial t} = \mathbf{Q} \quad (\text{B.2})$$

$$\begin{aligned}
K_{ij} &= \int_{\Omega} \kappa(\mathbf{r}) \nabla \psi_j(\mathbf{r}) \cdot \nabla \psi_i(\mathbf{r}) d\Omega, \\
C_{ij} &= \int_{\Omega} \mu_a(\mathbf{r}) c \psi_j(\mathbf{r}) \psi_i(\mathbf{r}) d\Omega, \\
B_{ij} &= \int_{\Omega} \psi_j(\mathbf{r}) \psi_i(\mathbf{r}) d\Omega \\
Q_j &= \int_{\Omega} \psi_j(\mathbf{r}) q_0(\mathbf{r}) d\Omega, \\
\Phi &= [\Phi_1(t), \Phi_2(t), \dots, \Phi_D(t)]^T \\
\kappa(\mathbf{r}) &= \frac{c}{3[\mu_a(\mathbf{r}) + \mu'_s(\mathbf{r})]},
\end{aligned}$$

where  $\kappa$  is the diffusion coefficient,  $\mu_a$  is the absorption coefficient,  $\mu'_s$  is the reduced scattering coefficient, and  $c$  is the speed of light.

The time-independent photon intensity is calculated as below:

$$\hat{\Phi}(\omega) \Big|_{\omega=0} = \frac{1}{\sqrt{2\pi}} \int_0^{\infty} \Phi(t) dt = [\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)]^{-1} \mathbf{Q} \quad (\text{B.3})$$

To introduce the concept of Photon Mean Time of Flight (PMTF), the Mellin transform needs to be introduced. The Mellin transform, which is the moment of the temporal distribution of an intensity, can be expressed as

$$\Gamma^*(s) = \int_0^{\infty} t^{s-1} \Gamma(t) dt \quad (\text{B.4})$$

where  $\Gamma$  is a time varying intensity.

Then, the mean time of the occurrence of the time varying intensity shown in Figure B. 1 is

$$\langle t \rangle = \frac{\int_0^{\infty} t \Gamma(t) dt}{\int_0^{\infty} \Gamma(t) dt} = \frac{\Gamma^*(2)}{\Gamma^*(1)} \quad (\text{B.5})$$

The definition of Fourier transform is

$$\hat{G}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega t} g(t) dt \quad (\text{B.6})$$

Differentiation by  $\omega$  results in

$$\frac{\partial \hat{G}(\omega)}{\partial \omega} = \frac{-i}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t e^{-i\omega t} g(t) dt, \quad g(t) = 0, \quad t < 0 \quad (\text{B.7})$$

When  $\omega$  is zero, the Fourier transform becomes

$$\hat{G}(\omega) \Big|_{\omega=0} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(t) dt \quad (\text{B.8})$$

$$\int_0^{\infty} g(t) dt = \sqrt{2\pi} \hat{G}(\omega) \Big|_{\omega=0} = G^*(1) \quad (\text{B.9})$$

When  $\omega$  is zero, the derivative of the Fourier transform becomes

$$\frac{\partial \hat{G}(\omega)}{\partial \omega} \Big|_{\omega=0} = \frac{-i}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t g(t) dt \quad (\text{B.10})$$

$$\int_0^{\infty} t g(t) dt = i\sqrt{2\pi} \frac{\partial \hat{G}(\omega)}{\partial \omega} \Big|_{\omega=0} = G^*(2)$$

Therefore, the mean time of flight can be expressed with the Fourier transform and its derivative as below:

$$\langle t \rangle = \frac{\int_0^{\infty} t g(t) dt}{\int_0^{\infty} g(t) dt} = \frac{G^*(2)}{G^*(1)} = \frac{i \frac{\partial \hat{G}(\omega)}{\partial \omega} \Big|_{\omega=0}}{\hat{G}(\omega) \Big|_{\omega=0}} \quad (\text{B.11})$$

Taking the Fourier transform of (B.2) results in

$$[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c) + i\omega\mathbf{B}] \hat{\Phi}(\omega) = \hat{\mathbf{Q}}(\omega) \quad (\text{B.12})$$

Differentiating with respect to  $\omega$  results in

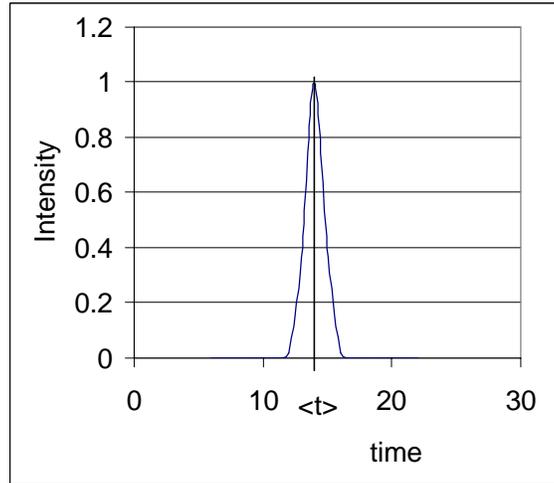
$$[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c) + i\omega\mathbf{B}] \frac{\partial \hat{\Phi}(\omega)}{\partial \omega} + i\mathbf{B}\hat{\Phi}(\omega) = \frac{\partial \hat{Q}(\omega)}{\partial \omega} = 0 \quad (\text{B.13})$$

When Q is an impulse, its Fourier Transform is a constant and the derivative is zero.

Evaluation of (B.12) and (B.13) at  $\omega = 0$  results in

$$[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)] \left. \frac{\partial \hat{\Phi}(\omega)}{\partial \omega} \right|_{\omega=0} = -i\mathbf{B}\hat{\Phi}(\omega) \Big|_{\omega=0} \quad (\text{B.14})$$

$$i \left. \frac{\partial \hat{\Phi}(\omega)}{\partial \omega} \right|_{\omega=0} = [\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)]^{-1} \mathbf{B}\hat{\Phi}(\omega) \Big|_{\omega=0} = \Phi^*(2) \quad (\text{B.15})$$



**Figure B. 1:** A time varying intensity. ( $\langle t \rangle$  is mean time of the occurrence of the intensity)

Mean time of flight is calculated as

$$\begin{aligned}
\langle t \rangle &= \frac{\int_0^\infty t\Phi(t)dt}{\int_0^\infty \Phi(t)dt} = \frac{\Phi^*(2)}{\Phi^*(1)} = \frac{i \frac{\partial \hat{\Phi}(\omega)}{\partial \omega} \Big|_{\omega=0}}{\hat{\Phi}(\omega) \Big|_{\omega=0}} \\
&= \frac{[\mathbf{K}(\kappa) + \mathbf{C}(\mu_a, c)]^{-1} \mathbf{B} \hat{\Phi}(\omega) \Big|_{\omega=0}}{\hat{\Phi}(\omega) \Big|_{\omega=0}}
\end{aligned}
\tag{B.16}$$

## VITA

Yong Yook Kim was born in Jeonju-si, South Korea in 1971. He entered Hanyang University in Seoul to study Civil Engineering in 1990. After two years of study in the Department of Civil Engineering in Hanyang University, he transferred to the Department of Mechanical Engineering and Energy Process in Southern Illinois University at Carbondale in August of 1993. After earning a Bachelor of Science Degree in Mechanical Engineering in 1996, he pursued graduate degrees in the Department of Aerospace and Ocean Engineering in Virginia Polytechnic Institute and State University, Blacksburg, VA. He earned a Master of Science degree in Aerospace engineering in May 2000. He is expected to graduate in January 2004 with a Ph. D in Aerospace Engineering. He will continue research on Finite Element Method and Structural Optimization in the Department of Aerospace and Ocean Engineering in Virginia Polytechnic Institute and State University as a postdoctoral research associate under Dr. Owen Hughes's supervision.