THE CHARACTERISATION OF MULTIPLE DEFECTS IN COMPONENTS USING ARTIFICIAL NEURAL NETWORKS

By

SIMON JOHN FARLEY

A thesis submitted in partial fulfilment of the requirements of Oxford Brookes University for the degree of Doctor of Philosophy

December 2011

ABSTRACT

This thesis investigates the use of artificial neural networks (ANNs) as a means of processing signals from non-destructive tests, to characterise defects and provide more information regarding the condition of the component than would otherwise be possible for an operator to obtain from the test data. ANNs are used both as pattern classifiers and as function approximators.

In the first part of the thesis, finite element analysis was carried out on a simple component containing a single defect modelled as a void, simulating three kinds of non-destructive test: an impact method that sent a stress wave through the component, an analysis of natural frequencies, and an ultrasonic pulse-echo method. The inputs to the ANNs were data from the numerical model, and the outputs were the x and y co-ordinates of the defect in the case of the impact and frequency methods, and the size and distance to the defect in the case of the ultrasonic method. Very good accuracy was observed in all three methods. Experimental validation of the ultrasonic method was carried out, and the ANNs returned accurate outputs for the position and size of a circular hole in a steel plate when presented with experimental data. When the ANNs were presented with noisy input data, their reduction in accuracy was small in comparison with published data from similar studies.

In the second part of the thesis, the case of two defects lying within one wavelength of each other was considered, where the reflected ultrasonic waves from each defect overlapped, partially cancelling each other out and reducing the overall amplitude. A novel ANN-based approach was developed to decouple the overlapping signals, characterising each defect in terms of its position and size. Optimisation of the ANN architecture was carried out to maximise the ability of the ANN to generalise when presented with previously unseen data.

Finally, an ANN-based general defect characterisation ‘expert system’ is presented, using data from an ultrasonic test as its input, and classifying cases according to the number of defects present. The system then characterised the defects present in the component in terms of their location and size, providing more information regarding the component’s condition than would be possible by existing techniques.

ACKNOWLEDGEMENTS

My thanks go firstly to the God of heaven, whose neural networks continue to amaze and inspire me.

EPSRC (Engineering and Physical Sciences Research Council) for funding the research.

My ever-patient wife, Hannah, who has been a constant source of support and encouragement.

My friends and family who have persuaded me to keep going when things were difficult.

My supervisors, Prof. John Durodola and Dr Neil Fellows, for their support and guidance through the project.

My colleagues and friends (the two groups are not mutually exclusive!), in particular Mike Hartman for his help with the experimental testing; Kieron Tew, Warwick Major, Martin Till, Chris Boram and Andy Slatter for their help in preparing specimens and advising on what can and cannot be machined; Daniel Bell and Mark Pollard for keeping me sane throughout the project and keeping my caffeine level sufficiently high.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 6

CHAPTER 7

LIST OF TABLES

CHAPTER 4

CHAPTER 7

SYMBOLS AND NOMENCLATURE

In certain cases, more than one definition is given, but the intended definition is clear from the context in the thesis.

A, B	Constants
$A_{0, 1, ...n}$	Antisymmetric Lamb wave (subscript denotes the particular mode)
ANN	Artificial Neural Network
b	Bias acting on a neuron
c	Bulk wave velocity in a liquid; Longitudinal wave velocity; Constant
c_1	Longitudinal wave velocity in an isotropic solid; Longitudinal wave velocity in material 1
c_{1L}	Longitudinal wave velocity in material 1
c_{1S}	Shear wave velocity in material 1
c_2	Shear wave velocity in an isotropic solid; Longitudinal wave velocity in material 2
c_{2L}	Longitudinal wave velocity in material 2
c_{2S}	Shear wave velocity in material 2
$c_{experiment}$	Speed of sound in the experimental specimen
c_{plate}	Longitudinal wave velocity in a plate
c_R	Rayleigh wave velocity
$c_{simulation}$	Speed of sound in the numerical model
c_T	Shear wave velocity
d	Distance; delay
E	Squared error between output and target values; Young's modulus
EMAT	Electro-Magnetic Acoustic Transducer
f	Function; Generalised frequency
f'	Derivative of function
fd	Frequency.thickness product
$f(y\_in)$	Activation function of neuron Y
FE, FEM	Finite Element (Method)
g	Function
$h_{corrected}$	Corrected peak height in ultrasonic signal
$h_{measured}$	Measured peak height in ultrasonic signal
I	Area moment of inertia
k	Bulk modulus; Wavenumber; Constant
K	Stiffness matrix
L	Length of beam

LMS	Least Mean Squares
m	Mass per unit length of beam; Number of neurons in output layer; Constant
M	Mass matrix
$^{M}, ^{N}$	Degrees of freedom
n	Number of neurons; Amount of noise added; number of cases
NDE	Non-Destructive Evaluation
NDT	Non-Destructive Testing
p	Number of neurons in hidden layer
R_{12}	Reflection coefficient between two materials
$RAND$	Random number between 0 and 1
R_{xy}	Cross correlation function
$S_{0, 1, ...n}$	Symmetric Lamb wave (subscript denotes the particular mode)
SCG	Scaled Conjugate Gradient
t	Target output value from a neuron; Time
T_{12}	Transmission coefficient between two materials
t_j, t_J	Target output value from a neuron (subscript indicates the particular neuron)
$\dot{u}$	Velocity in the x direction
$\ddot{u}$	Acceleration in the x direction
u, $U1$	Displacement in the x direction
Uy, $U2$	Displacement in the y direction
v	Weight between input and hidden neurons (subscript indicates the particular neurons)
v_0	Bias on hidden neuron
V_{max}	Maximum value in data set
V_n	Value with added noise
V_{norm}	Normalised value
V_o	Original value
V_{old}	Original value
V_P	Predicted value
V_T	Target value
w	Weight between hidden and output neurons (subscript indicates the particular neurons)
w_i, w_I	Weight between two neurons (subscript indicates the particular neuron)
w_{IJ}	Weight between ***I***th input neuron and ***J***th output neuron
W, W_1, W_2	Acoustic impedance (subscript denotes material)
$\bar{x}$	Mean of vector x
x, y, z	Cartesian co-ordinates

x,y	Vectors to be cross-correlated
X_1, X_i , X_n	Input neurons (subscript indicates the particular neuron)
x_i , x_I	Signal from a previous neuron (subscript indicates the particular neuron)
x_{min}	Distance across the smallest element
y	Output signal from neuron Y
$\bar{y}$	Mean of vector y
Y	Arbitrary neuron
y_in	Input signal to neuron Y
y_in_j, y_in_J	Input signal to an output neuron (subscript indicates the particular neuron)
Y_1, Y_k , Y_m	Output neurons (subscript indicates the particular neuron)
Z_1, Z_j , Z_p	Hidden neurons (subscript indicates the particular neuron)
α	Learning rate
Δ	Change in value
δ_in_j	Sum of delta inputs to neuron j
δ_j , δ_k	Error information term
θ	Initial lag (phase shift)
θ_1	Incident angle
θ_{1L}	Angle of longitudinal wave in material 1
θ_{1S}	Angle of shear wave in material 1
θ_2	Refracted angle
θ_{2L}	Angle of longitudinal wave in material 2
θ_{2S}	Angle of shear wave in material 2
θ_{cr_1}	First critical angle
θ_{cr_2}	Second critical angle
θ_g	Group velocity
θ_p	Phase velocity
μ	Momentum coefficient for training using backpropagation; Rigidity modulus; Constant for Gaussian function
ν	Poisson’s ratio
ρ	Density
σ	Constant for Gaussian function
σ_1	Incident stress
σ_R	Reflected stress
φ	Phase shift
φ^N	Eigenvector
ω	Angular frequency; Eigenfrequency
Ω	Electrical resistance

CHAPTER 1

INTRODUCTION

1.1 The importance of NDT in Quality Control

For centuries it has been desirable to assess the condition of components or manufactured items, to ensure that they meet the basic requirements for normal service. In fact, it could be said that quality control has been a consideration in the production and operation of components and structures since the beginning of civilisation. As structures became larger and more elaborate, so did the need to ensure that their structural integrity was sound. Aside from the obvious cost of remanufacture and rebuilding, an in-service failure of a critical component could have disastrous consequences in terms of loss of revenue, and of human life.

Non destructive testing (NDT) was developed as a means of testing a component for flaws without affecting the component's ability to be used in service. In the broadest terms, a non destructive test could be something as simple as a visual surface inspection, or something involving extremely expensive and specialised equipment such as a detailed x-ray inspection, but the goal of all methods of NDT is essentially the same: to identify flaws and anomalies that may affect the performance of the component under inspection. Once the severity of flaws in a component is known, it is possible to assess the component's condition and make a decision as to whether the component is suitable for use or not, or in some cases to predict the likely lifespan of the component.

1.2 Non-destructive testing (NDT) and non-destructive evaluation (NDE)

As described in section 1.1, the goal of non destructive testing is to locate flaws and anomalies in a component and, if possible, to give an indication of their location, size and nature. The output from the test would therefore typically be the size and location of any flaws. According to the process described by Halmshaw (1991), non destructive evaluation

(NDE) can be distinguished from non destructive testing (NDT) in that in NDE the information from the non destructive test is subsequently processed and a decision is made, generally as to whether or not the component is serviceable. The output from a non destructive evaluation, therefore, would often be a 'yes' or 'no' decision as to whether the component is fit for purpose, or possibly an indication of the remaining service life of the component. Rytter (1993) lists four levels of damage identification in civil engineering structures, although these could equally be applied to smaller components or structures. The four levels are given below:

Level 1: *Determination that damage is present in the structure*

Level 2: *Level 1 plus determination of the geometric location of the damage*

Level 3: *Level 2 plus quantification of the severity of the damage*

Level 4: *Level 3 plus prediction of the remaining service life of the structure*

In terms of the four levels listed above, levels 1 and 2 relate to NDT, and levels 3 and 4 to NDE. It is generally considered today (Silk, Stoneham *et al*. 1987) that a 'damage tolerant' approach to quality control is a safer option than the 'Elysian' approach used in earlier NDE, where the goal was to declare a component free from flaws. In the damage tolerant approach, NDE methods take into account the presence, size and position of flaws and a decision on the component's suitability is made. It is important to note at this point that the presence of flaws alone does not necessarily mean that a component is unsuitable for purpose, and for this reason it is vital to be able to locate and quantify any flaws present to be able to accurately assess the structural health of a component.

1.3 Rapid automated inspection of multiple components

In the manufacturing industry, production lines are capable of turning out components at a very high rate; far higher than would realistically allow a team of inspectors to assess every component for flaws using conventional methods. However, it is desirable to be able to get some indication of each component's condition before it leaves the factory, thus automated methods of NDE are attractive (Hands 1997). In order to minimise the time taken to conduct a test on a component, the test should be simple, repeatable, and require minimal human input. The output from the test would ideally class components into discrete satisfactory/unsatisfactory categories, but would also give an indication of the location, size and nature of any flaws present. Halmshaw (1991) lists the five main methods of non-destructive test: radiographic, ultrasonic, magnetic, electrical and penetrant. At the time of writing there are several additional methods, although the majority of these are extremely specialised and have been developed for certain applications or materials. Alleyne (1991) suggests that in order to conduct complete inspections of components, as opposed to inspections that will only reveal surface-breaking or near-surface flaws, radiographic and ultrasonic tests are preferable to the last three methods listed. These two methods of testing are used widely to locate both buried and surface-breaking flaws. They also lend themselves well to quantitative inspection, that is, to give an indication of a flaw's size and/or severity as well as its location. The ultrasonic method is widely employed in online inspection, and lends itself well to fully automated inspection as its output (typically a time-amplitude display) can be easily converted to be 'read' by a computer. Radiographic methods generally require more complex digital image processing in order for a computer to 'read' them, which means that the time taken for the automated test is increased. For this reason, as well as the

difficulties in gaining access to radiographic inspection equipment, the radiographic method was not considered for the purpose of this thesis.

In addition to the type of test, consideration must be given to the method of automated evaluation. Windsor (1995) suggests that automated inspection can prove more reliable than inspection relying on a human operator, as it removes the possible variables of fatigue, boredom, subjectivity, etc. from the process. However, the method of automation must be carefully chosen. As in many other fields, artificial neural networks (ANNs) applied to non-destructive evaluation have been shown to be able to derive more information from the test data than would otherwise be available to an operator (Sumpter and Noid 1996; Mahmoud and Abu Kiefa 1999). They also have advantages over conventional rule-based algorithms in that they are able to deal with noisy and/or incomplete data, and they are able to operate with minimal preprocessing of data (Windsor, Anselme *et al.* 1993). However, despite the recognised usefulness of ANNs in the area of NDE, previous work has used them predominantly for the purpose of defect classification rather than quantification, and their potential in this field has certainly not been fully explored. The majority of published work involving ANNs in NDE is in the classification of defects; that is, interpreting data from non-destructive tests and assigning each case into a discrete category. Relatively few studies have used ANNs to characterise defects; that is, to interpret data and return an estimation of the size and location of a defect. In particular, no published work has used ANNs to characterise cases where multiple defects occur in close proximity.

1.4 Aims of the research

ANN-based systems have been shown to produce accurate results for classification purposes, but very limited applications of ANNs for defect characterisation have been published. This

thesis explores the ability of ANNs to characterise defects by giving quantified information with regard to the location, size and number of defects present. Using a simple non-destructive testing approach with minimal feature extraction from test data, ANNs trained using the finite element method are assessed in their ability to characterise defects when presented with previously unseen numerical and experimental data. The underlying desire to use simple testing methods is based firstly on the need for the test to be rapid, and thus suitable for online testing at a high rate, and secondly on the desire to investigate the ability of ANNs to gather more data from the test data than would otherwise be available to an operator. Three main methods of non-destructive testing are investigated: impact, modal analysis and ultrasonic inspection.

Initially, the case of a single defect is considered, and several ANN/FEM/experiment systems are investigated to determine the size and location of the defect based on data from non-destructive tests. Following experimental validation, the thesis focusses on the ultrasonic pulse-echo method. This method is attractive as it is already widely used in industry (though often with a human operator). Using this method, the case of two defects lying in close proximity is investigated. This is of particular interest, as the ultrasonic wave reflections from two defects lying in close proximity can partially cancel each other out, thus making the defects harder to detect. An ANN system is attractive in this case as it can provide more information regarding the defects' location and size than would be available by conventional means. Finally, an ANN/FEM/experimental system is proposed that both classifies and quantifies defects in a simple component using ultrasonic A-scan data with minimal feature extraction.

1.5 Original contribution to knowledge

Following a demonstration of the effectiveness of an ANN-based approach to assessing components, a novel method of characterising two defects lying in close proximity is presented. In the case of the ultrasonic A-scan method, two defects lying within one wavelength of each other will return overlapping signals to the ultrasonic transducer, often with an overall amplitude lower than that which each defect would reflect on its own. Using minimal feature extraction from the raw ultrasonic signal, an ANN-based system is presented which is able to estimate the position and size of each defect with an accuracy comparable to that in many published works for systems characterising a single defect. This method is incorporated into a novel ANN-based 'expert system', which uses as its input raw data from an ultrasonic inspection and returns an accurate estimation of the number of defects present along with their size(s) and location(s).

1.6 Outline of the thesis

The thesis comprises nine chapters, which are summarised below.

Chapter 1 contains an introduction to the area of research undertaken.

Chapter 2 contains a detailed literature review of the use of ANNs in non-destructive evaluation. Particular attention is paid to ultrasonic inspection, and to the finite element method as a means to generate a training data set by simulating the ultrasonic test. The chapter concludes with a summary of the main points of the literature review, which inform the direction of investigation in this thesis.

Chapter 3 contains the background theory relevant to the research undertaken.

Chapter 4 describes various NDT/ANN/FEM methods to locate a single defect in a component of simple geometry. Three simple tests were investigated: impact (whereby a stress wave is sent through the component), modal analysis (whereby the natural frequencies of the component are determined) and ultrasonic pulse-echo inspection (whereby a stress wave is sent through the component and reflects off a defect). ANNs were trained and assessed with numerical data from a finite element model of each of these tests.

Chapter 5 describes the experimental work carried out during the research. Testing was carried out using impact and ultrasonic methods to validate the finite element models, and to assess the performance of the ANN-based systems when presented with experimental data.

Chapter 6 describes an ANN-based method to locate two defects using the ultrasonic pulse-echo method. The case was considered where two defects lie within one wavelength of each other, meaning that their reflected signals overlap. A preliminary study was carried out using synthesised data (generated using windowed sinusoidal functions) to assess whether the ANNs were capable of decoupling the overlapping signals. Following this preliminary study, attention was given to ultrasonic data generated using the finite element method.

Chapter 7 describes the overall ANN-based 'expert system' that identifies anomalies, and then classifies these into cases. Once classified, the size and location of each defect is estimated by the system. In this chapter, ANNs are used both as classifiers (to decide how

many defects are present in a signal) and as function approximators (to assess the size and position of any defects).

Chapter 8 contains a discussion of the various methods used, and comments on the issues surrounding online inspection. Results from the thesis are compared with previously published work, and the application of the complete system is discussed.

Chapter 9 presents a summary of the thesis, along with recommendations for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

As indicated in the introduction in chapter 1, the primary aim of this thesis is to investigate and demonstrate the effectiveness of an ANN system in classifying and quantifying cases of single and multiple defects using simple non-destructive methods. The review of literature presented in this chapter discusses previous work carried out by researchers using ANNs in NDE, with particular attention given to techniques involving elastic wave propagation such as ultrasonic inspection. Additionally, attention is given to the various methods previous researchers have used to extract features from test signals in order to present these data to ANNs, and the application of the finite element method to ANN-based NDE. Of particular interest are systems where ANNs have been trained using numerical data and then assessed with experimental data, although studies of this type are not common.

At the time of writing, no work was found by the author on the subject of using ANNs to detect and quantify multiple defects in close proximity using simple non-destructive techniques. However, an attempt is made to identify work that has been carried out in similar areas and to discuss its relevance to this thesis.

2.2 Early use of artificial neural networks in non-destructive evaluation

Interest in using artificial neural networks in NDE applications began in the mid to late 1980s, as the computer technology at scientists' disposal became not only more powerful, but also more affordable. Initial applications of ANNs were mainly in classifying defects into discrete categories, typically for cases of defects in welds or pressure vessel walls (Burch and Bealing 1986; Baker and Windsor 1989) and most published work used networks trained and assessed with relatively small experimental data sets (Perron 1988; Udpa and Udpa 1989).

ANNs were considered to be well suited to classifying real signals, as they have the ability to deal with noisy and/or incomplete data and would therefore perform better in the classification problem than a rule-based algorithm or mathematical model (Chen and Lee 1993; Ghaboussi and Banan 1994). In order to assess an ANN's ability to generalise, most studies relating to NDE have trained the ANN with a percentage of the available data (typically around 70-90%) and then used the remaining data to validate the network's performance once training is complete. If training has been successful, these previously unseen data will return correct values, i.e. the network output will match the desired target.

It soon became apparent that the use of ANNs in NDE could be exploited further than simply classifying flaws, and attention was given to the use of ANNs not just in classifying defect cases, but also in quantifying defects to give an indication of their size and/or location (Yagawa and Okuda 1996; Zgonc and Achenbach 1996). As available and affordable computer power increased even more rapidly during the mid 1990s, more and more attention was given to the application of ANNs in interpreting all kinds of data from non-destructive tests.

2.3 Early use of numerical/FE methods in non-destructive evaluation

The finite element method was already widely used around the late 1980s when researchers started to look seriously at ANNs in NDE. Early work in simulating elastic wave propagation had been going on for several years, with good correlation between experimental and numerical results claimed by several authors (Dewey 1983; Harker 1984). Dewey's paper (1983) detailed his work on the finite element modelling of ultrasonic flaw detection for transmitted and reflected waves. His work looked at three different integration algorithms,

namely Newmark-β, Wilson-θ and Houbolt integration, and their effects on the accuracy of the model's output when compared with a known analytical solution. He observed that when making a choice between integration algorithms there can be a trade-off between accuracy of frequency response and accuracy of amplitude response. This trade-off was considered to be due to the assumptions and conditions of the selected algorithms, where certain parameters had to be constrained to ensure stability of the solution. Dewey's conclusion was that different integration methods may be better suited to different types of problem. He recommended the Wilson-θ method for pulse-echo modelling due to its superior resolution in the time domain, whereas the Newmark-β method was recommended for modelling through-transmission of waves due to its superior modelling of amplitude, which would model attenuation accurately. However, Dewey concluded that the Houbolt method gave an acceptable level of accuracy in both the frequency and amplitude domains, which provided sufficiently accurate results to compare to experimental data. This method was recommended for general wave propagation simulations where transmission and reflection may both occur. The finite element method of modelling ultrasonic wave propagation was therefore considered robust, and Dewey concluded that if a suitable data set was obtained, the inverse method could be used to compare real flaws with those found by finite element simulation. By interpolating the FE results, real signals could be matched and the position and magnitude of the flaw estimated with reasonable confidence. Interest in the modelling of elastic wave propagation continued, and Harker (1984) used the finite difference method to model elastic wave scattering in plates. He investigated the effects of cracks on the behaviour of elastic waves, with a particular emphasis on Lamb waves emanating from a transducer and being scattered by cracks or slits. These results were correlated with photoelastic snapshots of experimental ultrasonic testing. The numerical study was in this

case restricted to two dimensions as with the available computer power a three dimensional model would have been too small in physical size to compare to an experimental sample. Harker's conclusion was that the finite difference method gave results very close to the theoretical results expected. Ludwig and Lord (1988) modelled ultrasonic transducers and wave propagation in a 2D plane strain model. The authors modelled a 'pitch-catch' type situation with one transmitter and one receiver, and although they generated realistic results, their conclusion was that a 3D model would be necessary to predict exact wave behaviour when scattered by a defect or feature. However, later work by several researchers (Zgonc and Achenbach 1996; Lowe, Cawley *et al*. 2002; Lowe and Diligent 2002; Hill, Forsyth *et al*. 2004) showed that a 2D simulation is often sufficient for suitably accurate modelling of ultrasonic wave motion, if no torsional waves are generated.

Zgonc and Achenbach (1996) demonstrated that the finite element method could be used to simulate ultrasonic wave movement in an aluminium plate with a circular hole, using a 2D FEM model in a similar arrangement to that used by Ludwig and Lord, although in Zgonc and Achenbach's work, each transducer acted both as a transmitter and receiver. In terms of the level of modelling accuracy, Ludwig and Lord's model used 12 elements per longitudinal wavelength in the primary direction of wave propagation, and 5 elements per wavelength in the secondary direction. This compares reasonably well with Zgonc and Achenbach's model, which used 10 elements per wavelength in the primary direction and 7 in the secondary. Subsequent work by Lowe *et al.* (Lowe, Cawley *et al*. 2002; Lowe and Diligent 2002) has shown that a minimum of around 7 elements per wavelength in the direction of wave motion must be used if the waveform is to be accurately modelled, whereas Hill *et al.* (2004) state that at least 20 elements per wavelength are necessary. Interestingly, Harker (1984) states that when 8 grid points per shear wavelength were used, the accuracy of the wave

propagation velocity was adversely affected and the numerical prediction lagged the results generated with 16 points per wavelength. With 16 points, the accuracy was greatly improved, and Harker recorded that no extra accuracy could be gained by further refining the grid size. What seems to be consistent among authors is that the Courant-Friedrichs-Lewy condition (Courant, Friedrichs *et al.* 1967) must be satisfied. That is, the maximum time increment for the dynamic modelling of an ultrasonic wave should be no greater than the time taken for the wave to travel across the smallest element in the model (Lowe, Challis *et al*. 2000; Lowe, Cawley *et al*. 2002; Lowe and Diligent 2002; Hill, Forsyth *et al*. 2004). As described in section 3.3.4, the mesh size should be suitable to adequately describe the wave's motion without being unnecessarily fine and thus increasing the computational requirements.

2.4 Application of artificial neural networks to non-destructive evaluation

ANNs lend themselves primarily to applications of function approximation or pattern recognition, and as such can be very suitable for use in NDE systems (Windsor 1995; Achenbach 2000; Ramuhalli, Udpa *et al*. 2003; Kuravsky and Baranov 2008). They are able to develop links between input data (of which there could be a very large amount) and the corresponding output data through a process known as training, described in section 3.2. Their application in non-destructive evaluation generally falls into three areas: classification of flaws into predetermined categories such as crack, void, lack of weld fusion, etc. (Chen 1989; Thavasimuthu, Rajagopalan *et al*. 1996; Margrave, Rigas *et al*. 1999; De Carvalho, Veiga *et al*. 2003), quantification of flaws to determine their severity (Aldrin, Cheng *et al*. 2000; Zapico, Gonzalez *et al*. 2003) and location of flaws (Darabi and Maldague 2002; Sahin and Shenoi 2003). These areas are not mutually exclusive, however, and it would be possible

for a single network to output results for some or all of these categories (Castellini and Revel 2000; Liu, Huang *et al*. 2002; Farley, Durodola *et al*. 2008; Luna-Aviles, Hernandez-Gomez *et al*. 2008). The input signals used to train the ANN(s) can either be from numerical data, such as a finite element simulation (Yagawa and Okuda 1996), from experimental data (Legendre, Massicotte *et al*. 2001), or a combination of both (Zgonc and Achenbach 1996). As ANNs are capable of both classification and quantification, an ideal ANN 'expert system' would, from the data recorded during a non-destructive test, be able to detect all flaws present in a component and give an indication of the nature, location and size of the flaw(s), along with a recommendation as to whether the component under inspection is fit for purpose (Udpa and Udpa 1997).

2.4.1 Artificial neural networks as flaw classifiers

The first use in the open literature of ANNs in experimental NDE was to classify test results into predetermined categories (Baker and Windsor 1989; Windsor, Anselme *et al*. 1993), i.e. to class the area or component under inspection as being fit or unfit for purpose, and, if unfit for purpose, to classify the flaw(s) present. Difficulties may exist, however, in representing the results of various non-destructive tests to the ANNs; what may be very obvious to a human operator looking at a screen could turn out to be quite difficult to represent numerically as an ANN input.

In an early ANN study, Baker and Windsor (1989) used experimental ultrasonic data to train and assess two different types of neural network: a standard back-propagation network (reported in Windsor, Anselme *et al.* 1993) and a Hopfield network (Hopfield 1982). The Hopfield network is *autoassociative*, meaning it is trained to recognise previously seen

patterns when presented with a partial, or noisy, representation of that pattern. This is in contrast to the *heteroassociative* backpropagation network, which is trained to associate an input pattern with a separate output pattern or class. The authors used a data set of 83 defects of known type (falling into four categories) from steel pressure vessels, and from the ultrasonic signals six separate feature parameters were extracted to represent the signals to the ANNs. It was shown that some of these features were linked to others, but that all were necessary in order for the Hopfield network to correctly classify cases. Interestingly, the authors commented that even though these feature parameters were all needed, the way in which they were presented to the networks was a result of trial and error to find the best method. The four defect cases considered were: smooth crack, rough crack, pores or slag. When the networks used in this study were trained and assessed with the same data set, the result was 100% correct classification in both cases, although this is to be expected, assuming the data were all correct, as the learning and testing sets were identical. When the learning set comprised one half of the data set, chosen at random, one or two cases from the full set were incorrectly classified, whilst the remainder were correctly classified. It is worth noting at this point that the Hopfield network is designed to recognise previously seen patterns, and will only perform well if the output classes are linearly separable. In this case linear separability was achieved only by using all six feature parameters. Also worthy of note is the fact that both networks required a considerable amount of preprocessing of the experimental data in order to correctly classify cases.

Windsor, Anselme *et al.* (1993) presented a method of using various ANNs to automatically classify weld defects from ultrasonic signals. The authors compared classical techniques for classifying defects with ANN approaches, and stated that one of the best opportunities for ANNs lay in their potential to analyse ultrasonic data without the need for detailed feature

extraction, which was invariably necessary for conventional 'rule-based' classification techniques. That said, in this case they concluded that some amount of prior knowledge of transformations of the ultrasonic signal would be necessary to ensure that the ANN could create meaningful decision boundaries between defect classes, so a small degree of pre-processing was deemed to be necessary. This was also shown by Udpa and Udpa (1989) to be the case when eddy current signals were used as inputs to multilayer perceptron ANNs. The signals were passed through a preliminary network, which computed a parametric representation of each signal using the Fourier descriptor method. This method meant that the representations of each signal were invariant under scaling, rotation and translation operations and therefore enabled the ANN to more easily distinguish between different cases. Udpa and Udpa (1997) describe this process as crucial for ensuring the classification process is insensitive to variations in test conditions.

In the case of ultrasonic testing of welds, Veiga *et al.* (2005) used experimental results from weld inspection to train and assess various ANNs in order to categorise signals into four discrete classes: no flaw, porosity, lack of penetration and lack of fusion. In this work, both the pulse-echo and time of flight diffraction (TOFD) methods of ultrasonic inspection were used, and all the ANNs used were of the multilayer feedforward type, trained using a supervised backpropagation training algorithm. Veiga found that the signals from the TOFD method benefited from pre-processing via a low pass filter, whereas the pulse-echo method's results were not significantly improved by this pre-processing. The outputs from the ANNs in this work were fairly accurate, with an average of around 72% of cases correctly classified, although the data set was once again of a fairly small size, a common issue when working with experimental data. However, Chen (1989) states that the sample size does not necessarily make a difference to the performance of an ANN for classifying defects, and

demonstrates that only a small improvement was made in classification accuracy when the data set increased from 30 to 105 samples.

2.4.2 Artificial neural networks as flaw quantifiers

Although previous work had shown that the size of a flaw could be inferred from an ultrasonic echo (Silk, Stoneham *et al*. 1987), it was not until ANNs had been extensively used for flaw classification that they were applied to the problem of flaw quantification. In their very comprehensive paper detailing the ability of neural networks to assist in various computational mechanics problems, Yagawa and Okuda (1996) used the finite element method to simulate ultrasonic inspection for locating crack tips. They used an ultrasonic frequency of 2.5MHz, modelled as a single period of a sinusoidal pulse, and simulated a steel plate in 2D using four-noded quadrilateral elements, with vertical cracks propagating from the base of the plate. Later work has shown that a more accurate pulse can be represented as a tone burst, with several periods of a sinusoidal function inside a Hanning window (Lowe, Challis *et al*. 2000). It is important to note, however, that Yagawa and Okuda's purpose in this work was more to demonstrate the ability of the ANN system than to accurately model ultrasonic wave propagation. The pulse was applied to a region on the top of the block, and throughout the simulation vertical displacements, *Uy*, were measured at seven separate nodes on the top of the block. The features used to represent each case to the ANN were the normalised time and height of the first peak of the ΔUy signal; that is, the *Uy* signal from a block without a crack subtracted from the *Uy* signal from a block with a crack. This method of comparing a 'healthy' or reference signal to a test signal has been widely used, and is particularly useful in identifying novelties or anomalies in multiple component inspection (Mustapha, Manson *et al*. 2005; Mustapha, Worden *et al*. 2007). Yagawa and Okuda's work

in this study was concerned with locating the tip of a vertical crack, with the outputs of the ANN being the (x,y) co-ordinate of the crack tip, although this technique has also been shown to be robust when locating voids, rather than cracks, in components (Hernandez-Gomez, Durodola *et al*. 2005; Farley, Durodola *et al*. 2008). Work carried out by Hernandez-Gomez, Durodola *et al.* (2005) involved sending a stress wave (modelled as an instantaneous pressure) along a steel bar, and measuring the lateral movement of the bar in several locations at the far end as the stress wave reached it. The presence of a void in the bar (between the point of pressure and the point at which lateral movement was measured) altered the response of the bar at the far end by interfering with the wave's motion. The resulting displacements at various nodes were used to successfully train ANNs which were able to output the (x,y) co-ordinate of the void. Multilayer backpropagation networks of three and four layers were used in their study, although accuracy did not improve significantly when a four layer network was chosen.

The neural network used by Yagawa and Okuda in their study to locate crack tips was of the multilayer feedforward type, trained using the enhanced backpropagation (EBP) algorithm and having three layers in total, i.e. one hidden layer. The number of neurons in the hidden layer was determined by trial and error, which seems to be a commonly used method (Yagawa and Okuda 1996; Sahin and Shenoi 2003). Bishop (1995) suggests that a good starting point for the number of neurons in the hidden layer should be equal to half of the total number in the input and output layers combined, but from this point it may be necessary to vary the number of neurons in the hidden layer and monitor the effect on the error output from the network. The error should reach a minimum when the optimum number of hidden neurons is reached, or should converge to a small value as the number of hidden neurons is increased, depending on the application of the ANN. Similarly, the initial weights and biases

of the network can have a direct effect on the network's convergence during training, which was the case in the study carried out by Yagawa and Okuda. They trained networks with twenty different initial states, and then took a mean value of the ten best results to be the initial state of the network finally used. However, although this may have saved some time during the training process for each ANN, it could be argued that this practice was not strictly necessary. Backpropagation algorithms will adjust weights iteratively until they reach values that minimise the error (Fausett 1994, Picton 2000), thus starting the training process with weights at random values will still lead to correct convergence, assuming a solution is possible, at the expense of a greater number of epochs required during the training process.

Two cases were investigated by Yagawa and Okuda (1996) using the data obtained from their FE simulation: the first case used data recorded at two nodes, and the second case used data recorded at seven nodes. The results obtained showed that accuracy of prediction of the crack tip location was significantly improved using a higher number of sensing points, a trend observed in similar work by Hernandez-Gomez, Durodola *et al.* (2005).

In a study to use an ANN method to detect and quantify cracks propagating from a circular hole, Zgonc and Achenbach (1996) used a five layer perceptron network with 13 output units, each output relating to a particular crack length. Essentially this was still a neural network classifier, rather than a quantifier, but Zgonc and Achenbach state that the use of a single output neuron was considered and discarded as the resolution in notch length evaluation was lost. In the network used, only one output neuron would have a positive value, thus identifying the particular crack length under inspection. It is interesting to note that Zgonc and Achenbach employed a five layer network, where the first two layers acted as a compact form of preprocessing and the remaining three layers formed the more common three-layer neural network.

The ability of ANNs to return accurate results when presented with noisy data was demonstrated by Sahin and Shenoi (2003) using vibration-based NDE methods coupled with neural networks. They showed that it was possible to accurately predict the location and severity of large flaws in composite beam-like structures, modelled using the finite element method. In their study, the inputs to the ANN were the first three undamped natural frequencies of the beam along with the corresponding mode shapes. Damage was modelled as a single localised area of reduced stiffness, reaching from the top to the bottom of the beam, thus its location could be expressed in terms of the length of the beam. Very good correlation was observed between the ANN outputs and the actual location and severity of the flawed area, even with data to which noise had been artificially added.

The practice of adding artificial noise was employed in a previous work by the author (Farley, Durodola *et al*. 2008) when locating single voids in a 2D steel bar using the ultrasonic A-scan method, and it was found that the accuracy of the ANNs' predictions of location and size of a defect were affected in proportion to the amount of noise added to the data. In this case, ten samples of noisy data were taken, and a mean value for each data point used as the input to the ANNs.

2.5 Numerical methods as a means of training artificial neural networks

From the open literature, it is clear that many researchers have made good use of numerical methods in order to generate large data sets without having to experimentally test large numbers of components. However, studies that train ANNs with numerical data and then assess their performance with experimental data are comparatively rare. Certainly when considering the necessity to train ANNs, the finite element method is particularly attractive as a tool for generating large, accurate data sets. Moser, Jacobs *et al.* (1999) demonstrated that an accurate FE model could be used to generate accurate data regarding the propagation of guided waves in simple components, and that when these results were validated with experimental results, the FE method could be considered robust for modelling components of more complex geometry. This method was presented as particularly attractive in cases where an analytical solution was not easily obtained. Similarly, Baskaran *et al.* (2007) showed that if the FE method of modelling ultrasonic inspection correlated well with experimental data in the case of a component of fairly simple geometry, it could also be assumed to be reliable in the case of a more complex component, and therefore the necessity for multiple experimental cases to be prepared would be eliminated. The work presented showed that using three ultrasonic probes, a component of complex geometry could be inspected for defects anywhere within its boundaries, i.e. the entire area could be inspected using three static probes, thus eliminating the need to move probes during testing.

In the majority of the open literature involving ANNs and NDE the ANNs were trained with data produced either by the FE method or by experimental means; only in a small minority of cases did researchers combine FE and experimental data to validate the capability of their ANN systems. However, as previously detailed in section 2.3, if good correlation between the FE model and experimental testing is observed in terms of the output values from the FE

model, it logically follows that the data generated by the FE method are reliable enough to train an ANN.

In both ultrasonic inspection (Yagawa and Okuda 1996; Zgonc and Achenbach 1996; Hernandez-Gomez, Durodola *et al*. 2005; Farley, Durodola *et al*. 2008) and vibration-based NDE (Sahin and Shenoi 2003; Luna-Aviles, Hernandez-Gomez *et al*. 2008) authors have shown the finite element method to be useful in creating parametric studies of components with flaws of various types and in various locations. The data recorded from the FE studies were then preprocessed and fed into ANNs.

Some authors (Zgonc and Achenbach 1996) have published work where combined FE and experimental data were used to train and assess ANNs, although such studies are not common. Given that a lot of work has been carried out by researchers in the area of accurately modelling non-destructive tests using the FE method (Lowe, Cawley *et al*. 2002; Hassan and Veronesi 2003; Bartoli, Lanza Di Scalea *et al*. 2005; Baskaran, Lakshmana Rao *et al*. 2007), it would seem to be widely accepted that if the guidelines given by previous researchers are used, an FE simulation of a non-destructive test is valid for training and assessing an ANN.

2.6 Summary of literature review

From the literature it is clear that ANNs can be used to interpret signals from various kinds of non-destructive test, and they have been shown to provide more information regarding the condition of the component under inspection than would otherwise be immediately available to an operator, or discernable using rule-based algorithms. ANNs have several advantages over conventional rule-based algorithms in that they have the ability to generalise, thus they

can be presented with noisy or incomplete data and still return useful results. Another claimed advantage of ANNs is in their potential ability to deal with data from non-destructive tests without extensive pre-processing (Windsor, Anselme *et al*. 1993), however almost all published work involving ANNs in NDE has required significant pre-processing and feature extraction of some kind. Although a thorough literature search was carried out by the author, no authoritative published work was found that addressed the question of how much pre-processing of data is required for an ANN-based system to accurately characterise defects. It follows that there is clearly scope to investigate methods of very simple and rapid feature extraction from non-destructive test data in order to present these data to an ANN system. This thesis looks at various methods of gathering data from non-destructive tests, and evaluates different methods of feature extraction and pre-processing of the data. In terms of generating large training data sets for an ANN system, the finite element method has been demonstrated as a valid method for producing data that correlate well with experimental results. Very few studies exist, in the context of non-destructive evaluation, where the finite element method has been used to train ANNs that are subsequently assessed with experimental data. This thesis investigates the extent to which this approach can be implemented, and comments on the feasibility of applying the technique to automated inspection. Despite the evidence from the literature that ANNs are able to provide additional information when presented with data from non-destructive tests, no published work was found on the characterisation of multiple defects in close proximity. Thus, this was identified as an area in which novel research could be conducted. This thesis investigates the problem of multiple defects in close proximity, and evaluates the usefulness of ANNs in characterising multiple defects in terms of their location and size.

CHAPTER 3

BACKGROUND THEORY

3.1 Introduction

As described in chapter 1, this thesis investigates the use of ANNs in NDE, with particular emphasis on using ANNs to provide more information from the non-destructive test data than would otherwise be available to an operator. This chapter presents the background theory relating to the research carried out.

3.2 Artificial Neural Networks

An artificial neural network is essentially an information processing tool designed to operate in a manner based on some of the functions of a biological brain. Like a biological brain, an artificial neural network is made up of a collection of individual neurons, connected in some way and able to pass signals between them. Additionally, like a biological brain, artificial neural networks require training in order to function efficiently. During training, an ANN draws links between input and output (target) data by iteratively adjusting the strength of the connections between neurons. Assuming the training process is successful, a trained ANN can provide accurate outputs when presented with both previously seen input data (memorisation) and previously unseen input data (generalisation). This would apply to pattern recognition (e.g. the ANN would correctly classify an input data set representing light and dark pixels to a letter of the alphabet), and to function approximation (e.g. the ANN would provide a correct output value, such as the location of a void, when presented with a set of input values, such as a series of numerical values representing a reflected ultrasonic pulse). It should be noted that when an ANN is used for generalisation, it is vital that the data set used for training is representative of the entire range of inputs that may be presented to the network in normal use. ANNs are very good at interpolating, but very poor at extrapolating

(Windsor 1995). Artificial neural networks offer advantages over conventional rule-based algorithms in the area of generalisation; in much the same way as a biological neural network, they are able to operate correctly even when presented with incomplete, noisy or corrupted input data.

According to Fausett (1994), the following assumptions are made regarding the operation of artificial neural networks:

> *"Information processing occurs at many simple elements called neurons.*
>
> *Signals are passed between neurons over connection links.*
>
> *Each connection link has an associated weight, which, in a typical neural network, multiplies the signal transmitted.*
>
> *Each neuron applies an activation function (usually nonlinear) to its net input (sum of weighted input signals) to determine its output signal."*

The similarity between a biological neural network and an artificial neural network can vary widely, but it is generally held (Fausett 1994; De Wilde 1997; Picton 2000) that an understanding of the function of a biological neural network can aid understanding of an artificial neural network. Biological neurons are found in any nervous system, which in a human comprises the brain and spinal column (the central nervous system) and various motor neurons in muscles and skin (the peripheral nervous system). Collectively these neurons, of which there are around 100 billion in the human nervous system (Guyton 2000), form a neural network, i.e. they are connected to each other in some way.

3.2.1 Operation of a neuron

The operation of neurons can vary, depending on their particular function, but they share many common features. In fact, the operation of an individual biological neuron is very similar between species; it is the architecture of connections between neurons that varies far more widely. A biological neuron, shown in figure 3.1, is primarily made up of three components: dendrites, soma and axon. The signal inputs to the soma are transmitted across synaptic gaps between the axons of the previous neurons, and subsequently along the dendrites.

Figure 3.1 A biological neuron
following Hernandez-Gomez, Durodola *et al*. (2005)

This transmission is achieved via a chemical process, which excites electrical impulses. Impulses are modified by the chemical transmitter so that they vary in intensity or frequency, in a similar manner to the alteration of signals by weights in an artificial neural network. The electrical impulses are summed by the soma, and, when a sufficient input signal is received, the soma 'fires', meaning that the summed signal is transmitted through the axon to the next layer of neurons. It would be logical at this stage to assume that the output from the soma is binary, i.e. either it fires or it doesn't fire, but variation in the frequency of firing means that the intensity of the output signal over a particular time window can be varied, so the output is

effectively variable. Neurons can be connected in an infinite number of ways, and the signals passed between them can be inhibitory (acting to prevent the receiving neuron from 'firing') or excitatory (to increase the likelihood of the receiving neuron's 'firing'). In the human nervous system, each neuron can in certain cases be connected to as many as 10,000 other neurons (Bloom 1986).

The mathematical model of a neuron is, fairly predictably, much simpler than its biological counterpart, but nonetheless is able to perform some similar functions. Each neuron has a set of inputs, which can be either from external sources (inputs to the network) or from other neurons within the network. Inputs to a neuron are weighted (i.e. assigned a multiplier), summed, and then a function is applied to the resultant value. This function is referred to as an activation (or transfer) function. Typical activation functions include step, sigmoid and linear functions, and are chosen depending on the particular application. A simple artificial neural network is shown in figure 3.2, with an individual mathematical neuron shown in figure 3.3.

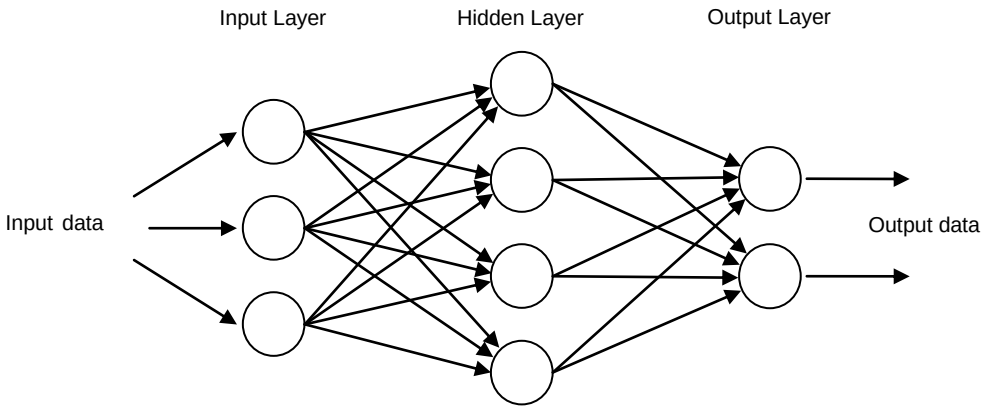


Figure 3.2 A simple artificial neural network

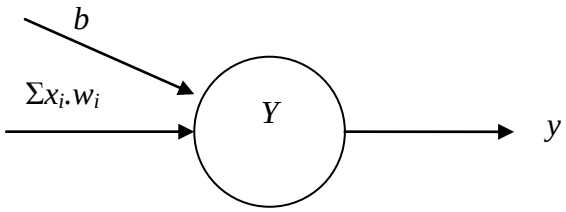


Figure 3.3 A mathematical neuron

The input signal to the neuron *Y* is given as

$$y\_in = b + \sum_{i=0}^{n} x_i.w_i \tag{3.1}$$

The output from neuron *Y* is given as

$$y = f(y\_in) \tag{3.2}$$

where *y_in* is the net input to neuron *Y*, *b* is the bias on unit *Y*, x_i is the signal from the previous neuron, w_i is the weight between the connected neurons, *n* is the number of other neurons passing their signal to neuron *Y*, and *y* is the output signal from neuron *Y*. It can be seen from equation 3.1 that the net input value depends both on the sum of the weighted signals from previous neurons, and also on the bias. In many ANNs, the weights (and biases) are adjusted during the training process to allow the network to 'learn' relationships between input and output data. The bias can be considered as an additional weighted signal into neuron *Y* with a constant signal value of 1, i.e. the signal value is independent of any input data although the weight acting on the bias can be adjusted in the same way as the weights between neurons. The purpose of the bias is to enable adjustment of the threshold at which the neuron fires. This concept is explained in more depth in section 3.2.2.1 and shown graphically in figures 3.4 and 3.5.

The activation function of the neuron will apply to the net input, *y_in*. Typical activation functions operate in the range [-1, 1], meaning that the output from the neuron *Y* will always be within this range. It has been shown that for very simple ANNs using step activation functions, binary inputs and targets (i.e. values of either 0 or 1) can yield good results, but in most cases bipolar inputs and targets (i.e. values of either -1 or 1) tend to produce better convergence of weights during training (Fausett 1994). This is because the bipolar signal allows weights to be modified even when the target is 'off', i.e. when the target is -1. For example, in the case of binary targets, it is not possible for a network trained using the Hebb rule (a simple algorithm for updating weights that multiplies the weight by the target value) to learn a pattern with a target of 0, as this does not produce any updating of the weights during training.

3.2.2 Types of network and their applications

Artificial neural networks can be designed and optimised for particular applications. An artificial neural network is essentially a network made up of layers of neurons, as shown in figure 3.2. The layout of the network is referred to as its *architecture*. Neurons within a given architecture can be connected in any number of ways, and can pass data forwards or backwards through the network. For the purpose of this thesis, only networks that pass data forward were considered, as the goal was to draw links between input data (information from a non-destructive test) and output data (such as the position of a defect) rather than perform any kind of internal computation. Networks of this kind are referred to as *feedforward* networks, so named because during normal operation data are fed only in the forward direction, from input to output. One of the main advantages of a neural network over a

'conventional' rule-based algorithm is in the ANN's ability to give a correct output when presented with previously unseen input data, or with corrupted or incomplete input data.

3.2.2.1 Linear separability

For classifier ANNs, it is important to address the issue of linear separability, that is, the ability for an ANN to correctly distinguish between different classes of pattern for all possible input vectors. For any classifier network, the desired output from a given output neuron is positive if the input pattern is a member of its particular class, or negative if the input pattern is not a member of that output neuron's class. If bipolar data are considered, a positive output would be represented by '1' and a negative output by '-1'. In the case of a classifier network, the output from each neuron needs to be bivalent, so a step activation (transfer) function is used. In this case, the threshold of the step function would be at zero, meaning that a positive input signal would result in an output of 1, and a negative input signal would result in an output of -1. This threshold is known as the *decision boundary*, and can be shown mathematically thus:

The input signal to the output neuron is given as

$$y\_in = b + \sum_{i=0}^{n} x_i.w_i \tag{3.3}$$

Knowing that the decision boundary must be at zero, this equation can be expressed so that *y_in*=0, giving

$$b + \sum_{i=0}^{n} x_i.w_i = 0 \tag{3.4}$$

This equation represents the decision boundary for the particular output neuron; for two inputs plotted against each other, this boundary would be represented by a line, for three inputs a plane, for four inputs a hyperplane, and so on. If a combination of weights and bias exists so that all the input vectors that require a positive output lie on one side of this line (or plane, hyperplane, etc.), and all input vectors that require a negative output lie on the other side of the line, the problem is said to be *linearly separable*. The implication of this is that a neural network using linear or step activation functions is able to correctly classify data that are linearly separable, thus the problem can be solved. An example is given below for the case of the logical AND, OR and XOR functions, all with two inputs and one output (following Fausett 1994).

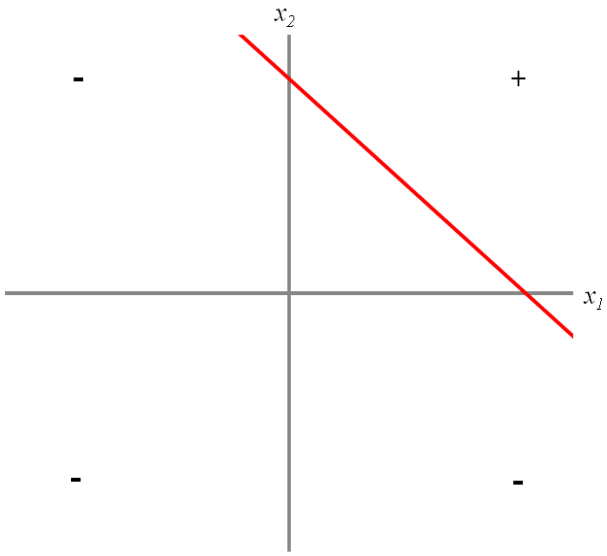
In this case, equation 3.4 can be expressed as:

$$b + x_1.w_1 + x_2.w_2 = 0 \tag{3.5}$$

It can be seen that equation 3.5 can be rearranged to make x_2 the subject as follows

$$x_2 = -\frac{w_1}{w_2}.x_1 - \frac{b}{w_2} \tag{3.6}$$

thus giving the equation of the line forming the decision boundary when x_1 and x_2 are plotted against each other. Figures 3.4 to 3.6 show the desired output of the logical AND, OR and XOR functions with a possible decision boundary plotted. The region where the output should be positive is indicated on each graph by '+', and negative by '-'.



$x_2 = -x_1 + 1$

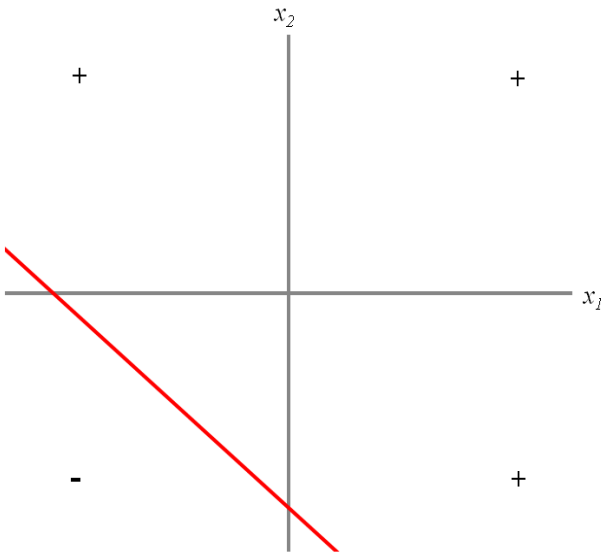
thus suitable weights could be :

$w_1 = 1$

$w_2 = 1$

$b = -1$

Figure 3.4 A possible decision boundary for the AND function



$x_2 = -x_1 - 1$

thus suitable weights could be :

$w_1 = 1$

$w_2 = 1$

$b = 1$

Figure 3.5 A possible decision boundary for the OR function

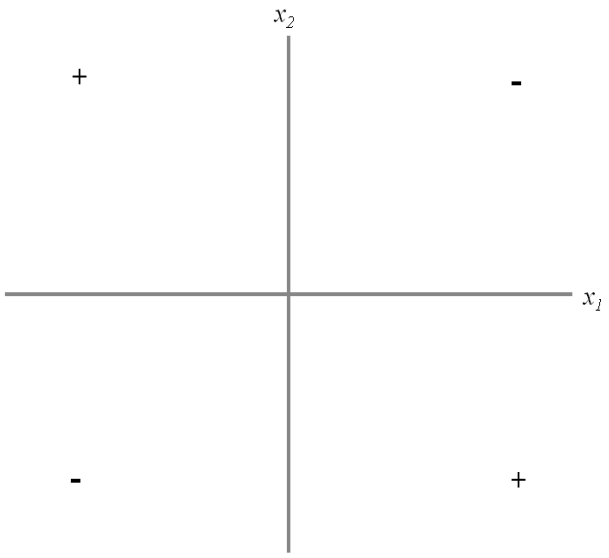


Figure 3.6 Desired output for the XOR function – a single line cannot create a suitable decision boundary in this case

As can be seen from figures 3.4 to 3.6, the AND and OR functions are linearly separable and thus can be solved by a linear neural network, as long as a bias is employed. The bias acts to provide an offset for the decision boundary so that it can cross either axis at any point. Without a bias, the decision boundary would be forced to pass through the graph's origin, possibly making the problem unsolvable. The XOR function is not linearly separable, even with the use of a bias, and thus is an example of a problem that cannot be solved by a single layer neural network.

3.2.2.2 Training using the delta learning rule

As described in section 3.2.1, in order for an ANN to operate correctly, it first needs to be trained. This process generally involves passing data through the network and comparing the network's output with the desired target output, then modifying weights and biases to reduce the error between output and target. This process is repeated iteratively; each passing of the data set through the network is referred to as an *epoch*.

The basis for most weight updating algorithms in multilayer, feedforward ANNs is the delta learning rule, also known as the least mean squares (LMS) or Widrow-Hoff rule (Fausett 1994). The delta rule for weight updates aims to minimise the squared error between the input to the output neuron, *y_in*, and the target, *t*, for all patterns in the training set, and makes use of a *learning rate*, α, which can be adjusted to optimise the rate at which weights are updated. A high value of α may reduce training time if the weights manage to converge to a stable solution, but there is the possibility that weights will be adjusted too much, and convergence will not be achieved. Conversely, a low value of α will increase training time, but will lead to better convergence of weights.

For a network with a single output neuron, the delta rule for each weight update can be expressed as follows:

$$\Delta w_{\boldsymbol{I}} = \alpha(t - y\_in).x_{\boldsymbol{I}} \tag{3.7}$$

This rule also holds true for the bias, which can be assumed to be another weight with a constant input of 1, as described in section 3.2.1. Following Fausett (1994), the upper case subscript ***I*** is used here and in the subsequent derivation to indicate a particular (but arbitrary) weight being adjusted, whereas the lower case subscript *i* is used in the summation nomenclature.

The delta rule can be derived by expressing the squared error for a single training pattern as follows:

$$E = (t - y\_in)^2 \tag{3.8}$$

As the error is a function of all the weights associated with the particular value of *y_in*, the error gradient can be expressed as the partial derivative of *E* with respect to each weight, w_i, where i = 1 to n, the number of weighted connections. In order to minimise the error most rapidly, it can be seen that the desired modification of weights should be in the direction opposite to the error gradient (whose direction would increase the error most rapidly). Thus, for each weight, w_I, the desired direction of modification can be expressed as

$$-\frac{\partial E}{\partial w_I} \tag{3.9}$$

As $y\_in = b + \sum_{i=1}^{n} x_i.w_i$ (from equation 3.3), it follows that

$$\frac{\partial E}{\partial w_I} = -2(t - y\_in).\frac{\partial y\_in}{\partial w_I} \tag{3.10}$$

which yields

$$\frac{\partial E}{\partial w_I} = -2(t - y\_in).x_I \tag{3.11}$$

Thus, the opposite gradient to minimise the squared error can be expressed (using the learning rate, α, as a positive constant) as

$$\Delta w_I = \alpha(t - y\_in).x_I \tag{3.12}$$

In the case of a network with multiple output neurons it can be shown that the delta rule can still be applied during training. Again following Fausett (1994), the delta rule for adjusting the weight from the ***I***th input unit to the ***J***th output unit is given as

$$\Delta w_{IJ} = \alpha(t_J - y\_in_J).x_I \tag{3.13}$$

The subscript ***IJ*** denotes the weight between input unit ***I*** and output unit ***J***. Again, the uppercase subscripts are used to denote particular (but arbitrary) units; the lowercase subscripts are used in the summation nomenclature.

The weight adjustment expression can be derived by expressing the squared error for a particular training pattern as

$$E = \sum_{j=1}^{m}(t_j - y\_in_j)^2 \tag{3.14}$$

where *j* represents each of *m* output units. Again it can be seen that as the error is a function of all the weights, the fastest reduction in error will be to modify each weight w_{IJ} in the direction $-\frac{\partial E}{\partial w_{IJ}}$. In order to derive a formula for $\frac{\partial E}{\partial w_{IJ}}$ for the particular weight w_{IJ}, it must be noted that because the weight w_{IJ} only affects the error at output unit Y_J, the error gradient can be expressed as:

$$\frac{\partial E}{\partial w_{IJ}} = \frac{\partial}{\partial w_{IJ}}\sum_{j=1}^{m}(t_j - y\_in_j)^2 \tag{3.15}$$

$$= \frac{\partial}{\partial w_{IJ}}(t_J - y\_in_J)^2 \tag{3.16}$$

If the net input to output unit Y_J, y_in_J is expressed as

$$y\_in_J = b + \sum_{i=1}^{n} x_i . w_{iJ} \tag{3.17}$$

then the derivative becomes

$$\frac{\partial E}{\partial w_{IJ}} = -2(t_J - y\_in_J)\frac{\partial y\_in_J}{\partial w_{IJ}} \tag{3.18}$$

$$= -2(t_J - y\_in_J).x_I \tag{3.19}$$

Thus the direction in which the individual weight, w_{IJ}, should be adjusted to minimise error can be expressed as

$$\Delta w_{IJ} = \alpha(t_J - y\_in_J).x_I \tag{3.20}$$

Fausett (1994) states that by minimising the error for a training set in this way (it is possible that weights may not exist to fully solve a problem, thus minimal error is the next best solution), the error for previously unseen data will also be minimised, provided that the training data are representative of the entire data set.

3.2.2.3 Multilayer networks and backpropagation

As can be inferred from the previous few sections, the more neurons and hidden layers that are present in a neural network, the more complex and computationally demanding are the weight modifications during training. During the training process of multilayer neural networks (excluding the MADALINE, which cannot be trained using backpropagation techniques (Winter and Widrow 1988) due to its non-differentiable activation function) weights are modified starting with those acting on the output layer, then moving to those

acting on the hidden layer closest to the output layer, and so on. As the weights are modified in this order (from output to input), the term *backpropagation of errors*, or often simply *backpropagation*, is used to describe the training process. For the majority of multilayer ANNs, this process takes place using what is referred to as the *generalised delta rule*. A typical backpropagation network with one hidden layer is shown in figure 3.7. Following Fausett (1994), there are n units in the input layer, p units in the hidden layer and m units in the output layer. The bias acting on each neuron in the hidden and output layer is again presented as an additional input with a constant value of 1 that is transmitted to the receiving neuron via an adjustable weight. The backpropagation algorithm presented is suitable for any real integer values of n, p and m, and essentially has three stages. The first stage is the feeding forward of the information through the network, the second stage is the backpropagation of the errors (or values representing the errors) and the final stage is the adjustment of the weights. In order for the backpropagation algorithm to be implemented, activation functions must be "*continuous, differentiable and monotonically non-decreasing*" (Fausett 1994). Typical activation functions for backpropagation networks include the binary sigmoid and bipolar sigmoid, which return a value in the range [0, 1] or [-1, 1] respectively for any value of input. The binary sigmoid function is shown graphically in figure 3.8, with its equation and derivative given in equations 3.21 and 3.22 respectively. The bipolar sigmoid function is shown in figure 3.9, with its equation and derivative given in equations 3.23 and 3.24 respectively.

Figure 3.7 A backpropagation network with one hidden layer following Fausett (1994)

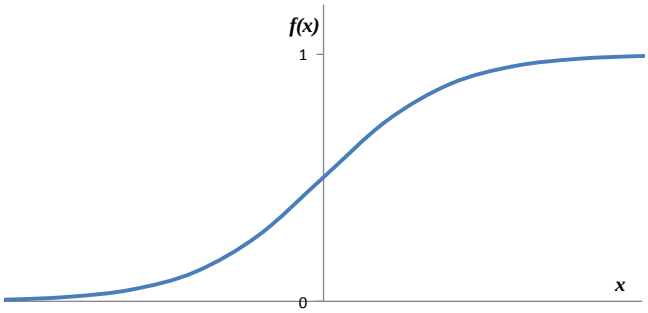


Figure 3.8 The binary sigmoid function

The equation for the binary sigmoid (log-sigmoid) is given as

$$f(x) = \frac{1}{1+\exp(-x)} \tag{3.21}$$

and its derivative is

$$f'(x) = f(x)[1-f(x)] \tag{3.22}$$

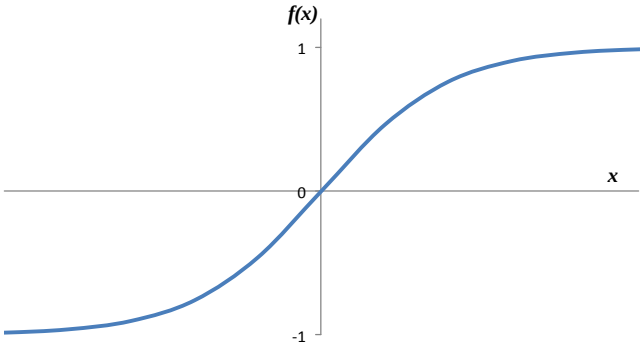


Figure 3.9 The bipolar sigmoid function

The equation for the bipolar sigmoid (tan-sigmoid) is given as

$$f(x) = \frac{2}{1+\exp(-x)} - 1 \tag{3.23}$$

and its derivative is

$$f'(x) = \frac{1}{2}[1+f(x)][1-f(x)] \tag{3.24}$$

Figure 3.7 shows the data transfer during the feedforward stage of a typical multilayer network. As can be seen, data are fed from the inputs toward the outputs in one direction only. In this case, each input unit, X_i, i=1 to n, receives an input signal and passes this value to all units in the next (hidden) layer. Each hidden unit, Z_j, j=1 to p, sums its inputs and computes its activation according to equations 3.25 and 3.26, then passes this value to all units in the output layer. In this example the subscript ij relates to the weighted connection **from** unit i **to** unit j.

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i.v_{ij} \tag{3.25}$$

$$z_j = f(z\_in_j) \tag{3.26}$$

Each output unit Y_k, k=1 to m, again sums its inputs and computes its activation following equations 3.27 and 3.28. The outputs y_k, k=1 to m, make up the output from the network for the given input pattern.

$$y\_in_k = w_{0k} + \sum_{j=1}^{p} z_j.w_{jk} \tag{3.27}$$

(3.28)

Once the output from each unit is determined, the error information term, δ_k, k=1 to m, for each of the output units can be expressed as

$$\delta_k = (t_k - y_k).f'(y\_in_k) \tag{3.29}$$

From this error information term, the weight correction term Δw_{jk}, j=1 to p, for each of p weights acting on output unit Y_k, can be calculated as

$$\Delta w_{jk} = \alpha.\delta_k.z_j \tag{3.30}$$

and the bias correction term as

$$\Delta w_{0k} = \alpha.\delta_k \tag{3.31}$$

The weights are not immediately updated, as the error term needs to be backpropagated to the units in the hidden layer before weight adjustment takes place. This is done in a similar manner to the feeding forward of data, albeit in the opposite direction, and is achieved thus (for hidden unit Z_j, j=1 to p). Firstly, each hidden unit, Z_j, sums its delta inputs from the units in the output layer:

$$\delta\_in_j = \sum_{k=1}^{m} \delta_k.w_{jk} \tag{3.32}$$

then multiplies this value by the derivative of its activation function to calculate its error information term.

$$\delta_j = \delta\_in_j.f'(z\_in_j) \tag{3.33}$$

The weight correction term, Δv_{ij}, i=1 to n, can then be calculated as

$$\Delta v_{ij} = \alpha.\delta_j.x_i \tag{3.34}$$

and the bias correction term, Δv_{0j}, as

$$\Delta v_{0j} = \alpha.\delta_j \tag{3.35}$$

The weights and biases can then all be updated simultaneously, and the next pattern presented to the network. For added clarity, this process is shown as a flowchart in figure 3.10. In some cases it is more convenient to update the weights and biases after several training patterns have been presented, or after each epoch. This is known as *batch updating*, and can be employed to smooth out the weight updating to avoid large changes to weights. In the

case of batch updating, each weight correction term is the average of the accumulated weight correction terms in the particular batch.

Figure 3.10 Training algorithm for feedforward, backpropagation network with three layers following Fausett (1994)

3.2.2.4 Momentum in backpropagation

It can be shown that the backpropagation rule makes use of the error gradient (Fausett 1994), as described earlier regarding the delta rule. In the case of a multilayer, backpropagation network, the weights will be adjusted (assuming other stopping criteria are not met) until the error reaches a minimum value. However, there is a possibility that this value will be a local minimum rather than a global minimum, and thus a more appropriate combination of weights could exist for the given input and target vector pairs. Adding a momentum term to the weight updating algorithm means that the algorithm adjusts weights not in the direction of the gradient to most effectively reduce the error, but in the direction of a combination of the current gradient and the gradient from previous iterations. The momentum term is denoted μ, and must lie in the range [0, 1]. The relationship for updating weights with momentum is given in equation 3.36, for training step *t+1*. The equation shown would adjust the weights acting on the output layer (following the previous nomenclature), but similar equations can be expressed for the weights on the hidden layer. The weight updates at steps *t* and *t-1* must be stored during training.

$$w_{jk}(t+1) = w_{jk}(t) + \alpha.\delta_k.z_j + \mu[w_{jk}(t) - w_{jk}(t-1)] \tag{3.36}$$

which can be expressed as

$$\Delta w_{jk}(t+1) = \alpha.\delta_k.z_j + \mu.\Delta w_{jk}(t) \tag{3.37}$$

3.2.2.5 Training using the scaled conjugate gradient method

The standard backpropagation algorithm, even with a momentum term included, has several disadvantages. Firstly, although the weights are adjusted in the negative direction of the

steepest gradient at each iteration, this does not necessarily produce the fastest convergence for a given data set. Secondly, the parameters α and μ must be specified by the user, and generally can only be found by trial and error (Moller 1993). Inappropriate selection of these values can lead to poor convergence, and optimising these values for a particular network can be very time consuming. Thirdly, the learning rate, α, is a fixed value throughout training, thus the length of the weight update (the distance moved along the gradient) is proportional to a single value of α. Conjugate gradient methods of training improve upon the standard backpropagation algorithm by conducting the line search (the direction in which weights should be modified) along conjugate directions rather than the negative gradient direction. This has been shown (Hestenes and Stiefel 1952; Demuth and Beale 1998) to generally produce much faster convergence for a given data set. The step size, or distance moved along the conjugate direction, can be adjusted at each iteration, meaning that a suitable step size can be chosen to minimise the error function along the line that is being searched.

Various conjugate gradient methods exist to train ANNs, but for speed of convergence the scaled conjugate gradient method is attractive. The scaled conjugate gradient (SCG) method was presented by Moller (1993) and makes use of a scalar to approximate the second derivative of the error function, thus removing the need for a line search. Although the SCG method has been shown to require more iterations to converge when compared to other conjugate gradient methods, the computing power in each step is significantly reduced due to the absence of the line search, thus convergence can be achieved in a shorter time. Moller (1993) reports speed improvements of around 30 times when the SCG method is compared to the standard backpropagation algorithm.

3.3 Ultrasonic Inspection

Ultrasonic inspection is widely used in NDE, as detailed in chapter 1. This section covers the key concepts in ultrasonic inspection along with the modelling of ultrasonic wave motion using the finite element method. The theory of wave motion applies to impact-based inspection methods as well as ultrasonics, although with impact methods it is not possible to control the frequency at which the wave is oscillating. The implications of this are discussed in section 3.3.1.4.

3.3.1 Elastic wave motion

When an object is placed under a condition of stress, the response of the object is not to instantaneously distribute this stress equally; the stress will propagate through the object from the point of application. If the response of the object over a very short time is to be investigated, the precise motion of the stress wave from the point of application must be defined.

3.3.1.1 Bulk waves

In any unbounded fluid (i.e. a medium that cannot sustain shear), the only type of wave that can propagate through the fluid is a bulk wave, which travels at uniform velocity (assuming conditions throughout the fluid are constant) in all directions from the point of application. Its velocity can be expressed as the speed of sound in the medium, and is given by

$$c = \sqrt{\frac{k}{\rho}} \tag{3.38}$$

where c is the velocity of the wave, k is the bulk modulus of the fluid, and ρ is the density of the fluid.

In the case of a solid medium (i.e. a medium that can sustain shear), in addition to this primary bulk wave is a shear wave whose direction of propagation is perpendicular to that of the primary wave. The primary and secondary waves are generally denoted P-wave and S-wave respectively. The velocity of the P-wave in an extended isotropic solid is given as

$$c_1 = \sqrt{\frac{k + \frac{3}{4}\mu}{\rho}} \tag{3.39}$$

where c_1 is the velocity of the P-wave and μ is the rigidity modulus. P-waves are sometimes referred to as dilatational waves, as they act to dilate the medium as they travel through it.

The velocity of the S-wave, c_2, can be expressed as

$$c_2 = \sqrt{\frac{\mu}{\rho}} \tag{3.40}$$

using the same nomenclature as above. S-waves are sometimes referred to as distortional waves, as they act to distort the medium through which they are travelling. For the purpose of analysis, it can assumed that P-waves are irrotational (thus the dilatation is in a predetermined plane) and that S-waves are equivoluminal (i.e. there is no dilatation) (Kolsky 1963).

3.3.1.2 Wave interaction at a boundary

When a wave reaches a boundary, such as an edge or a flaw in a solid, its behaviour is modified. When a plane wave reflects from a normal boundary, the stress reflected and transmitted can be calculated using the reflection and transmission coefficients. In figure 3.11, a plane wave, σ_1, is shown travelling in the x-direction through medium 1 and reaching the boundary that joins medium 1 to medium 2.

The reflection coefficient can be expressed as the ratio of the stress reflected to the incident stress, and is given as

$$R_{12} = \frac{\sigma_R}{\sigma_I} \tag{3.41}$$

where R_{12} is the reflection coefficient between medium 1 and medium 2, σ_R is the reflected stress and σ_1 the incident stress.

Figure 3.11 Wave reflection following Rose (1999)

The reflection coefficient, R_{12}, can be calculated using the acoustic impedance of both media. Rose (1999) states that the acoustic impedance of a material can be considered as a characteristic material property, and gives acoustic impedance, W, as

$$W = \rho.c \approx \sqrt{\rho.E} \tag{3.42}$$

where ρ is the material's density, c is the longitudinal wave velocity in the material and E is the Young's modulus. Thus, the reflection coefficient can be expressed as

$$R_{12} = \frac{W_2 - W_1}{W_1 + W_2} \tag{3.43}$$

where W_1 is the acoustic impedance of material 1 and W_2 is the acoustic impedance of material 2.

Similarly, the transmission coefficient, T_{12}, can be expressed as

$$T_{12} = \frac{2W_2}{W_1 + W_2} \tag{3.44}$$

using the same nomenclature.

In practice, the wavefront does not necessarily act normal to an interface. If the wave reaches an interface at an angle, there can be a reflection, a refraction and a mode conversion (depending on the angle of incidence). For an angle of incidence θ_1, Snell's law states that the angle of refraction can be determined, as long as the longitudinal wave velocities are known in both media. Snell's law is given as

$$c_1 \sin\theta_2 = c_2 \sin\theta_1 \tag{3.45}$$

where θ_1 is the incident angle, θ_2 is the refracted angle, c_1 is the longitudinal wave velocity in material 1, and c_2 is the longitudinal velocity in material 2. Note that c_1 and c_2 in this nomenclature both refer to longitudinal waves in materials 1 and 2 respectively, rather than the P-wave and S-wave in a single medium as in equations 3.39 and 3.40. Figure 3.12 shows Snell's law applied to a simple refraction situation at a boundary between two materials. For clarity, only the longitudinal wave is shown, although as Snell's law is derived from analysis of wave speed, the angle of a reflected and refracted shear wave can also be calculated. In the example shown, the speed of sound in material 2 is greater than in material 1, that is, $c_2 > c_1$. It can easily be seen that for cases when the speed of sound is lower in material 2 than material 1, the angle θ_2 will be less than θ_1.

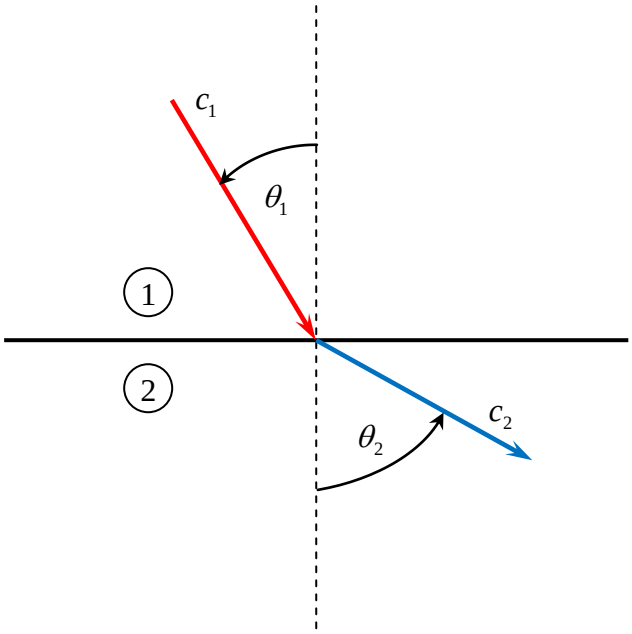


Figure 3.12 Graphical illustration of Snell's law showing refracted longitudinal wave

As stated above, at a boundary it is possible that mode conversion (longitudinal to shear and vice versa) will take place, thus several reflected and refracted waves will be present. Figure 3.13 shows all of the possible mode conversions at a boundary between two materials, where a longitudinal wave is reflected, refracted and mode converted. The reflected angle of the longitudinal wave will be the same as the incident angle, as its velocity in material 1 must be the same, regardless of its direction of propagation. The relationship between the incident angle and the angle of the reflected shear wave and the refracted longitudinal and shear waves can be expressed as follows:

$$c_{1L} \sin\theta_{2L} = c_{2L} \sin\theta_{1L} \tag{3.46}$$

$$c_{1L} \sin\theta_{2S} = c_{2S} \sin\theta_{1L} \tag{3.47}$$

$$c_{1L} \sin\theta_{1S} = c_{1S} \sin\theta_{1L} \tag{3.48}$$

where c_{1L} is the longitudinal wave velocity in material 1, c_{2L} is the longitudinal wave velocity in material 2, c_{1S} is the shear wave velocity in material 1, and c_{2S} is the shear wave velocity in

material 2. The angles, θ_{1L}, θ_{2L}, etc. relate to the angles these waves make with the vertical using the same subscript nomenclature.

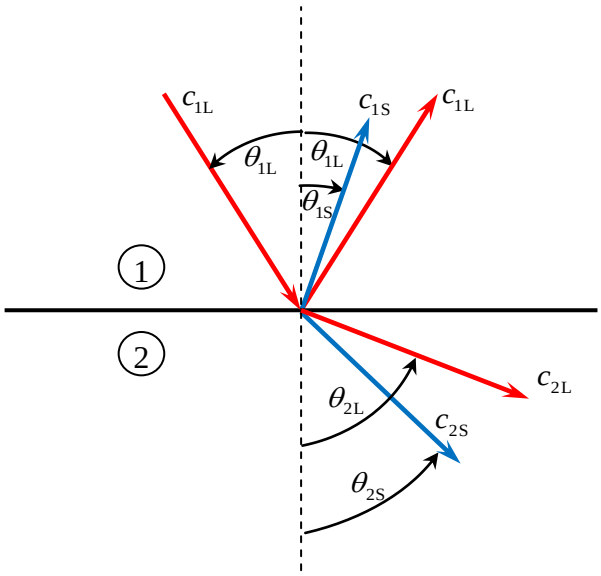


Figure 3.13 Mode conversion at a boundary between two materials

It is clear from Snell's law and from figure 3.13 that an angle of incidence could exist that would make the refracted longitudinal wave's angle or the refracted shear wave's angle equal to 90°. These angles are known as *critical angles*. The first critical angle, θ_{cr_1}, is defined as

$$\theta_{cr_1} = \sin^{-1}\left(\frac{c_{1L}}{c_{2L}}\right) \text{ when } \theta_{2L} = 90° \tag{3.49}$$

and is the incident angle that produces only a refracted shear wave in material 2. All of the longitudinal energy is either reflected (as c_{1L}, figure 3.14b) or converted into an interface wave that travels along the interface between the two materials.

The second critical angle, θ_{cr_2}, is defined as

$$\theta_{cr_2} = \sin^{-1}\left(\frac{c_{1L}}{c_{2S}}\right) \text{ when } \theta_{2S} = 90° \tag{3.50}$$

and produces no longitudinal or shear wave in material 2. These critical angles are illustrated in figure 3.14.

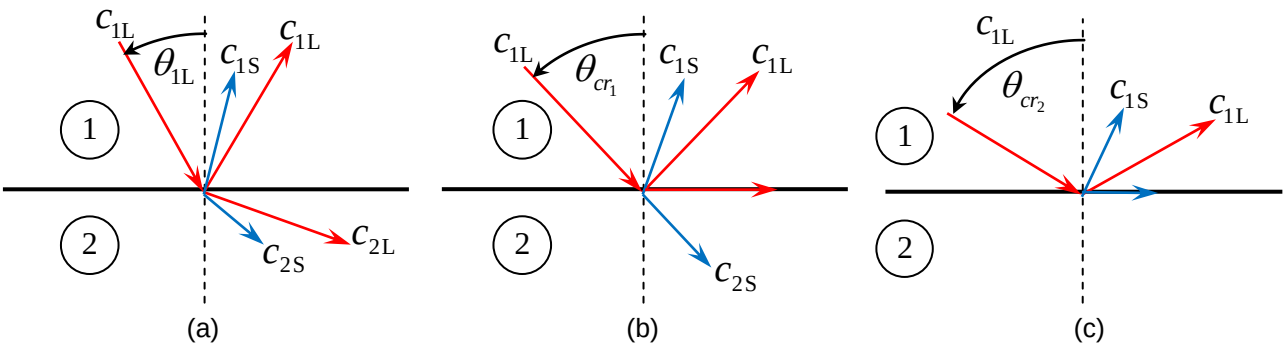


Figure 3.14 Mode conversions showing critical angles: general (a), first critical angle (b) and second critical angle (c)

3.3.1.3 Lamb Waves

In the case of a bounded medium, such as a plate, the boundaries of the medium act as channels to guide the motion of the wave(s) present. These guided waves in plates are often referred to as Lamb waves. The velocity of Lamb waves varies depending on several parameters, but particularly the frequency of the vibration that generates them. Guided waves are governed by the same equations as bulk waves (see section 3.3.1.1) but must also have boundary conditions applied due to the presence of edges in the medium through which they are travelling. As waves contact boundaries, there will be some reflection and refraction of these waves, which leads to mode conversion between longitudinal and shear waves.

In a medium that is unbounded except for an interface between two materials, Snell's law can be applied to analyse the transmission, reflection and refraction at the interface, as only the longitudinal and shear modes can be present. In the case of guided waves in a bounded

medium, and infinite number of modes can be present, thus mode conversion can be difficult to predict analytically (Rose 1999). As mode conversion takes place, the different modes can propagate at different velocities. Lord Rayleigh (1945) observed that in a group of waves of similar frequency, the velocity of the individual waves in the group can be greater than the overall group velocity. The individual waves seemed to appear at the rear of the group, propagate forward and then disappear as they reached the front of the group. The velocity of these individual waves is known as *phase velocity*, whereas the velocity of the overall group is known as *group velocity*. If the phase velocity does not match the group velocity, the waves are said to be dispersive. The relationship between the two velocities can be expressed by looking at the phase velocity at various frequencies. The wavenumber, *k*, is given by

$$k = \frac{\omega}{c} \tag{3.51}$$

where ω is the angular frequency of the wave, and c is the wave velocity (phase velocity). If the angular frequency, ω, is plotted as a function of wavenumber, the group velocity can be expressed as the derivative of the function. Figure 3.15 shows how the group velocity changes with phase velocity, the velocities in this case being represented graphically by the angle, θ_g for group velocity and θ_p for phase velocity. The phase velocity can be expressed as

$$c_p = \frac{\omega}{k} \tag{3.52}$$

and the group velocity by

(3.53)

Figure 3.15 Group velocity variation with phase velocity following Rose (1999)

As Lamb waves can have multimode characteristics, each mode must be evaluated in terms of phase and group velocity to enable a complete dispersion curve to be produced. An example of such dispersion curves is shown in figure 3.16, where the various modes are shown as functions of the frequency.thickness product, *fd*. The velocity of each mode is dependent not only on the frequency of excitation, but also on the thickness of the plate through which the wave is travelling. The cutoff frequencies shown in figure 3.16 show that at a given value of *fd*, only certain modes may be present. Decreasing the *fd* value to below a particular mode's cutoff frequency will mean that the mode cannot be present in the system under investigation. The modes are annotated as S_n and A_n, $n \geq 0$, to represent the symmetric and antisymmetric modes respectively. The fundamental symmetric and antisymmetric modes are referred to as S_0 and A_0, with the first harmonic modes referred to as S_1 and A_1, the second as S_2 and A_2, and so on.

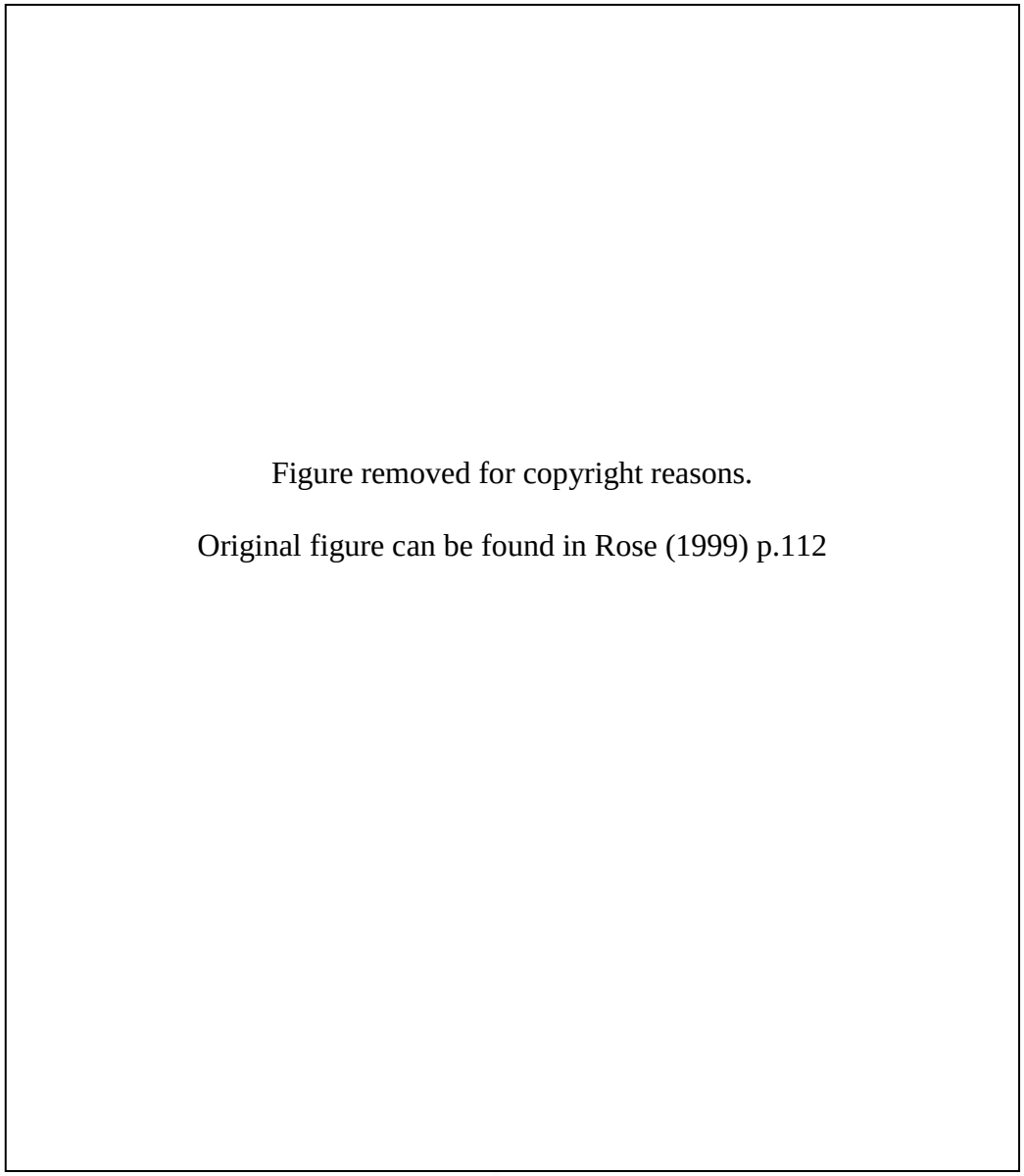


Figure 3.16 Dispersion curves for a traction-free aluminium plate (Rose 1999)

It can also be seen in figure 3.16, when $fd = 0$, that the initial phase velocity of the S_0 mode is equal to the classical 'long wavelength' value for wave velocity in a plate, which is given as

$$c_{plate} = \sqrt{\frac{E}{\rho(1-\nu^2)}} \tag{3.54}$$

where E is the Young's modulus, ρ is the density of the material and v is the Poisson's ratio. Additionally, it can be seen that as the fd product increases, the phase velocity of the S_0 and A_0 modes converges to the Rayleigh (surface) wave velocity in the material, c_R, whereas the other modes' phase velocities converge to the shear wave velocity, c_T. Note that c_T (using the nomenclature in figure 3.16) is equal to the S-wave velocity, c_2, from section 3.3.1.1.

It can be seen from figure 3.16 that at low values of fd, only the fundamental S_0 and A_0 modes are present, and the phase and group velocities are very close in value. This is beneficial in terms of ultrasonic inspection, as dispersion is minimised and the additional harmonic modes are not present to complicate the wave sent into a specimen under test.

3.3.1.4 Wave motion using impact methods

As mentioned in section 3.3, wave motion induced by a single impact (as opposed to some kind of oscillation) can be analysed using the same governing equations as waves generated at ultrasonic frequencies. However, when using impact methods in bounded media it should be noted that all frequencies will be excited in the specimen under test, thus every mode may be present, as indicated in figure 3.16. If the frequency of excitation is not controlled, a very large number of wave modes will be generated, propagating at different velocities. This can make analytical solutions very difficult, and can present problems experimentally when trying to track a particular wave's motion within a component. In practice, the fastest moving mode is generally monitored, and care is taken to take measurements before subsequent modes (or reflections of the fastest moving mode) arrive at the point of measurement (Ishak, Liu *et al*. 2002). In many cases using impact-based tests, the frequency *response* of the component under inspection is analysed, as exciting a component via impact

will allow it to vibrate at its natural frequency or at harmonics of this natural frequency. Inspection using natural frequency is discussed further in section 3.4.

3.3.2 Generating and receiving ultrasonic waves

Ultrasonic waves can be generated in a medium by causing the particles of the particular material to vibrate, and this vibration (if we consider only the P-wave for now, acting in a homogeneous and isotropic medium) will propagate at the speed of sound in the medium. Generally, the term *ultrasonic* refers to vibrations at any frequency above around 20kHz, the upper limit of the human hearing range (Halmshaw 1991). Considering the previous section regarding dispersion and wave modes, it is desirable to induce ultrasonic waves with as narrow a frequency band as possible in order to control the available modes and velocities; a single strike with a hammer, for example, will send a stress wave through the medium at the speed of sound, but every frequency will be excited, thus the wave's behaviour will be very difficult to accurately predict. In order to maintain control over the available frequencies present, ultrasonic pulses sent into materials tend to be of several cycles in duration. The higher the number of cycles (that is, the longer the pulse), the more precise the frequency of inspection (Halmshaw 1991). Of course, a larger number of cycles means that the wavefront will move further before the pulse finishes, thus a compromise is often required between the need to be able to inspect areas close to the point of wave application and the need to maintain accuracy in the frequency domain. Ultrasonic pulses in practice are generally sinusoidal functions, which are windowed so that the amplitude increases from the start to the centre of the pulse, then decreases towards the end of the pulse, as shown in figure 3.17. This windowed pulse, (sometimes referred to as a *tone burst*) is essentially several periods of a sinusoidal function multiplied by a bell-shaped curve function to alter the amplitude. A

typical bell-shaped curve is the Hann function, and the window is commonly referred to as a *Hanning window* (Lowe, Challis *et al*. 2000). Windowing a pulse in this way serves to reduce the spread of frequencies present (Ho 2005), thus helping to maintain predictable wave motion.

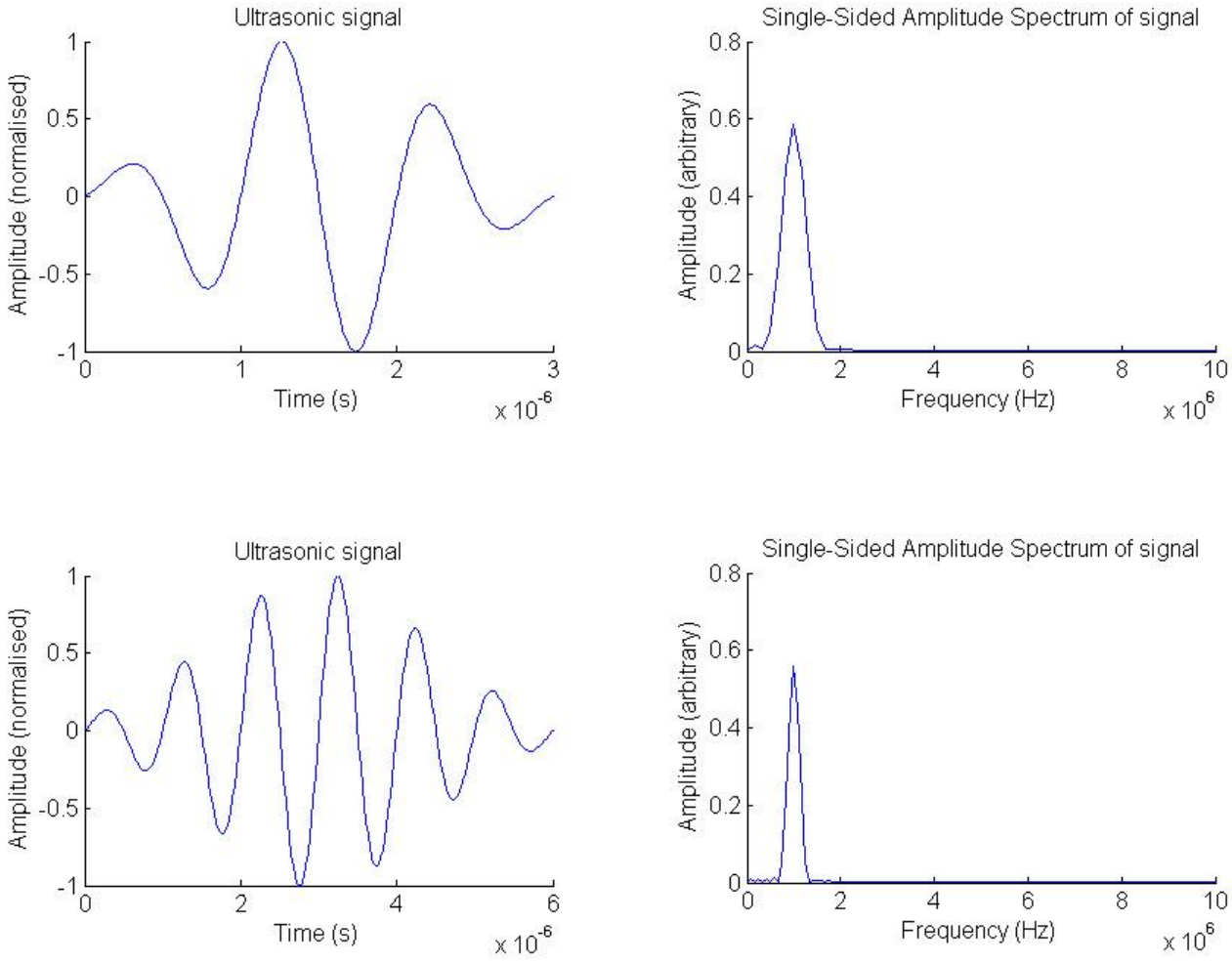


Figure 3.17 Windowed sinusoidal functions (left) and their corresponding Fast Fourier Transforms (right)

In order to insonify the component under inspection, that is, to induce the ultrasonic wave, the particles on the surface of the component must be caused to vibrate. This can be achieved

through a variety of means, such as using a piezoelectric transducer, an electromagnetic acoustic transducer (EMAT) or a laser. The most common method of inducing and receiving ultrasonic waves is the piezoelectric transducer (Halmshaw 1991). These transducers operate using the *piezoelectric effect* (for receivers), whereby the piezoelectric crystal in the transducer produces an electric potential when deformed either in compression or tension. The amount of electric potential and the mechanical pressure applied to the crystal are proportional. In a similar manner, the *inverse piezoelectric effect* is exactly the opposite: an applied electric potential causes the crystal to deform. If the electric potential is alternating, the crystal will vibrate, thus inducing a wave of a particular frequency. Typically, piezoelectric transducers are designed to operate at the natural frequency of the crystal, although many are able to operate at harmonic frequencies that are multiples of the natural frequency (Halmshaw 1991). Piezoelectric transducers can be very small in physical size, thus can achieve good spatial resolution or be used where access to the component is limited. However, they must be physically connected to the medium with a couplant (typically a type of grease or oil), or used in an immersion tank, where the ultrasonic wave is sent through water before reaching the component under inspection. In the case where a couplant is used, differences in coupling conditions between tests can lead to inconsistencies, although solutions to these issues have been proposed (Zgonc and Achenbach 1996). A typical piezoelectric transducer is shown in figure 3.18.

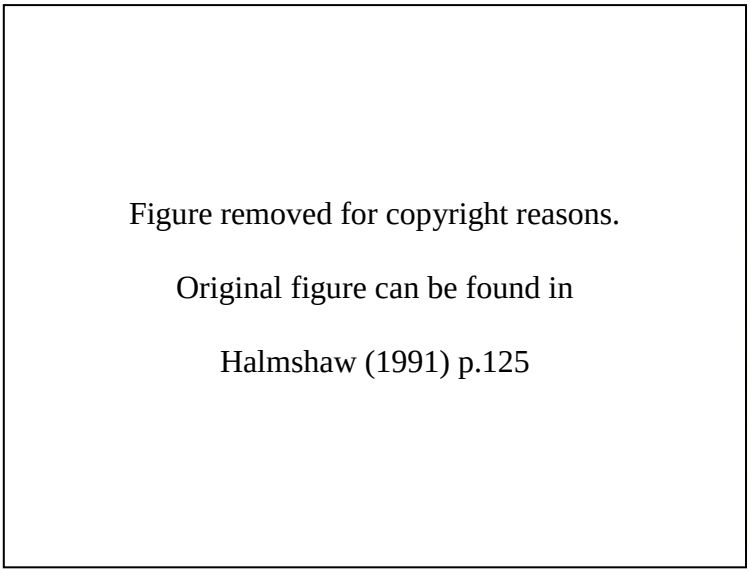


Figure 3.18 A compression wave piezoelectric transducer (Halmshaw 1991)

Piezoelectric transducers can be used to generate and receive ultrasonic waves, and often the same probe is used to transmit and receive waves (referred to as the pulse-echo method). This is achieved by first sending the ultrasonic pulse into the component under inspection (the pulse shape being as per figure 3.17), then switching the transducer to a receiving mode for a given length of time. During this receiving mode, the ultrasonic waves reflected from the boundaries of the component, as well as from any flaws, will cause the transducer to output an electric potential as they reach its surface. The transmitter pulse repetition rate can be adjusted depending on the particular application to ensure that reflected signals are not affected by new pulses being sent through the component.

EMATs are able to induce vibrations in a metal component without needing to be physically coupled, which makes them more suitable for use in situations involving high temperature, moving parts, or on components with rough surfaces (Halmshaw 1991). The EMAT

principle (illustrated in figure 3.19) involves a coil of wire carrying a current at radio (ultrasonic) frequency, and a magnet which provides a steady magnetic field. As the alternating current passes through the wire coil, eddy currents are induced in the surface of the component under inspection, and these currents interact with the magnetic field to produce Lorentz forces in the specimen's surface. As the current in the wire coil is alternating at a high frequency, the Lorentz forces also alternate at this frequency, thus generating an ultrasonic wave that travels through the component. For the example shown in figure 3.19, the magnetic field is horizontal, thus the Lorentz forces act in the vertical plane, producing a compressional wave. EMATs have the advantage of operating without the need for physical coupling, but have the disadvantage of being larger and much less efficient than piezoelectric transducers.

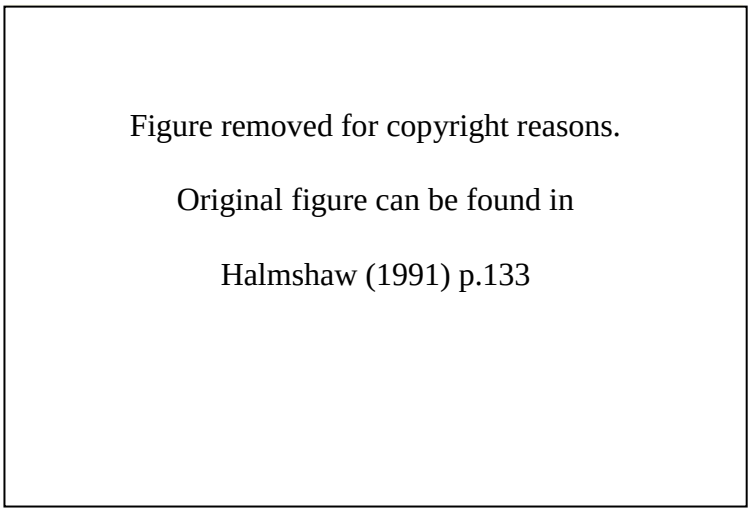


Figure 3.19 An EMAT for generating compressional waves (Halmshaw 1991)

EMATs can be operated in much the same way as piezoelectric transducers, in that they can either act as transmitters, receivers, or combined transmitters and receivers.

Laser induced ultrasound makes use of the thermoelastic effect. As a laser beam reaches the surface of the component under inspection, the electromagnetic radiation is partially absorbed

by the component's surface. This causes a local rise in temperature leading to thermal expansion of the medium at this localised point. The laser beam is typically varied in intensity at the desired ultrasonic frequency, thus the thermal expansion and contraction of the medium cause a wave to propagate through the medium at this frequency. The waves produced are P-waves (compressional waves) parallel to the component's surface. This method has advantages in that it is contactless, and very short pulse durations can be achieved, although the equipment is expensive and time consuming to set up.

3.3.3 Displaying results of ultrasonic inspection

As has been covered in section 3.3.2, various methods exist for sending and receiving ultrasonic waves. In all cases, when the waves reach the receiver, they need to be converted to an appropriate format to enable the operator to understand the outcome of the inspection. Typically, ultrasonic scan results are referred to as A-scan, B-scan, C-scan and D-scan. An A-scan is the fundamental output of an ultrasonic test, and is a two-dimensional display showing time on the x-axis and amplitude of signal on the y-axis. In the case of a single transducer acting as transmitter and receiver, the A-scan will show the initial pulse applied, and then any reflections from features of the component under inspection. A typical A-scan is shown in figure 3.20, where the time taken for the ultrasonic pulse to reach the transducer relates to the position of a flaw or feature, and the amplitude of the reflected pulse relates to the flaw or feature's size.

B-scan makes use of the A-scan data, but creates a two-dimensional representation of the component under inspection. Each A-scan can only take place at one transducer location, whereas if the transducer is moved along a straight line (i.e. in one dimension), a B-scan can

be created by using all of the A-scan data along the line of inspection. Thus, a B-scan is a 'side-on' view of the component, showing flaws and/or features as shown in figure 3.21.

C-scan involves moving the transducer over the entire plan view of the component, and again makes use of the A-scan data at each point to create a 'plan view' map of the component, showing areas where flaws and/or features exist. B and C-scans are often rendered in colour using specialist computer software to indicate the severity (size) of any features detected.

D-scan is similar to B-scan in its appearance, but is a scan taken in the transverse rather than the longitudinal plane at a particular longitudinal location.

Figure 3.22 shows how the various scans relate to each other in the case of a weld inspection.

Figure removed for copyright reasons.

Original figure can be found in

Hull and John (1989) p.68

Figure 3.20 A-scan ultrasonic method showing test (a) and display (b) (Hull and John 1989)

Figure removed for copyright reasons.

Original figure can be found in

Hull and John (1989) p.69

Figure 3.21 B-scan ultrasonic method (Hull and John 1989)

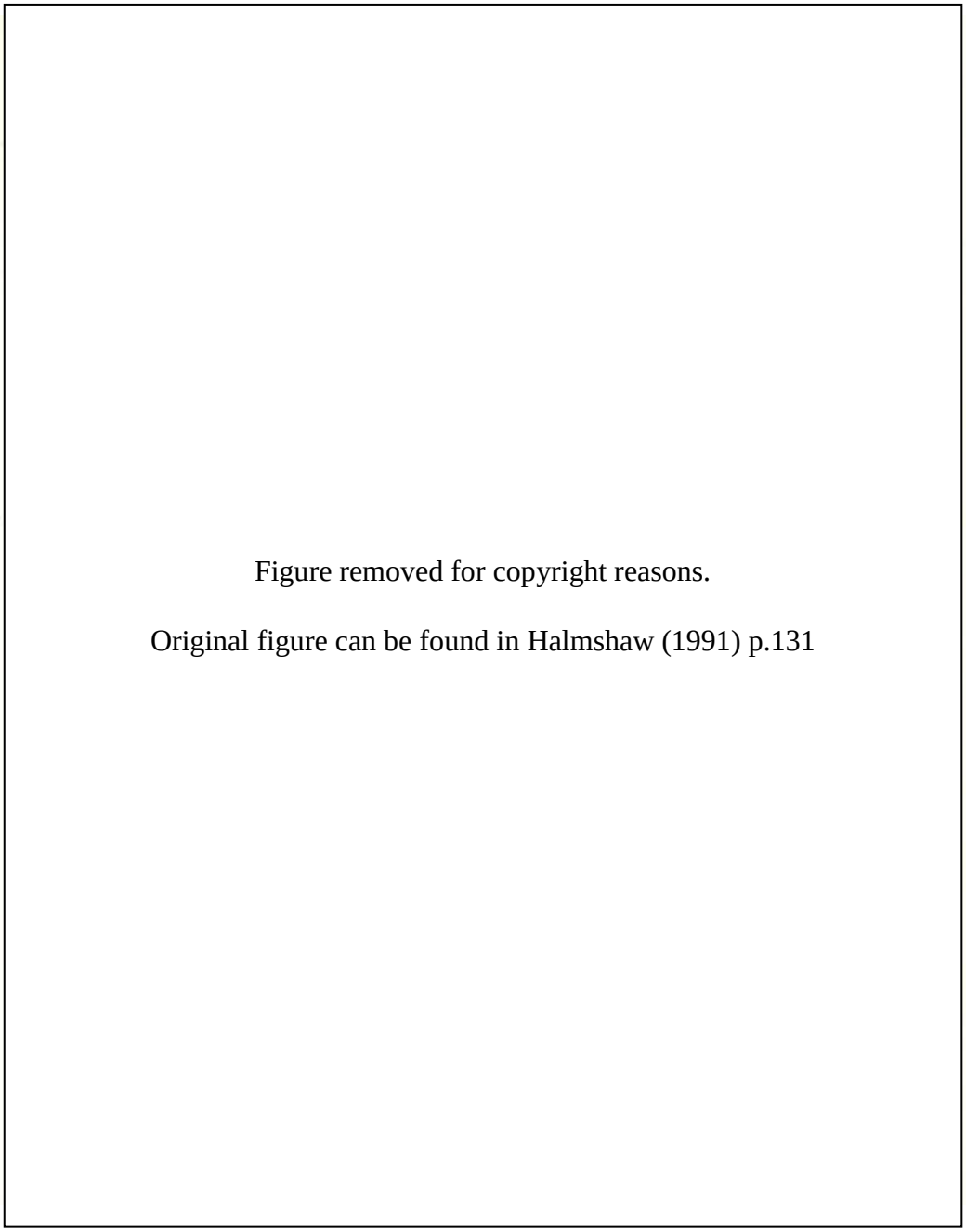


Figure 3.22 Generation of A-, B-, C- and D-scans for weld inspection (Halmshaw 1991)

3.3.4 Finite element modelling of ultrasonic inspection

In order to model an ultrasonic wave, consideration must be given to the type of solver used. The explicit solver is almost universally used for dynamic problems as, unlike the implicit solver, it will not try to arrive at a steady-state solution for each timestep of the simulation by inverting the stiffness matrix, thus computational requirements are reduced. Integration is generally performed using the explicit central-difference method. To find displacement u at time $t_{(i+1)}$, the equation is given as

$$\dot{u}_{(i+\frac{1}{2})} = \dot{u}_{(i-\frac{1}{2})} + \frac{\Delta t_{(i+1)} + \Delta t_{(i)}}{2} \ddot{u}_{(i)}, \tag{3.55}$$

$$u_{(i+1)} = u_{(i)} + \Delta t_{(i+1)} \dot{u}_{(i+\frac{1}{2})} \tag{3.56}$$

where u is the displacement, $\dot{u}$ is the velocity, $\ddot{u}$ is the acceleration, i relates to a particular time instance, $i+\frac{1}{2}$ relates to the midway point between i and i+1 and Δt is the timestep. Timesteps must be chosen to satisfy the Courant-Friedrichs-Lewy condition (Courant, Friedrichs *et al*. 1967) which states that the timestep must be smaller than the time taken for the wave to travel from one node to another. Thus, the maximum timestep can be expressed as

$$\Delta t_{\max} = \frac{x_{\min}}{c_1} \tag{3.57}$$

where Δt_{max} is the maximum timestep, x_{min} is the distance across the smallest element, and c_1 is the longitudinal wave velocity in the material, as given in equation 3.39. As can be inferred, this relationship means that as the element size decreases, the timesteps must decrease proportionally, thus there is a desire to optimise the mesh size to faithfully model

the wave's motion whilst minimising computational demand. Studies have shown that around 10 to 20 elements per longitudinal wavelength are required for good resolution; these studies are discussed in more depth in section 3.3.

3.4 Natural frequency extraction

Every object will have a particular natural frequency at which it will resonate. In addition to this natural frequency, harmonic frequencies (that is, frequencies above the natural frequency) can also cause the object to resonate in different modes. Damage to the object will cause the natural frequency to alter, thus defects can be detected by measuring the modal response of a component.

3.4.1 Determination of natural frequency

The natural frequency of a particular mode of a cantilever beam can be calculated as

$$f = \frac{k}{2\pi L^2}\sqrt{\frac{EI}{m}} \tag{3.58}$$

where f is the natural frequency of the beam (in cycles/second), k is a constant depending on the particular mode of vibration, L is the length of the beam, E is the Young's modulus, I is the area moment of inertia and m is the mass per unit length of the beam. As can be inferred from the equation, a flaw in the beam will mean there will be a region with a lower stiffness than the main beam, thus the overall stiffness will be lowered and the natural frequency will be reduced. In practice these changes in natural frequency can be very slight for small defects, thus to measure the frequencies at different modes experimentally would require very sensitive measuring apparatus.

3.4.2 Finite element modelling to determine natural frequency

In order to extract the natural frequencies (eigenfrequencies) from an undamped finite element model, the following equation is used:

$$\left(-\omega^2 M^{MN} + K^{MN}\right)\phi^N = 0 \tag{3.59}$$

where ω is the eigenfrequency, M is the mass matrix (which is symmetrical and positive definite), K is the stiffness matrix, φ^N is the eigenvector and ^{M} and ^{N} are degrees of freedom.

3.5 Summary of background theory

In this chapter, the underlying theory behind the techniques used in this thesis has been presented. Care has been taken to ensure an adequate description of the fundamentals of the operation of artificial neural networks whilst restricting the writing to the types of ANN used in the course of the research. Additionally, the theory behind the inspection of components using ultrasonic, impact and modal-based methods has been covered.

CHAPTER 4

SINGLE DEFECT IDENTIFICATION ANALYSES

4.1 Introduction

This chapter describes the methods used in this thesis to locate single dominant defects in steel bars, namely an impact method inducing a stress wave into the bar, a modal analysis that extracts eigenfrequencies and an ultrasonic pulse-echo method. All of these have been used before as non-destructive testing methods, where an operator would evaluate the data gathered, but in this thesis they are used as methods to train and assess the ability of ANNs to locate and quantify flaws in components. The use of ANNs was considered following Windsor, Anselme *et al.* (1993), who stated that the best future use of ANNs was in their potential to produce accurate results without the need for extensive feature extraction from test data. Therefore, an ANN approach to locating and quantifying defects was desirable to show not only that ANNs could be used as an alternative to simple wave mechanics in NDE but also to demonstrate their ability to produce accurate results relating to defects' presence, position, type and severity with minimal pre-processing of test data. In the context of this thesis, the term 'dominant' refers to a single detectable defect (i.e. above detectable size) in the component under inspection. As with all flaw detection methods, it is possible that flaws may exist below the detectable size, though these would be unlikely to be of concern to the inspector assuming the inspection method was correctly chosen. Identifying and locating a defect depends both on the defect's position and its size. If it is located close to an edge, or shielded by a geometric feature of the component, certain types of inspection would prove more suitable than others to reliably detect it. Similarly, if the defect is below a certain size it may not be detected at all. In this chapter, techniques using stress wave propagation and dynamic modal response of the component are described, and the results presented. The impact and modal analysis methods were considered as starting points for the investigation as feasible techniques to be used for NDE, and the analysis carried out shows the advantages

and limitations of these two methods. Additionally, the ultrasonic A-scan method, which is by far the most commonly used in practice of the three methods presented in this chapter, was considered.

4.2 Impact method

This method involves impacting a component at one end, and tracking the resultant stress wave through to the far end of the component (Hernandez-Gomez, Durodola *et al.* 2005). As the stress wave interacts with a defect, its behaviour changes, and by extracting features from the resultant wave it is possible to train ANNs to determine the (x,y) position of the defect.

4.2.1 Description of component and test

In this test, a steel bar was modelled in two dimensions using ABAQUS 6.5-1 finite element software, with values for Young's modulus, Poisson's ratio and density taken as 210GPa, 0.3 and 7800kg/m^3 respectively. The dynamic explicit method was used with the program's default timesteps. The material was assumed to be homogeneous and isotropic, and four-noded linear quadrilateral plane strain elements of the type CPE4R were used, following Thomas**,** Drinkwater *et al.* (2005). Cases were analysed using a plain bar and also a bar with a notch of varying depth near the top, as shown in figure 4.1. The notch was added to act as a stress concentrator near the top of the bar, to alter the stress wave's shape and introduce an antisymmetric component to the wave's propagation. The system's response to a more complex waveform passing over the region containing the defect was analysed in order to demonstrate that accurate results could be generated regardless of the type of waveform present.

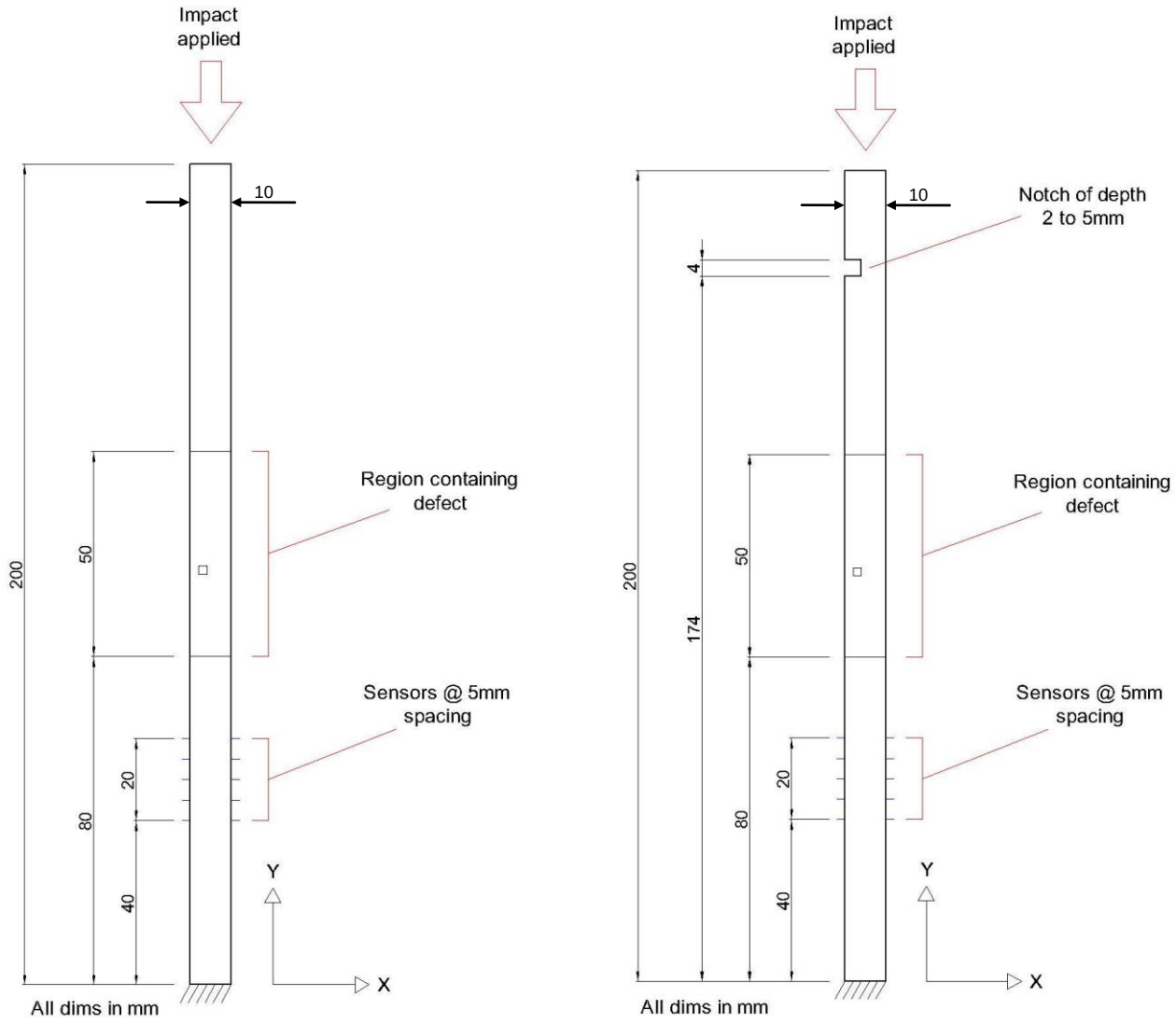


Figure 4.1 Steel bars (a) without notch, and (b) with notch

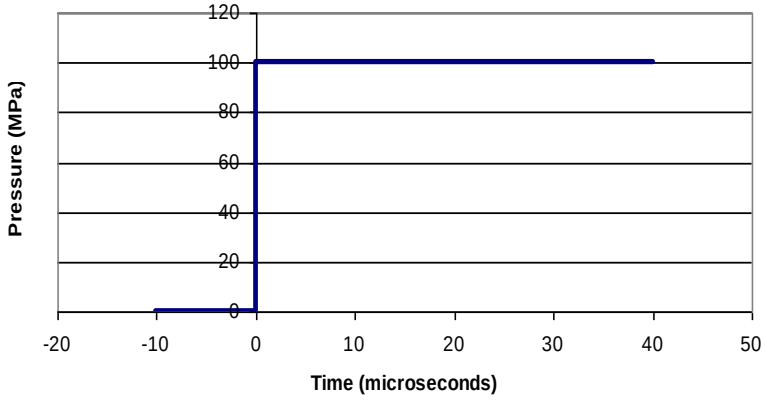


Figure 4.2 Instantaneous step load function

An impact was modelled by applying an instantaneous step pressure of 100MPa to the top of the bar, as shown in figure 4.2, and the base of the bar was constrained in all three degrees of freedom using the 'encastre' boundary condition. Elements of size 1mm x 1mm were used, giving 2000 elements in total. As the impact was applied, a stress wave propagated along the length of the bar and passed through the region where a single defect, modelled as a void, was located. The presence of the void modified the behaviour of the wave, and the resultant wave shape was captured by measuring the instantaneous horizontal displacements, U1, at ten sensors (nodes) near the base of the bar (five nodes on each side), at a time 40μs after the initial impact. The time of 40μs was chosen as this was the time required for the bulk wave to propagate through the entire length of the bar. Parametric studies were conducted with the void's position varied within the region indicated in figure 4.1, so that a data set of 500 cases was obtained. The study was conducted initially with voids of 1mm x 1mm, but later expanded to investigate the effect of smaller (0.5mm x 0.5mm) and larger (2mm x 2mm) voids. A typical 'raw' displacement signal from the simulation of a bar with no defect is shown in figure 4.3. The range in which measurements were taken is between 0.04m and 0.06m in the vertical direction, to correspond with the region containing the sensors as shown

in figure 4.1. The displacement on each side of the bar is shown at time t=40μs, where the dilatation of the bar can clearly be seen. From this raw signal, the displacement at 5 sensor points (at 5mm intervals as shown in figure 4.1) was used to represent each case to the ANNs.

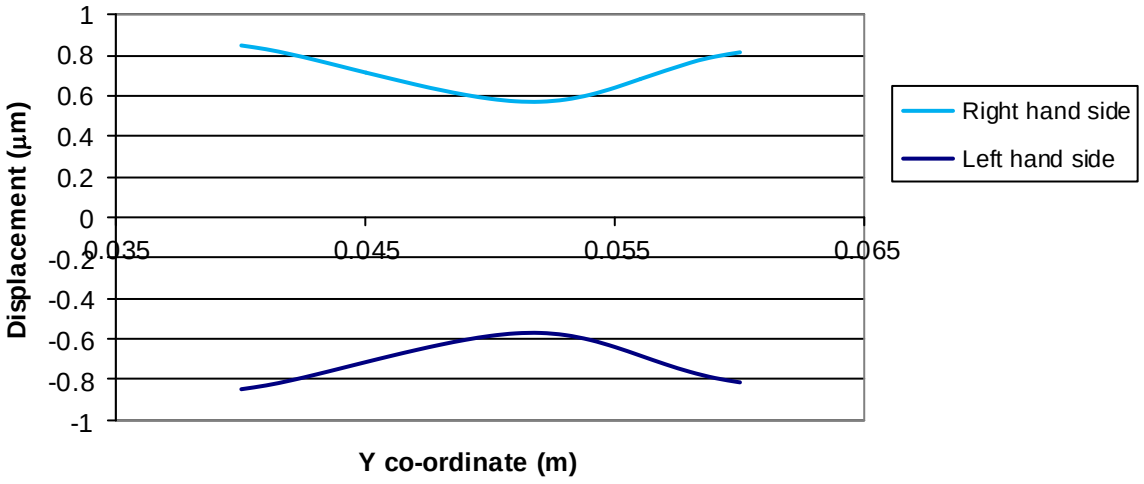


Figure 4.3 Horizontal displacement in sensor region at t=40μs (plain bar with no defect)

A mesh convergence study was carried out, using square elements of 2mm, 1mm and 0.5mm. The results from 5 sensor points, representing the sensor positions in figure 4.1, are shown in figure 4.4, where it can be seen that there was little increase in the resolution when the element size was reduced further than 1mm. Thus, a 1mm mesh was considered sufficiently accurate for this study.

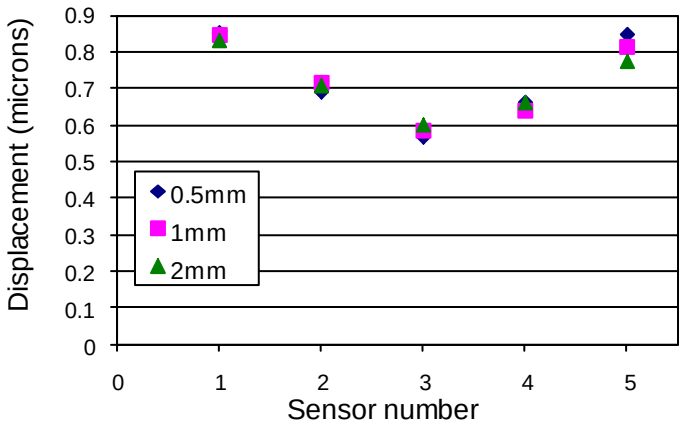


Figure 4.4 Displacement at 5 sensor points as mesh size is reduced

The two sets of data collected (horizontal displacements at ten nodes and the (x,y) co-ordinate of the centre of the void for each case) were both normalised to give each data set a range of [-1, 1] before being used to train and subsequently assess the ANNs. It is worth noting that the method of measuring displacement at just two sensors (one on each side of the bar) over a time window of 40μs was also investigated. However, the method of using multiple sensor points at a single time instant was found to give more meaningful data to use as inputs to the ANNs.

4.2.2 Description of ANNs

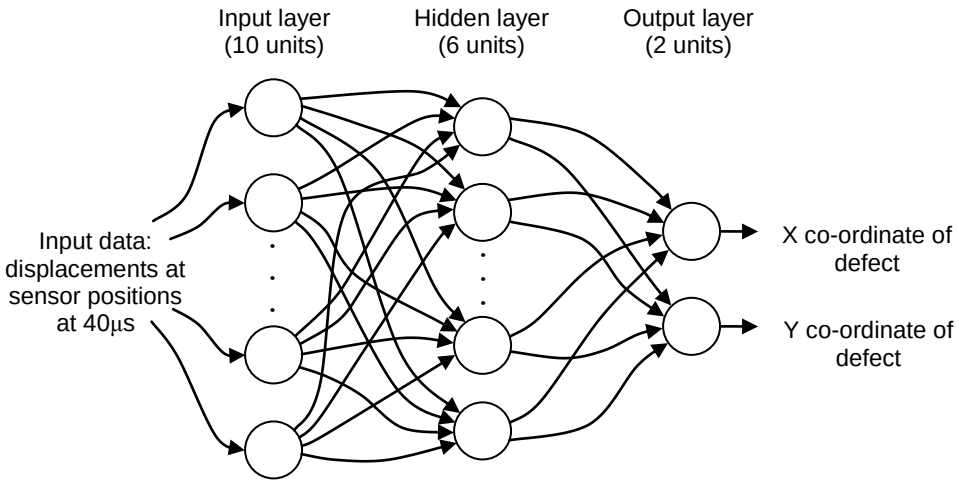
In this study, three different configurations of ANN were used as function approximators to provide a numerical output for the position of the void when presented with the data from the various sensors. All ANNs used were of the feedforward, backpropagation type, which have been shown to be well suited to heteroassociative function approximation. Training took place using the scaled conjugate gradient (SCG) method for 20,000 epochs using the neural

network toolbox in MATLAB 7.1. The target error value and minimum gradient were both specified as zero, thus the training would take place for the full number of epochs regardless of performance. Training and assessment of the ANNs were carried out using data from between 2 and 10 sensors (between 1 and 5 sensors on each side of the bar). Of the available data, up to 90% were used to train the ANNs, and the remainder used to assess the ANNs. The data were arranged in random order before the training/assessment split was made, in order to provide the ANNs with a training data set that adequately represented the full range of values to be expected. Three different configurations of ANN were used in this study, all with one hidden layer and two output neurons. The use of a single hidden layer was considered sufficient as a three-layer ANN is capable of approximating any nonlinear function (Kolmogorov 1957). Having two output neurons was a non-negotiable constraint, as it was necessary for all the ANNs in this study to output a separate value for the x and the y co-ordinate. All ANNs in this study used tan-sigmoid activation functions in the first two layers, and linear functions in the output layer. Whilst the tan-sigmoid functions returned outputs in the range [-1, 1], the linear functions used in the output layer meant that erroneous outputs from the ANN (in both training and assessment modes) could be easily identified as there was no upper or lower limit to the output value. The three different configurations of ANN are listed in table 4.1, and shown graphically in figure 4.5. It should be noted that when using the neural network toolbox in MATLAB 7.1, all input data are sent to each neuron in the input layer, thus the number of neurons in the input layer does not have to match the number of input data values.

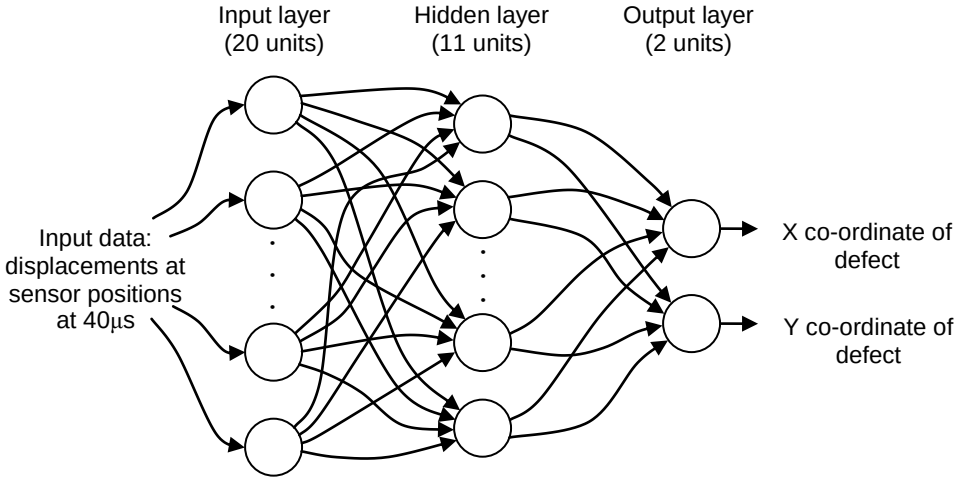
Configuration	No. of neurons in input layer	No. of neurons in hidden layer	No. of neurons in output layer
A	10	6	2
B	20	11	2
C	20	40	2

Table 4.1 ANN configurations

Network A: 10-6-2



Network B: 20-11-2



Network C: 20-40-2

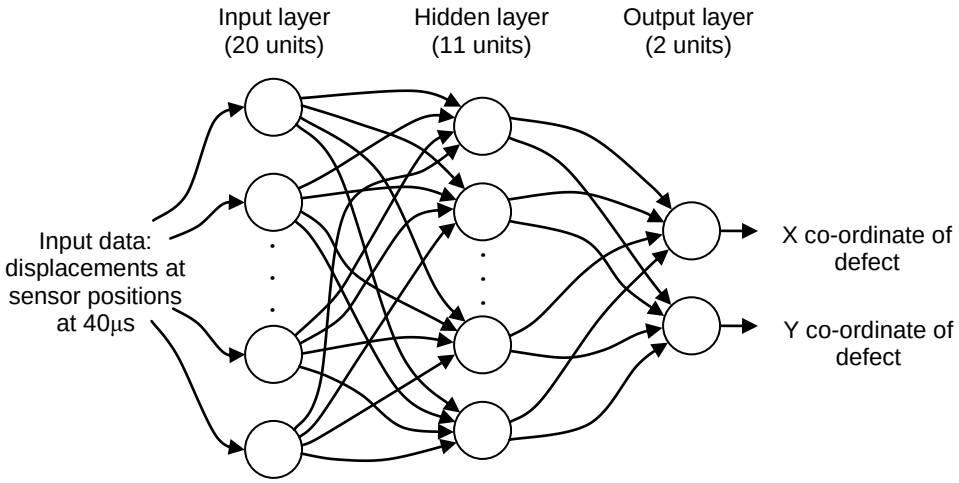


Figure 4.5 ANN configurations

Initially ten neurons were chosen for the input layer, to correspond to the ten inputs. The output layer had to contain 2 neurons, and the literature suggests that a good starting point when designing a three-layer ANN is to use the sum of the input and output layers divided by 2 to get the number of neurons in the hidden layer (Bishop 1995). This rule was followed again for configuration B, where twice as many neurons were used in the input layer. The value of 20 neurons was chosen to demonstrate whether doubling the number of neurons in the input layer had a significant effect on the quality of the output data. In this case, the same rule was applied to choose the number of neurons in the hidden layer. In configuration C, the number of neurons in the hidden layer was increased greatly to 40, in order to demonstrate whether a very large hidden layer had a significant effect on the quality of the output data. For this part of the investigation, three ANN layouts were considered sufficient, as with the configurations chosen these would give a good indication of whether greater accuracy in prediction could be achieved by altering the configuration at a later stage. The output (displacement) data from the FE simulation typically lay in the region of ~1μm, meaning the magnitude of difference between values was very small if a range of [-1, 1] was used. The data were therefore normalised using the equation

$$V_{norm} = \frac{V}{V_{\max}} \tag{4.1}$$

where V_{norm} is the normalised value, V is the value before normalisation and V_{max} is the maximum absolute value of the data set. By normalising in this way, all the data were made to lie in the range [-1, 1] using the full range up to a maximum value of 1. It is important to note at this stage that the normalisation process took place across the entire displacement data set, to include both the training and the assessment data. Similarly, the x and y co-ordinate (target) data set was normalised in the same manner. Although normalisation does nothing to

alter the ratio of one value to another, it enables the ANNs to make more meaningful connections between layers as the values presented will occupy the full range of [-1, 1]. With the tan-sigmoid function, input values lying outside the range [-1, 1] or values very close to zero tend to produce outputs of 1 and 0 respectively; by normalising the data the full range within the tan-sigmoid function can therefore be used.

As outlined in section 3.2, the training process involved presenting a percentage of the sensor data to the ANNs along with the corresponding target data (co-ordinates of the defect), and the assessment of the ANNs involved presenting the ANNs with the sensor data alone. The output of the ANNs to this previously unseen input data was then compared to the actual targets, and the deviation from the target measured using the following formula:

$$Mean\ deviation = \frac{\sum_1^n |V_P - V_T|}{n} \tag{4.2}$$

where n is the number of cases, V_P is the predicted value (ANN output) and V_T is the target value (co-ordinate of defect). Using this mean value allowed all cases to be objectively compared by providing a numerical method of assessing the overall quality of the output data. The inputs to the ANNs were the normalised nodal displacements from the finite element model. In order to determine how much information was required to enable the ANN to 'learn' the relationship between input and output data, between 2 and 10 sensors were used as ANN inputs. Of the available data, either 80% or 90% were used to train the ANNs, and the remaining 10% or 20% used to assess the ANNs' performance. The possible permutations are shown in table 4.2. It can be seen that there are 120 possible combinations, and all were investigated.

Variable	Possible values
Number of sensors used	2, 4, 6, 8, 10
Amount of training data (%)	80, 90
Notch depth (mm)	0, 2, 4, 5
ANN configuration	A, B, C

Table 4.2 Possible permutations of case

4.2.3 Results using impact method

This section presents the results obtained from the impact method study, and shows a selection of results that demonstrate the observed trends. The various values were selected in order to demonstrate the effect of the amount of information recorded from the non-destructive test and also the effect of the amount of data used to train the ANNs.

4.2.3.1 The effect of number of sensors

As can be seen from the graphs in figures 4.6 and 4.7, the accuracy of the ANNs' predictions increased as the number of sensor points increased. In this case accuracy was measured as the mean absolute deviation from the target of all data presented to the ANNs for assessment, as shown in equation 4.2. This increase in accuracy along with the number of sensor points used was to be expected; since the sensor readings were taken at one particular time instant, the modified stress wave's shape would be described more accurately as the number of points increased. Figure 4.6 shows the mean deviation from the target as the number of sensors was

increased for the case of a plain bar using network configuration B with 90% training data. Two sets of data were generated, to represent the x and y co-ordinates separately, and the mean deviation in this case was taken for the normalised values rather than the absolute values. The red lines in figure 4.7 represent a theoretical perfect match between ANN outputs and target data; if all data lay on the red line, the match between ANN output and target output would be 100%.

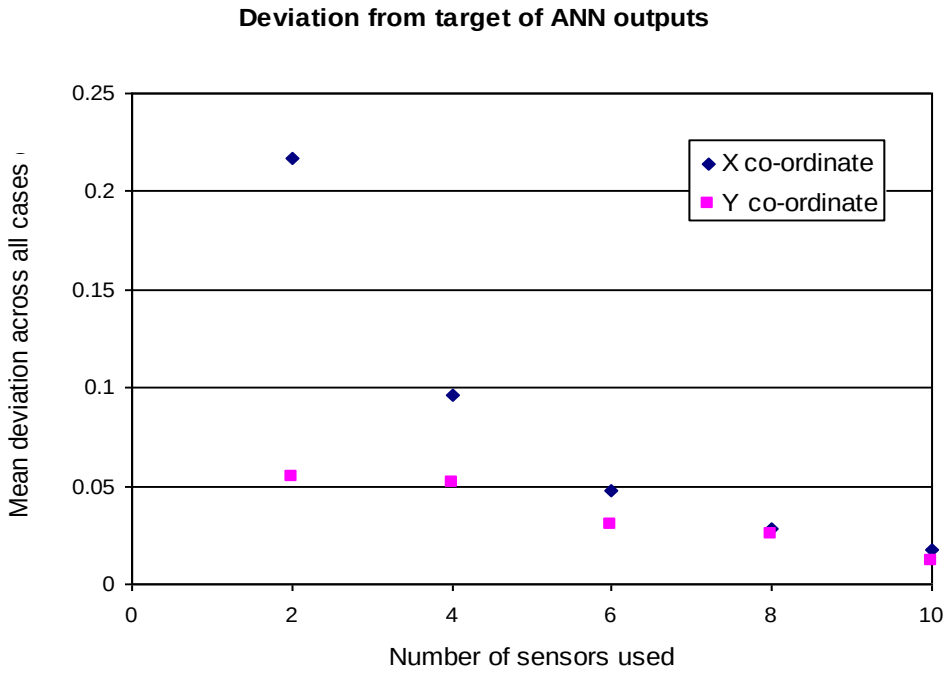


Figure 4.6 ANN performance with number of sensors

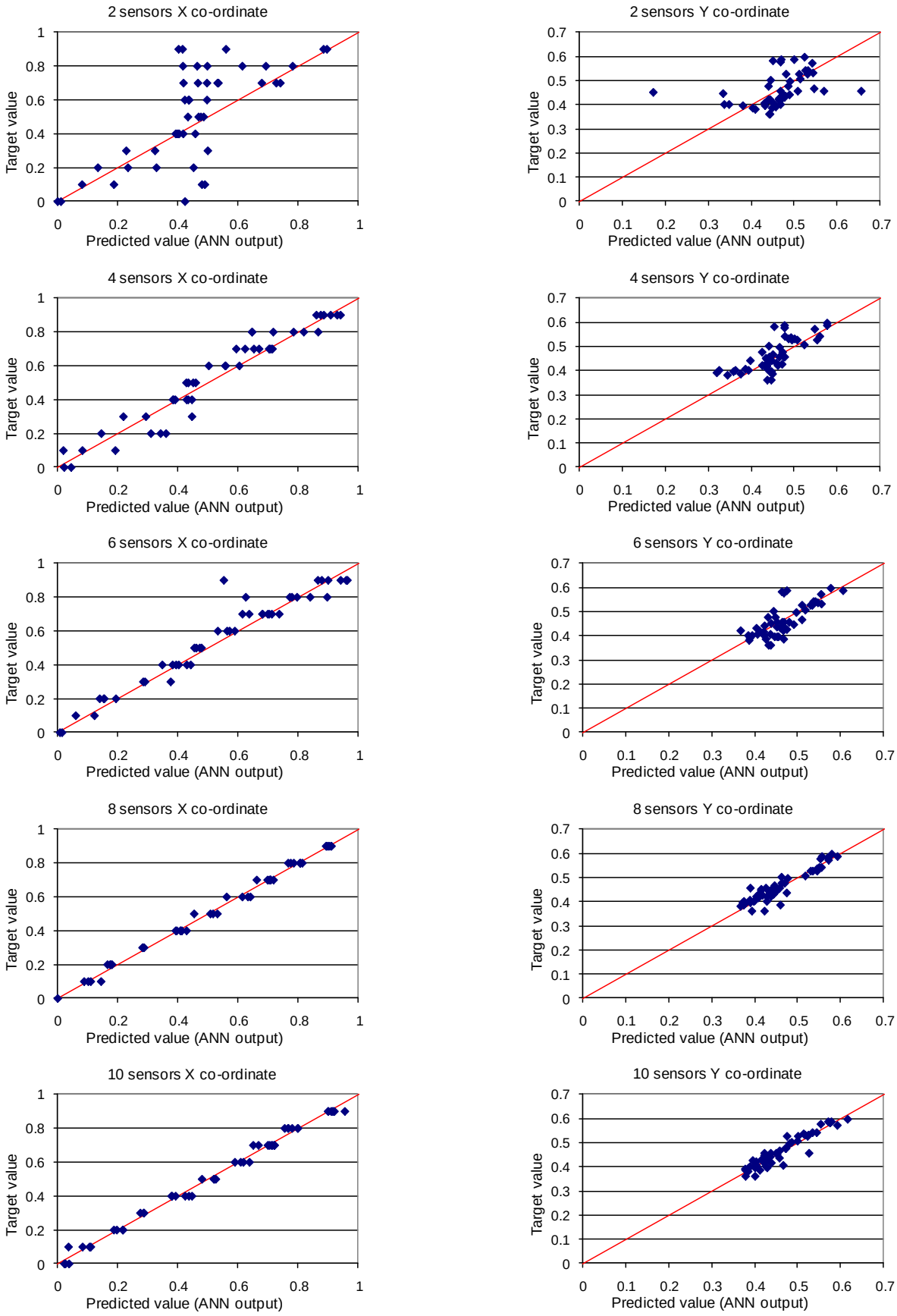


Figure 4.7 ANN outputs with number of sensors – no notch, 90% training data, network B

4.2.3.2 The effect of the amount of training data

ANNs were trained using either 80% or 90% of the available data set, which corresponds to 400 or 450 defect positions from the 500 simulated. Four cases are presented in this section, which are representative of the overall trends observed. The particular cases are given in table 4.3.

Case	Notch depth (mm)	No. of sensors used	ANN configuration
1	4	6	B
2	4	8	C
3	no notch	8	C
4	2	10	C

Table 4.3 Selected cases to demonstrate the effect of the amount of training data

The accuracy of the ANNs' predictions for both x and y co-ordinate generally improved when a larger amount of training data was used. Figure 4.8 shows the overall accuracy of each selected case (the mean absolute deviation from the target), and figures 4.9 and 4.10 show the accuracy of the x and y co-ordinate predictions respectively. In case 3, it can be seen that an opposite trend was observed, that is, the error increased as the amount of training data increased. Intuitively one would assume that a larger training data set would produce more accurate results from an ANN, though attention must be given to the *quality* as well as the *quantity* of the data used to train the ANN. If the data set presented for training does not provide a good representation of the full range of values in the data set, then when previously unseen data are presented there is a risk of inaccurate outputs. As stated in section 3.2,

artificial neural networks, interpolate well but do not extrapolate well (Windsor 1995). Additionally, it is possible to 'overtrain' an ANN so that it does not provide accurate interpolation between values used during training. Such cases are more common with networks containing many layers and/or neurons per layer. Overtraining effectively means that the network 'learns' the training set so rigidly that all it is capable of doing is memorising the training patterns. During overtraining, a network derives overly complex relationships between input and output data so that, when presented with previously unseen data, it attempts to 'over fit' the input data, often returning highly erroneous outputs. An overtrained network memorises well (previously learned patterns are recalled with high accuracy) but is not able to generalise effectively, thus performs poorly when presented with previously unseen data. In case 3, figures 4.8 to 4.10, it is likely that a combination of a poorly selected training data set that did not provide an accurate representation of the overall data set (the training data were selected at random from the complete data set) and an amount of overtraining using these data were responsible for the poorer performance relative to the other configurations shown.

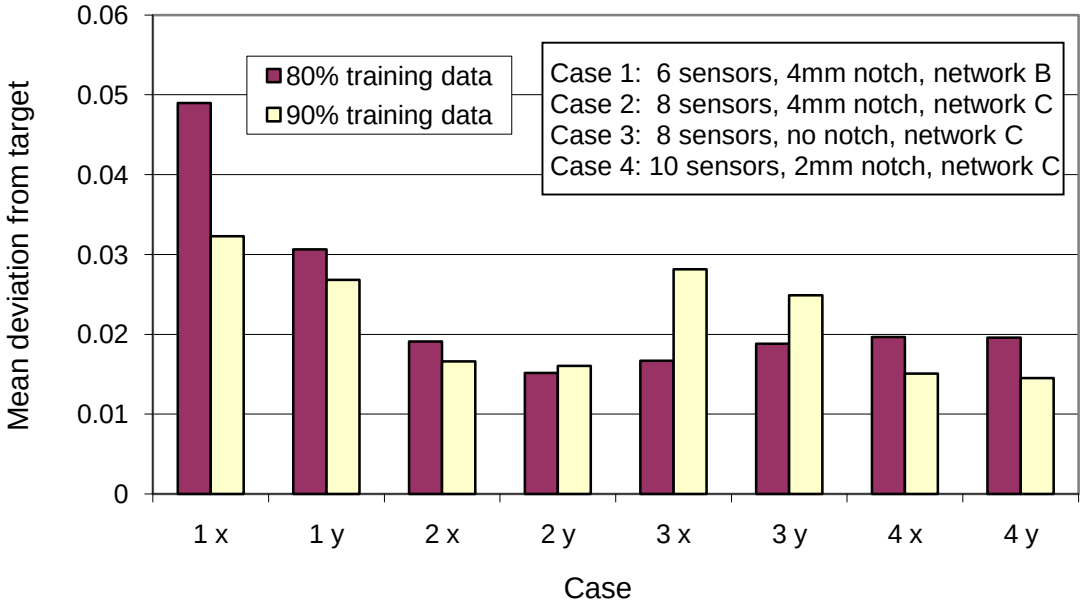


Figure 4.8 ANN performance with amount of training data for four selected cases

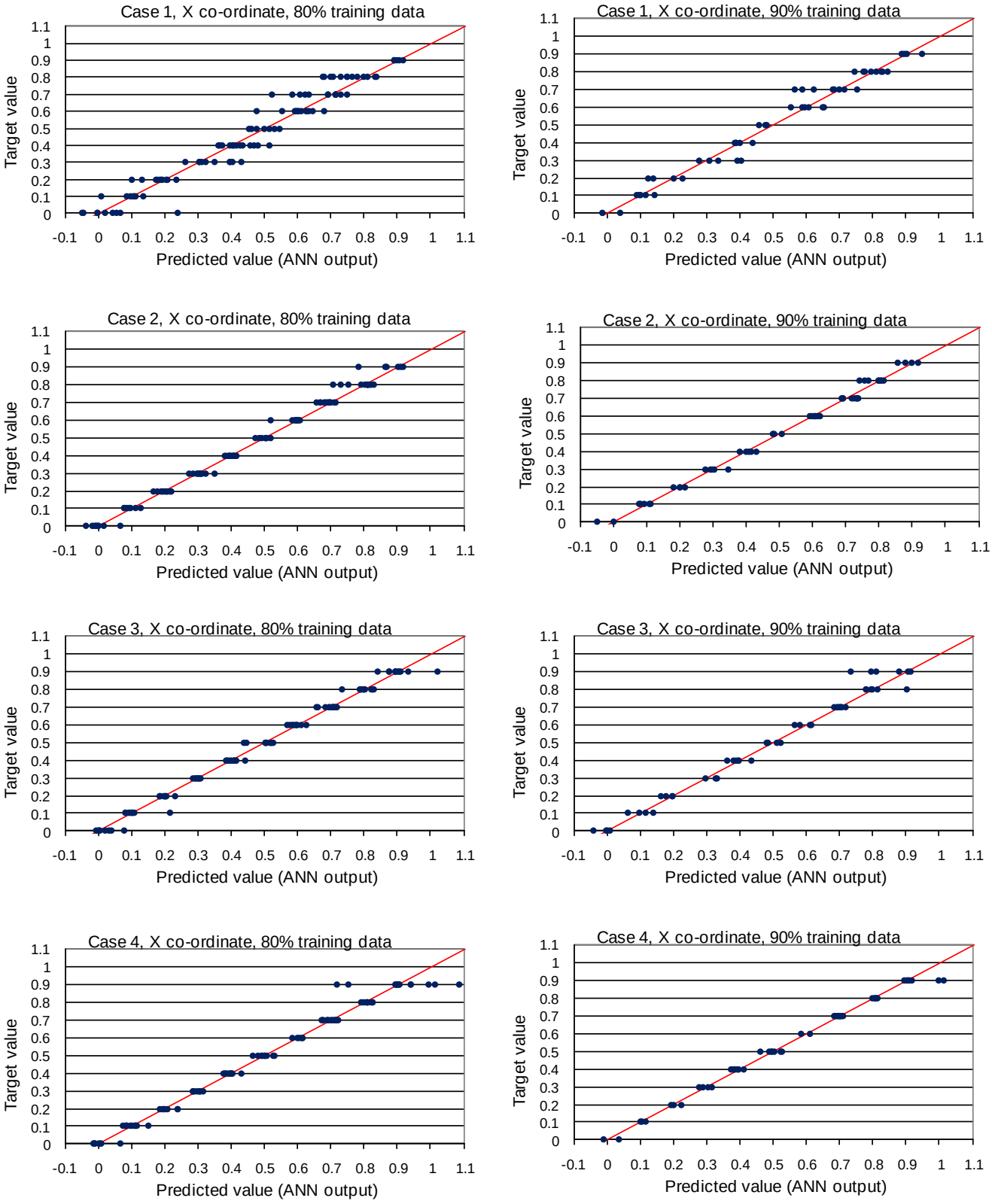


Figure 4.9 ANN outputs for selected cases' X co-ordinate, trained using 80% of available data (left) and 90% (right)

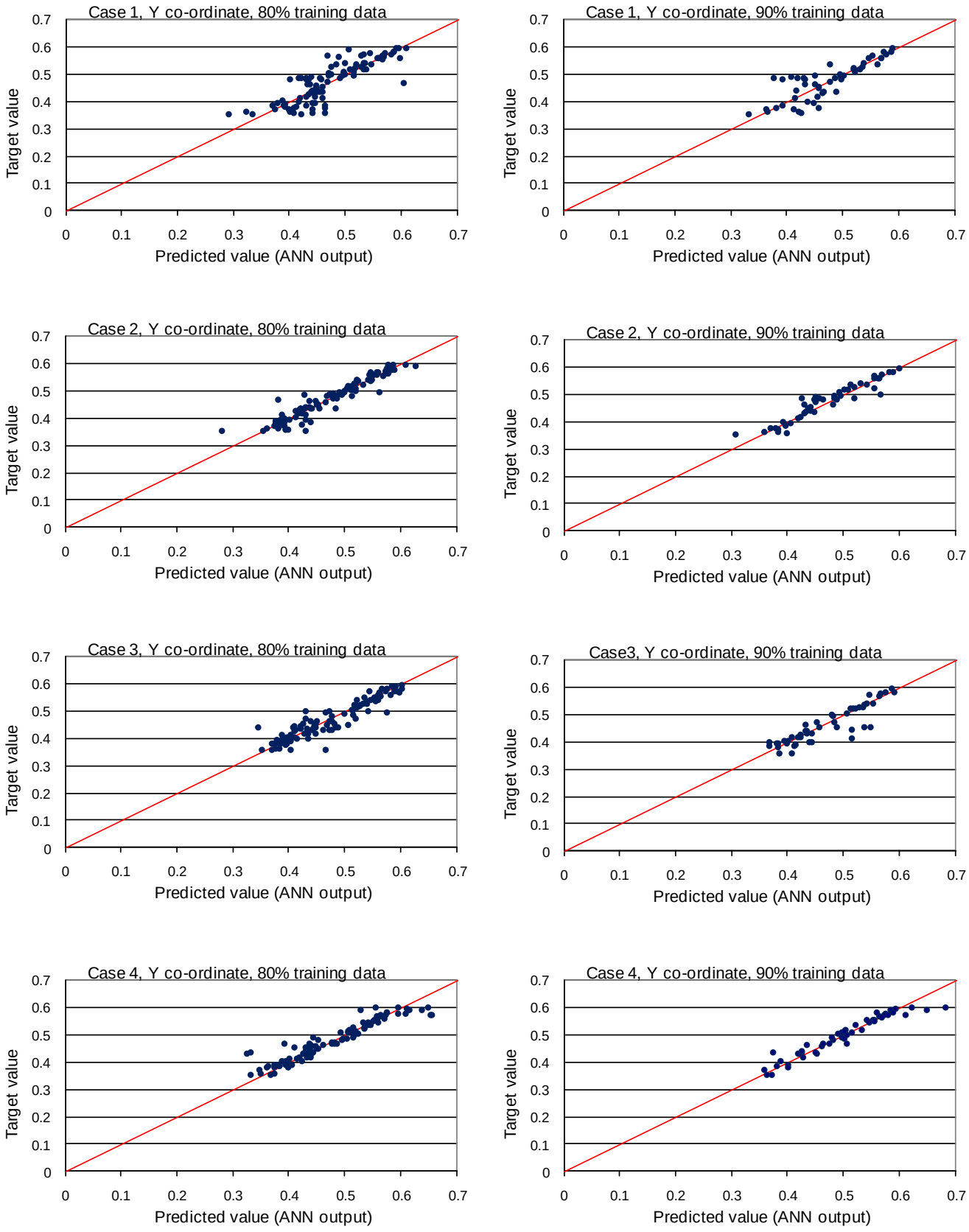


Figure 4.10 ANN outputs for selected cases' Y co-ordinate, trained using 80% of available data (left) and 90% (right)

4.2.3.3 The effect of ANN configuration

Using the three network configurations described in section 4.2.2, ANNs were trained using a percentage of the available data and then assessed using the previously unseen remainder of the data. The overall accuracy of the outputs improved as the number of neurons in the ANNs was increased. Three selected cases, listed in table 4.4, are chosen to demonstrate the trends observed.

Case	Notch depth (mm)	No. of sensors used	Training data (%)
1	5	6	80
2	no notch	8	80
3	4mm	10	90

Table 4.4 Selected cases to demonstrate the effect of network configuration

Figure 4.11 shows the three cases, with the mean normalised deviation from the target for each case shown. As can be seen, the trend is that the deviation reduces as the networks get larger. Individual results for each case are shown in figures 4.12 to 4.14.

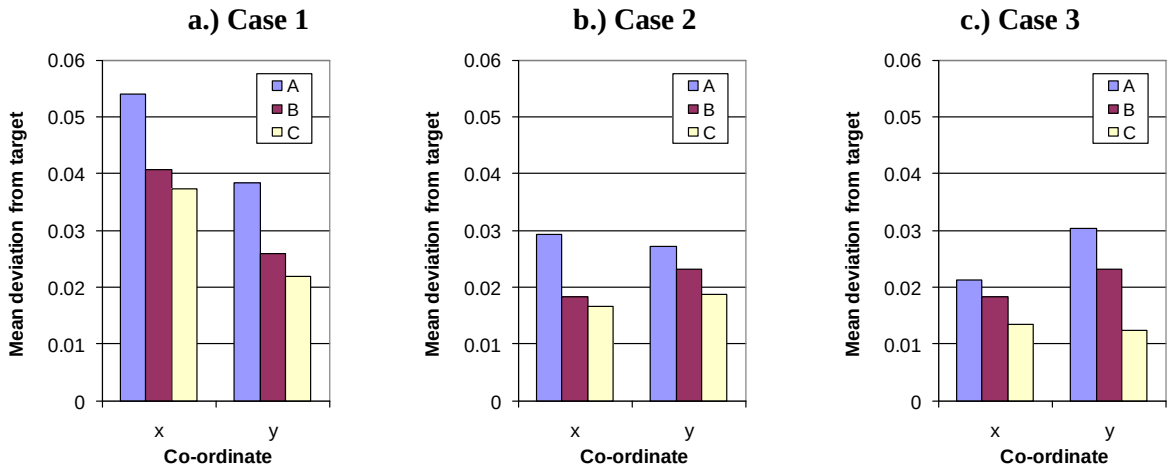


Figure 4.11 ANN performance with network configuration

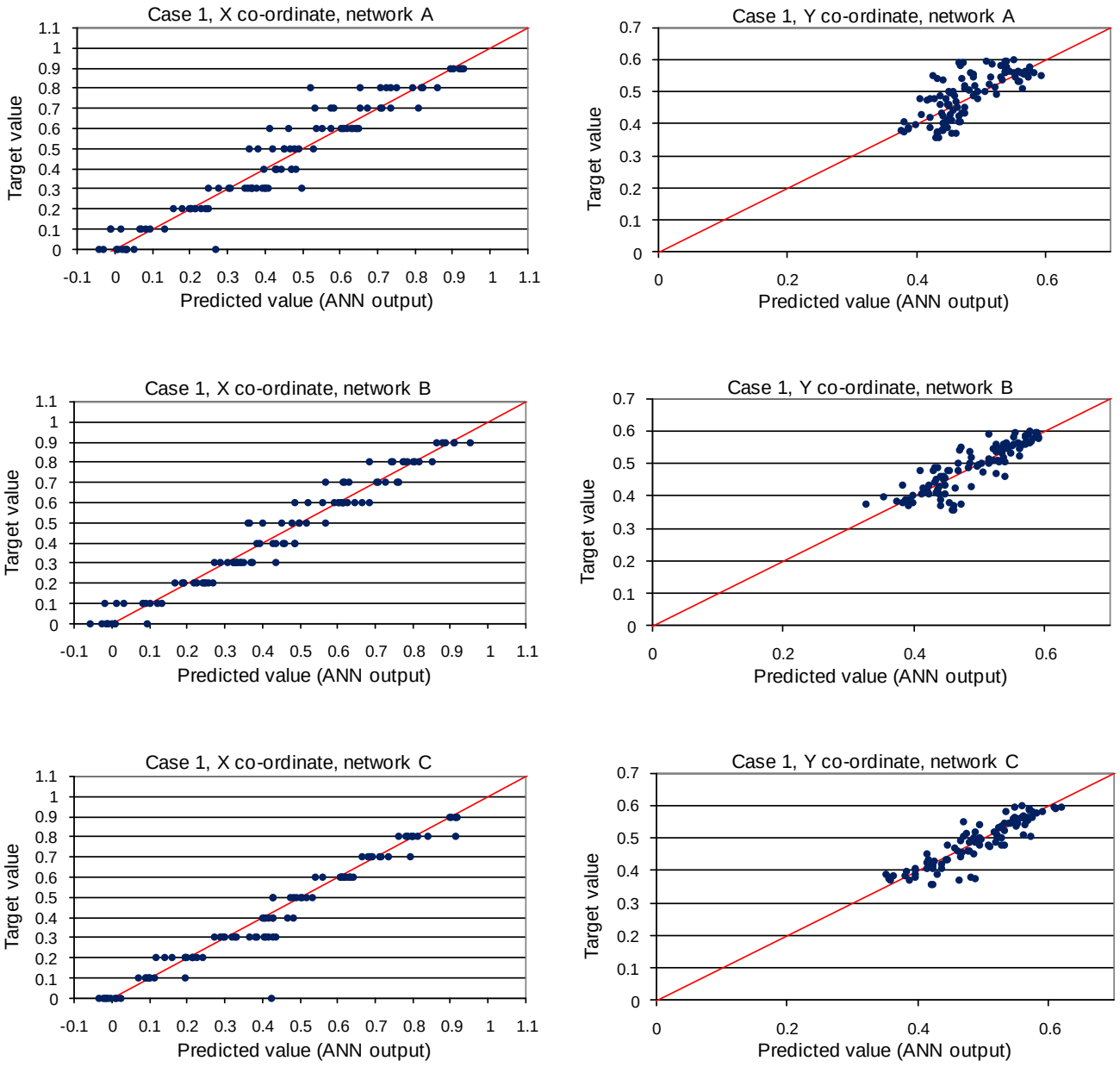


Figure 4.12 ANN outputs with network configuration – case 1

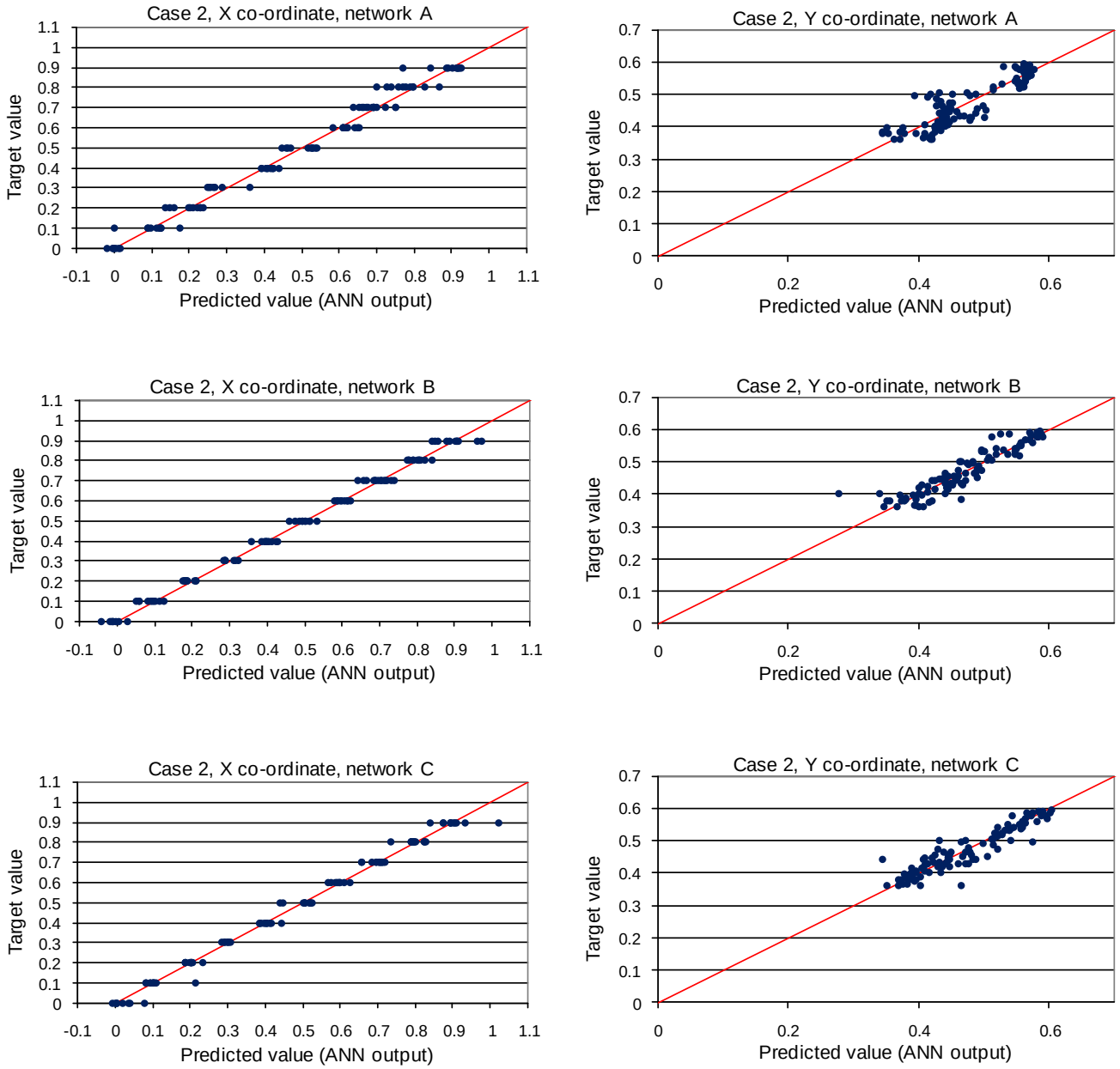


Figure 4.13 ANN outputs with network configuration – case 2

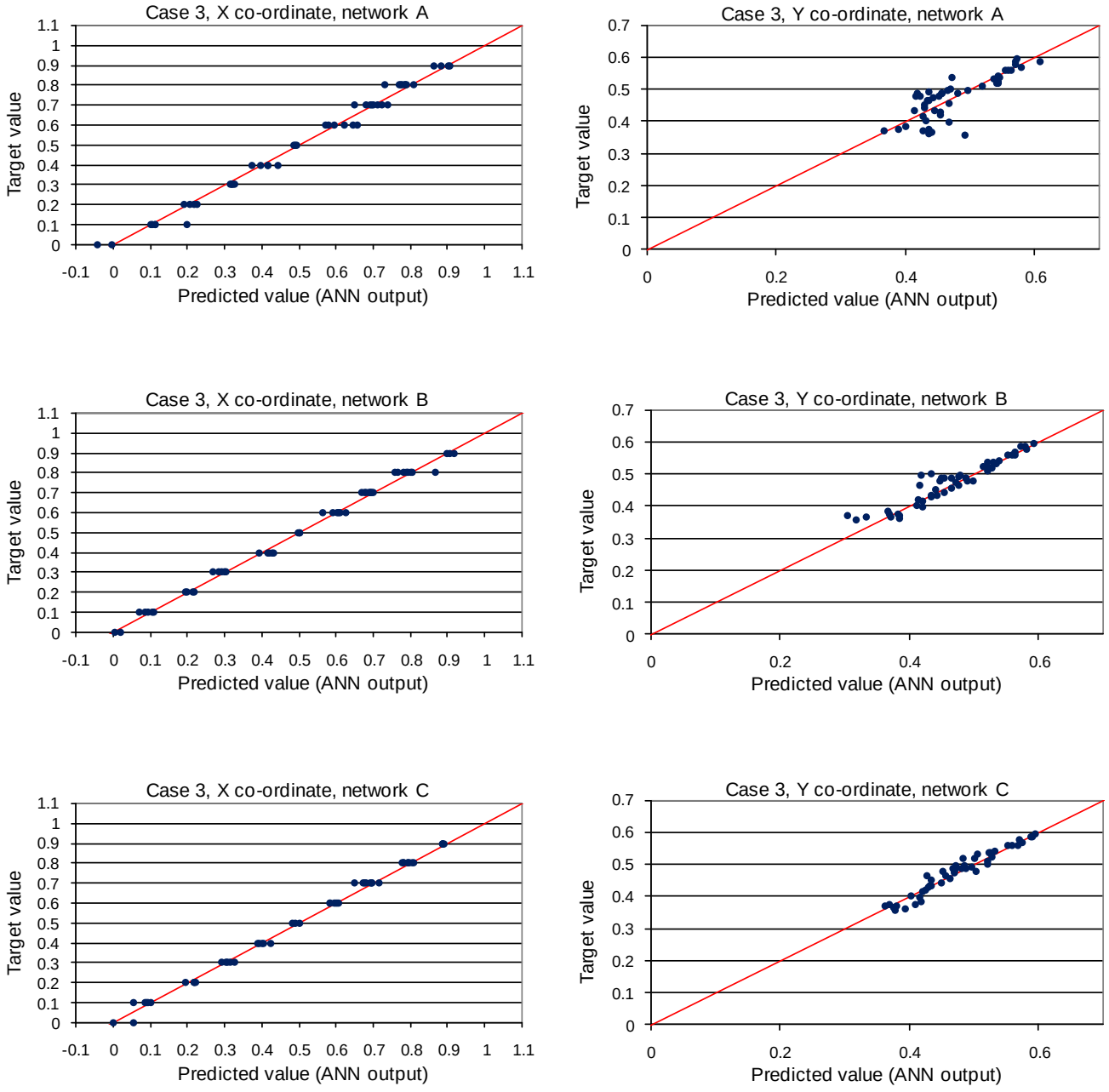


Figure 4.14 ANN outputs with network configuration – case 3

4.2.3.4 The effect of notch depth

Simulations were run to include a notch in the bar of varying depth, acting as a stress concentrator and modifying the incident wave. The presence of a notch near the top of the bar did not significantly affect the ability of the ANNs to locate the position of the void. As described in section 3.3.1.2, the presence of a boundary in a component will have the effect of reflecting the incident wave. In the case of a notch, the boundary only covers part of the medium through which the wave is travelling, thus the incident wave will be partially reflected whilst the remaining part of the wave will continue to propagate through the component. As this study was collecting data from the first wave to arrive at the base of the bar, the presence of a notch reduced the amplitude of this wave and also introduced an antisymmetric component to the wave. Without a notch or defect, the wave would theoretically be symmetric as the pressure was applied evenly across the top of the bar and the component was symmetrical.

Figure 4.15 shows the mean absolute deviation from the target expressed as a normalised value (equation 4.2) for several selected cases as the notch depth was increased.

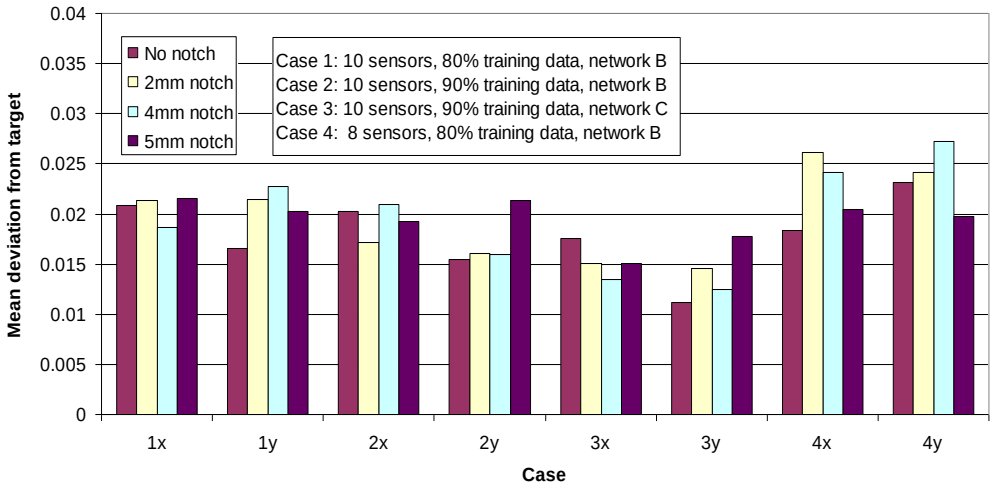


Figure 4.15 ANN performance with notch depth

4.2.4 Larger and smaller defects

Several simulations were conducted to investigate the effects of larger (2mm x 2mm) and smaller (0.5mm x 0.5mm) voids. Although the results were very similar in all cases, it was noted that in the case of smaller voids, a higher accuracy was observed across all the ANNs' outputs. This was considered to be due to the larger training data set, which was four times the size of the set used in the main impact method study, and would therefore provide a much more comprehensive sample when training the ANNs. In this case it was much more likely that a previously unseen defect would lie within one defect's width of a defect used for training, thus minimising the interpolation required from the ANN.

4.2.5 Summary of impact method

A method has been presented to locate defects in a simple component using a stress wave induced by an impact and ANNs to interpret the displacement data collected from the FEM simulation. The ANNs' predictions of defect position remained reliable even with the presence of a notch acting as a stress concentrator in the component. This method was considered attractive because it demonstrated the usefulness of ANNs in providing more information about the defect than would otherwise have been available to an operator measuring displacement. However, although the method worked well with numerical data it was considered that some issues lay in the measurement of experimental data. Accurately measuring very small displacements at a precise time after the impact could be difficult to achieve in practice, especially if, as desired, the testing was to be simple, rapid and repeatable. Although ANNs have been shown to be able to deal with noisy data, or data sets with values missing, if the quality of experimental data is not of the same order as that of the

data gathered from the numerical simulations, issues may arise in the ANNs' ability to interpret those data.

4.3 Modal analysis method

Following the promising results from the impact method, the modal response of the bar was investigated, following work by Salawu and Williams (1995) and Sahin and Shenoi (2003) who used the modal response of components to assess structural condition and to locate flaws. As with many other areas of non-destructive evaluation, ANNs using frequency based data have been shown to provide more information regarding a component's condition than would otherwise be available to an operator, thus there was a desire to evaluate their performance in this area. This method involved extracting the first ten eigenfrequencies of a component, and using these to train and assess the various ANNs. The eigenfrequencies from a non-defected component were stored as reference values, and then compared to those extracted from a component with a defect. The percentage differences between measured and reference eigenfrequencies were used to represent each case to the ANNs, and, once again, the target data were the (x,y) co-ordinates of the defect.

4.3.1 Description of component and test

The component used in this study was identical to that used in the impact method (see section 4.2.1 and figure 4.1), and was modelled in ABAQUS 6.5-1, this time using the linear perturbation method with a Lanczos eigensolver. Ten eigenfrequencies were determined in the analyses with no boundaries imposed on the frequencies. The bar was constrained at its base in all three degrees of freedom using the 'encastre' boundary condition, and no load was

applied. The component was meshed using 2000 plane strain elements of size 1mm x 1mm and type CPE4R, following the successful mesh refinement study in section 4.2. A parametric study was conducted in a similar manner to that in section 4.2. A plain bar and bars with notches of various depths were considered, and for each of these bars 500 defect positions were simulated. The presence of the defect (modelled as a void) reduced the stiffness of the bar in the region of the defect, thus altering the dynamic properties of the component. Depending on the defect's position, the values of the first ten eigenfrequencies were altered, and when these were compared with the values obtained from a component with no defect the resultant differences in values were used as data to train and assess the ANNs. Table 4.5 shows the first ten natural frequencies from a non-defected bar, along with those from bars with a defect near the top, middle and bottom of the region in which defects were investigated, following figure 4.1. The displacement in the first ten modes is shown graphically in figure 4.16.

Mode	Defect position						
	No defect	Top right		Middle		Bottom left	
	Hz	Hz	% change	Hz	% change	Hz	% change
1	**1290.18**	1288.14	-0.16	1280.36	-0.76	1288.65	-0.12
2	**3522.21**	3516.17	-0.17	3521.18	-0.03	3519.70	-0.07
3	**6508.17**	6507.65	-0.01	6490.65	-0.27	6503.44	-0.07
4	**6663.50**	6662.73	-0.01	6631.10	-0.49	6659.42	-0.06
5	**10583.0**	10568.8	-0.13	10579.5	-0.03	10567.1	-0.15
6	**15231.4**	15216.7	-0.10	15145.9	-0.56	15227.9	-0.02
7	**19391.9**	19371.7	-0.10	19360.0	-0.16	19399.1	0.04
8	**20661.2**	20649.0	-0.06	20632.9	-0.14	20632.4	-0.14
9	**26621.7**	26584.1	-0.14	26514.9	-0.40	26605.2	-0.06
10	**32085.4**	32093.3	0.02	32078.8	-0.02	32050.5	-0.11

Table 4.5 Comparison of natural frequencies of a bar with 2mm notch for selected cases

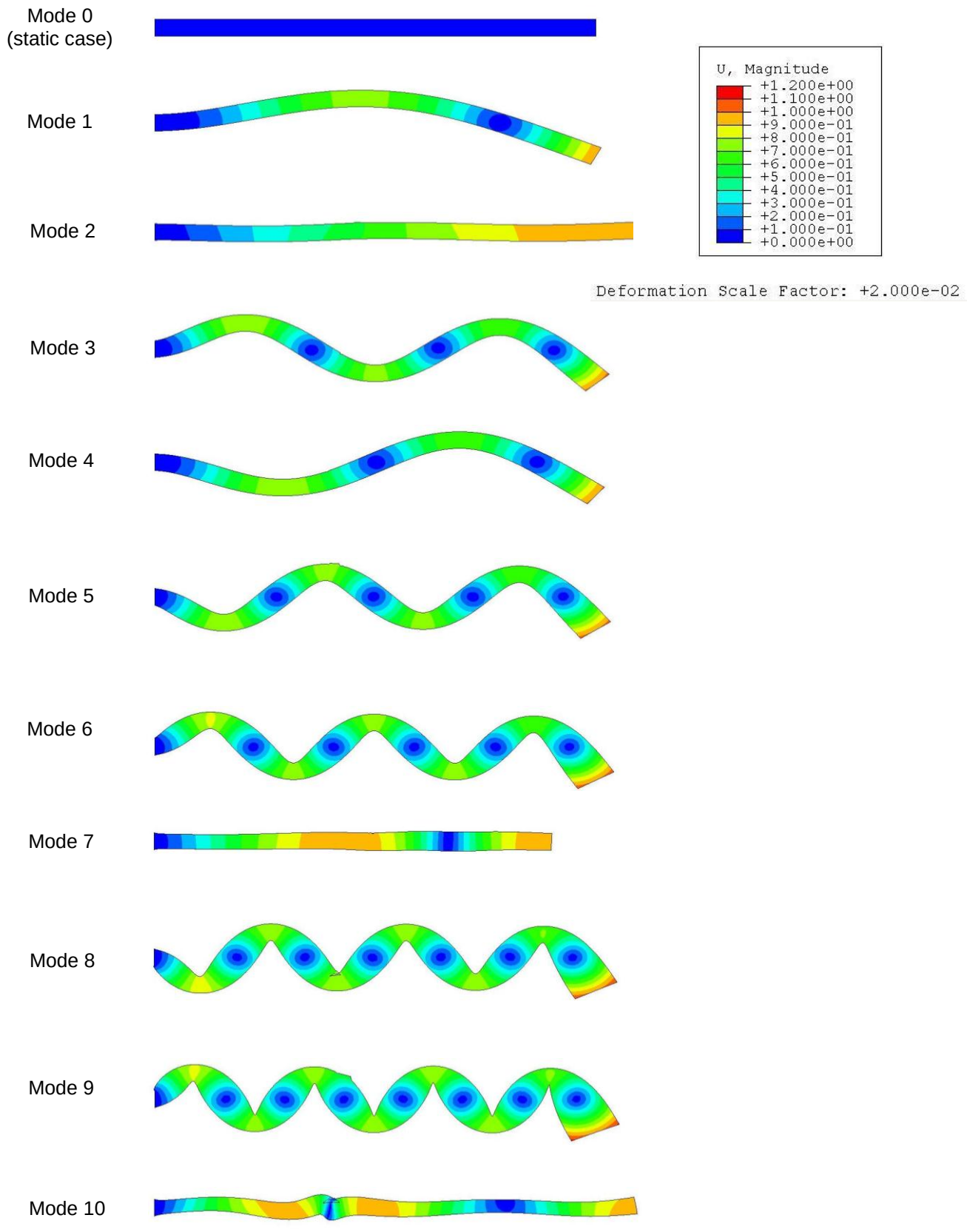


Figure 4.16 Displacement in the first ten modes of natural resonance for a plain bar

4.3.2 Description of ANNs

In this study, the same ANNs were used as in the previous study (see section 4.2.2). The ANNs were trained with up to 90% of the available data and assessed with the remainder of the data. Again, the ANNs were trained using the SCG method for 20,000 epochs. The inputs to the ANNs were the percentage differences between the eigenfrequencies of a defected and a non-defected component, which were then normalised using the formula in equation 4.1 to lie in the range [-1, 1]. The targets were the normalised x and y co-ordinates of the defects.

4.3.3 Results of modal analysis method

As in the impact method study, the same 120 permutations of case were investigated, and representative results are presented in this section. On the whole, the accuracy of the ANNs was much better using the modal analysis method than the impact method, although in a practical situation the impact method would be more likely to produce useful measurements; there are many ways to track an elastic wave's motion through a component, but measuring eigenfrequencies to a high enough degree of accuracy would be very difficult, given that the difference in eigenfrequencies due to the presence of a flaw was in the region of ~0.5%. The measured eigenfrequencies were in the range of 200Hz to 27kHz.

4.3.3.1 The effect of number of eigenfrequencies

As with the previous study, the accuracy of the ANNs' predictions improved as the number of eigenfrequencies used was increased. Values from 2 to 10 eigenfrequencies were used as inputs to the ANNs, and the accuracy of the ANNs was evaluated using the mean deviation

from the target values, as shown in equation 4.2. Figure 4.17 shows the performance of the ANNs as the number of eigenfrequencies was increased for a plain bar and for a bar with a notch of 2mm depth. It can be seen that the accuracy in determining the x co-ordinate is far better for a bar with a notch. This is because a plain bar is a symmetrical shape, so the presence of a defect at a certain distance from the centreline will affect the eigenfrequencies in the same way regardless of its direction from the centreline; the local reduction in density will be equal in terms of the dynamic response of the component. In the case of a bar with a notch, because the component is asymmetrical, the dynamic response is affected in different ways depending on the direction from the centreline of a defect's position. As can be seen from figures 4.18 and 4.19, the accuracy of the modal analysis method in locating the exact position of each defect was far higher than that of the impact method. It should be noted that in all figures comparing ANN outputs with target values, the normalised values are used, following equation 4.1.

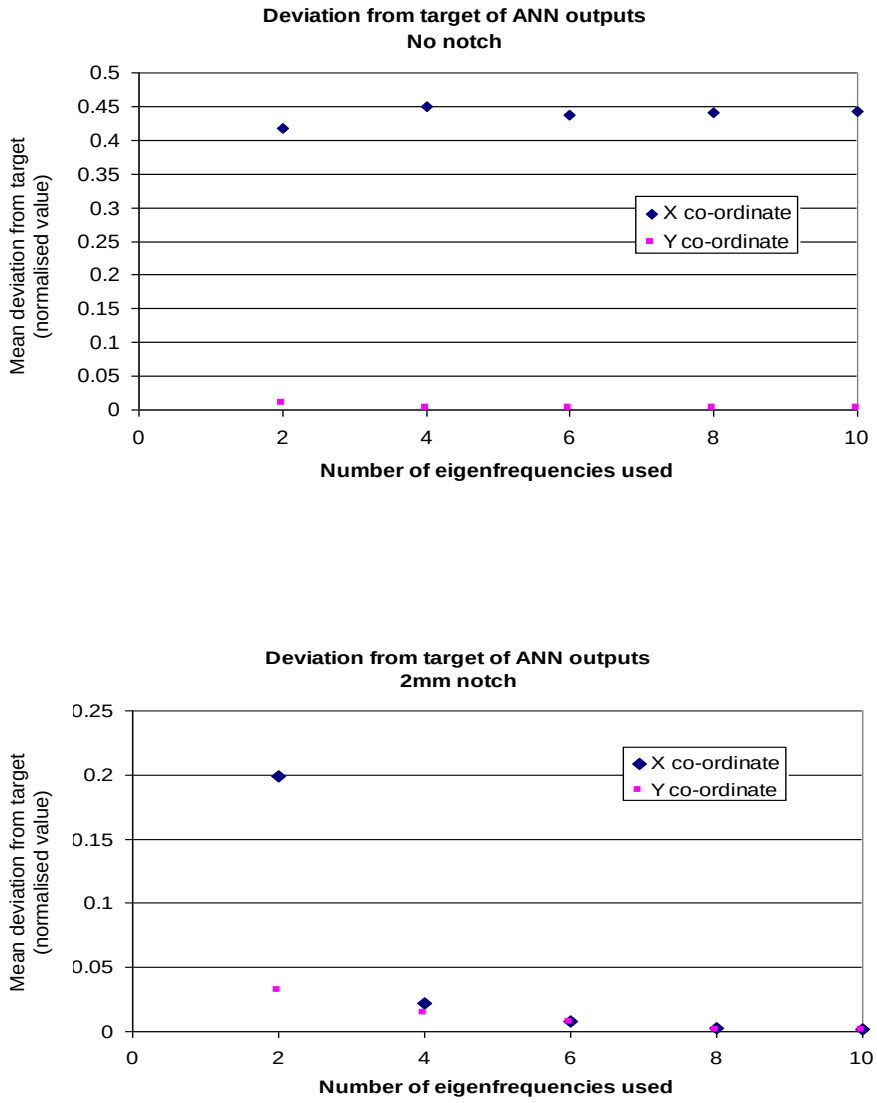


Figure 4.17 ANN performance with number of eigenfrequencies for bars with and without a notch

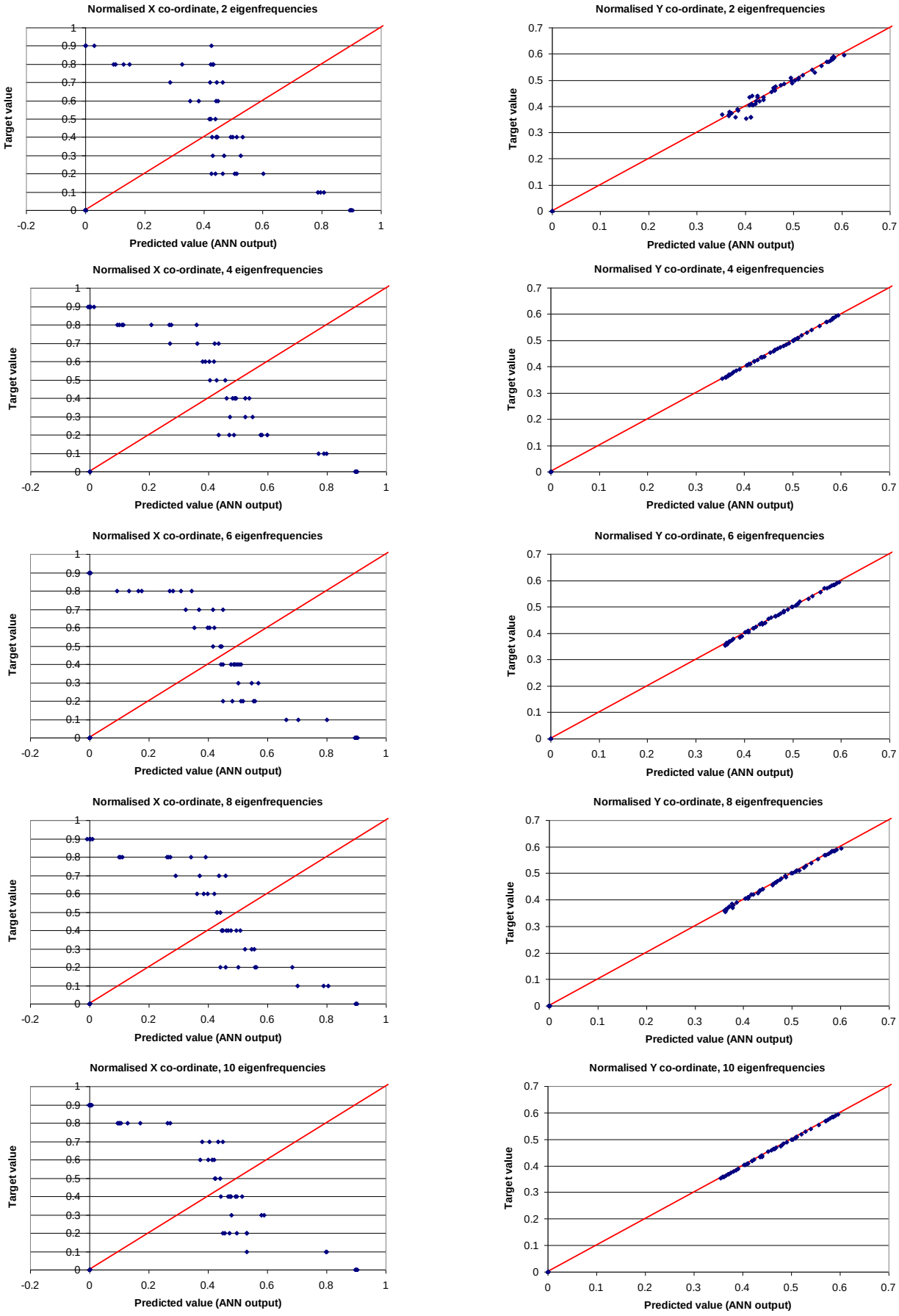


Figure 4.18 ANN output with number of eigenfrequencies
Plain bar, 90% training data, network C

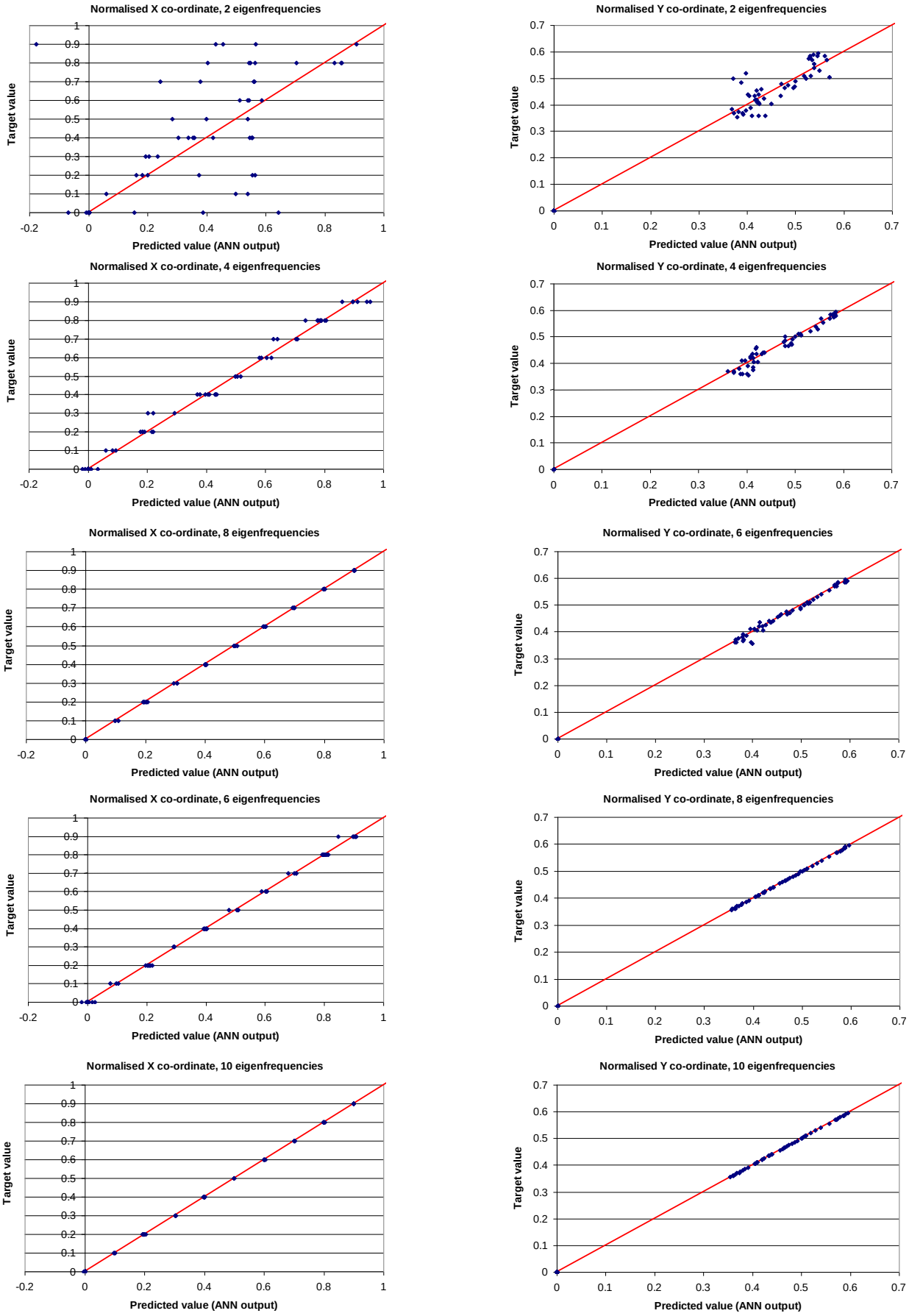


Figure 4.19 ANN output with number of eigenfrequencies
2mm notch, 90% training data, network A

4.3.3.2 The effect of the amount of training data

In almost all the cases considered, the accuracy of each ANN was increased as the amount of training data was increased, as can be seen in figure 4.20. The only exception to this trend was the x co-ordinate of the plain bar case, where the accuracy was consistently low for all cases, regardless of the amount of training data. It is worth noting that the accuracy of the ANNs when presented with data based on dynamic response was much higher than when presented with data based on the stress wave propagation using the impact method.

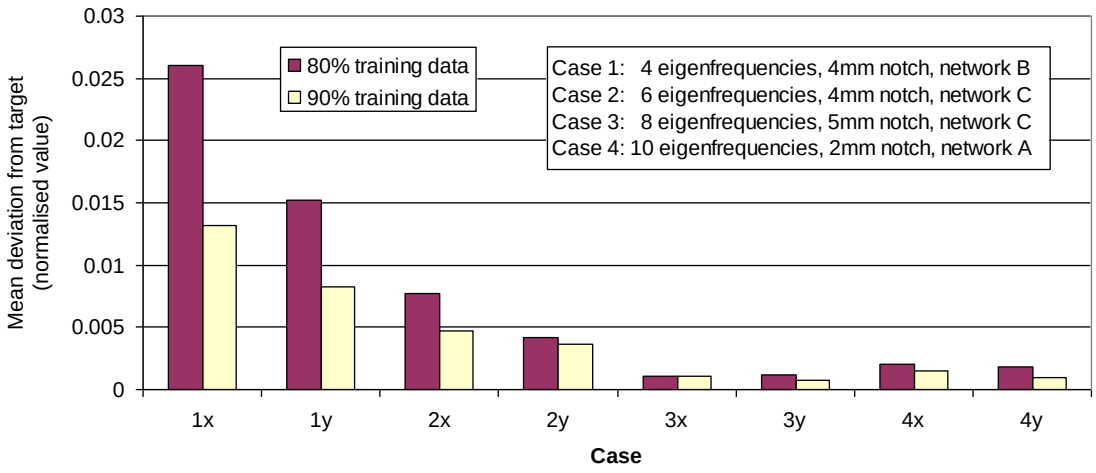


Figure 4.20 ANN accuracy with training data

4.3.3.3 The effect of ANN configuration

As with the impact study, the accuracy of the ANN output increased as the size of the ANN increased, as can be seen in figure 4.21. The general trend observed was that there was a noticeable improvement when moving from network A to network B, but adding more neurons in the hidden layer did not produce a proportional increase in the accuracy of the outputs.

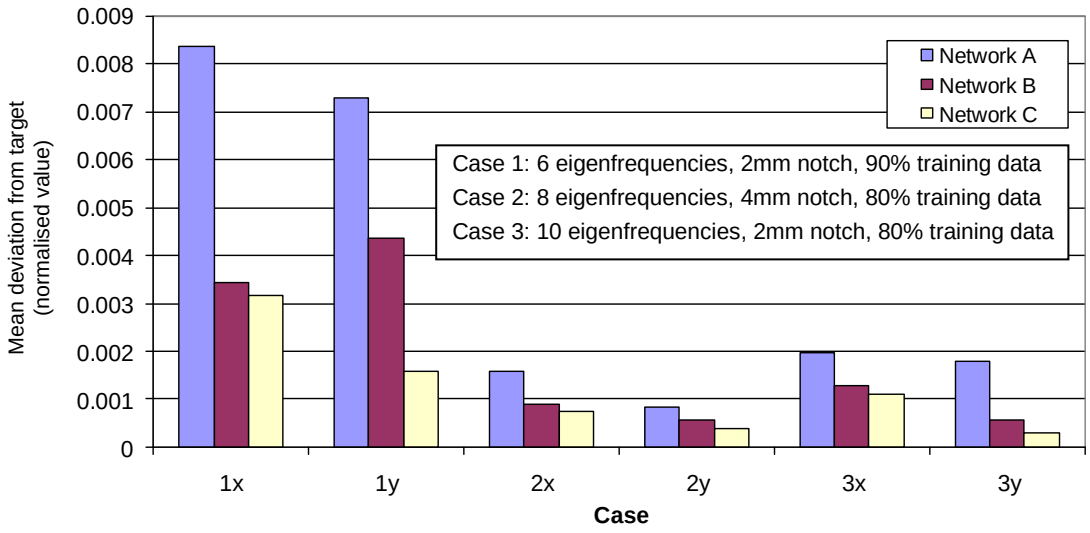


Figure 4.21 ANN performance with network configuration

4.3.3.4 The effect of notch depth

As can be seen in figure 4.22, the presence of a notch only had a significant effect on the ANNs' accuracy when locating the x co-ordinate of the defect in the plain bar case. It can also be seen that when using 8 eigenfrequencies or more, the accuracy of prediction was very high for any case where a notch was present in the bar.

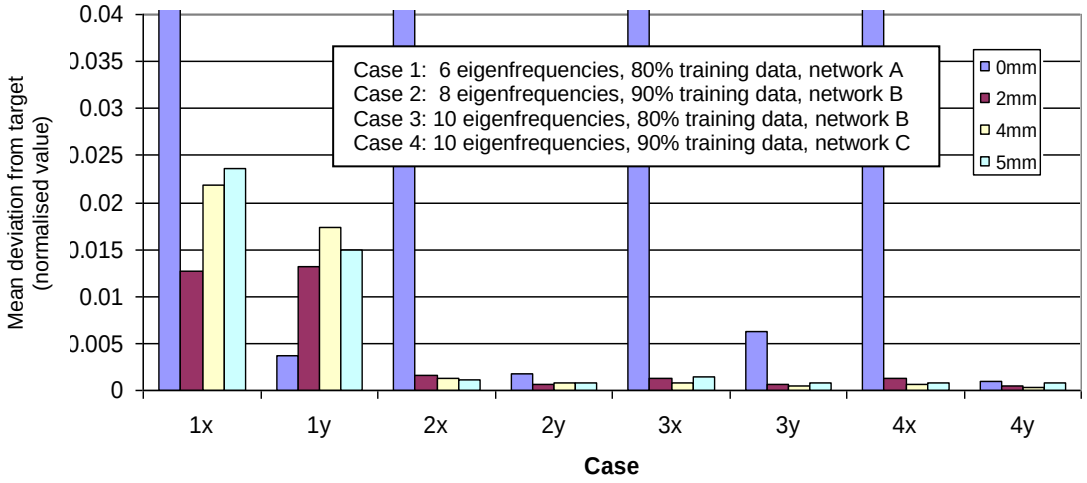


Figure 4.22 ANN performance with notch depth

4.3.4 Summary of modal analysis method

A method to locate flaws based on the modal response of a component has been presented. As with the method using an impact at the top of the bar, the modal analysis method demonstrates the usefulness of ANNs in providing more information regarding the position of a flaw than would be available otherwise. Excellent results were seen in the ANNs' outputs except in the case of bars without a notch, where the x co-ordinate could not be accurately determined due to the symmetry of the bar. This method was considered to have severe limitations in practical terms, however, as it relies on measuring the first ten eigenfrequencies to a very accurate degree, which would be very difficult to do in practice. Published work using this method is generally applied to large structures such as bridges (Salawu and Williams 1995), or used to locate comparatively large regions containing flaws (Sahin and Shenoi 2003) rather than to accurately locate very small flaws. The difference in natural frequency between a bar with a defect and one without was small enough that the bars would

have to be manufactured to an extremely high tolerance to allow defects to be located in this manner. A variation in the length and width of 0.01mm would alter the natural frequency by around 0.22%, which would be sufficiently large to adversely affect the performance of the ANN. Although the method worked well with numerical data, it was considered that it would not be possible to gather experimental data to a high enough degree of accuracy using the available resources. For this reason, the method was not pursued any further in this thesis.

4.4 Ultrasonic method

In order to create a model that could be easily validated experimentally, the FE model used in the impact method was adapted to represent a pulse-echo ultrasonic A-scan inspection. This was much easier to validate experimentally than either of the two methods previously described, as equipment to do so was already available at Oxford Brookes University. Given that the ultrasonic pulse-echo method is essentially a one-dimensional inspection method, it was decided that the ANN outputs for this study would be the depth to a defect (distance from the ultrasonic transducer) and size of a defect (length perpendicular to the incident P-wave); the x co-ordinate of the defect was not considered in this case. The technique demonstrated by Yagawa and Okuda (1996) and Lowe and Diligent (2002) was used to simulate an ultrasonic pulse being applied to the top of the bar in place of the step impact. This would ensure a controllable frequency.thickness product, as the frequency of the applied pulse could be precisely specified. In the case of an instantaneous step load all frequencies would be excited at the same time, thus making it potentially very difficult to distinguish between different wave modes.

4.4.1 Description of component and test

The same vertical steel bar was modelled using ABAQUS, as in the previous studies, and constrained in all three degrees of freedom at the base. 4-noded plane strain elements of size 1mm x 1mm and type CPE4R were used, giving 2000 elements in total. The ABAQUS dynamic explicit solver was chosen with time steps of 0.1μs to ensure that each time step was less than the time taken for the stress wave to propagate across one element (Lowe, Challis *et al*. 2000; Hill, Forsyth *et al*. 2004). To represent the ultrasonic input pulse, two periods of a sinusoidal function inside a Gaussian window were used as shown in figure 4.23 and detailed in equation 4.3. The final function was normalised to give a range of [-1, 1].

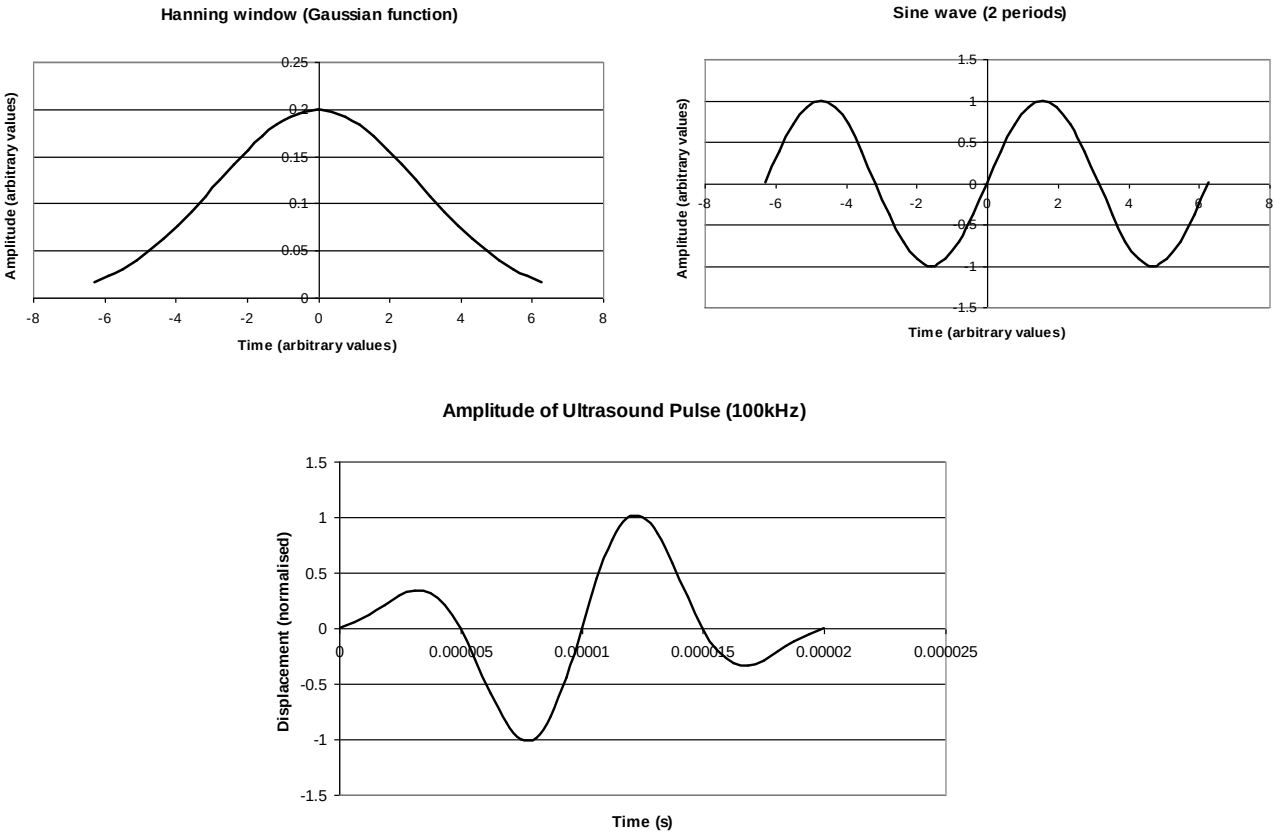


Figure 4.23 Creation of the ultrasonic pulse

The Gaussian function is given by equation (4.3):

$$f(x) = \frac{e^{-(x-\mu)^2 / 2\sigma^2}}{\sigma\sqrt{2\pi}} \tag{4.3}$$

where μ and σ are parameters that affect the shape of the curve. In this study, values of $\mu=0$ and $\sigma=2$ were used, with a range in x of $[-2\pi, 2\pi]$. The resultant curve was then scaled to start at t=0 and to last the duration of the ultrasonic pulse.

A frequency of 100kHz was chosen to give a frequency-thickness product of 1MHz.mm, meaning that only the fundamental flexural (antisymmetric) wave A_0, and the fundamental compressional (symmetric) wave S_0 would be present (Rose 1999). The ultrasonic pulse was applied to the top surface of the bar as pressure using the tabular amplitude function in ABAQUS, with a maximum value of 10MPa. Square defects of between 1mm and 6mm in height were considered, lying in the same region of the bar as in the previous studies. 1650 cases were investigated in total, with the study being restricted to internal defects; it was considered that a visual inspection would be sufficient to locate surface flaws, whereas an ultrasonic inspection would be more useful in locating internal flaws. Throughout each simulation, vertical displacements (U2) of each node on the top surface of the bar were recorded, and a mean value across nodes used as the representative signal. The presence of a defect caused the incident wave to be partially reflected, thus causing a disturbance at the top of the bar at a time before the main wave returned after being reflected from the base. A data set was created using representative features of the wave response measured at the top of the bar. For this study, the signal from a bar with no defect was subtracted from the signal from the bar with a defect so that the resultant wave showed the difference between signals, following Yagawa and Okuda (1996). From this signal, an algorithm was developed using

MATLAB 7.1 (see appendix D) to locate the time and the height of the first peak in the differential signal. These two values were then normalised and used to represent each case to the ANNs. The time of the peak provided information regarding the defect's position (y co-ordinate), and the height of the peak provided information regarding the defect's size (Halmshaw 1991). A typical reflected signal from a bar with no defect is shown in figure 4.24, along with the signal from a bar a defect present and the resultant feature signal from which the time and height of the first peak was determined.

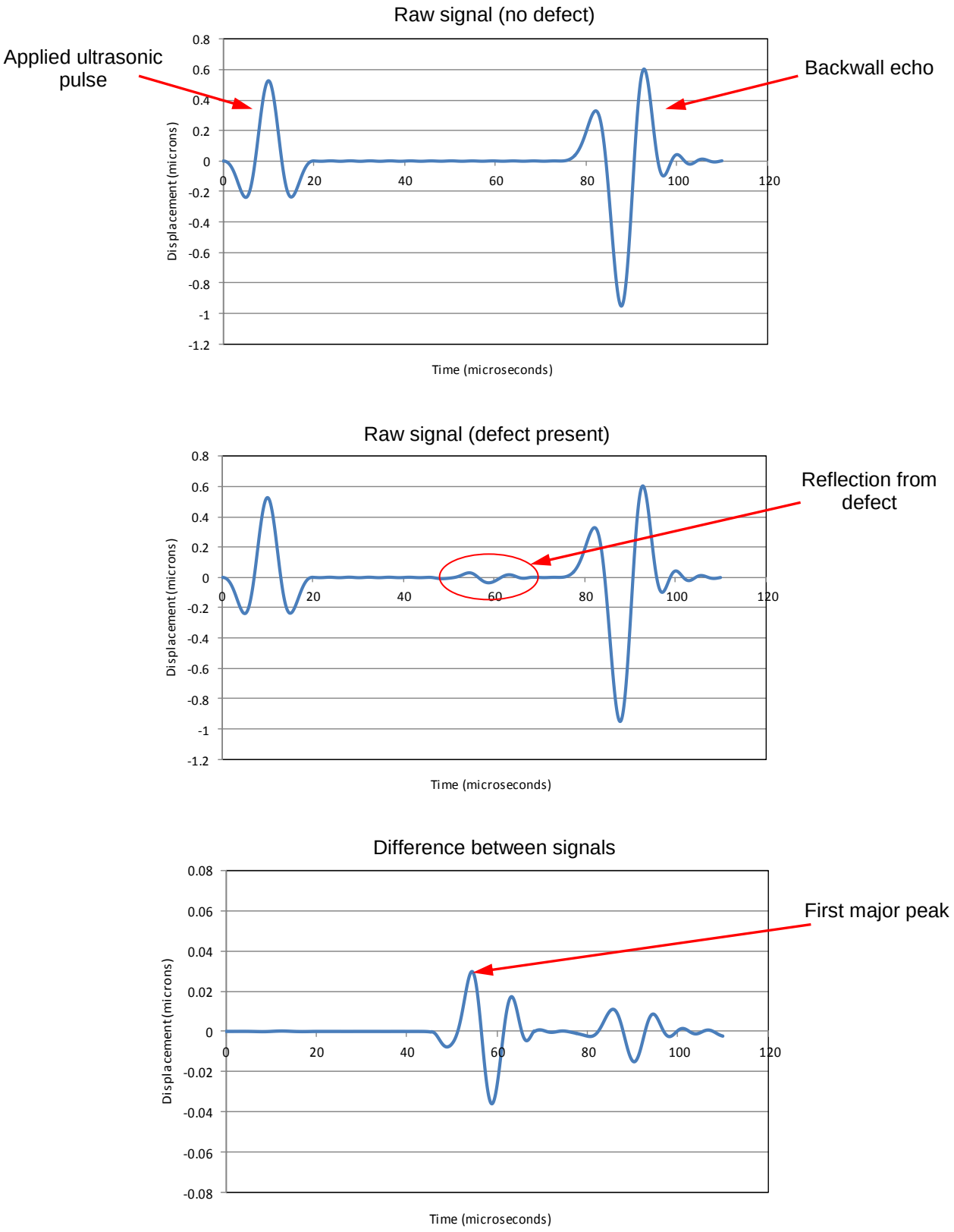


Figure 4.24 Feature extraction from the ultrasonic signal

4.4.2 Description of ANNs

The same three ANN configurations were used as in the previous two studies (see section 4.2.2), and trained with between 5% and 90% of the available data, arranged in random order. To enable an objective comparison between cases, when the data were arranged in random order this was done in groups dependent on defect size. This way, a training set of, for example, 80% of the available data would contain 80% of the data for defects of 1mm width (ordered at random), 80% of the data for defects of 2mm width and so on. The target data (ANN outputs) were the normalised y co-ordinate and the size of the defect. As in the previous studies, the accuracy of the ANNs was measured using the mean deviation from the target values (equation 4.2).

In this study, the ability of the ANNs to deal with noisy data was also investigated. Networks were trained with data without noise, and then assessed with data to which varying amounts of artificial noise had been added. Noise was added using the formula presented by Sahin and Shenoi (2003), shown below in its modified form to increase the range to [V_o-n, V_o+n].

$$V_n = V_o \times \left[1 + (RAND - 0.5).(2n)\right] \qquad (4.4)$$

where V_n is the value with noise added, V_o is the original value, *RAND* represents a random number between 0 and 1, and n is the amount of noise to be added. Noise of 1 to 40% (in 1% increments) was added to all data points in the assessment set (a different value of *RAND* for each point) ten times, and the mean of these ten values taken for each point to represent the case to the ANNs.

4.4.3 Results of ultrasonic method

As in previous studies, a large number of simulations were carried out and data collected. The amount of data used to train the ANNs varied from 5% to 90%, with values from 10% to 90% going up in increments of 10%. Three ANN layouts were considered, as described in section 4.2.2, and noise was varied from 0% to 40% (n=0 to n=0.4) using equation 4.4. The results shown in this section, as in the previous sections, are a selection representative of the overall findings of the study.

4.4.3.1 The effect of the amount of training data

As with the previous two studies, the networks generally performed better with a larger training data set. However, as the training data set was larger than in previous studies it was possible to achieve reasonable ANN performance with fairly small amounts of training data. It was noted that the largest ANN had very poor accuracy compared to the other two when trained with a very small data set. This is a good example of undertraining of large networks; for a large network capable of drawing more complex links between input and output data, an insufficient training set can lead to incorrect predictions when the network is presented with previously unseen data, especially if the values of that data lie outside the bounds of the training data set, thus requiring the ANN to extrapolate. Figures 4.25 and 4.26 show the accuracy of the ANN outputs for network C as the amount of training data was increased. It can be seen that when 30% or more of the available data was used for training, the accuracy in predicting both defect size and position was very good. Figure 4.27 shows the mean absolute error in all three ANN configurations, and again it can be seen that there is little improvement in the accuracy of the ANNs' outputs once around 20-30% of the data were

used for training. As in previous figures, the mean absolute error in terms of the normalised values is used, as presented in equation 4.2. It is interesting to note that when 60% of the available data was used to train network A (figure 4.27), the estimations for defect size were significantly less accurate than in most other cases. The most likely explanation for this is that the random combination of training data did not provide a full representation of the spread of values, thus as the network derived relationships between the input and output data during training, an amount of overtraining occurred as described in section 4.2.3.2. Given the good performance of network A at all other values of training data above 10%, overtraining is the most likely cause of the increased error when 60% of the data were used to train the ANN. An additional point worthy of mention is that all the errors are very small, thus even though a spike is visible in the error observed, it relates to a normalised error of around 0.0065, which equates to 0.039mm in terms of estimating the size of the defect.

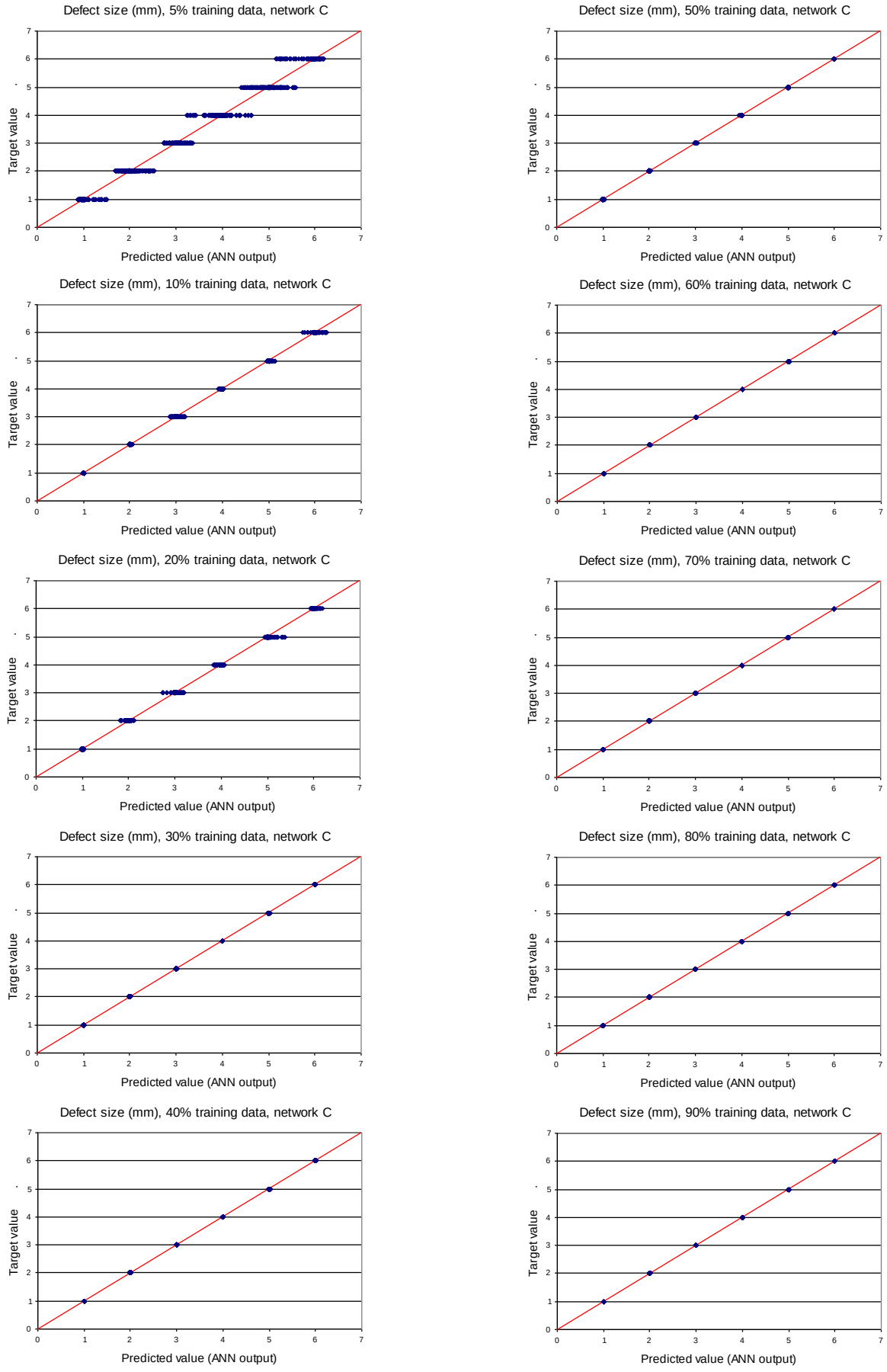


Figure 4.25 ANN output for defect size with amount of training data

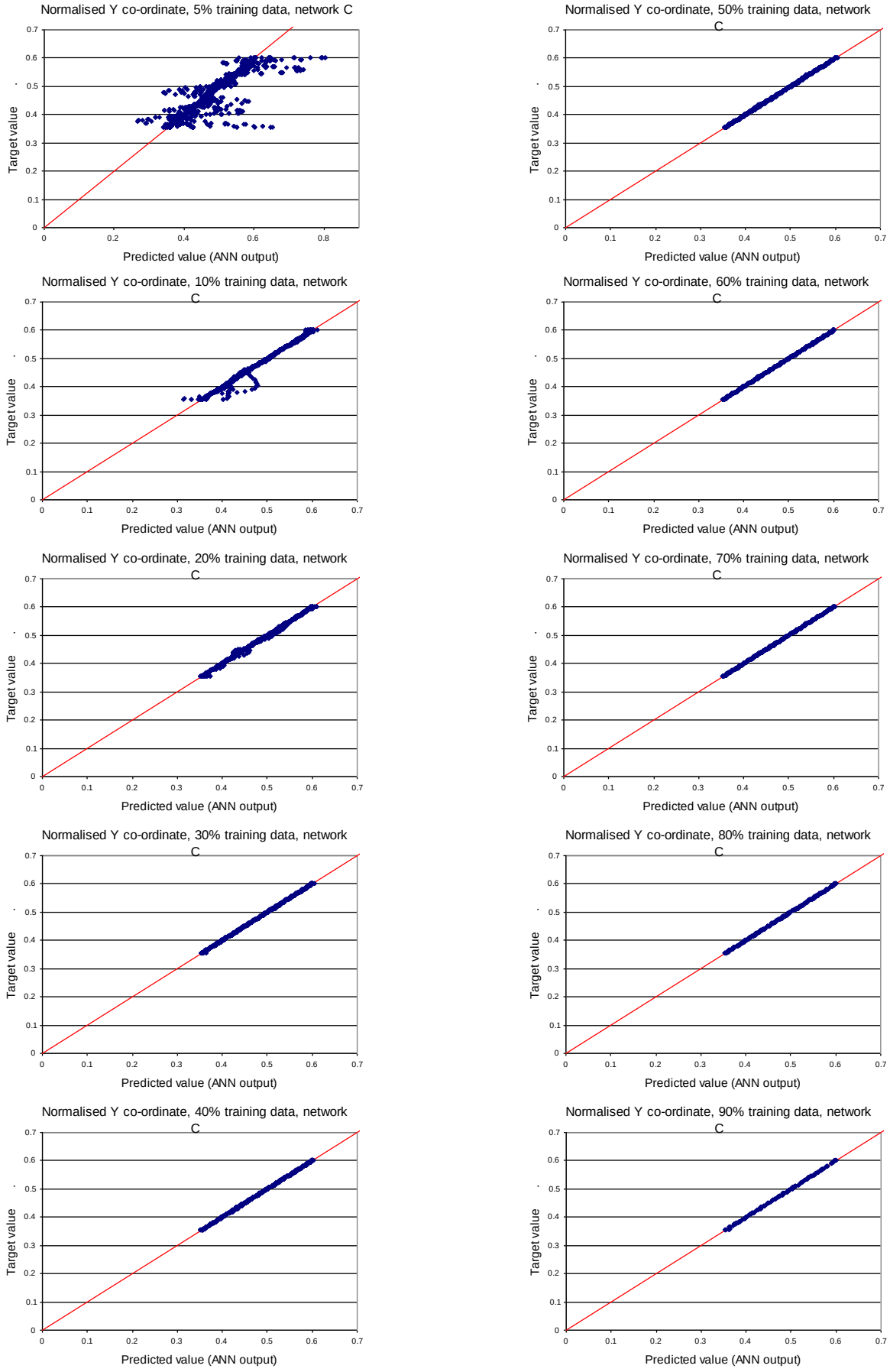


Figure 4.26 ANN output for defect position with amount of training data

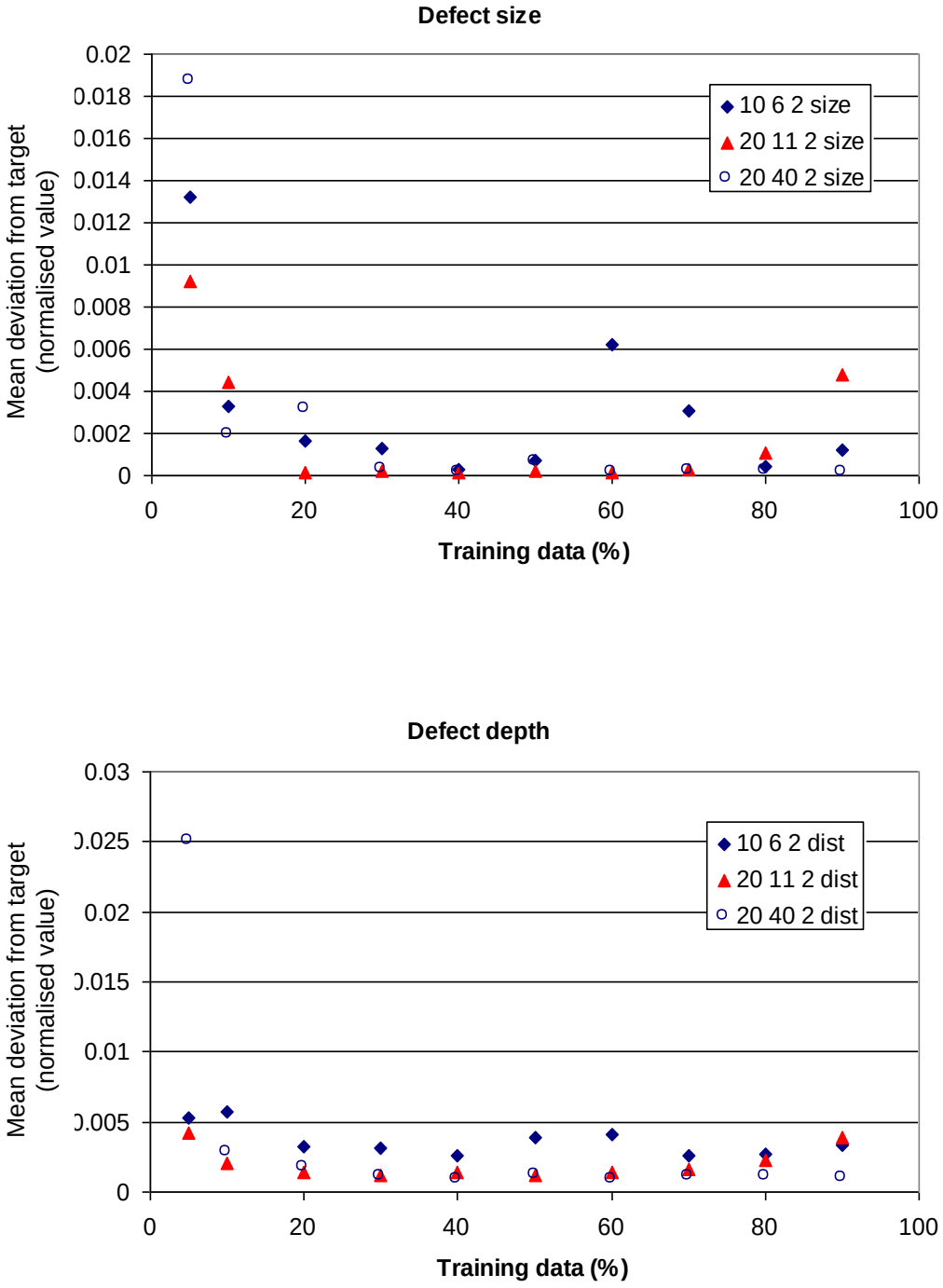


Figure 4.27 ANN performance with amount of training data

An interesting point to note from figure 4.27 is that for the two smaller ANN configurations (10-6-2 and 20-11-2), overtraining seems to have occurred when 90% of the data were used to train the networks, i.e. the error increases slightly. However, the errors are very small; a mean absolute error of 0.005 in sizing the defect (which corresponds to an error of 0.03mm) and 0.004 in locating the defect (which corresponds to an error of 0.8mm) suggest that the system is performing at a sufficiently high level of accuracy.

4.4.3.2 The effect of ANN configuration

In general, the results show that increasing the number of neurons in the ANN produced better results, but this was only the case where a sufficient amount of training data was used. For cases with very small amounts of training data the largest network produced the least reliable results. Figures 4.28 and 4.29 show that in the case where 10% of the available data were used to train the networks, the best results were obtained with network A (for defect size) and network B (for defect position), whereas when 70% of the available data were used for training, network C produced the most accurate results for both defect size and position.

It can be seen from figures 4.28 and 4.29 that, when very small amounts of training data were used, the accuracy of the largest ANN was significantly lower than that of the other two. This is a good example of *undertraining*, where an ANN is not able to fully derive the relationship between input and output data. With the smaller architectures, less complex relationships would be derived, hence they still performed well with small amounts of training data. With network C, much more complex relationships between input and output data were possible, thus with small amounts of training data the network was not able to fully derive an adequate relationship between input and output data.

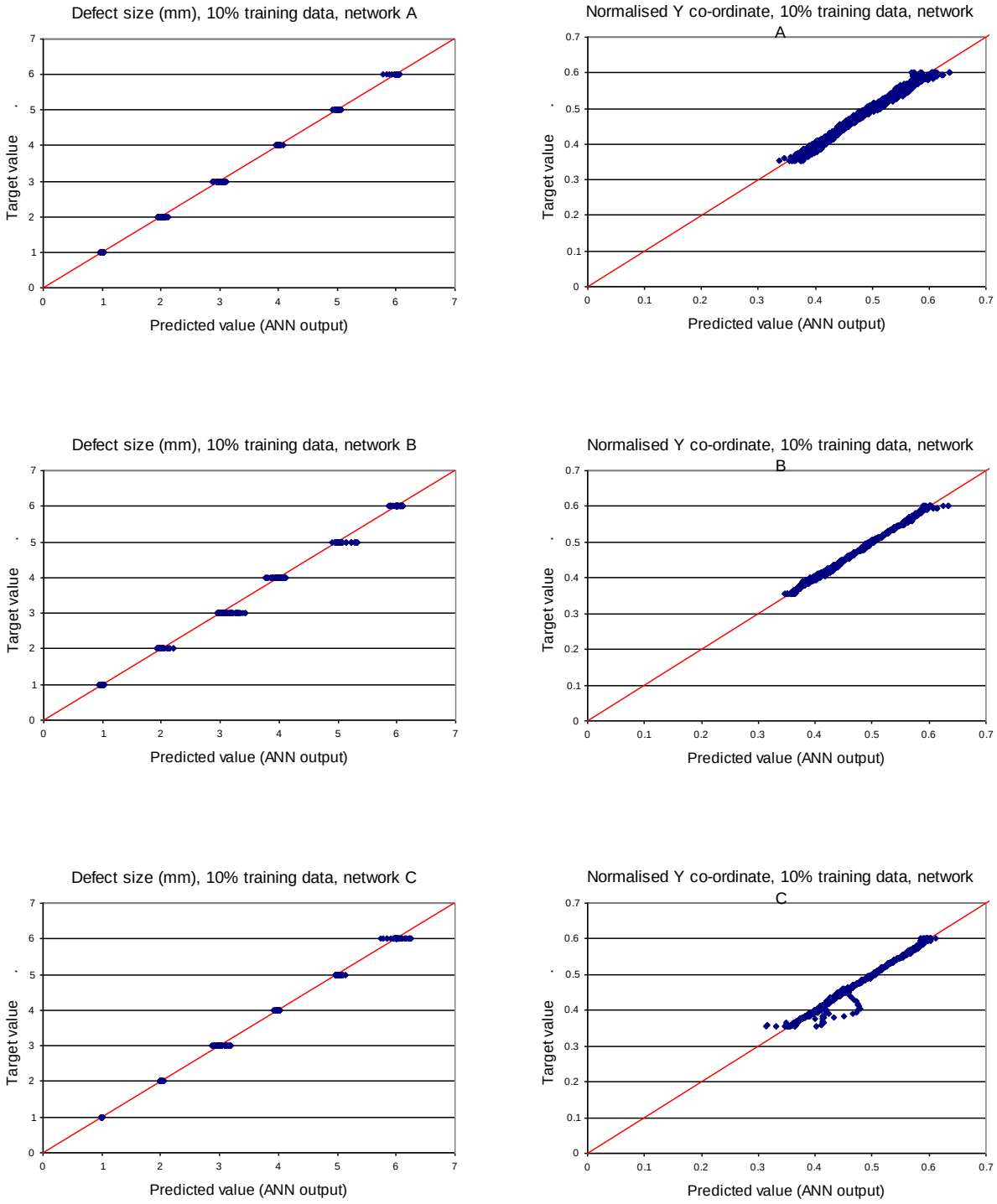


Figure 4.28 ANN performance with network configuration (10% training data)

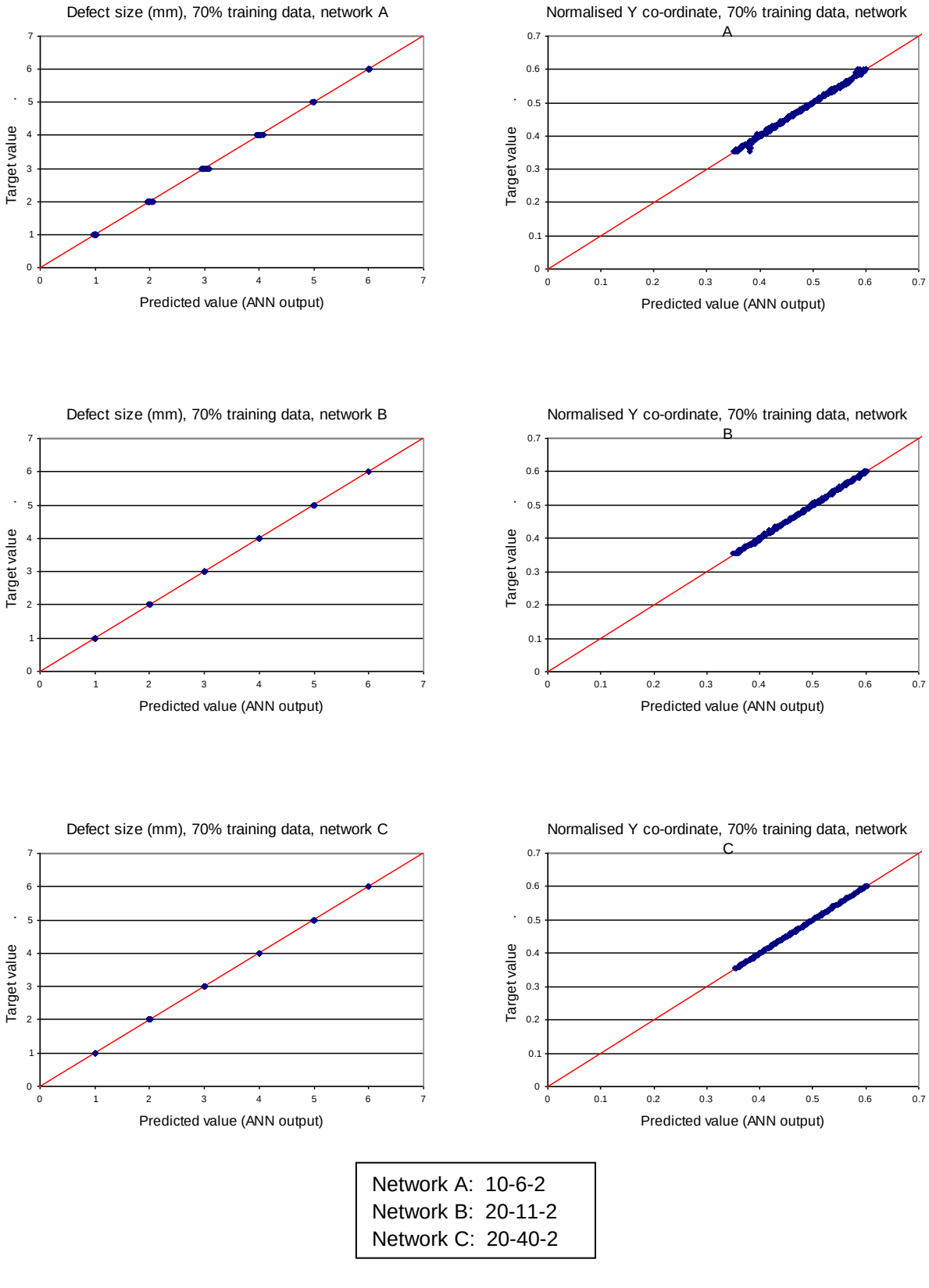


Figure 4.29 ANN performance with network configuration (70% training data)

4.4.3.3 The effect of added noise

The addition of noise to the data sets presented to previously trained ANNs reduced the accuracy of the ANNs' predictions in a roughly linear manner. Figure 4.30 shows a typical ANN output when presented with noisy input data, in this case using network configuration B which was trained with 90% of the available (noise-free) data. As the networks were trained with noise-free data, it was expected that the accuracy would decrease in proportion to the amount of noise added to the assessment data set, which was generally true for all cases investigated. Figure 4.31 shows 3D plots of mean deviation against added noise and amount of training data for all cases. It can be seen that with low amounts of added noise, all networks perform very well, but as the amount of noise increased the performance deteriorated significantly. It is also noticeable that the ANNs returned more accurate predictions for defect depth than for defect size. This can be explained by the use of normalised values, which express all values as a fraction of the maximum value. As the maximum value of defect depth is 200mm and the maximum value of defect size is 6mm, it follows that a normalised deviation from the target will not represent the same distance for each case. As can be seen from figure 4.31, the deviation of the defect size is in the region of 20 times the magnitude of the deviation of the defect depth. As the maximum value for defect depth is around 30 times the maximum value for defect size, it is clear that the data presented to the ANNs contain roughly the same degree of resolution for both size and depth if the absolute deviation in terms of distance is used as a measure.

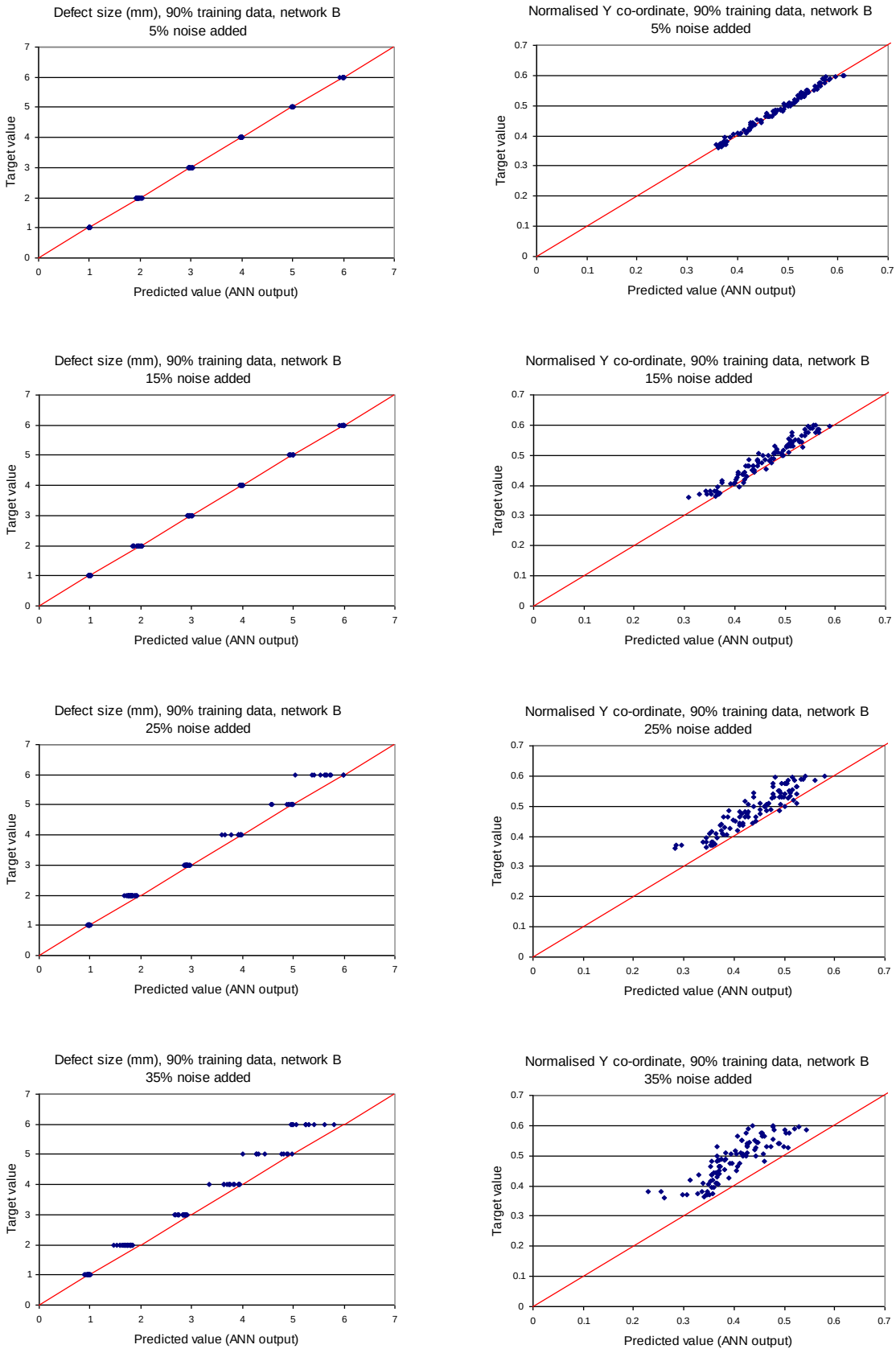


Figure 4.30 ANN performance with amount of added noise (90% training data)

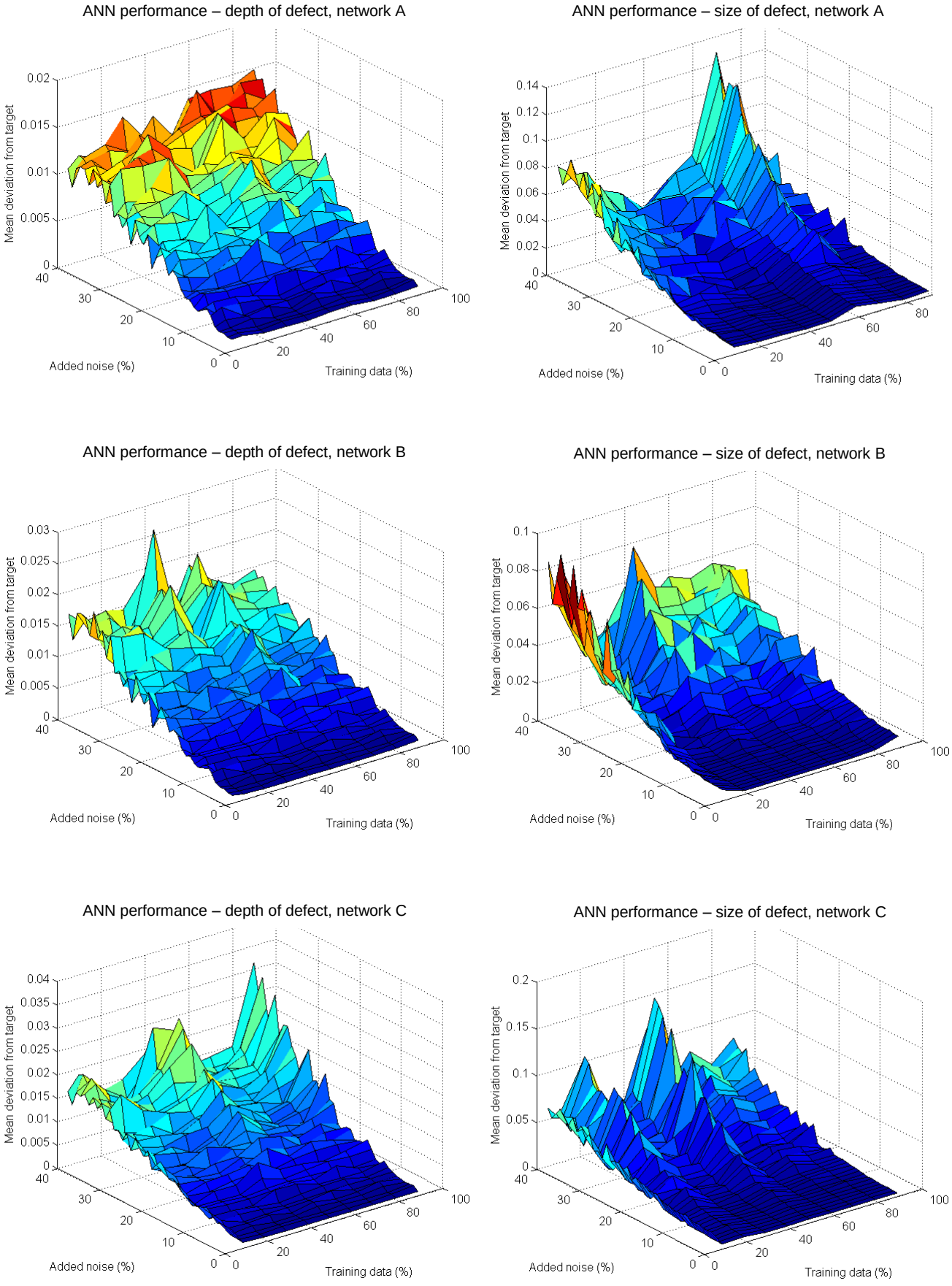


Figure 4.31 ANN performance with added noise and training data

4.4.4 Summary of ultrasonic method

A method has been presented to extract features from ultrasonic signals and use ANNs to process these features to return information regarding the position and size of a defect. In this case, the ANN was operating as a simple function approximator, drawing links between the time and height of the first peak of the reflected pulse and the depth and size of a defect. Although this was not a particularly demanding role for an ANN (the relationships between time and depth, and height and size, are roughly linear), it was considered an important step in the process of creating a complete ANN-based system that could interpret ultrasonic data for a variety of cases and return information regarding the size and location of defects.

4.5 Summary of single defect study

Three different methods of locating defects with the aid of artificial neural networks have been presented in this chapter: an impact method, modal analysis, and an ultrasonic pulse-echo method. Each method has been shown to be able to produce sufficient data to train and assess ANNs, that can subsequently provide information on a defect's position and/or size. In practice, the ultrasonic pulse-echo method seemed most appealing as a simple and rapid testing method, as a single transducer can be used to inspect an entire component provided the geometry is simple. The impact method was also considered to be worth pursuing, although it was anticipated that multiple sensors may be required to achieve enough resolution to describe the stress wave's motion through the component. Although the modal analysis method produced excellent results, the cost of equipment and set-up in order to measure natural frequencies in engineering components to a suitable degree of accuracy was considered prohibitive. The modal analysis method's usefulness was considered to lie more

with the inspection of large structures, or to locate larger defects such as areas of delamination in composites, thus it was not pursued any further.

The effect of ANN layout has been shown to make an obvious difference to the performance of the system, but it must be borne in mind that the accuracy of the ANNs' predictions is directly related to the quality of the data presented. Highly noisy data from experiments could lead to erroneous outputs from the ANNs, although the ANNs used in this chapter have been shown to be reasonably robust when presented with noisy data.

CHAPTER 5

EXPERIMENTAL IMPLEMENTATION

5.1 Introduction

Following the success of the ANN-based systems described in chapter 4, experimental testing was carried out using the impact and ultrasonic methods. Both of these methods were attractive in terms of being simple tests, from which data could be gathered and passed to an ANN system which would provide as much detail as possible regarding any defects present.

5.2 Impact method

The impact method aimed to reproduce the numerical simulation by applying an instantaneous pressure to the end of a steel bar by means of an impact from a mass attached to a pendulum. This section describes the experimental procedure and presents the results.

5.2.1 Experimental procedure

A mounting jig and pendulum were manufactured in the Oxford Brookes University School of Technology workshop, shown in figure 5.1 holding a sample bar. The jig was designed to clamp the bar in all degrees of freedom at the far end (in a similar manner to the 'encastre' boundary condition in the numerical model in section 4.2.1), and to hold it in the correct position so that the pendulum would hit the end in the correct orientation, that is, along the length of the bar.

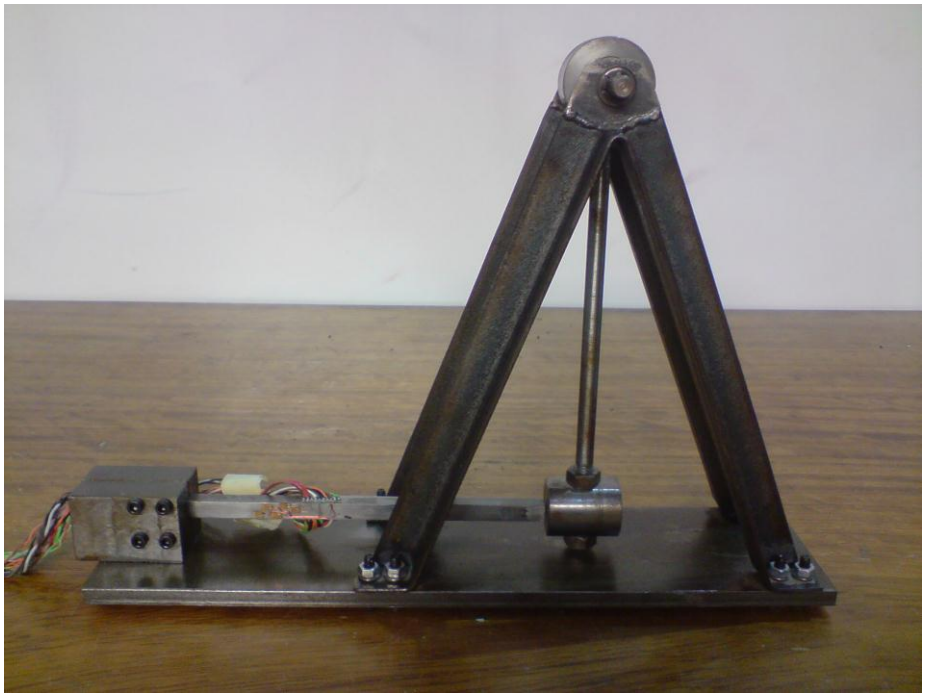


Figure 5.1 Mounting jig and pendulum for impact test

The mass of the pendulum was 200g, which was considered sufficient to introduce a stress wave of meaningful magnitude when hitting the end of the bar after being dropped from the vertical position. The vertical position in this case refers to the position where the pendulum was raised to its maximum height. As the pendulum was mounted on roller bearings, starting the pendulum swing from the vertical position with no velocity was considered to be a repeatable initial condition that would give a consistent impact force on the specimen.

Three bars were prepared with strain gauges at the sensor locations described in section 4.2.1, and a single hole of 1mm diameter was made in two of the bars. The remaining bar was kept as a non-defected control specimen. Some difficulties arose in placing strain gauges close enough together to lie on the exact points measured in the numerical model due to the size of the gauge backing. This was overcome by mounting the gauges slightly off-centre, meaning

they were placed at the correct distance from the end of the bar. Figure 5.2 shows a gauged bar, where the staggering of the gauge positions can be seen.

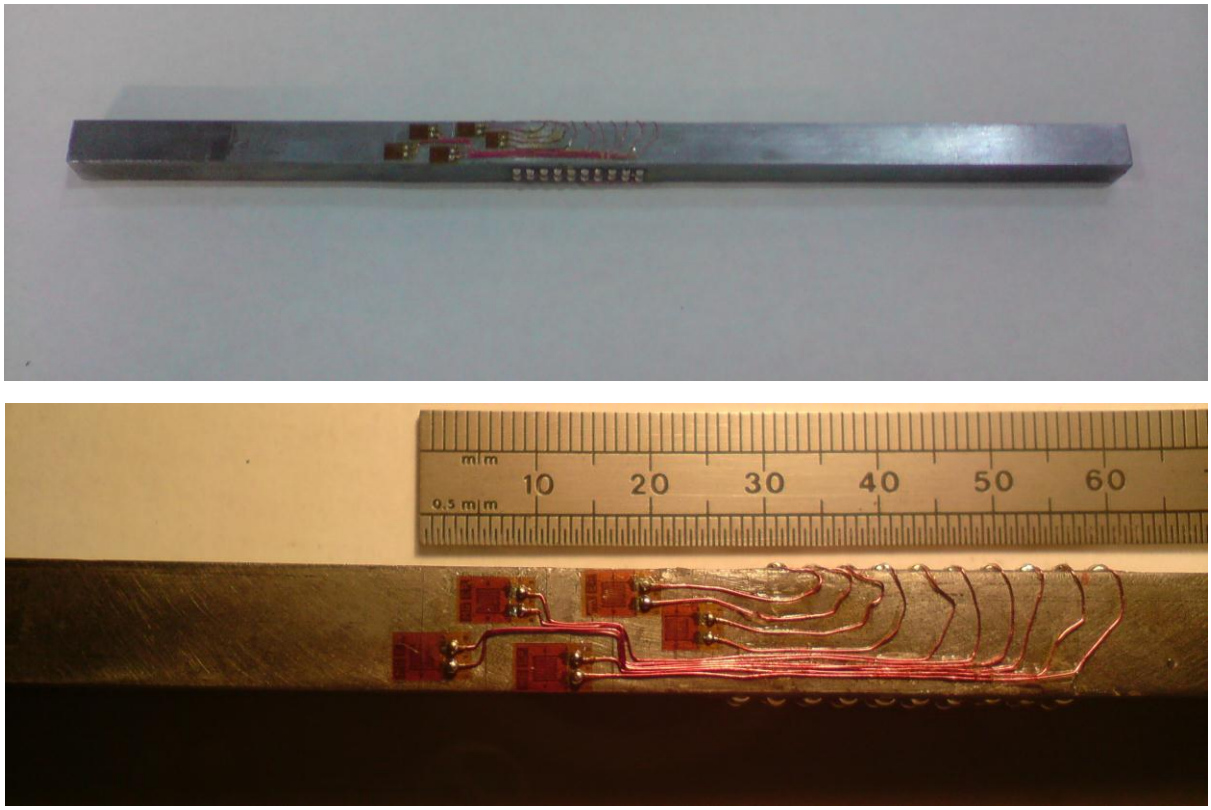


Figure 5.2 Test specimen with strain gauges

The strain gauges used were Vishay EA-06-060LZ-120/E temperature compensated units, with an active length of 0.06in (1.52mm) and a resistance of 120Ω. The strain gauges were connected to a Fylde H359-TA high speed strain amplifier in order to obtain a signal of a sufficient magnitude. Each gauge was connected as a quarter Wheatstone bridge, which was balanced before the outputs from the amplifier were connected via a PCI card to a laptop PC running LabView 7.1. A LabView program was created to record voltage outputs from the strain amplifier at a sampling rate of 500kHz, which was triggered by the first gauge's amplified voltage exceeding 0.2V. The experimental equipment schematic is shown in figure 5.3.

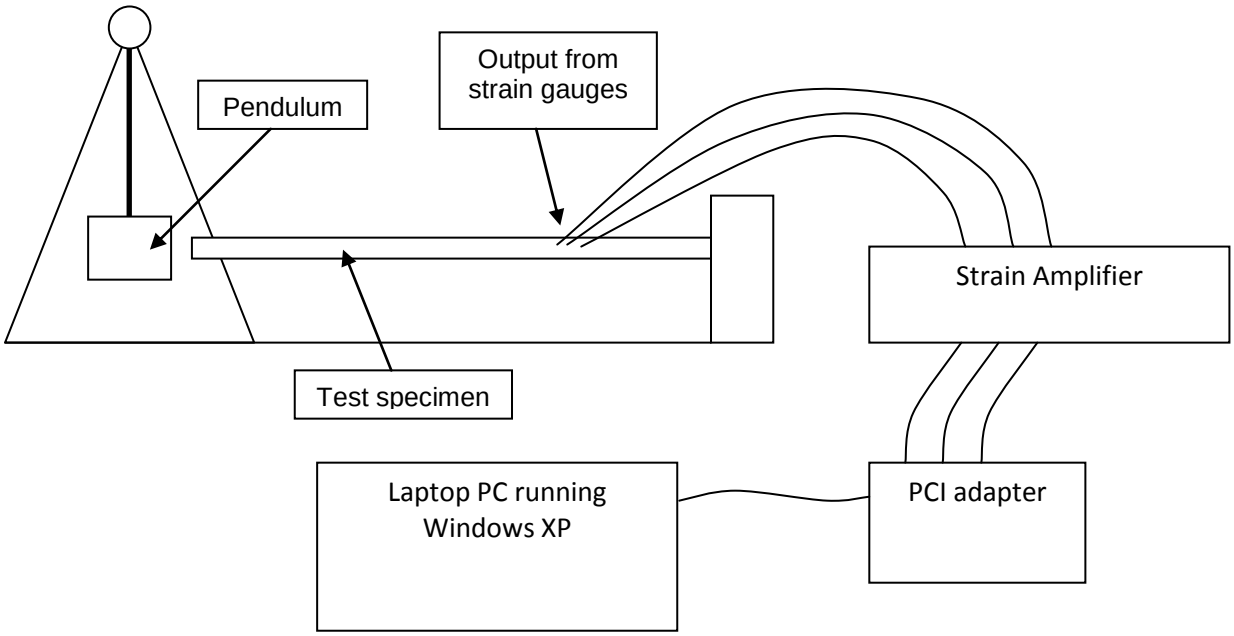


Figure 5.3 Experimental apparatus for impact method

Difficulty was initially experienced in balancing the bridges in the strain amplifier due to voltages drifting. Following investigation it was found that the internal resistors were not of a high enough tolerance for the application. By replacing these resistors with units of a much more strict tolerance, the problem was solved and results could be recorded. Additionally, difficulty was experienced in achieving communication between the PCI interface and the LabView software. Although 8 channels were available to log data, it was not possible to record these data simultaneously, thus only 4 channels could be used at any one time. This was not considered to be too big a problem, as the finite element model could be adapted to give the nodal displacement at fewer points but over a time window rather than just at one particular time instant. In this manner, it was reasoned that the data from experiment could be compared to the numerical data.

Each channel of the strain amplifier was trimmed to provide the same level of amplification for each gauge. This was achieved by placing a series of known loads on the bar and ensuring the output voltages were the same across all gauges for each load. The voltages were zeroed before testing took place, thus all voltages recorded were directly proportional to the amount of strain present.

5.2.2 Results from impact method

The results from the experiment were initially encouraging. It was clear that the stress wave's influence on the bar's shape could be measured using strain gauges, and the dilatation of the bar was clearly visible. Figure 5.4 shows a typical response from four gauges.

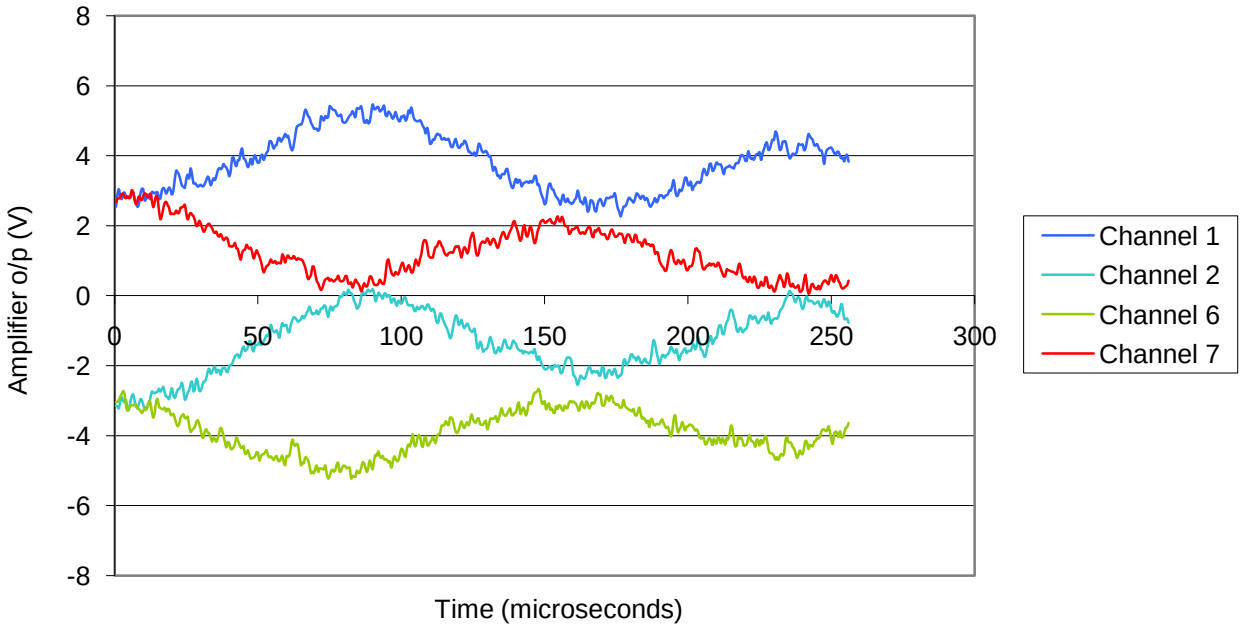


Figure 5.4 Typical output from bar with no defect (4 channels)

As can be seen from figure 5.4, the wave travelling through the bar is symmetric in that the bar is straining outwards as the wave passes. Channels 1 and 2 are on one side of the bar, and 5mm apart, and channels 6 and 7 are in the same longitudinal position on the opposite side of the bar. Despite the promising nature of this method from initial testing, it was discovered that repeatability of results presented significant difficulty. Figure 5.5 shows the variation of readings for identical tests

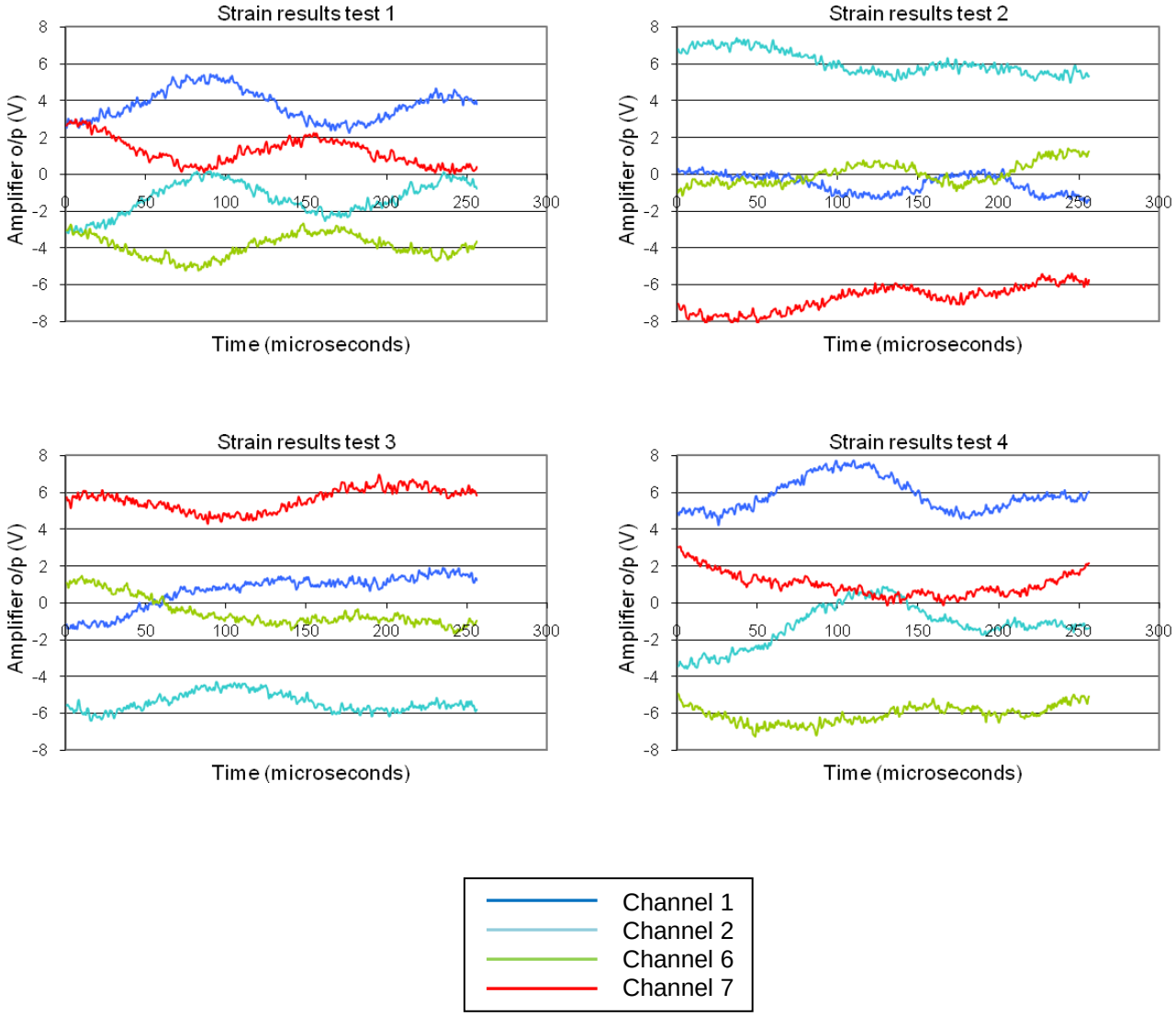


Figure 5.5 Variation in strain gauge outputs for identical tests (bar with no defect)

5.2.3 Summary of impact method

Despite many efforts to increase the repeatability of the experimental impact test, it was considered to have too much variation to be able to directly compare the results with those generated numerically, thus the experimental data were not able to be presented to the ANNs for validation. Typically, in published work using an impact method, small displacements are measured using a scanning laser vibrometer (Ishak, Liu *et al*. 2002), a technique which was not available for the purpose of this investigation. Due to the variability of the experimental results and the lack of a sufficiently accurate measuring method, the impact method was not pursued any further for the course of this thesis.

5.3 Experimental ultrasonic method

Following the difficulty in achieving repeatable conditions using the impact method, attention was given to the ultrasonic method, as described in section 4.4. Experimental apparatus was set up in the Stress Analysis Laboratory at Oxford Brookes University in order to validate the findings of the ultrasonic simulations. Experimental data were preprocessed to locate the time and height of the first peak of the reflected waveform, then normalised and calibrated before being presented to ANNs trained with FEM data.

5.3.1 Experimental procedure

Due to limitations on available equipment a higher ultrasonic frequency was used in experiment than in the simulation. In the simulation, a frequency of 100kHz was chosen as it kept the frequency.thickness product of the system below 1MHz.mm as detailed in section 4.4.1. The available testing equipment operated at a frequency of 5MHz, but was considered

valid based on the data that needed to be gathered from the test. As the chosen method used the time taken to reach a defect, and the magnitude of the reflected wave from the defect, it was reasoned that any discrepancy in wave velocity between the simulation and the experiment could be dealt with by calibration, i.e.

$$c_{simulation} = k.c_{experiment} \tag{5.1}$$

where c is the velocity of the longitudinal stress wave and k is a constant to be determined during calibration. In a similar way, using this method of calibration it would be possible to conduct experiments on different materials and correlate these results to the ANNs trained on a steel component using equation 5.1 to correct for the variation in the speed of sound in the material. In order to minimise the effect of variable coupling conditions in the experimental situation, the amplitude of the wave reflected from the defect was expressed as a percentage of the amplitude of the backwall echo (Zgonc and Achenbach 1996) before being calibrated against the data collected by FEM, and subsequently normalised for presentation to the ANNs.

A Socomate USPC-3100 flaw detector card was used in conjunction with a laptop PC running LabView 7.1 in Windows XP. The transducer used was a NDT Systems Nova type C11 with a centre frequency of 5MHz and an element diameter of 0.375" (9.5mm). This transducer was used as both a transmitter and receiver in the flaw detector's pulse-echo mode. The LabView program that was supplied with the Socomate hardware was not able to write the data to a file, so a modification was made to the software to locate the array used to display the graph, and output this as a data file. A screen shot of the modified LabView software is shown in figure 5.6.

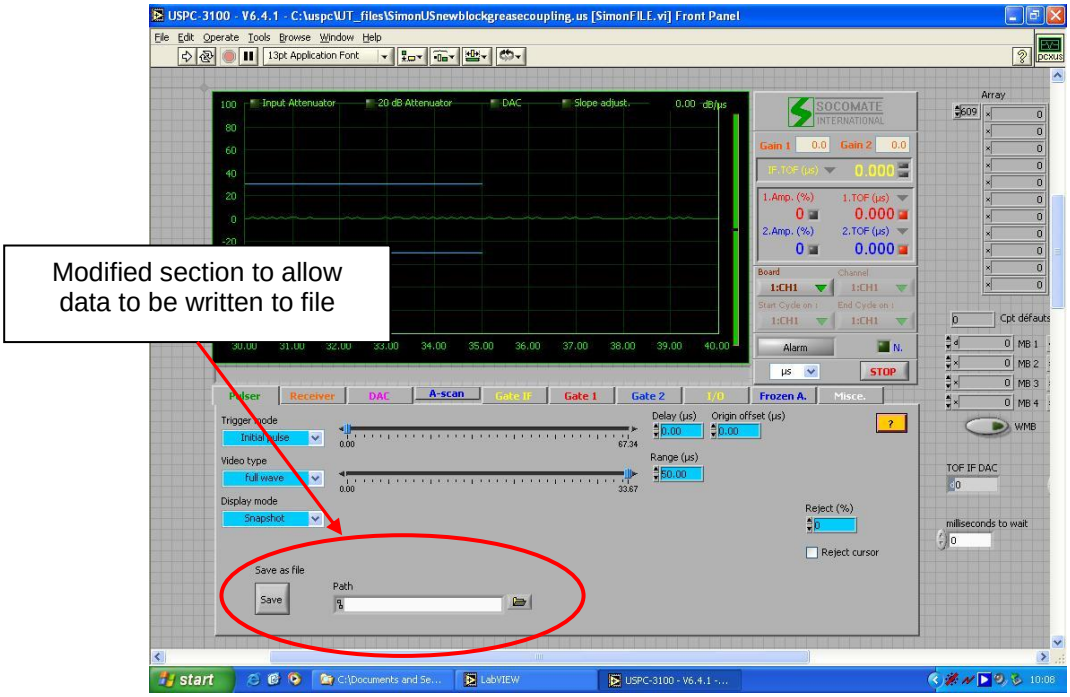


Figure 5.6 Screenshot of modified LabView program

A series of steel plates were machined to size (100mm x 200mm x 10mm), and in each plate a hole was drilled through the thickness at a specified location as shown in figures 5.7 and 5.8. The transducer was coupled to the top edge of the plate using water-based ultrasonic coupling gel and held in place by resting a small weight on the top as shown in figure 5.9. (Kumar, Gupta *et al.* 2005).

The reflected waveform was recorded in tabular form as a generic data file, then imported into MATLAB 7.1 where the algorithm previously developed for the FEM data (see section 4.4.1 and appendix D) was used to locate the time and height of the peaks in the signal reflected from the hole, and also in the reflection from the backwall. The heights of the peaks in the signal were expressed as a percentage of the height of peaks in the backwall echo in order to minimise the effect of variation in coupling conditions between tests.

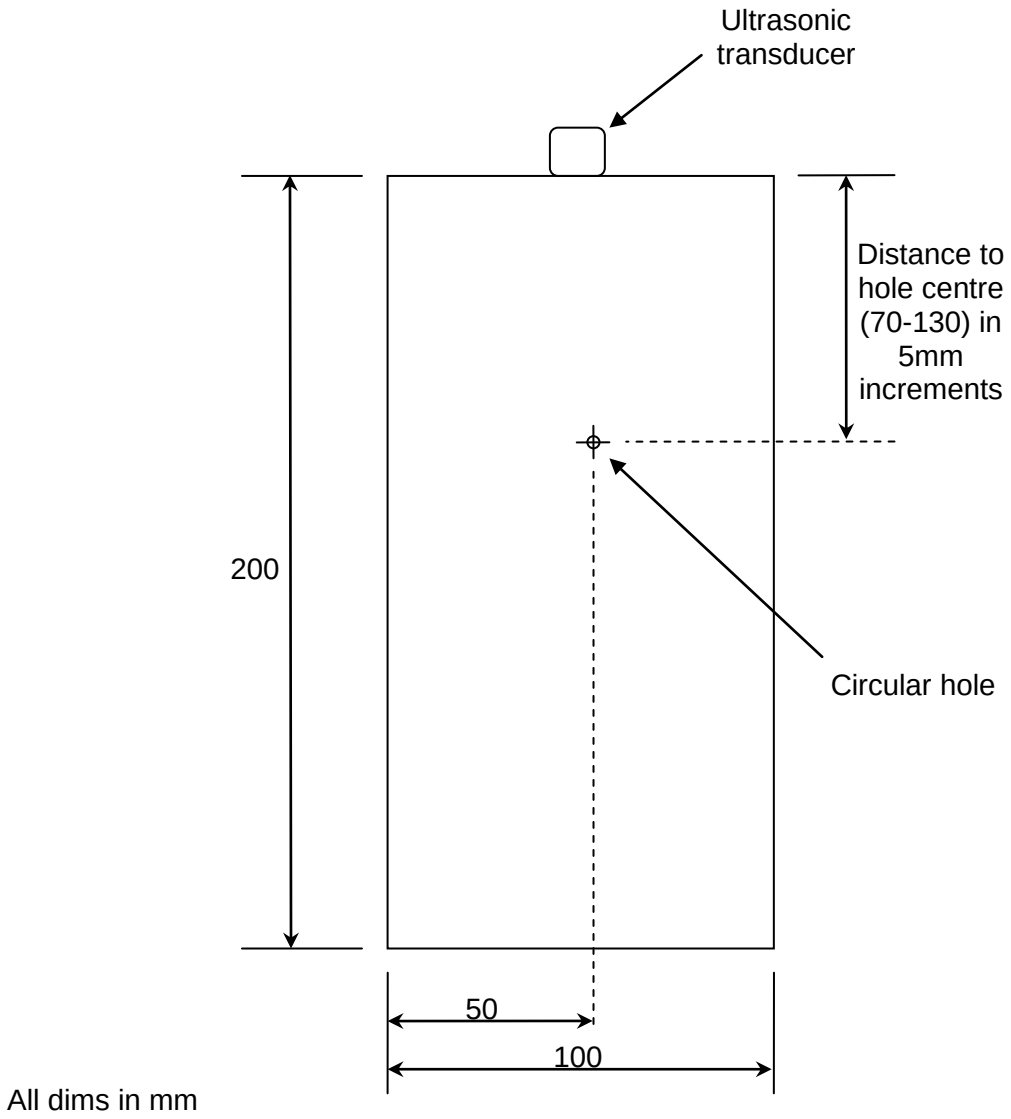


Figure 5.7 Steel plate dimensions for ultrasonic testing

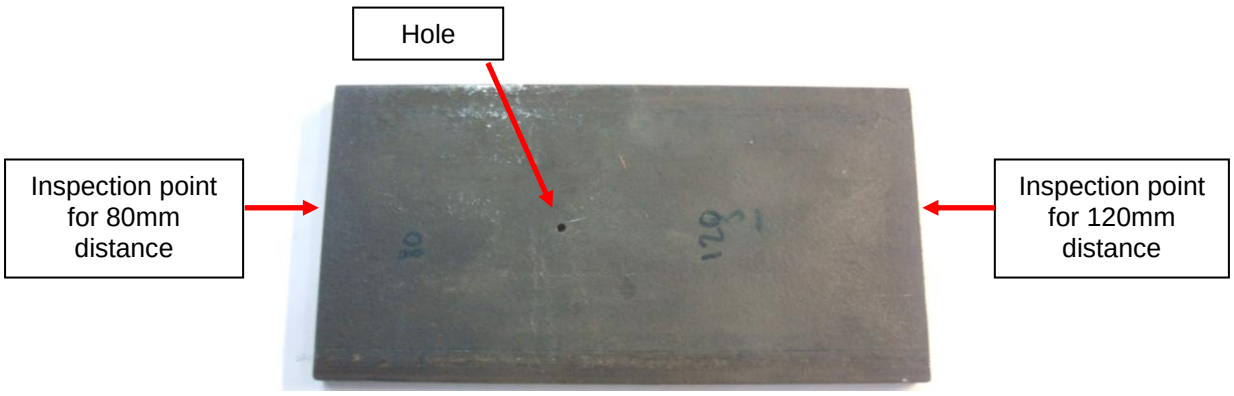


Figure 5.8 Test specimen for ultrasonic method, showing hole drilled 80mm from one edge and 120mm from the other edge

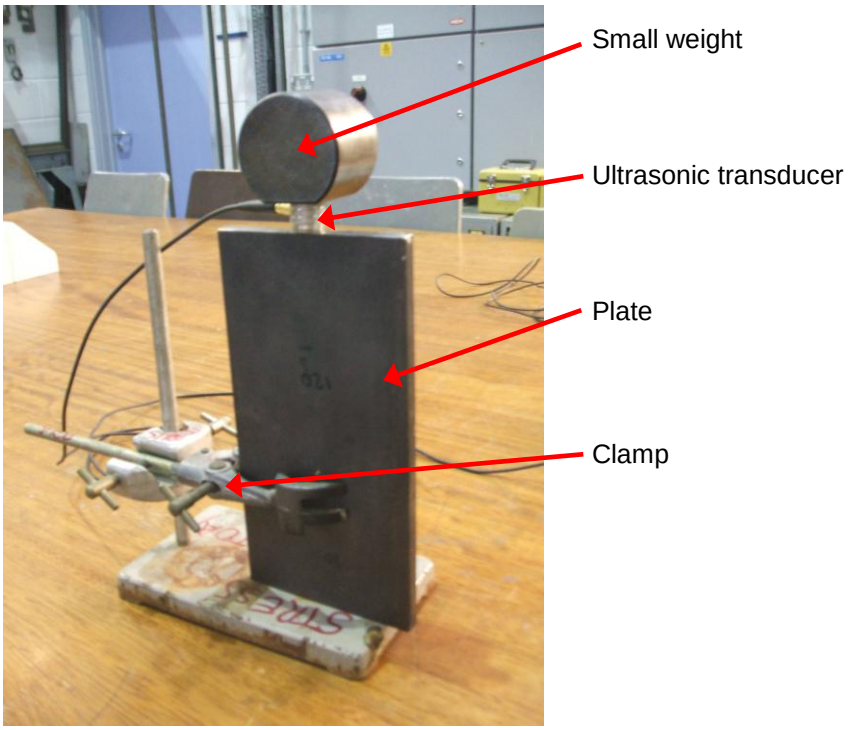


Figure 5.9 Ultrasonic transducer attached to specimen

5.3.2 Experimental Results

Experimental data were gathered using plates with holes of 1, 2 and 3mm diameter at distances of between 70 and 130mm from the edge of the plate, in increments of 5mm as shown in figure 5.7. Some difficulty was observed in ensuring the amplitude of the reflected ultrasonic wave was reasonably consistent, mainly due to the surface finish on the edges of the test pieces. By using the gain control on the Socomate software it was possible to compensate for this, and a satisfactory signal to noise ratio was achieved across all cases. Typical waveforms from the experimental testing are shown in figures 5.10 to 5.12, where the overview, close-up of the reflection from the defect, and the backwall echo can be seen. What is apparent from the overview is that there are multiple reflections from the hole. This was at first thought to be caused by reverberation of the column of air in the hole as the wave

hit it, but following subsequent investigation it transpired that additional wavefronts were being generated by the presence of head waves (Fan 2010). In order to examine this phenomenon, the finite element model described in section 4.4.1 was modified to operate at an ultrasonic frequency of 5MHz. This required an element size of 0.1mm in the x and y directions to provide around 12 elements per longitudinal wavelength, and a time step of 10^{-8}s in order to satisfy the Courant-Friedrichs-Lewy condition (Courant, Friedrichs *et al.* 1967), thus the memory requirements for the simulation became significantly higher than the simulations at 100kHz. Due to the higher frequency used in the experimental testing, the frequency.thickness product of the system was 50MHz.mm, meaning that many modes and different group and phase velocities could be present, as explained in section 3.3.1.3. In addition to the dilatational and distortional waves, head waves were also created by the interaction of the longitudinal (dilatational) waves with the surfaces of the plate, thus creating additional incident wavefronts that hit the hole at a time shortly after the initial wavefront. Figure 5.13 shows the development of head waves from the finite element simulation. It can be seen that the head waves cause a series of wavefronts across the thickness of the plate that repeat with regular frequency and, after the first head wave, decrease in intensity as they repeat. These wavefronts, when they arrive at a defect, will all send a reflection back to the ultrasonic transducer as can be seen in figure 5.10. In the case of simulations at 100kHz, the head waves were not generated, thus the phenomenon was not observed.

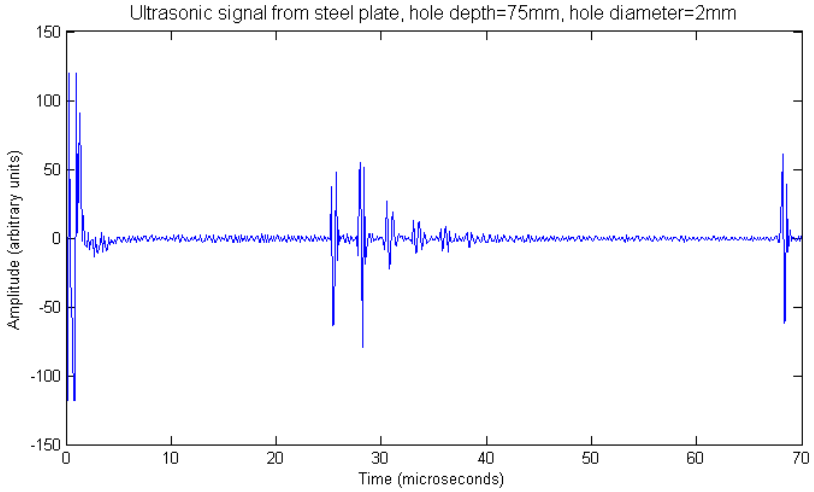


Figure 5.10 Typical ultrasonic signal from experimental testing

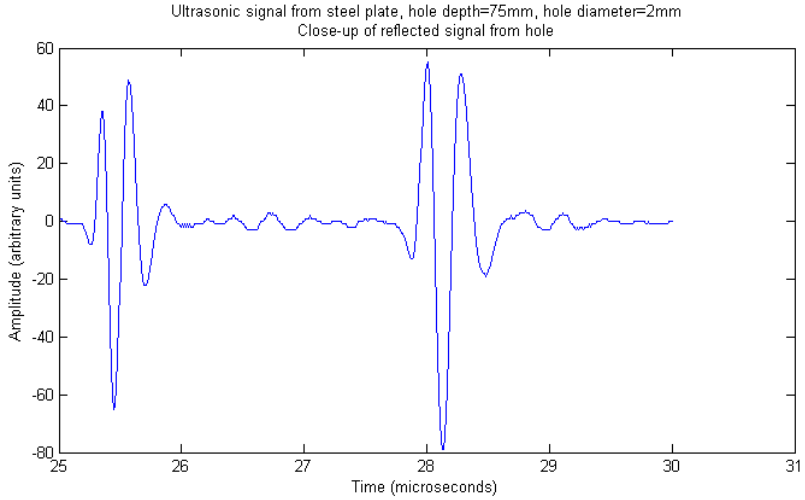


Figure 5.11 Close-up of reflected signal from hole showing incident wave (left) and first head wave (right)

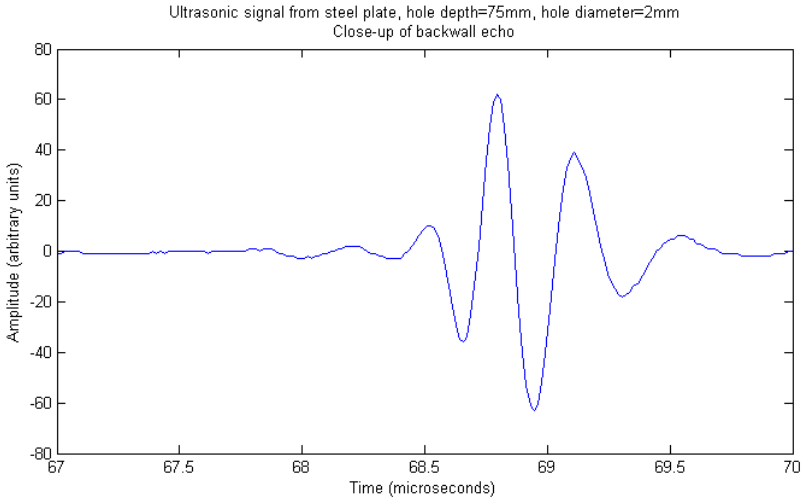


Figure 5.12 Close-up of reflected signal from backwall

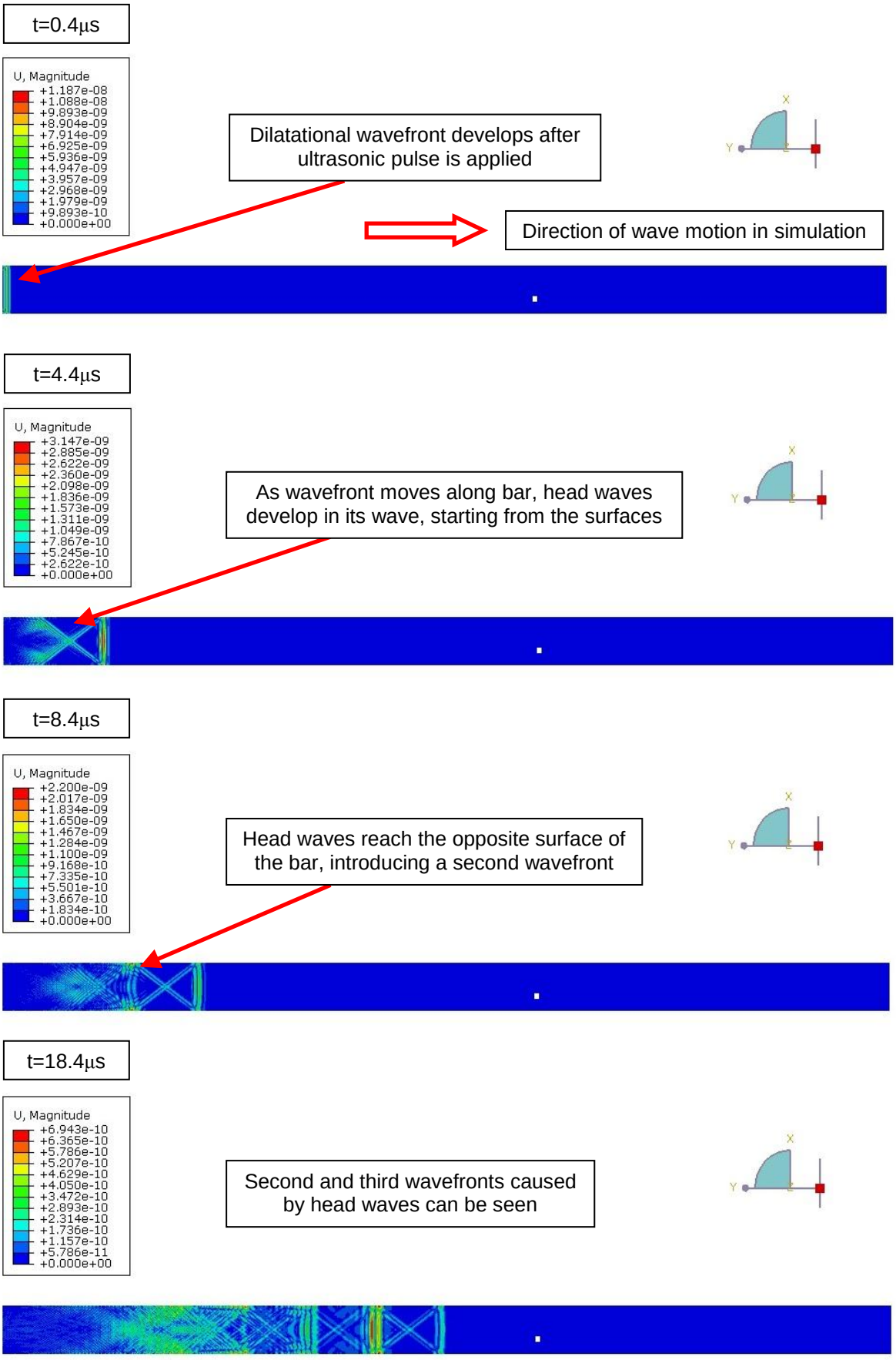


Figure 5.13 Development of head waves – simulation at 5MHz

5.3.3 Data processing for ANNs

The data gathered from experimentation were imported into MATLAB 7.1, where the peaks were located using the algorithm developed by the author in section 4.4.1 and included as appendix D. As with the FEM simulation of an ultrasonic pulse, the time and the height of the first reflected peak were intended to be used to represent the case to the ANNs. However, due to the variation in coupling conditions between cases it was decided to measure the height of each peak in the first reflected signal and to compare these with the heights of each peak in the backwall reflection. A mean value was then taken in order to express the defect signal's height as a percentage of the backwall echo. Note that the backwall reflection was the inverted waveform of the reflection from the defect, so the magnitude of each peak was used to ensure all values were positive. Figure 5.14 shows the variation in peak height with defect depth and diameter. It can be seen that although there is a spread of points in the data set, the data generally lie in three discrete groups.

Much more attenuation was observed in the experimental results than in the FEM simulation. Halmshaw (1991) states that if the wavefront is assumed to be spherical (or, for a 2D simulation, circular), the intensity of the wave varies inversely with the square of the distance travelled, whereas for a plane wavefront the intensity varies linearly with distance. The numerical model assumed plane strain conditions in the z-direction, i.e. the width of the plate, thus the wavefront could be considered plane as it was bounded by the two surfaces 10mm apart, and could not propagate in the z-direction. In the case of the experiments conducted, the wavefront could be considered spherical as waves could propagate in all three directions, thus the amplitude of the reflected wave would be expected to vary with distance to the defect according to the 'inverse square law'.

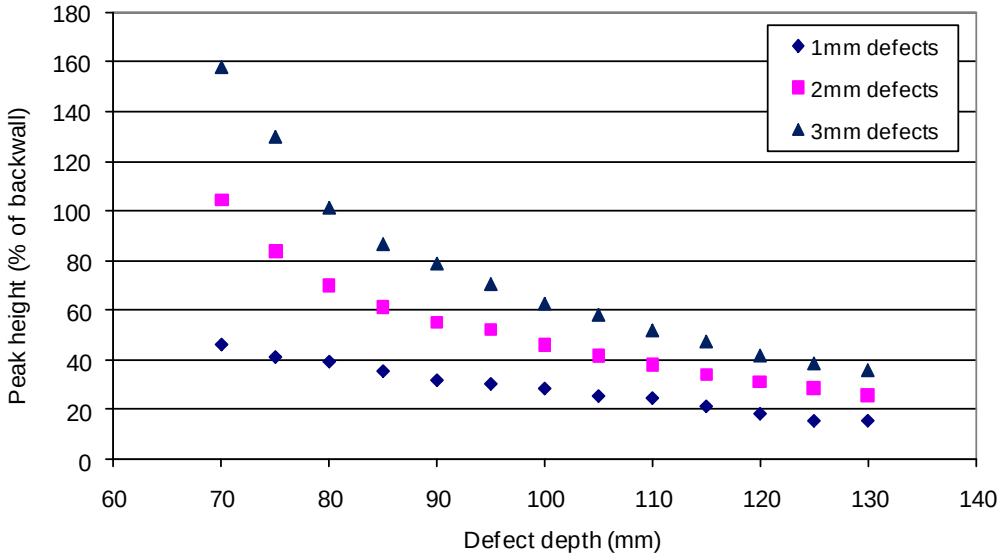


Figure 5.14 Peak height of reflected signal expressed as a percentage of backwall echo

The effect of attenuation in the experimental data was considered worthy of investigation, so an effort was made to compensate for this. It is important to note that the aim of this thesis was to determine whether an ANN trained with FE data can be assessed with experimental data, rather than to necessarily produce a completely accurate FE model. For this reason, a method of adjusting the experimental data to compensate for attenuation was preferable over refining the FE model and running all parametric simulations again.

The amplitude of an ultrasonic wave can be considered to be inversely proportional to the square of the distance travelled, although other effects such as scattering and absorption will also reduce the signal reflected by a feature (Halmshaw 1991). In order to determine the exact relationship between distance and amplitude in this series of experiments, the data were normalised by dividing the peak height by the defect size. These results were plotted on a graph, shown in figure 5.15, where it can be seen that the relationship between distance and normalised peak height is asymptotic. All the normalised data were seen to follow the same

trend. The equation of the trend line of this graph was found using the TRENDLINE function in Microsoft Excel 97, and the coefficients of the equation were then used to correct the peak heights to remove the effect of attenuation. As can be seen from figure 5.15, the amplitude of the reflection from each defect is proportional to the distance raised to the power of -2.08, very close to the value of -2 that would be expected using the inverse square law.

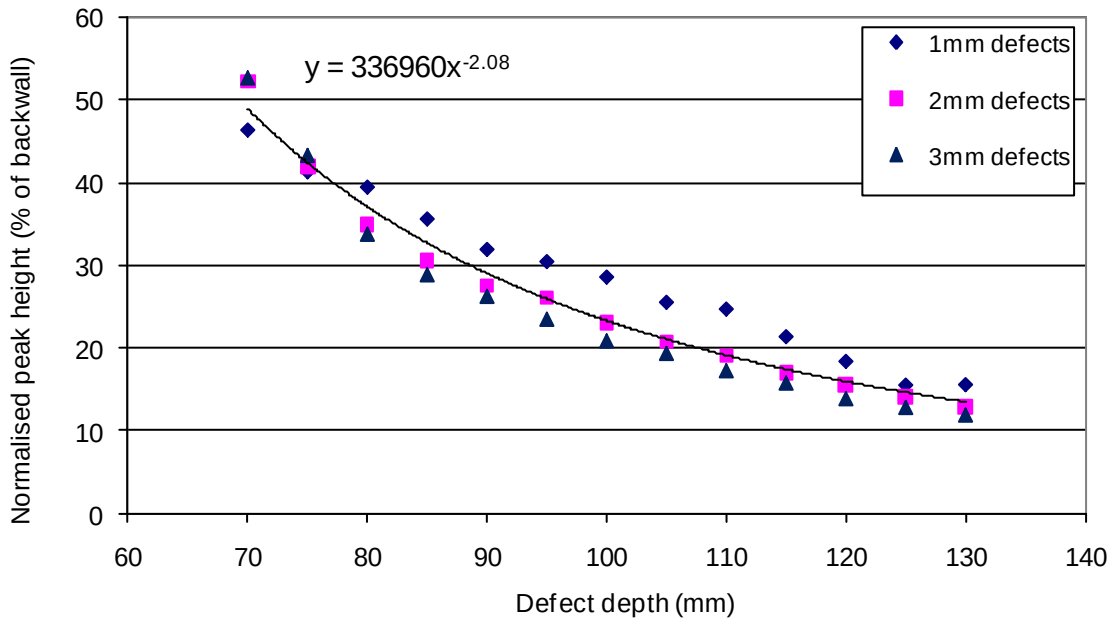


Figure 5.15 Normalised peak height of reflected signal (% of backwall echo divided by defect size)

The peak heights were corrected using equation 5.2.

$$h_{corrected} = h_{measured} \times \frac{d^{2.08}}{k} \qquad (5.2)$$

where $h_{corrected}$ is the peak height corrected for the effects of attenuation, $h_{measured}$ is the peak height from experiment, expressed as a percentage of backwall echo, d is the distance to the defect, measured in mm, and k is an arbitrary constant to scale the results.

The corrected peak heights are shown in figure 5.16, where it can be seen that, although the data for each defect size are not uniform in magnitude, distinct grouping of each size is clearly discernable.

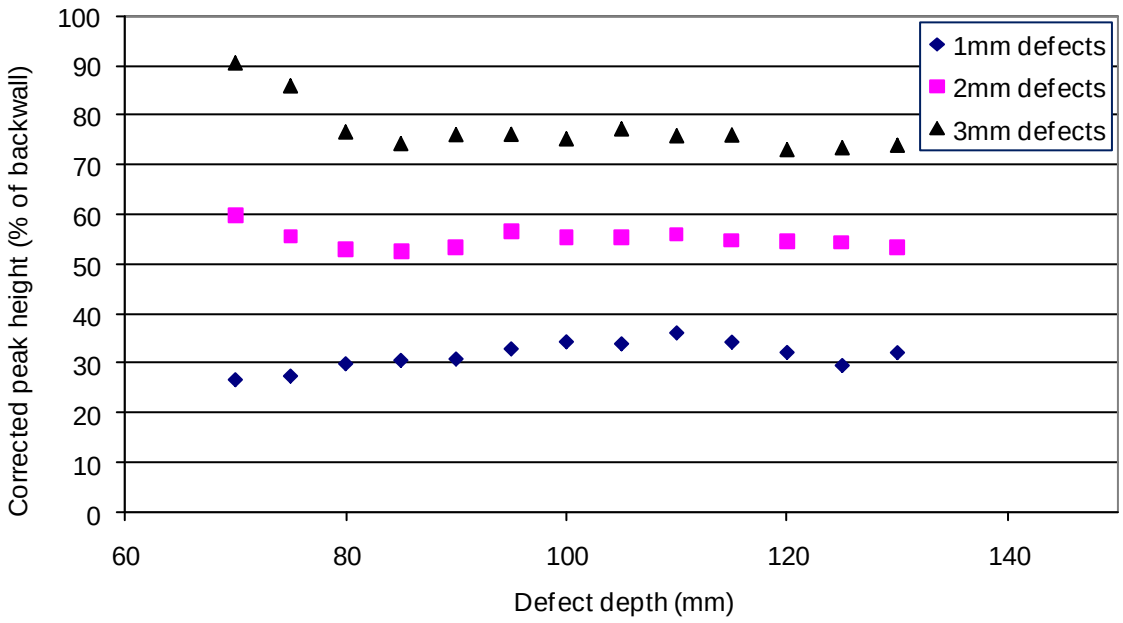


Figure 5.16 Corrected peak heights with defect depth

Once the time of the first peak and the corrected height of the reflected signal were determined, it was necessary to calibrate these against the data gathered using the finite element method so that they could be presented to the ANNs that had already been trained with the finite element data. Calibration was achieved by comparing the experimental data to similar cases generated using the finite element method, and determining a pair of coefficients that could be used to scale and shift the experimental data. This procedure was required as the speed of sound in the experimental specimen differed slightly from that used in the finite element model. Additionally, with the ultrasonic testing equipment used, the time at the start of recording was not the time that the initial pulse started to be generated, as was the case in the simulation. Calibrating in this manner meant that these effects were

negated. In the case of the peak time, this was done by plotting a graph of time (of the first peak in the reflected signal) for the FE data in the y-axis and the experimental data in the x-axis. The equation of a best fit line was then expressed in terms of

$$y = mx + c \tag{5.3}$$

where m is the scaling coefficient and c is the shifting coefficient. Figure 5.17 shows the relationship between the FE data and the experimental data. It can be seen that the coefficients in equation 5.3 are slightly different for each of the three sizes investigated, so a mean value for each coefficient was chosen based on all three data sets. It should be noted that the small spread of values between data sets is due to the diameter of the holes. The measured distance (see figure 5.7) was to the centre of the hole, whereas in practice and in the simulations the ultrasonic wave was reflected from the top of the hole. It can be seen that in the case of a hole of larger diameter the ultrasonic wave takes slightly less time to return to the transducer than for a hole of smaller diameter.

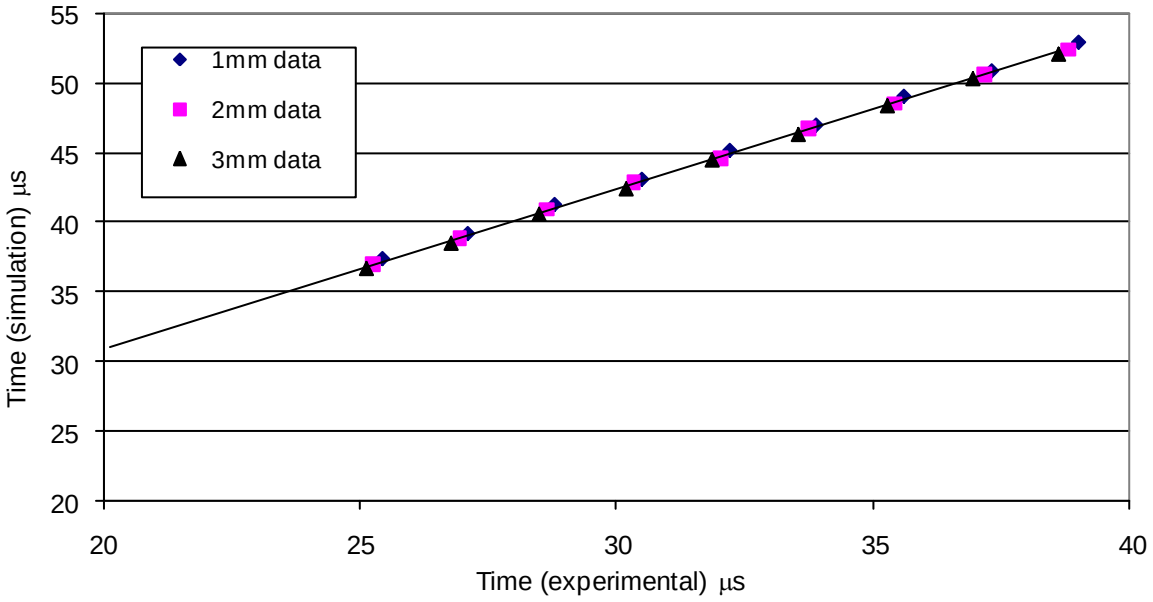


Figure 5.17 Calibration graph for peak time

Using the SLOPE and INTERCEPT functions in Microsoft Excel 97, the coefficients of the trend line were found to be m=1.15 and c=8. Using these coefficients meant that the peak times were directly comparable to the data gathered in the FE model of the pulse-echo inspection, as detailed in section 4.4, and could therefore be normalised in the same manner and presented directly to the ANNs trained with FE data.

For the peak height the procedure was similar, but there was a much greater variation in the results recorded during experimental testing. Figure 5.18 shows the final results for the peak height values for experimental data (on the x-axis) against data generated in the FE simulation (on the y-axis). It can be seen that even after scaling and calibrating there is a variation in the height of the reflected signal, although clear clustering of defect sizes can be seen.

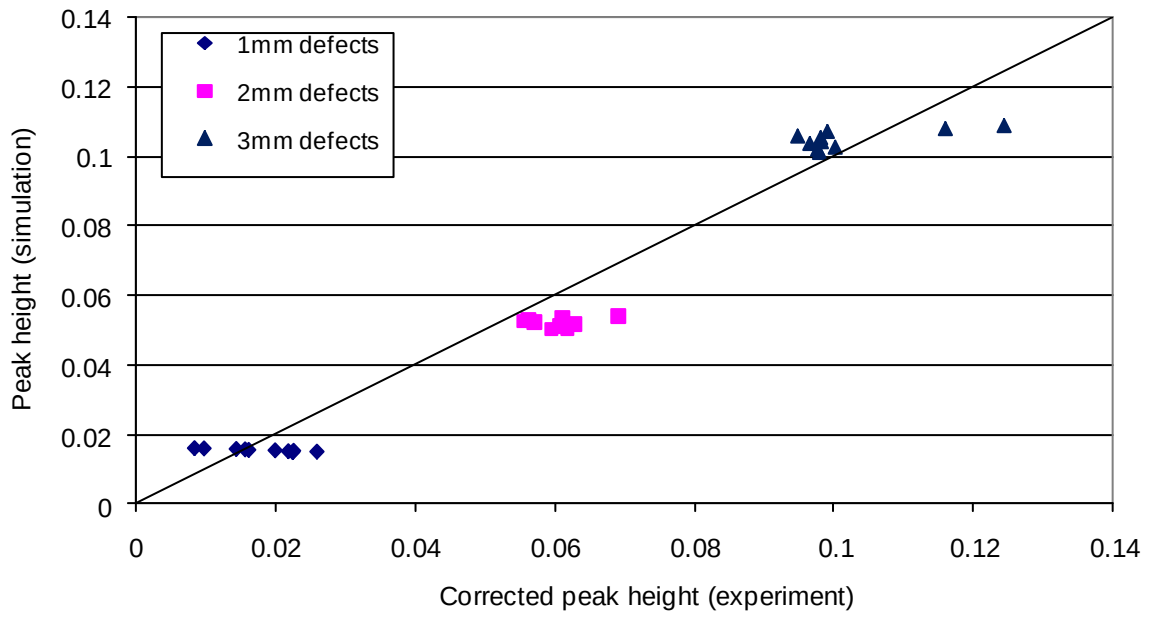


Figure 5.18 Comparison of peak heights from experiment and simulation

These results were normalised to lie in the range [-1, 1] using the same method as used for the data set generated using the FE simulation, and presented to the various ANNs.

5.3.4 Results from ANNs

The same ANN configurations were used for the experimental data as were used for the numerical data from simulations, as described in section 4.4.2. Networks trained with 70%, 80% and 90% of the FE data were used to assess the experimental data, and the results are presented in this section. As can be seen from figures 5.19 and 5.20, the ANNs generally performed well when trying to locate the distance to the defect, but not quite so well when trying to determine the defect's size. It is also apparent that the accuracy of the ANNs decreased rapidly at normalised defect depths of above 0.6. This was the maximum value used during training, and so the ANN algorithm would be extrapolating from the training data set above this value, hence the poorer results. In practice this issue could be overcome by providing a more comprehensive training set representative of defects in any position in the component under inspection.

In terms of the accuracy in estimating the defect's size, the results from the ANNs show clear clustering for each discrete defect size, and although there is a fairly large spread of estimations in most cases, most defects have been sized to within around ± 0.3mm of the target value.

From figures 5.19 and 5.20 it can be seen that when the ANNs were trained with 70% of the numerical data, accuracy in locating and sizing defects was significantly lower than when 80% or 90% of the data were used. This would suggest that 70% of the available numerical data was too little to allow the ANNs to draw sufficient links between input and output data and still be robust when presented with experimental data, thus the networks were undertrained. When 80% of the numerical data was used to train the ANNs, all three ANNs gave their best performance, indicating that the ANNs were neither undertrained nor overtrained with this amount of training data. With 90% of the numerical data used to train

the ANNs, networks A and C performed well in terms of estimating the distance to a defect, but reduced in accuracy when estimating the defect's size. This would suggest that overtraining had started to occur, albeit not severe at this stage. Network B performed better than networks A and C, but again showed the highest accuracy with 80% training data, reducing slightly in accuracy when 90% of the data was used for training.

From these results it can be inferred that between 70% and 90% of the numerical data must be used during the training process, and that the best of the three ANN layouts, based on the performance when presented with experimental data, was network B. This would suggest that, taking into account the range of values to be expected from an experimental situation, the architecture of network B was neither too small to allow meaningful links to be developed between input and target data during training, nor so large that overly complex links were developed between inputs and targets, reducing the ability of the ANN to generalise.

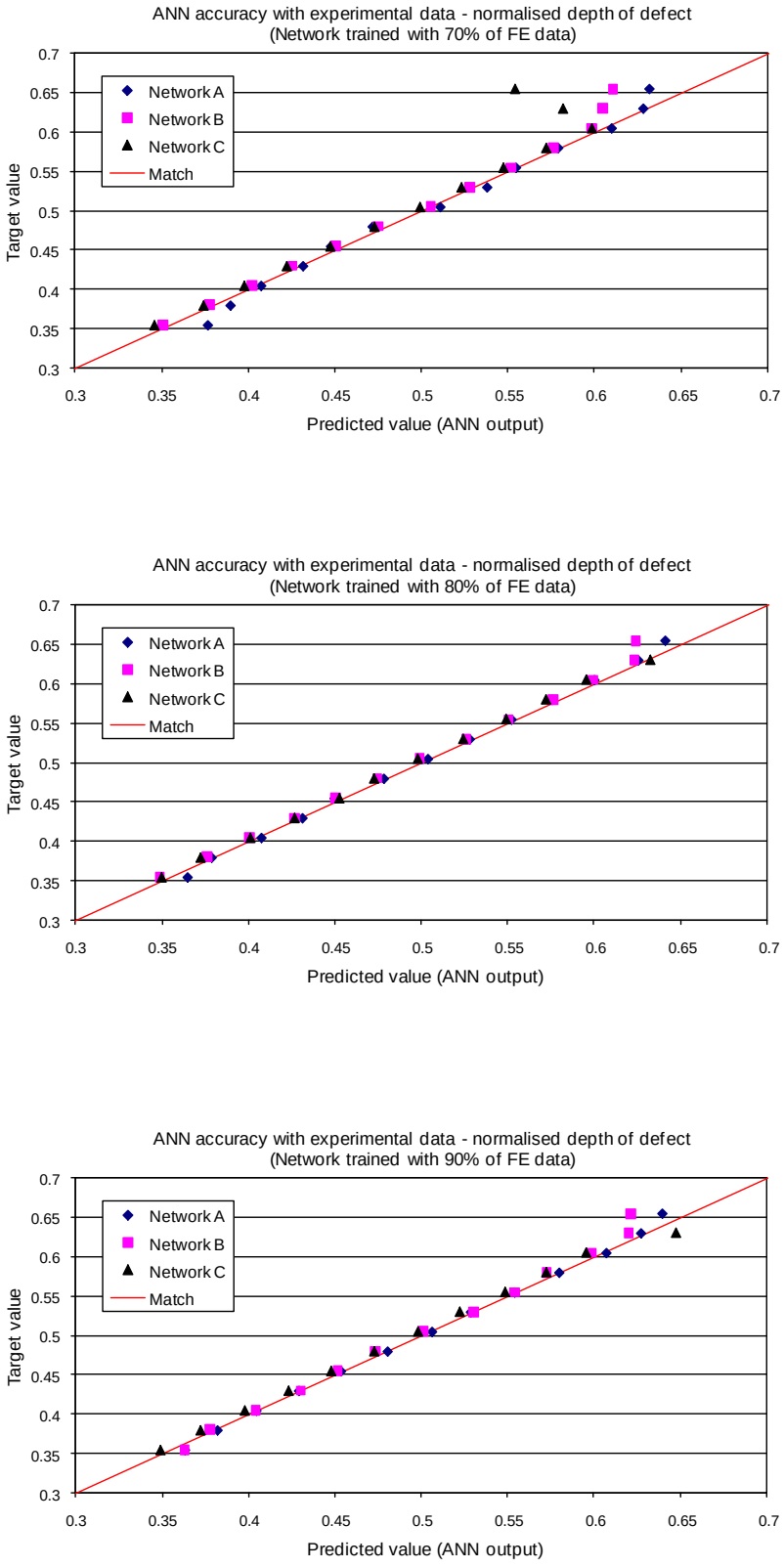


Figure 5.19 ANN results for depth of defect using experimental data

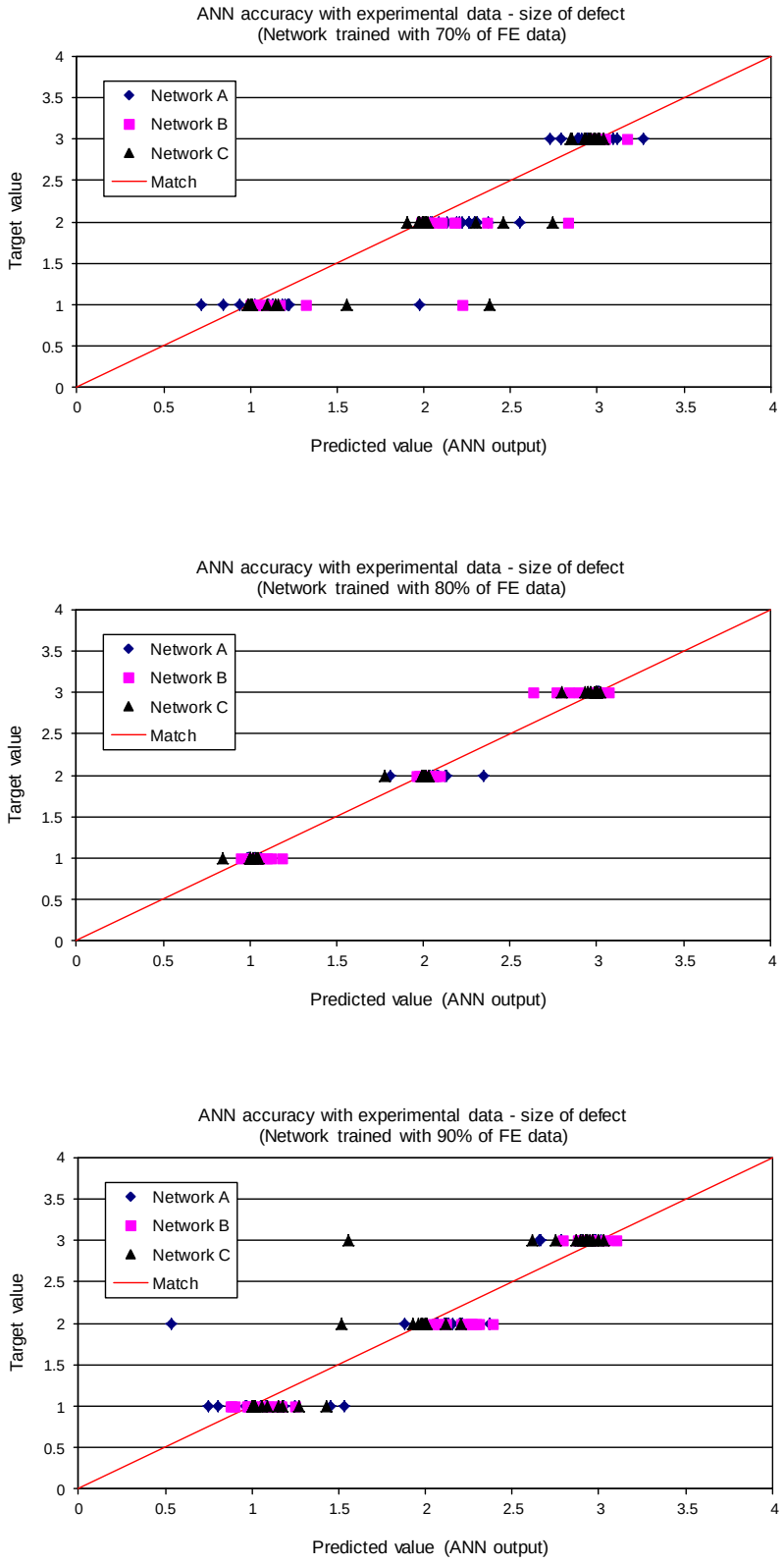


Figure 5.20 ANN results for size of defect using experimental data

5.3.5 Summary of ultrasonic method

Experimental testing has been carried out to validate the ANN method for sizing and locating defects using the ultrasonic pulse-echo method. Although in this case some preprocessing was required to compensate for attenuation in the test specimen, the experimental method produced very good results in the case of locating the defect's position. Published work indicates that a tolerance of 5-7% is acceptable for ANN-based systems (Thavasimuthu, Rajagopalan *et al.* 1996, Zgonc and Achenbach 1996); typical error of the ANNs used in this chapter is of the order of around 1-2% for locating the distance to the defect. As expected, accuracy was seen to decrease sharply when the ANNs were required to extrapolate from their trained values, as was the case for the highest normalised values in the estimations for the depth of the defect. In terms of estimating the size of a defect, the ANNs returned results with a typical error of around 3-5% when trained with 80% of the available FE data. Although this error increased to around 10-19% with the ANNs trained with 70% or 90% of the FE data, the results show that an ANN method for locating and quantifying defects can return results that are sufficiently accurate to be used for an automated system. However, care may need to be taken in the choice of ANN architecture and the amount of training data used; the training data set needs to be representative of the entire range of possible situations to ensure that the ANN is interpolating between previously seen cases when presented with previously unseen data. In the cases where the defect's size was not accurately estimated, the main cause for this was considered to be the variation in coupling conditions between different tests, despite great care having been taken to make the tests repeatable. With the equipment available it was not possible to investigate non-contact ultrasonic inspection, which could possibly offer a means of overcoming this issue.

5.4 Summary of experimental implementation

Two methods of experimental testing were investigated, using an impact and an ultrasonic transducer to create stress waves in a test specimen. The impact method was attractive due to its simplicity, but it was found that to measure displacement to a satisfactory degree of accuracy would require more advanced equipment than was available, which would mean that the method would cease to be so simple. For this reason, the impact method was not pursued further. Very good results were obtained using the ultrasonic method in terms of using ANNs to estimate the location of a defect, and acceptable results were obtained in the estimation of a defect's size. The method has produced repeatable results, and satisfies the criterion set out at the start of the thesis that a simple test should be able to provide information regarding the structural condition of a component. Although the variation in coupling conditions caused some issues with the testing equipment available, it is considered that the use of an EMAT or non-contact ultrasonic transducer would reduce this variation considerably, thus the method could be used for looking at more complex situations.

CHAPTER 6

MULTIPLE DEFECT IDENTIFICATION ANALYSES

6.1 Introduction

This chapter describes the application of ANNs in detecting and quantifying multiple defects when presented with ultrasonic pulse-echo A-scan data containing multiple reflected signals. Although it would be feasible for several defects to be present in a component, the area of particular interest in this study was the case where two defects occurred within one wavelength of each other. This case incurred the possibility that, if both reflected ultrasonic signals were out of phase, they would partially cancel each other out. This could result in the maximum amplitude of the reflected signal being below the threshold at which the component would normally be rejected, even if the two defects were of a size where the amplitudes of their reflected signals would individually be above the threshold. Cases involving multiple defects whose signals did not overlap could be considered as two separate single defect cases, and thus analysed using the techniques described in chapter 4. Although a situation could occur where three or more defects were present in close proximity, the research was restricted to the case of two defects in order to ascertain that an ANN approach was a viable means of analysis. The investigation of more than two defects in close proximity is considered as further work (see section 9.3).

Figure 6.1 shows a typical ultrasonic signal, and the drop in amplitude that results when two signals overlap at a phase difference of a half wavelength. It can be seen from figure 6.1 that, in a conventional threshold-based ultrasonic inspection, if the threshold was at a value above around 0.35, the second signal would pass the inspection, even though it contains two defects which individually would not pass. The aim of this study was to investigate whether an ANN-based method would provide more information than would be available to an operator in cases where the A-scan technique was employed using a single frequency.

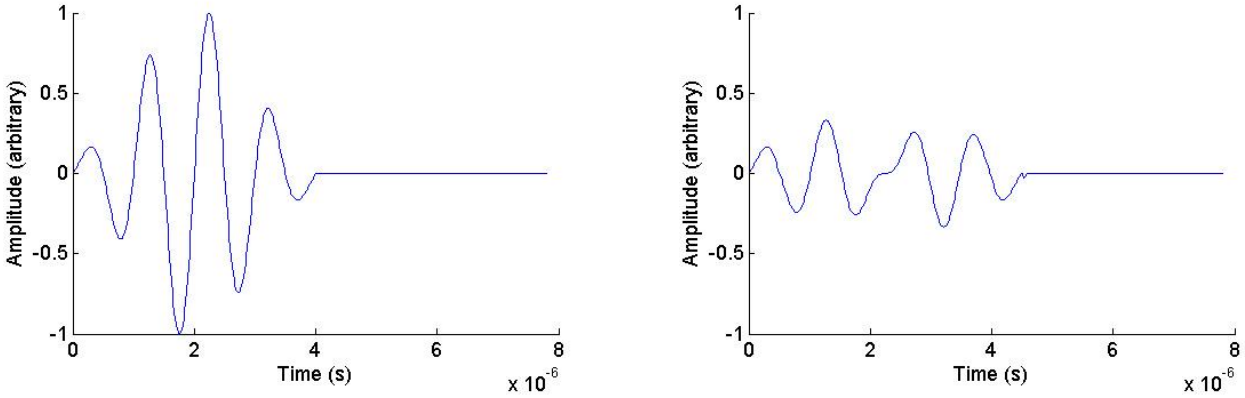


Figure 6.1 Typical ultrasonic signal (left) and the effect of two identical signals combining at ½ wavelength (right)

Instead of simply stating whether a feature signal, such as that from a defect, had an amplitude over a particular threshold, an ANN system was investigated as a means to decouple two overlapping signals, thus providing information on both defects. As stated in chapter 4, the ultrasonic pulse-echo method was considered for this thesis as it satisfied the criterion of being a simple, rapid and repeatable test.

6.2 Preliminary numerical study

In order to assess whether an ANN-based system was capable of separating two overlapping signals, an initial numerical investigation was conducted using two overlapping sinusoidal functions in MATLAB 7.1. The resultant function was given as

$$g(x) = A.f(x) + B.f(x+\phi) \tag{6.1}$$

where *g(x)* is the combined function, *A* and *B* are constants to scale the functions, *f(x)* is a sinusoidal function of *x*, and *f(x+φ)* is the same sinusoidal function of *x* with a phase shift

given by φ. For the first part of the study only one phase shift existed, whereas subsequent work employed an initial lag so that the feature signal started at some point after the global signal began. In this subsequent work, a separate phase shift on each function was used in order to assess the ability of the ANN system to determine the time at which both feature signals occurred, relative to the global signal presented to the ANN.

6.2.1 Initial study with one phase shift

The task of the ANNs in this study was to return the correct values for *A*, *B*, and φ (from equation 6.1) when presented with data representing the combined signal, *g(x)*. This work was conducted in order to verify that the chosen method of feature extraction was robust with synthesised data before being used with numerical and experimental ultrasonic data. Additionally, this method was appealing as the data set could be very quickly generated without the need for parametric FE studies or exhaustive experimental testing. As with previous work, multilayer, feedforward backpropagation networks were used as function approximators, so that their output would be continually variable. Function approximator ANNs were preferred over those working as classifiers, which would assign outputs into discrete groups, as in reality it would be desirable to determine the exact size and location of any defects in a component. To represent the ultrasonic signal, 2 periods of a sine function inside a Gaussian window were used, as was the case for the FE representation of the ultrasonic pulse described in section 4.4.1 and shown in figure 6.2. The input signal was scaled to have a range in amplitude of [-1, 1]. An ultrasonic frequency of 100kHz was used, with timesteps of 156.25ns, giving a total input signal length of 128 steps (20μs).

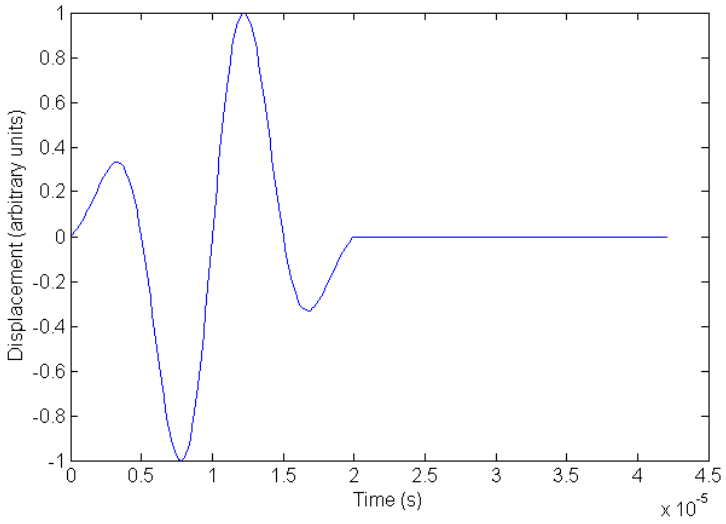


Figure 6.2 Ultrasonic input pulse

6.2.1.1 Implementation of initial study

For the initial study only one phase shift was considered, as it was reasoned that it would be possible to use a method such as a zero-crossing technique (Kazys 2004) to locate the start of the feature signal, i.e. the portion of the overall recorded signal that contained reflections from defects. Integer values of 1 to 3 were used in equation 6.1 for the scaling coefficients *A* and *B*, and integer timestep values of 4 to 136 in multiples of 4 (to represent 34 discrete phase shifts) were used for the phase shift coefficient, φ. As the overall signal would contain two overlapping individual signals, a total signal length of 270 steps (42.03125μs) was used, with zero padding employed to keep all cases at a uniform array length. Values for φ were chosen to vary the overlap of the two signals from 625ns to 21.25μs, meaning the range would cover the case of signals almost completely overlapping (figure 6.3) to the case of signals just discretely separated (figure 6.4). A data set was generated using MATLAB 7.1, where all possible permutations of *A*, *B* and φ were recorded. This data set had 306 cases.

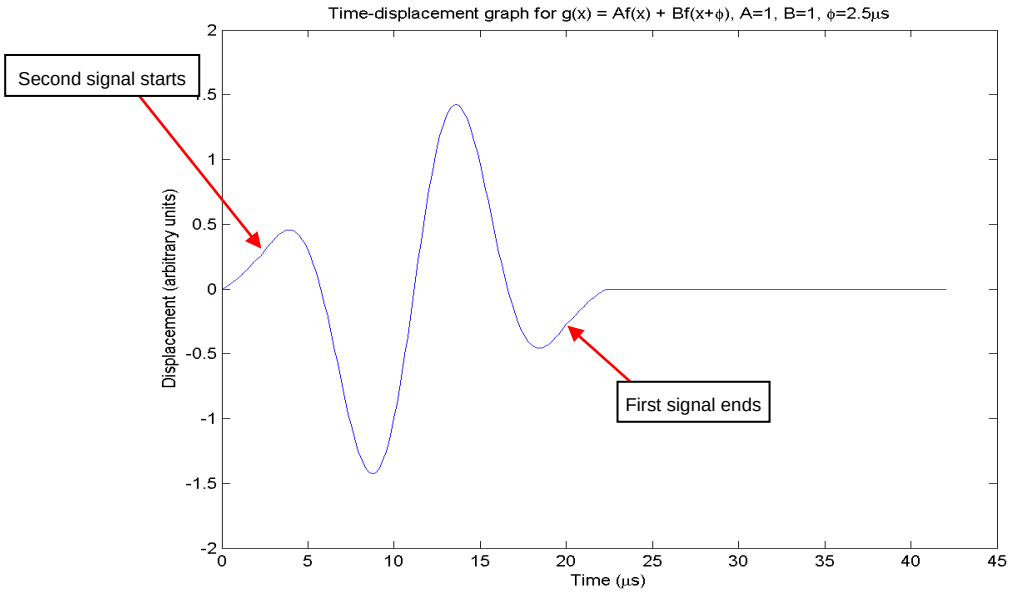


Figure 6.3 Combined ultrasonic pulses with short phase shift: φ=2.5μs

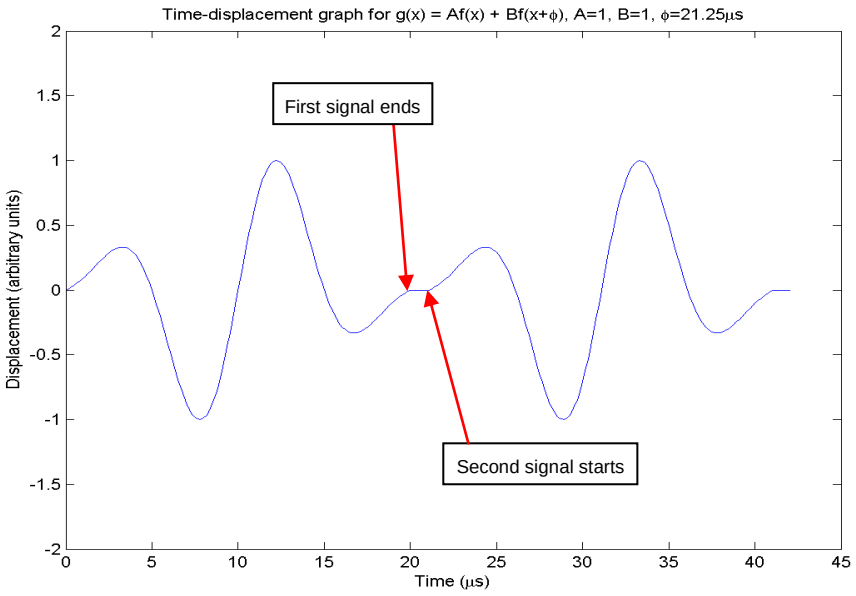


Figure 6.4 Combined ultrasonic pulses with long phase shift: φ=21.25μs

In a similar manner to the case of locating and sizing a single defect, described in chapter 4, the time and height of each peak of the combined signal were recorded and used to represent

the signal to the ANNs, following Yagawa and Okuda (1996). The primary area of investigation in this study was to determine if the time and height of each peak in the combined signal contained enough information to allow the ANN to decouple the two individual signals. A representation of this type was particularly desirable as it did not require a great deal of pre-processing of the raw data which was often necessary when attempting any kind of automated processing and interpretation of ultrasonic data (Windsor, Anselme *et al.* 1993).

The input data (times and heights of all peaks) and the target data were normalised to lie in the range [-1, 1] before being presented to the ANNs. Initially, all data were used to train and subsequently assess the ANNs, as the purpose of this initial study was to determine whether a multilayer ANN could draw adequate links between the input and output data, and therefore successfully decouple two overlapping signals. The outcome of this initial investigation was then used as a basis for further research.

A three layer feedforward, backpropagation ANN was used, with 16 neurons in the first layer, between 1 and 40 neurons in the hidden layer, and 3 neurons in the output layer. Training took place for 20,000 epochs using the scaled conjugate gradient method. The ANN outputs were the normalised predicted values for A, B and φ. As with previous work in chapter 4, the measure of the ANNs' accuracy was determined by the mean absolute deviation from the target value for each parameter, as shown in equation 4.2. In this study, in order to achieve the best performance with minimal computer resources, the performance of the network was monitored as the number of neurons in the hidden layer increased, as shown in figure 6.5. It can be seen that there was no significant improvement in network performance once the hidden layer reached 17 neurons, so this was the number chosen for the remainder of this initial study.

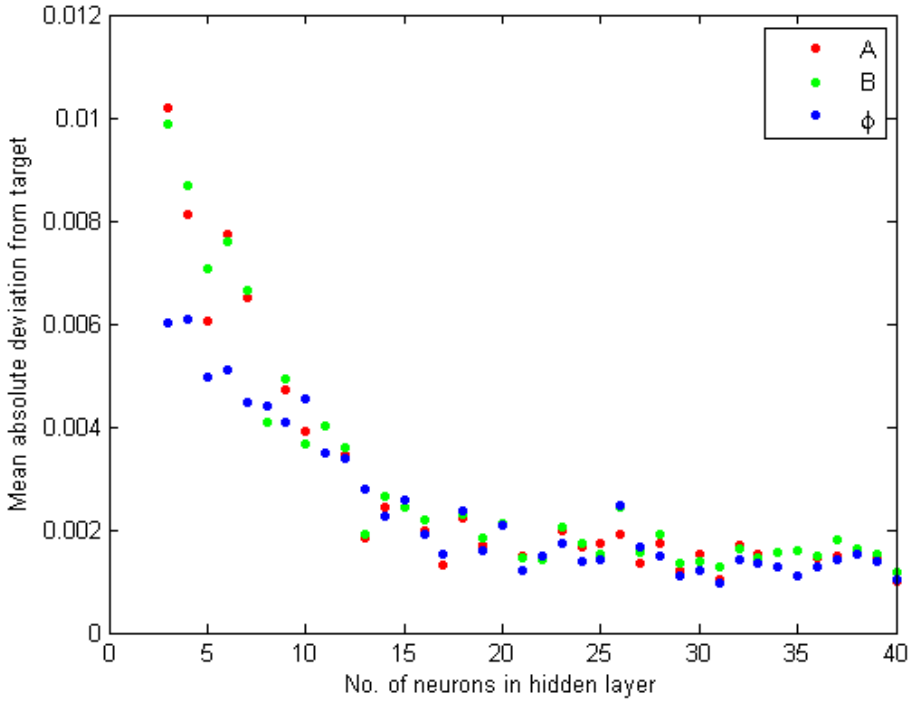


Figure 6.5 ANN performance as number of neurons in hidden layer increases for coefficients *A*, *B* and *φ*

6.2.1.2 Results from initial study

Using an ANN of layout 16-17-3, all data (100%) were initially used to train and assess the network, to determine whether the input data contained enough information to enable the ANN to decouple the two signals. The decision to initially use 100% of the data was based on a desire to see whether the ANN could draw meaningful links between input and target data, rather than to test its ability to generalise. The results are shown in figure 6.6, where it can be seen that the network performed very well in determining the three coefficients, demonstrating that the features extracted from the overlapping signals contained enough information to allow the ANN to distinguish between cases when 100% of the data was used. However, when trained with only 90% of the data set and assessed with the remaining 10%, the performance of the ANN was not as good when trying to determine the scaling

coefficients, *A* and *B*, as shown in figure 6.7. This would suggest that the information available in the input data was only partially sufficient to allow the trained ANN to interpolate between cases, thus causing inaccurate outputs. The implication of this was that further pre-processing of the data would be needed to ensure a greater reliability of the ANN's outputs.

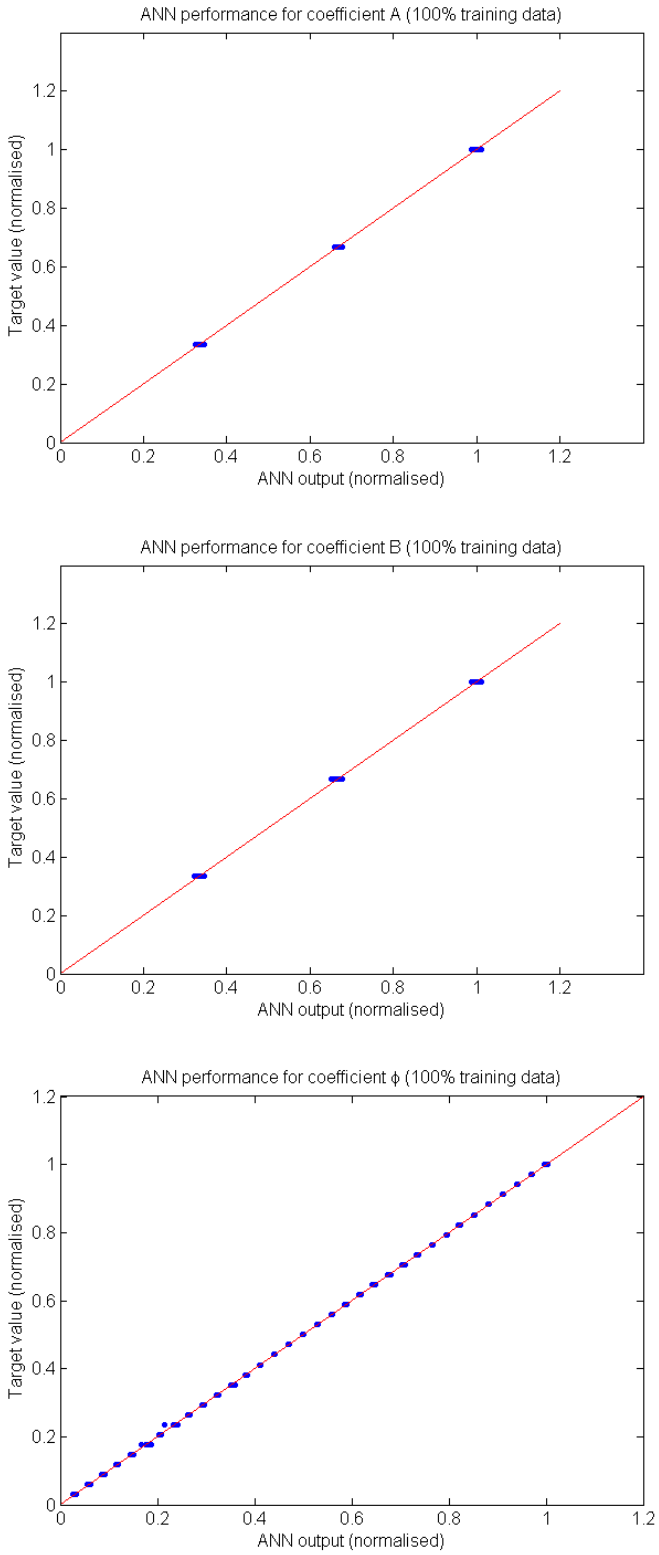


Figure 6.6 ANN performance for *A*, *B* and φ with 100% training data

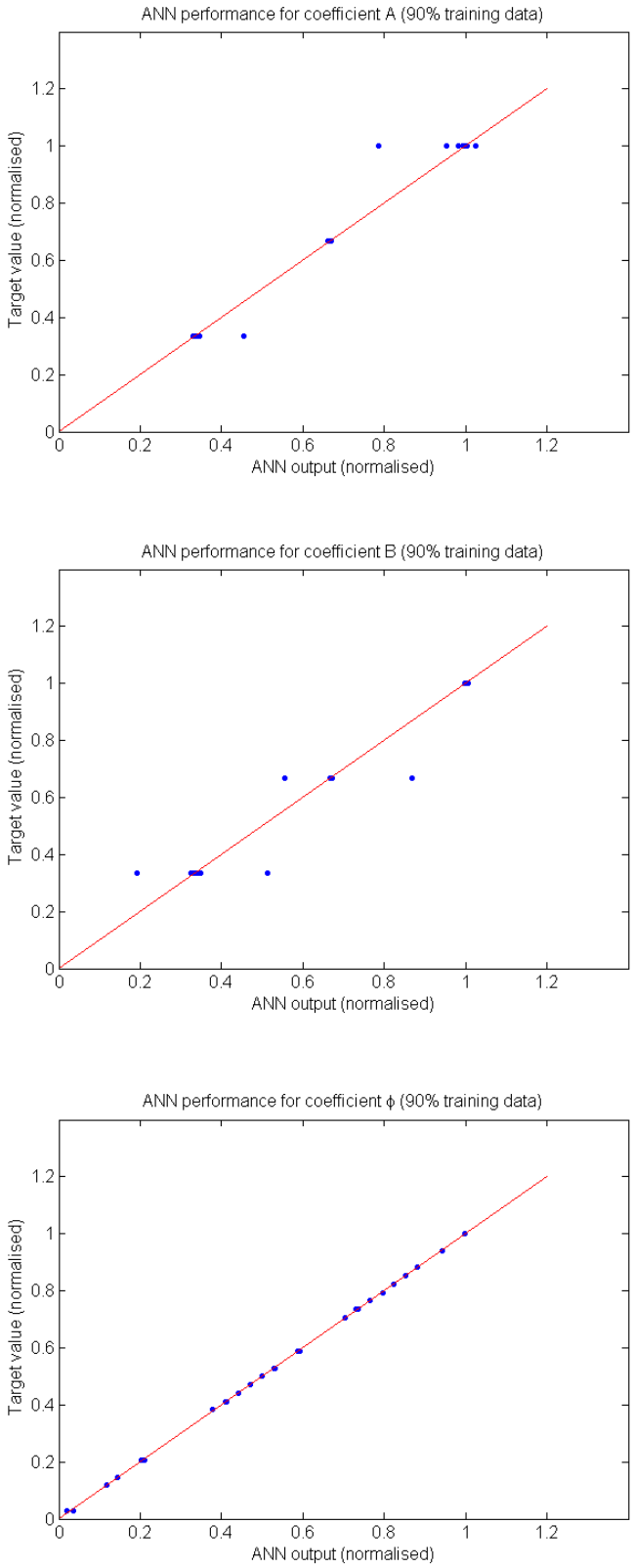


Figure 6.7 ANN performance for *A*, *B* and φ with 90% training data

6.2.1.3 Cross-correlation of data

As was observed from the initial study, the ANN's performance significantly reduced in terms of returning the scaling coefficients, *A* and *B* when trained with 90% of the available data and assessed with the remaining 10%. One of the limitations of the method used to represent the signals to the ANNs was that the order in which the data were stored was critical. As signals could contain between 4 and 8 peaks, the position of each peak's data in the input array was very important in order for the network to operate correctly. The algorithm used to find the peaks ordered them into an array in the order they occurred, i.e. a peak at the beginning of the signal would be assigned to column 1, the next peak to column 2 etc. However, in the event of the algorithm being unable to locate a particular peak (for example, if it was very small, a saddle point or very close to another peak) and therefore missing a peak in the signal, the subsequent peaks would appear in the wrong columns and would provide incorrect data for the ANNs. One way around this problem was to cross-correlate the overlapping signal, *g(x)*, with the input signal, *f(x)*, which provided a smoother signal from which peaks could be located (Mahmud and Ryoji 2004). This was done using the *xcorr* function in MATLAB 7.1. The cross-correlation method essentially examines a particular signal (in this case the global signal containing two overlapping signals), and provides information on points where the signal resembles another signal (in this case the input signal). The cross-correlation function is given as

$$R_{xy} = \frac{\sum_i \left[\left(x_i - \bar{x} \right) \times \left(y_{i-d} - \bar{y} \right) \right]}{\sqrt{\sum_i \left(x_i - \bar{x} \right)^2} \sqrt{\sum_i \left(y_{i-d} - \bar{y} \right)^2}} \tag{6.2}$$

where R_{xy} is the cross-correlation at delay *d*, *x* and *y* are two vectors to be cross-correlated, $\bar{x}$ is the mean value of vector *x* and $\bar{y}$ is the mean value of vector *y*. In the case of the cross-

correlation performed on the two signals, vectors *x* and *y* represent the amplitude of each signal, *g(x)* and *f(x)*, as a function of time. It can be seen from equation 6.2 that if at delay *d* there is a high correlation between both vectors, the numerator will have a high value, thus indicating a strong correlation at that point. The denominator terms (the standard deviations of each signal at that value of *d*) act to normalise the output. An example of the raw and cross-correlated signals is shown in figure 6.8.

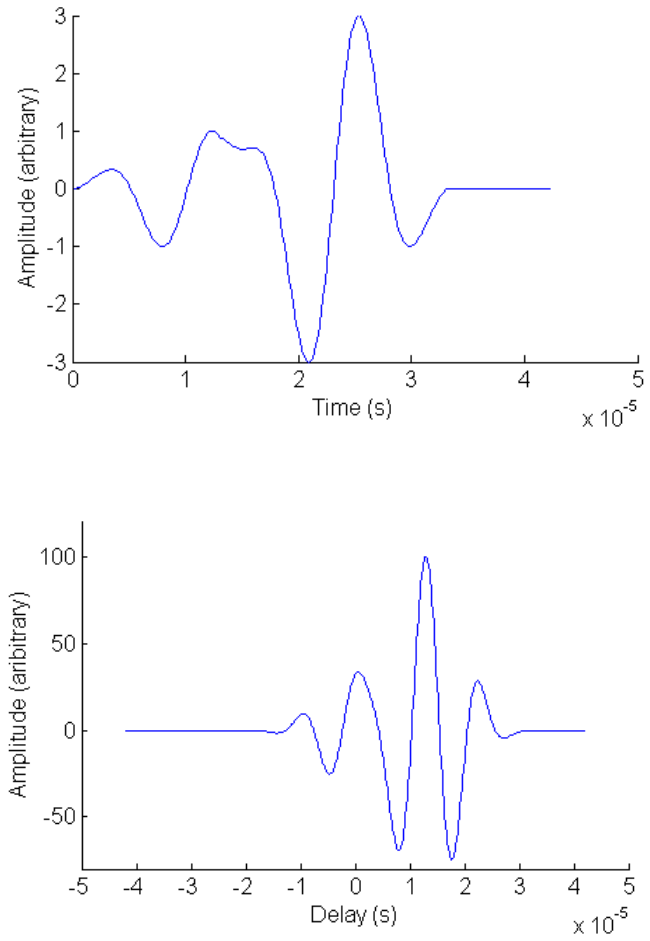


Figure 6.8 Raw signal (upper) and cross-correlated signal (lower)

The ANN developed in section 6.2.1.1 was trained and assessed using the times and heights of all the peaks in the cross-correlated signal. 90% of the available data were used to train

the ANN, and the remaining 10% to assess its performance. The results are shown in figure 6.9.

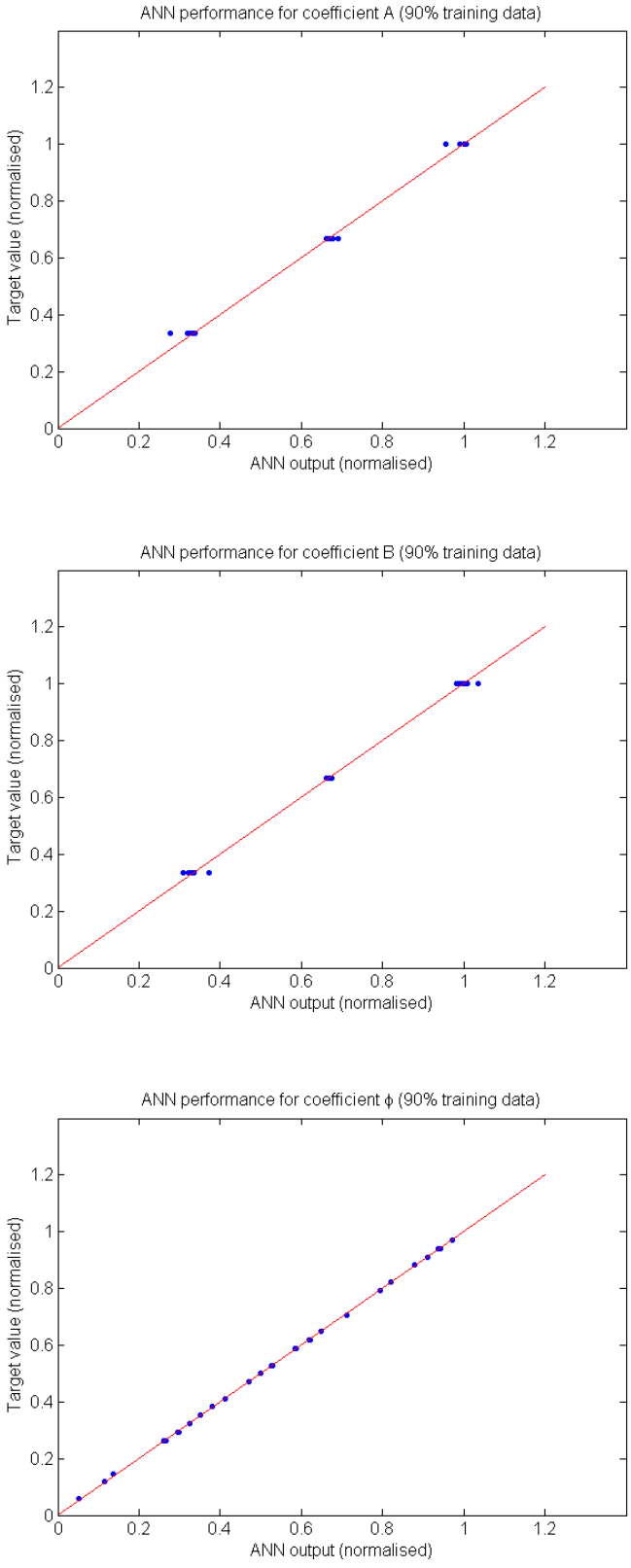


Figure 6.9 ANN performance with 90% training data, cross-correlated signal for coefficients *A*, *B* and *φ*

As can be seen from figure 6.9, considerably better results were achieved in the estimation of scaling coefficients A and B than were seen using the raw data. Thus, the cross-correlation technique was considered a suitable pre-processing tool for further study.

6.2.2 Secondary study with phase shift and initial lag

After ensuring that the problem of decoupling two overlapping signals was solvable using an ANN method, the case where the feature signal did not start at the same time as the global signal was investigated, that is, where the first sinusoidal function started some time after *t=0*. The expression for this function becomes

$$g(x) = A.f(x+\theta) + B.f(x+\phi) \tag{6.3}$$

where *g(x)* is the overall function of *x*, *A* and *B* are the scaling coefficients, θ is the initial lag and φ is the phase shift between signals. This was more representative of a real situation, as the global signal from an ultrasonic test would not necessarily start at the same time as the first feature signal contained within it.

6.2.2.1 Implementation of secondary study

Following the successful method demonstrated in section 6.2.1.3, the times and heights of all peaks in the cross-correlated signal were again used to represent the signals to the ANNs. In this case, however, the global signal was longer to incorporate the extra phase shift given by the lag coefficient θ. Time steps of 156.25ns were used, and each case had 410 steps in total, giving a total signal length of 64.0625μs. Integer values of 1 to 3 were used for the scaling coefficients *A* and *B*, and integer values between 4 and 136 in multiples of 4 were used for

the lag coefficient θ. Integer values between 8 and 272 in multiples of 4 were used for the phase shift coefficient φ, maintaining the boundary condition $\varphi>\theta$, as the second signal had to occur at a later time than the first. This gave a total of 10,404 cases in the data set.

In this study, the ANN had four output neurons, to represent the four function coefficients, *A*, *B*, θ and φ. 22 input neurons were used (a value chosen as there were up to 11 peaks in the signal, each requiring a value for time and height). As in section 6.2.1.1, the ANN's performance was monitored as the number of neurons in the hidden layer was increased, as shown in figure 6.10. This optimisation of the ANN layout was conducted using 100% of the available data to train and assess the ANN. Training once again lasted for 20,000 epochs and used the scaled conjugate gradient method.

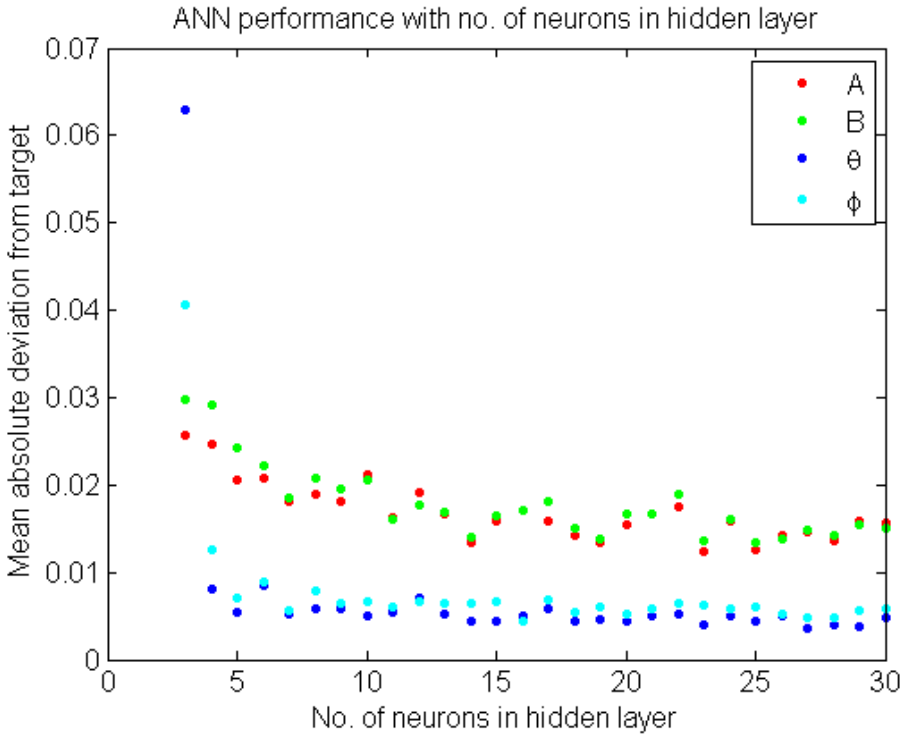


Figure 6.10 ANN performance as number of neurons in hidden layer increases for coefficients *A*, *B*, θ and φ

As can be seen from figure 6.10, there was little improvement after the hidden layer reached 12 neurons, thus this was the value chosen for the hidden layer. The final network had a layout of 22-12-4. It was trained with 90% of the available data, selected at random, and assessed with the remaining 10%.

6.2.2.2 Results from secondary study

The results from the ANN are shown in figure 6.11. The accuracy of the ANN's predictions for the scaling coefficients A and B was noticeably better in this case than it was in the case of a single shift, presented in sections 6.2.1.2 and 6.2.1.3. This is considered to be due to the larger data set generated, meaning that the training data set more accurately represented the data set as a whole.

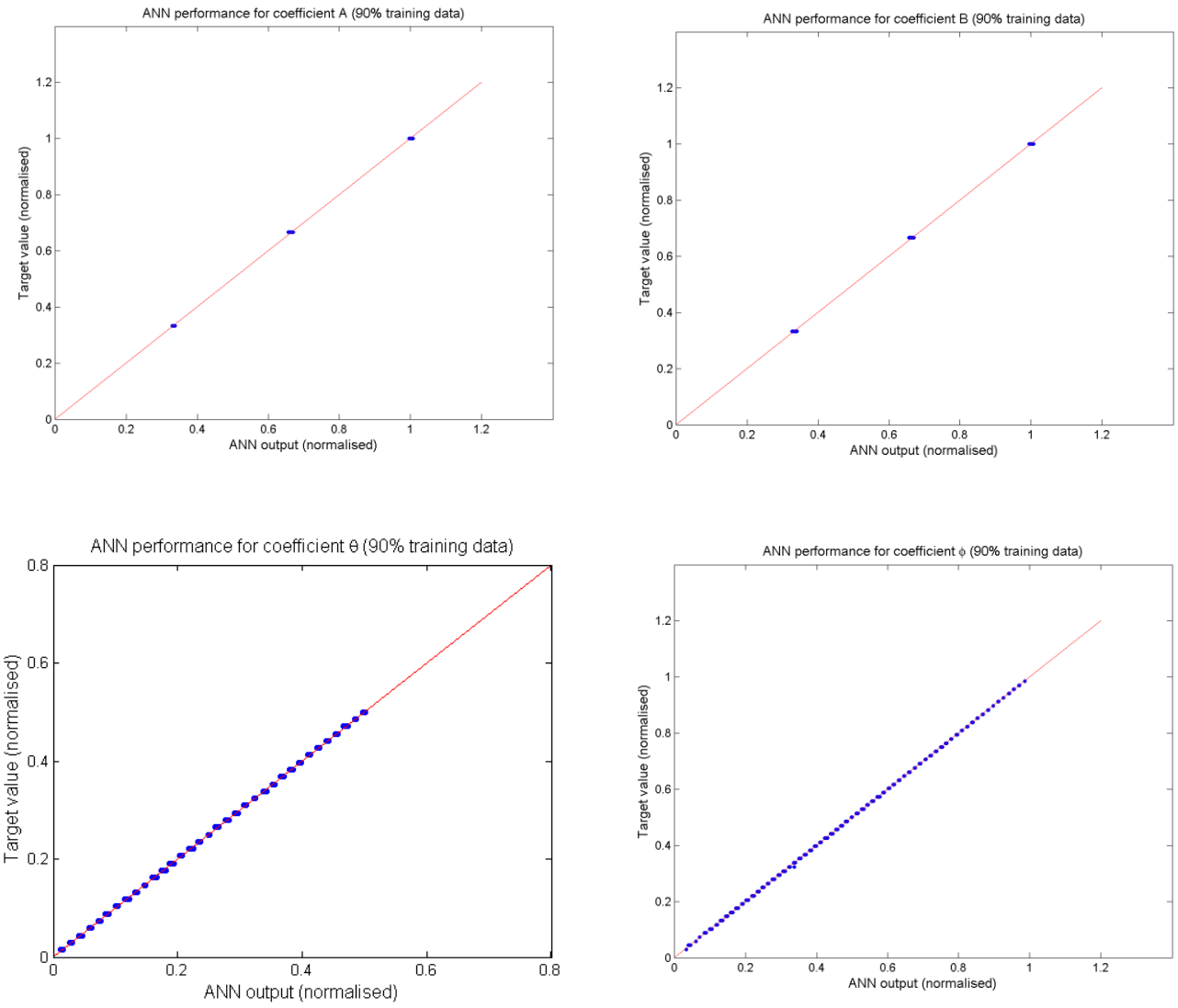


Figure 6.11 ANN performance with 90% training data, cross-correlated signal for coefficients *A*, *B*, θ and φ

6.2.3 Summary of preliminary studies

The preliminary studies showed that, in the case of synthetic, mathematically generated signals, an ANN system was able to decouple two overlapping signals and return the phase shifts and magnitudes of each signal when the overall function was cross-correlated with a sample signal. The cross-correlated signal was easier for the peak finding algorithm to interpret, thus more reliable input data were able to be created. The performance of the ANNs was limited if the input data were not in a specified order, e.g. if the peak finding algorithm missed a peak, thus the importance of presenting the peak times and heights in uniform arrays was highlighted.

Overall, the method showed very promising results, and demonstrates that the times and heights of each peak in a cross-correlated signal of this type contain sufficient data to allow an ANN to decouple the two signals.

6.3 Ultrasonic method

Successful methods have been presented in this thesis demonstrating the use of ANNs in locating and sizing a single defect, and in decoupling multiple waveforms. These techniques were applied to the ultrasonic inspection of a component with two discrete defects using data generated by the finite element method. As stated in the introduction to this chapter in section 6.1, particular attention was paid to the case of two defects lying longitudinally within one wavelength of each other, causing their respective reflected signals to overlap. Experimental data were not possible to gather using the equipment available, as the higher frequency of the experimental ultrasonic transducer meant that the wavelength in the test samples was around 1.2mm. This meant that it was not possible to get two defects of

between 1 and 6mm size within one wavelength, thus the investigation was a purely numerical study.

6.3.1 Development of data set using the finite element model

As with the case of a single defect, a steel bar of thickness 10mm and length 200mm was the component under inspection. The component is shown in figure 6.12. A sinusoidal pulse of centre frequency 100kHz and duration 2 cycles, inside a Gaussian window, was used as the ultrasonic input pulse. The pulse-echo method of ultrasonic inspection was used, meaning that only one transducer was modelled.

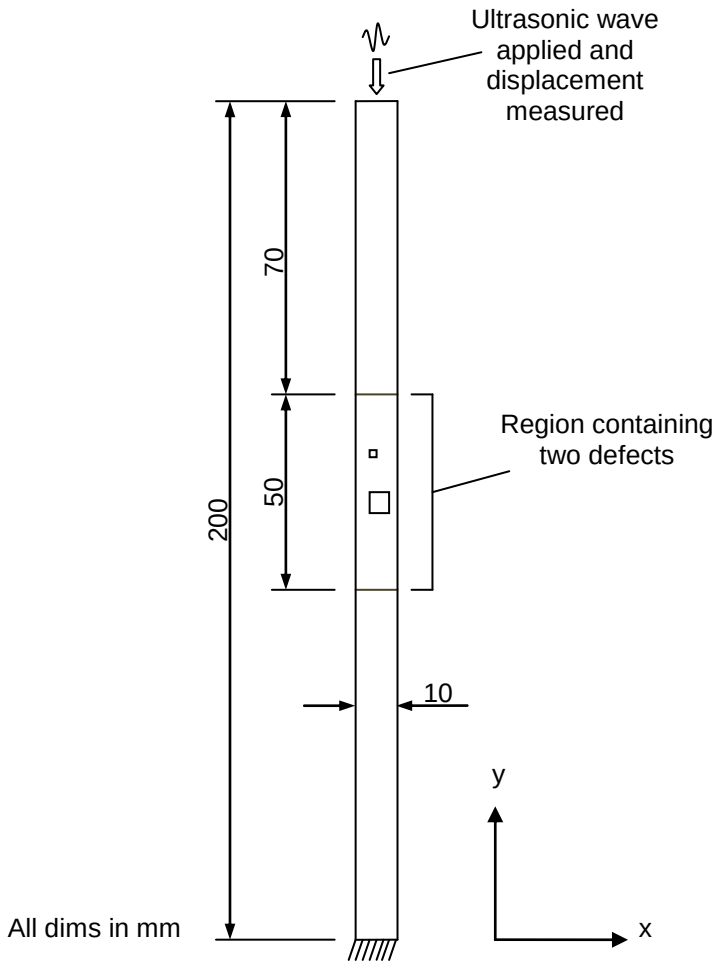


Figure 6.12 Steel bar with multiple defects

The finite element model used was identical to that described in section 4.4.1, except in this case the python script (see appendix B) that provided the element number for the parametric study was modified to contain two variables rather than one. These variables corresponded to the element numbers that would be removed from the model. In this case, two square defects were removed from the model for each case in the parametric study. The ABAQUS input file was used as a template, and the python script created the files for the parametric study from this. Due to memory constraints, 500 cases were processed at a time. A typical mesh, showing the square defects, is shown in figure 6.13. A Fortran program was used to interrogate the output files and return the vertical nodal displacements at the top of the bar. This provided displacement at 11 nodes for each case. The mean value of all 11 nodes was taken as being representative of the transducer output, thus this mean value at each of 500 timesteps was used as the signal from which the ANN input data would be determined. The start time for the recorded signal was 30μs from the start of the simulation, with timesteps of 0.1μs. This starting point and timestep value were sufficient to capture the complete wave reflected from the defects. Defects were considered in the same region as in the single defect case, that is, between 70mm and 120mm from the top of the bar. Given that the ultrasonic frequency was 100kHz, and the P-wave velocity in steel is around 5810m/s (Hull and John 1989) using equation 3.39, this corresponds to a wavelength of around 58mm, meaning that all the defects would lie within one wavelength of each other. Element numbers for the two defects were chosen at random, but with a constraint applied so that the defects did not touch, thus all cases involved two separate defects. The overall data set contained 3055 cases, with defects ranging from 1mm to 4mm in size. In all cases, at least one of the defects was 1mm in size. It was considered that a system capable of providing information on defects as small as 1mm in size would be able to provide similar information on larger defects. Additionally,

given that the system analysed two overlapping ultrasonic signals, it was reasoned that a system capable of providing information on the size of each defect would be robust in performing the same task if the size of both defects was scaled up or down. In this case, the overall magnitude of the resultant signal would be increased, but the shape of the resultant signal would remain the same. Cases were considered with the smaller defect closer to the top of the bar than the larger defect, and vice versa to make sure the data set was representative of all possible situations.

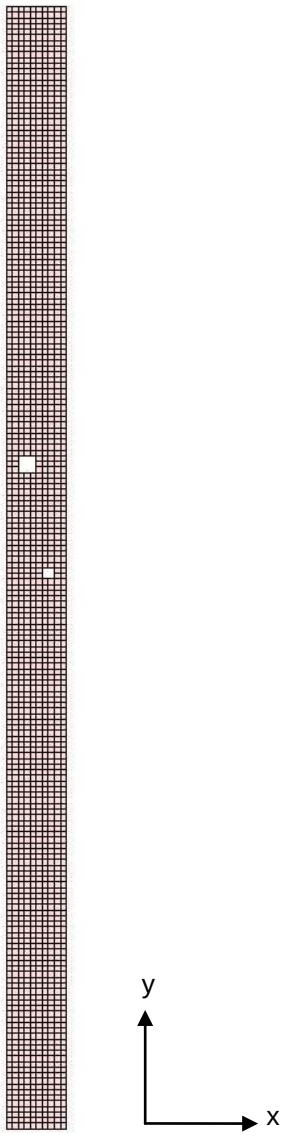


Figure 6.13 FE mesh on steel bar showing example defect positions

6.3.2 Feature extraction

Following the successful results of the single defect study, the method of Yagawa and Okuda (1996) was again used to subtract a signal from a non-defected component from that of a defected component. Thus, the signals used in this study were the difference between the measured signal and a reference signal from a bar with no defect. As with the purely numerical data considered earlier in this chapter, the signals from the test were cross-correlated with the transmitted ultrasonic pulse, in order to determine the points at which the two signals were similar. The times and heights of each peak in the cross-correlated signal were normalised and used as features to represent each case to the ANNs.

6.3.3 ANN architecture and optimisation

Heteroassociative multilayer backpropagation networks were used as function approximators to draw links between the input data (the normalised times and heights of all peaks in the cross-correlated data) and the output data (the normalised location and size of each defect). Across all cases the maximum number of peaks in the cross-correlated data was 13, so an ANN with 26 input neurons and 4 output neurons was chosen as a sensible starting point. The 4 neurons in the output layer was a non-negotiable constraint, as the target data had four components: distance to defect 1, distance to defect 2, size of defect 1 and size of defect 2. In order to determine the optimum ANN layout, the number of neurons in the hidden layer was varied, and the performance of the ANN monitored using the measure of mean absolute deviation from the target, as given in equation 4.2. Up to 90% of the available data were used to train the ANNs, and the remaining data used to assess the ANNs' performance when presented with previously unseen data. The ANNs were trained for 20,000 epochs using the

scaled conjugate gradient method, as was the case for the single defect study in chapter 4. Figure 6.14 shows the mean absolute deviation from the target as the number of neurons in the hidden layer was increased. As can be seen from figure 6.14, it was apparent that a hidden layer of 6 neurons would provide as much accuracy as a much larger hidden layer. The optimised ANN layout is shown graphically in figure 6.15.

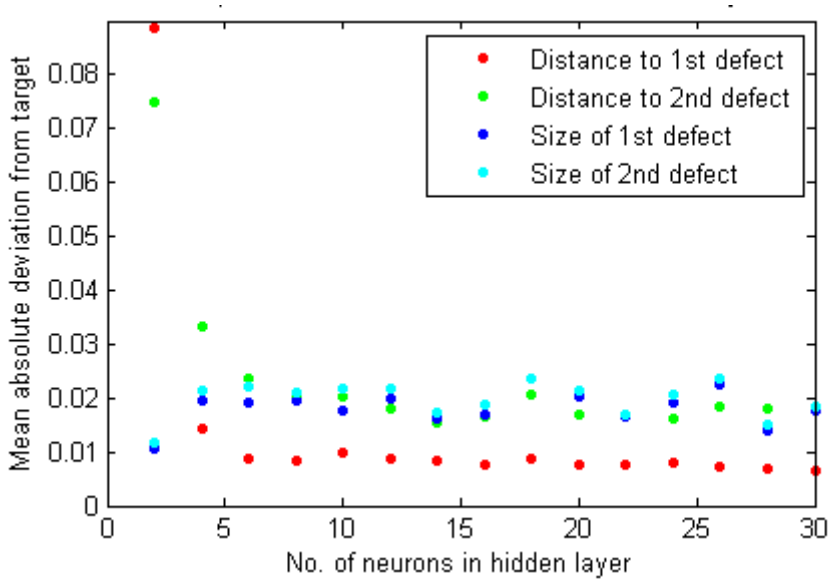


Figure 6.14 ANN performance as number of neurons in hidden layer increases for distance and size of both defects

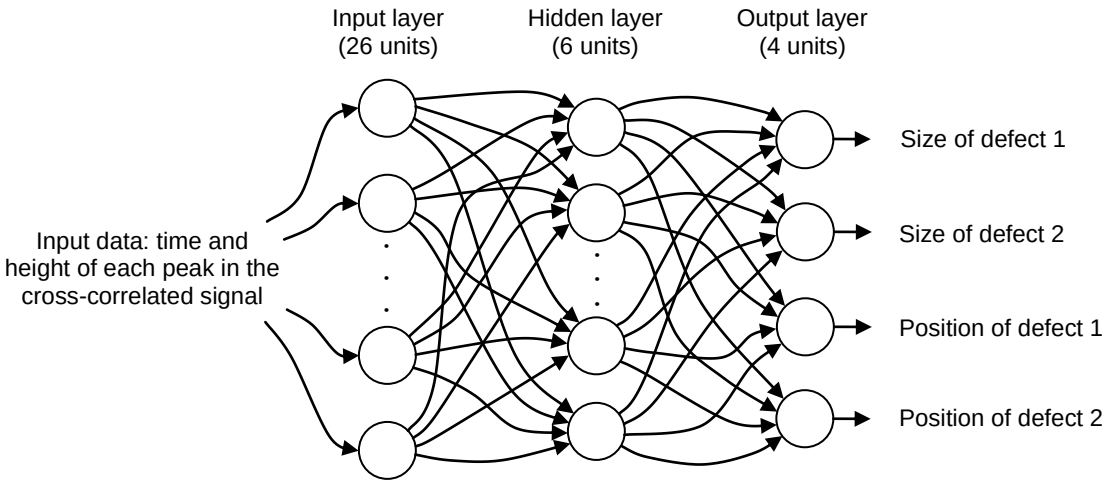


Figure 6.15 ANN layout after initial optimisation

Although the network appeared optimised when a hidden layer of 6 neurons was used, when tested with the data set the results were not as accurate as was initially hoped. Figure 6.16 shows the output from a 26-6-4 ANN trained using 90% of the available data, where it can be seen that although the distance to the first defect is fairly accurate, the other parameters show much more variation between the estimated and target values.

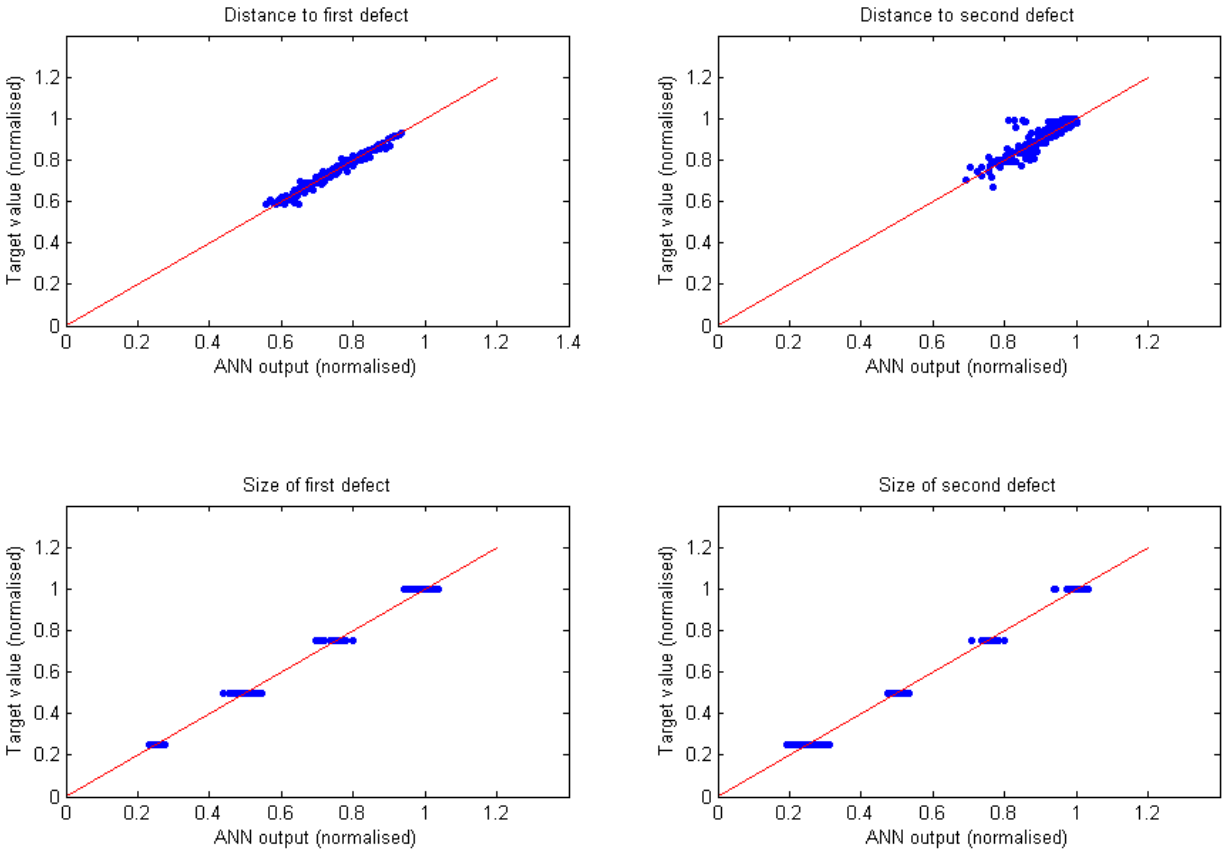


Figure 6.16 ANN performance for 26-6-4 network, trained with 90% of available data

Following the results obtained in chapter 4, where it was shown that a larger ANN generally gave more accurate estimations of parameters when presented with useful input data, an ANN layout with large input and hidden layers was considered. This ANN had a layout of 40-50-4, and was chosen to determine whether there was any advantage in greatly increasing the number of neurons in the input layer. As is widely reported in the literature, finding the optimum ANN layout for a given data set is a matter of making educated guesses to create a starting point, and then iteratively adjusting the architecture by trial and error to determine

the most efficient layout (Demuth and Beale 1998). The results from the 40-50-4 ANN, trained with 90% of the available data, are shown in figure 6.17, where it can be seen that the accuracy of the ANN improved over that of the 26-6-4 layout in locating the distance to the first defect, but reduced slightly in estimating the distance to the second defect and the size of both defects. Neater clustering of the majority of cases was observed in the distance to the second defect when the 40-50-4 layout was employed, but this improvement was negated by the much larger error in the few outlying cases.

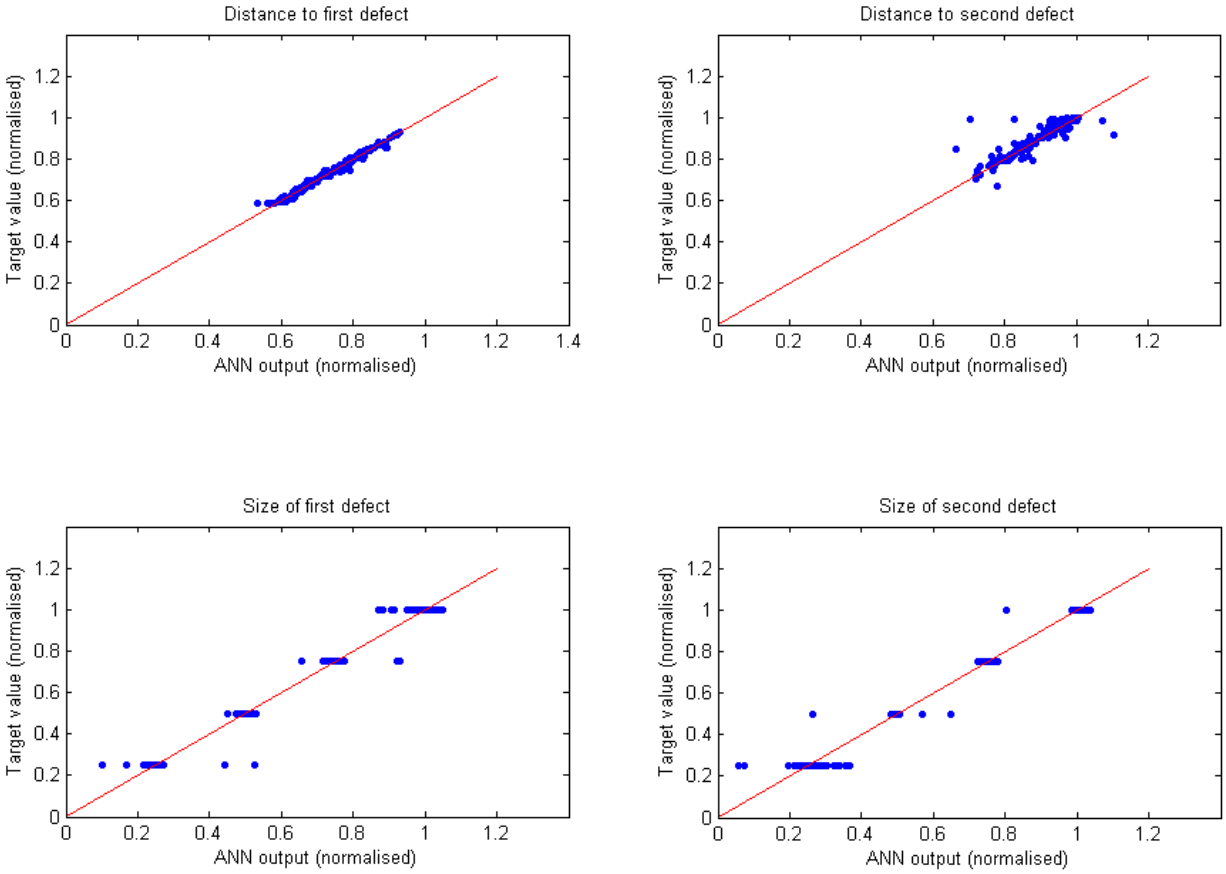


Figure 6.17 ANN performance for 40-50-4 network, trained with 90% of available data

From these results, it was evident that there was no overall advantage to be gained from the larger ANN architecture, and that even if a small advantage was possible, this may be outweighed by the larger ANN's need for longer training times and its possible tendency to 'overtrain', thus making it potentially unable to deal so well with noisy data. However, in order to inform the final decision of the optimum ANN architecture for the application, where noisy data would be considered, both the 26-6-4 and the 40-50-4 layout were used for the remainder of the study to characterise two separate defects. It was reasoned that their performance with lower amounts of training data and with noisy data would aid in assessing their ability to generalise, thus demonstrating whether overtraining was taking place in the larger network.

6.3.4 Results from ultrasonic simulations

As with previous results sections, the results presented here are representative of the observations as a whole. Both ANN configurations were presented with between 70 and 90% of the available data for training, and assessed with the remaining data. The ANNs were also assessed with data to which artificial noise had been added, following the procedure described in section 4.4.2, as this was considered to be more representative of experimental data. As stated in section 6.3, it was not possible to experimentally validate this study with the available experimental equipment.

6.3.4.1 The effect of the amount of training data

Generally, the results from the ANNs became more accurate as the amount of training data increased, although this increase in accuracy was fairly slight. Results for both ANN

configurations are shown in figures 6.18 to 6.21, where it can be seen that the majority of the ANNs' estimations are fairly close to the target regardless of the amount of training data used, but by increasing the amount of training data the number of highly erroneous predictions reduced.

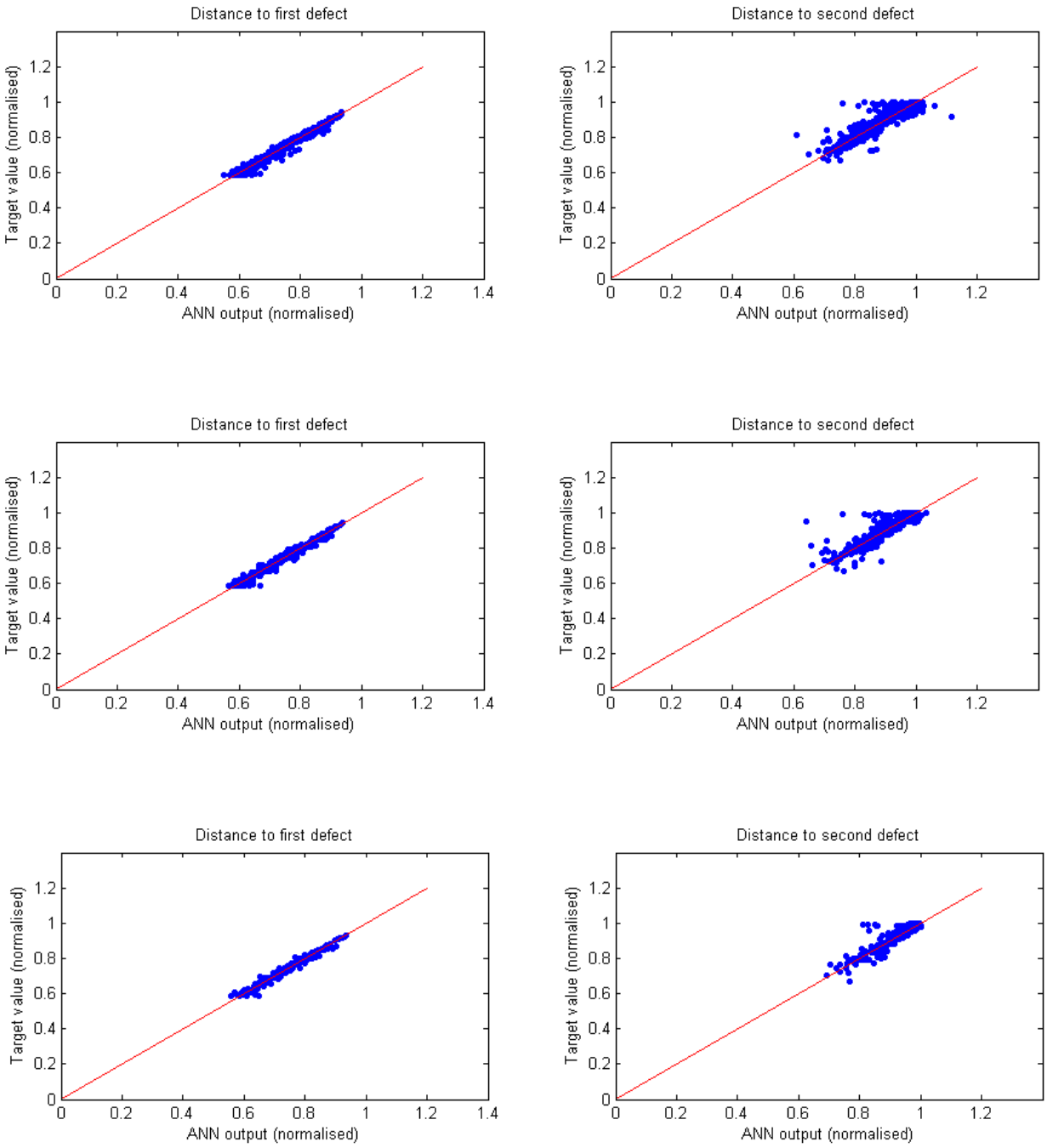


Figure 6.18 ANN performance (distance to defect) for 26-6-4 network, trained with 70% of available data (top), 80% (centre) and 90% (bottom)

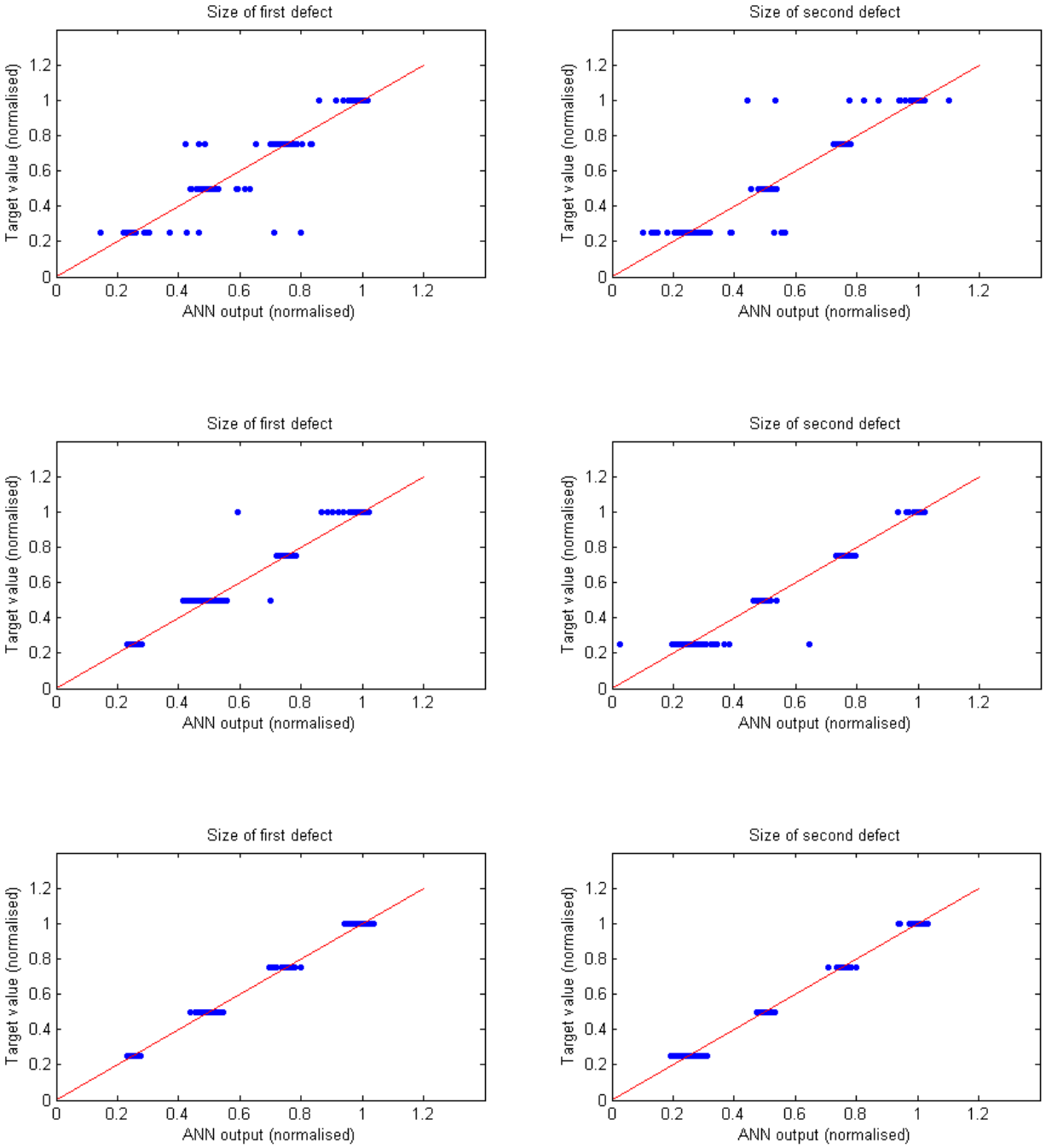


Figure 6.19 ANN performance (defect size) for 26-6-4 network, trained with 70% of available data (top), 80% (centre) and 90% (bottom)

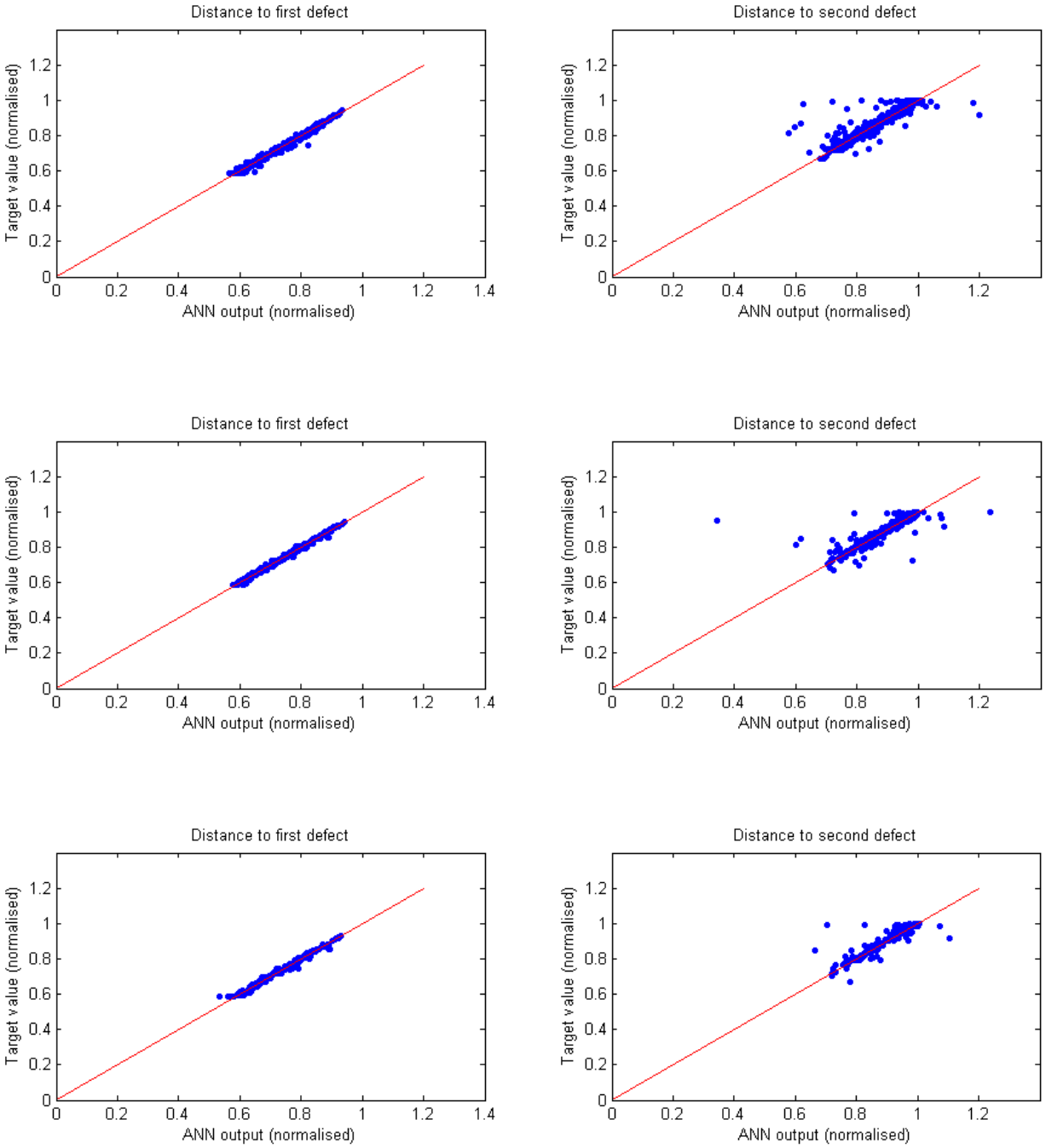


Figure 6.20 ANN performance (distance to defect) for 40-50-4 network, trained with 70% of available data (top), 80% (centre) and 90% (bottom)

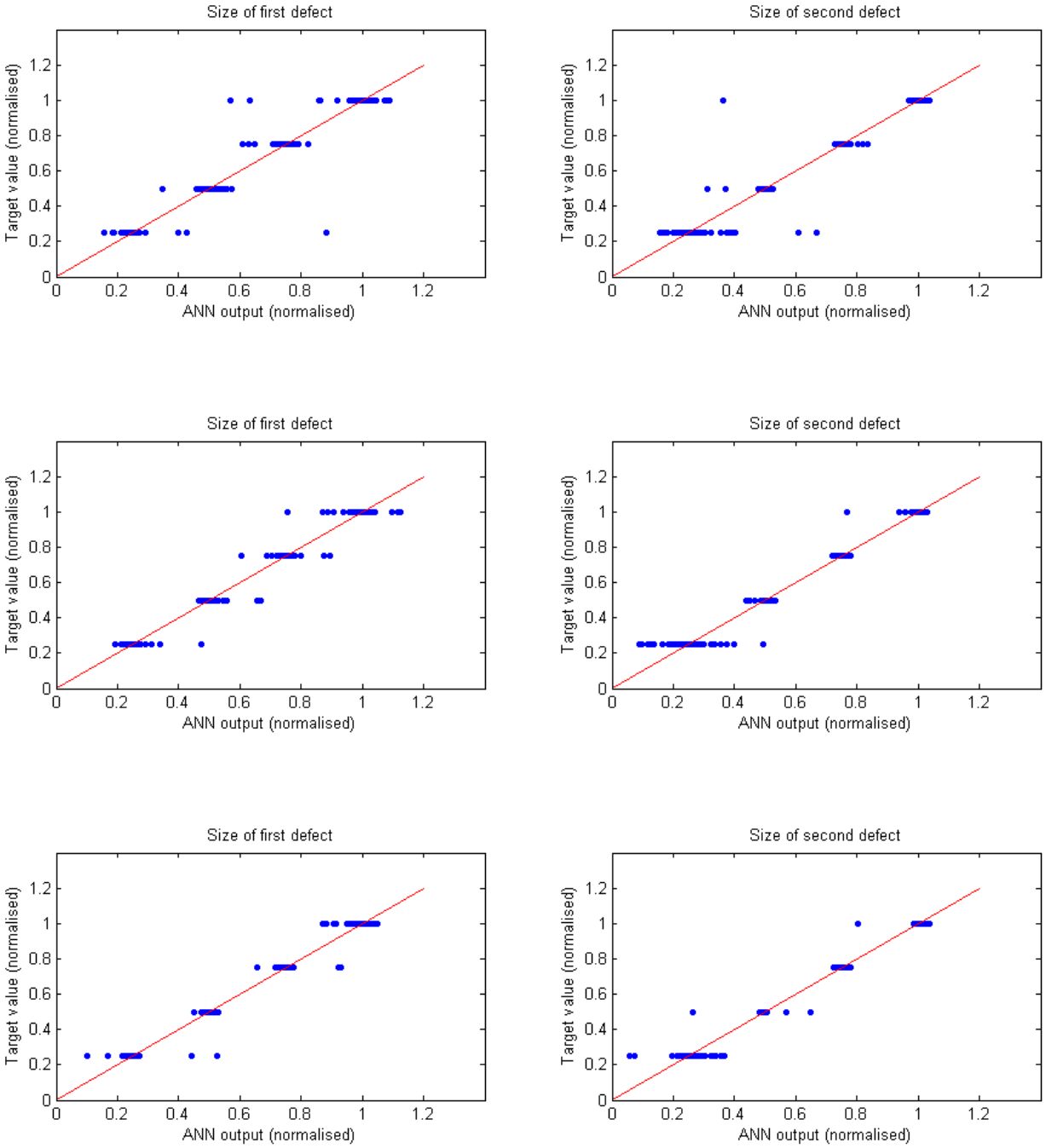


Figure 6.21 ANN performance (defect size) for 40-50-4 network, trained with 70% of available data (top), 80% (centre) and 90% (bottom)

6.3.4.2 The effect of added noise

As was the case in the single defect study in chapter 4, artificial random noise was added to the data using equation 4.4. Ten random values for noise were taken for each piece of input data, and the resultant noise was the mean of these ten values, multiplied by a scaling factor to alter the amount of noise added. The addition of noise affected both ANNs in a similar manner, and the error in estimating the four parameters increased as the amount of noise was increased. Both ANNs proved to be reasonably robust when returning estimates for the distance to each defect, although the addition of noise introduced much larger errors in the ANNs' estimations of the sizes of the defects. Based on examination of the experimental results presented in chapter 5, it was found that the typical variation in the time and height of a peak in the reflected ultrasonic signal was of the order of ±5%, thus the ANNs' performance with 5% added noise was focussed upon. Figure 6.22 shows both ANNs' outputs when presented with data to which 5% of artificial noise had been added. In this case the ANNs were trained with 90% of the available data, although a similar variation between estimation and target was observed when training was conducted with 80% of the available data. When training was conducted with 70% of the available data, the performance of both ANNs with noisy data was noticeably less accurate than in the case of 80% or 90% training data.

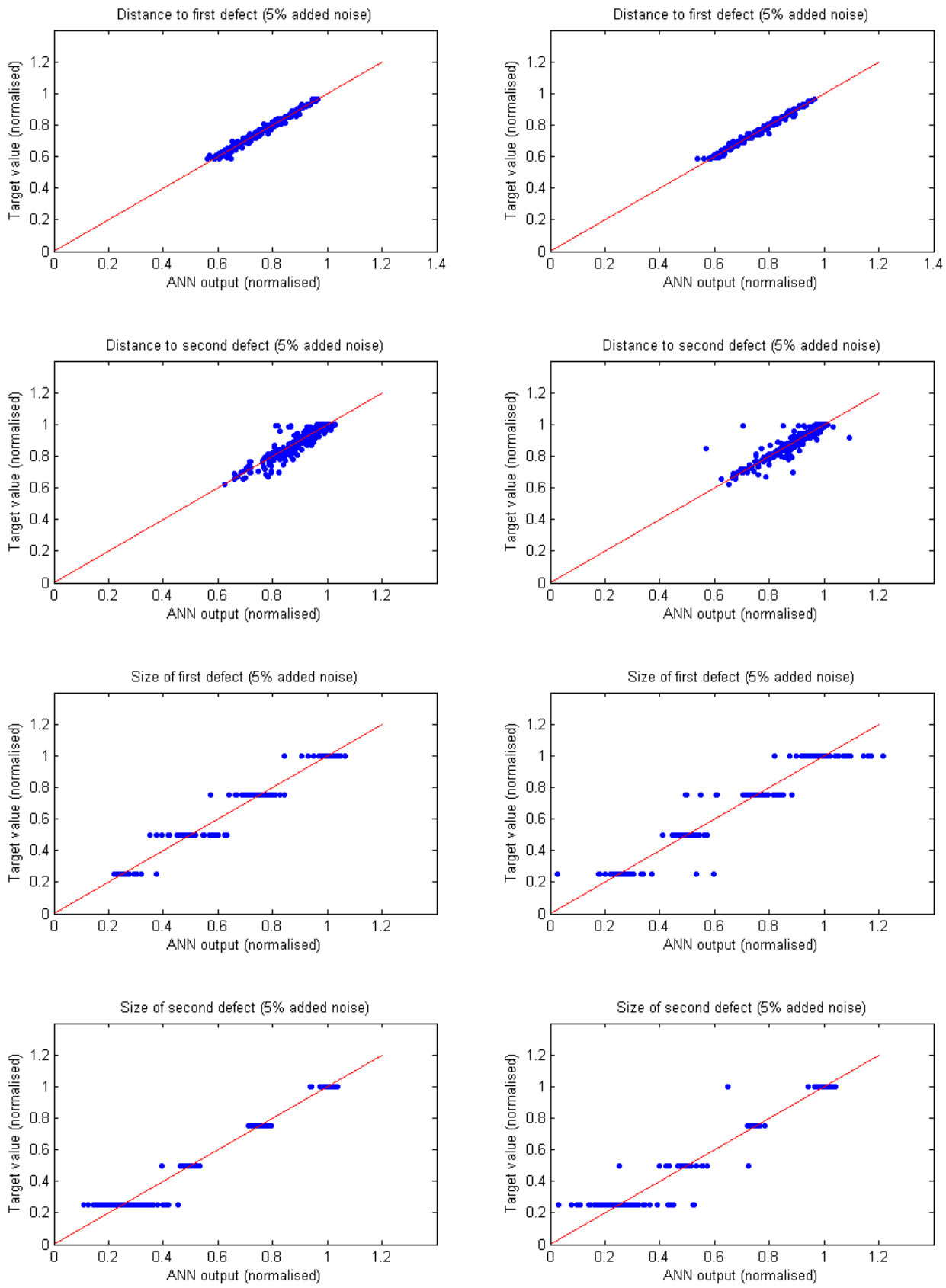


Figure 6.22 ANN performance with 5% added noise using 90% training data, 26-6-4 configuration (left) and 40-50-4 configuration (right)

Both ANNs seemed to be affected by noise in the same way, where it was observed that their errors increased roughly linearly with the amount of noise added, although for very small amounts of noise (<~2%) the performance of both networks did not significantly change. Figures 6.23 and 6.24 show the mean absolute deviation from the target for each network for 70-90% of training data, and added noise of up to 20%. It can be seen that for the 26-6-4 ANN trained with 70% of the available data there is a larger error in the estimation of defect size, when noise is added, than when the network is trained with either 80% or 90% of the available data. This would suggest that 70% of the data did not provide a sufficiently representative sample of all cases to enable the smaller network to derive the relationships between input and output data, thus when using 70% of the available data the network was undertrained. Performance with added noise for the same network trained with 80% or 90% of the available data was measurably better.

For the 40-50-4 network, a similar linear relationship between the amount of noise added and mean error was observed. Using this ANN architecture, the effects of added noise were very similar regardless of whether 70, 80 or 90% training data were used.

Although the difference was fairly slight, it can be seen from figures 6.23 and 6.24 that the 40-50-4 layout was affected more severely by added noise than the 26-6-4 layout. This would suggest that overtraining had occurred, and that the larger ANN was not so capable of generalising when presented with previously unseen data. Further evidence to indicate that overtraining had taken place can be seen in figure 6.22, where, although the main clustering of cases for the 40-50-4 ANN was similar to the 26-6-4 ANN, there were many more outlying data points when the larger ANN was used. This would suggest that the larger network was less able to generalise when presented with previously unseen data.

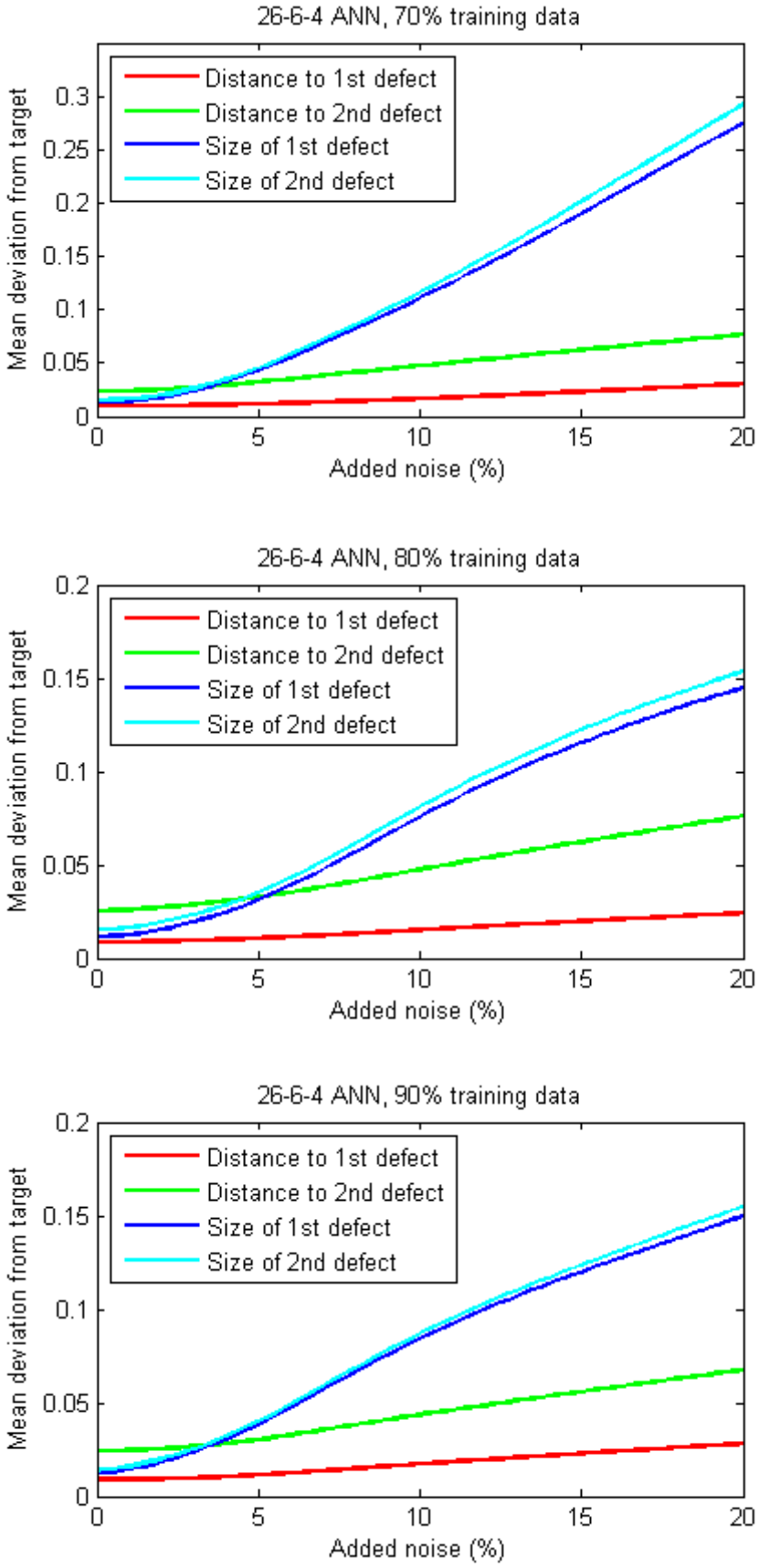


Figure 6.23 ANN performance with added noise, 26-6-4 configuration

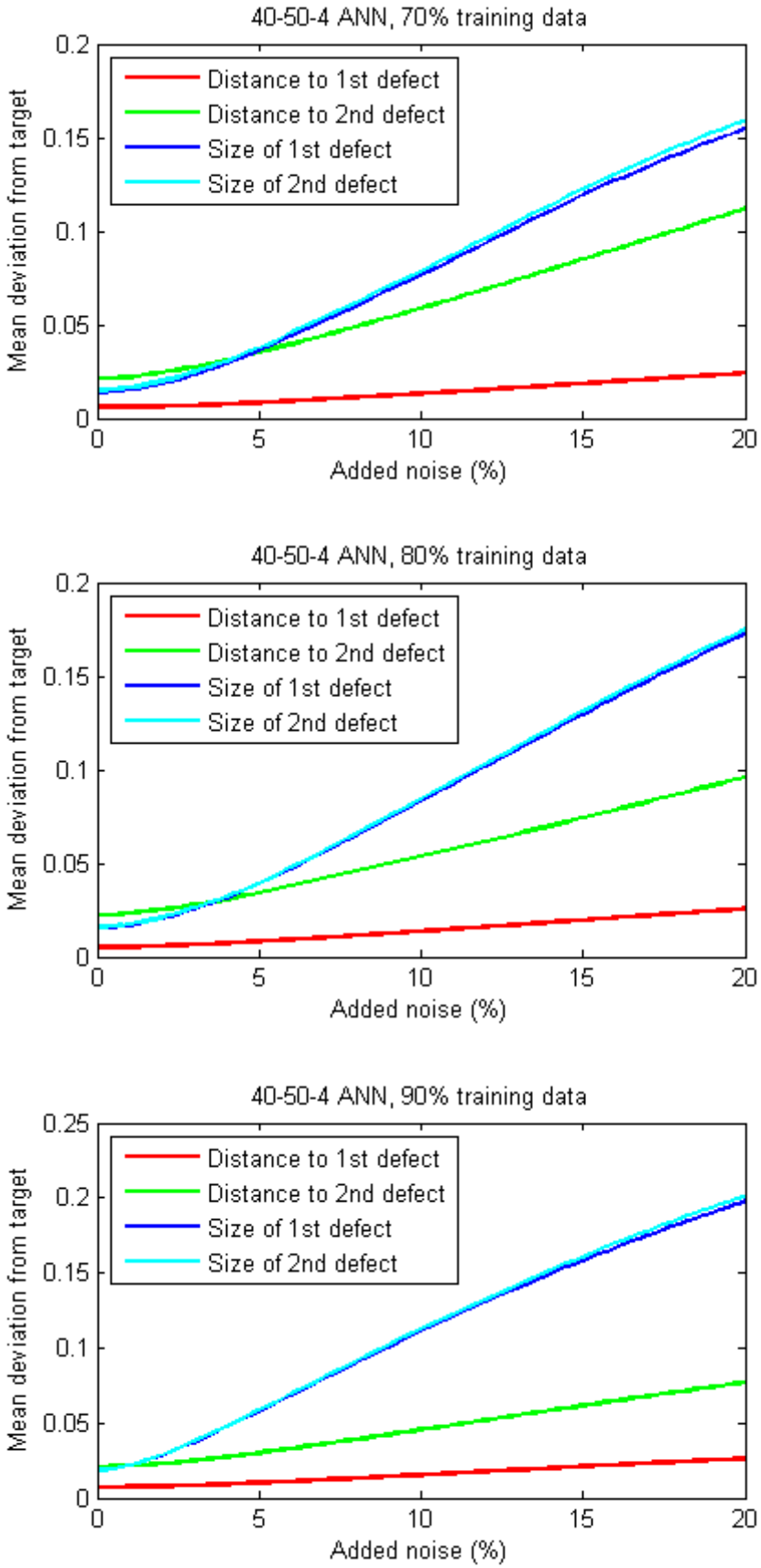


Figure 6.24 ANN performance with added noise, 40-50-4 configuration

6.3.5 Further ANN investigation

Following the results presented in section 6.3.4.2, it was clear that the addition of noise severely affected the ANNs' ability to accurately estimate the sizes of the defects, and less severely the ability to accurately estimate the location of the defects. To investigate whether a more detailed representation of the pulse-echo signal would make an ANN more robust when presented with noisy data, a network was designed that used the entire 'raw' signal from the ultrasonic simulations as its input. All signals in the data set were normalised so that the data points lay in the range [-1, 1]. Using this method, each case had 500 data points, corresponding to the vertical displacement at the transducer location at 0.1μs intervals. Although requiring an ANN with a much larger input layer (Margrave 1999), this method was considered as it required no feature extraction from the ultrasonic data. An ANN with 500-252-4 architecture was used, trained for 10,000 epochs using the scaled conjugate gradient method. The accuracy of this ANN was very similar to that of the previous ANNs presented in sections 6.3.3 and 6.3.4 in terms of locating the two defects, and slightly less accurate when estimating the defects' sizes. However, the larger ANN using the full input signal performed measurably better than the smaller networks when noise was added to the input data. Further investigation in this area, including the optimisation of the ANN architecture, is considered as further work in section 9.3.

6.3.6 Summary of ultrasonic method

A method has been presented to extract features from ultrasonic A-scan signals and present these to two different ANNs in order to determine the distance to two defects and the size of each defect. The method has produced encouraging results, although it has been noted that a

data set representative of the entire range of possible inputs, and thus potentially of the order of several thousand cases, may be required to provide adequate training for the ANNs. Estimations for the distance to the first defect were generally accurate, with a mean error of around 0.01 of the normalised value, whereas estimations for the distance to the second defect showed a mean error of around twice this value. Similarly, estimations for both defects' sizes typically had a mean normalised error of around 0.02. Two ANN layouts were considered, and both performed significantly better when trained with over 80% of the available data. The performance of the smaller network was slightly better than the larger network when presented with noisy data, although it was observed that the addition of noise made a significant contribution to error. Neither network performed well when large amounts of noise were added to the data set, particularly in terms of estimating the defects' sizes.

A trial investigation using the full ultrasonic signal with no feature extraction was undertaken. An ANN trained and assessed with these 'raw' data returned very similar accuracy to the two ANN layouts used in the main investigation, but was more robust when presented with noisy data.

6.4 Summary of two defect study

An ANN-based method of decoupling two overlapping windowed sinusoidal signals has been presented. It has been demonstrated that the time and height of each peak in the cross-correlated combined signal contain enough information to allow an ANN to estimate each signal's magnitude and starting point. Based on this success, an investigation was carried out using the same method to decouple two overlapping signals from an ultrasonic pulse-echo

test. The results from the investigation show that an ANN can estimate the position of the first (closer) defect to a good degree of accuracy, and although estimation of the second defect's position and the size of each defect was less accurate, the mean normalised deviation from the target values was around 0.02. The results show that an ANN is able to provide more information that would otherwise be available to an operator, and that the method can provide realistic estimates of defects' sizes and locations.

CHAPTER 7

GENERAL DEFECT IDENTIFICATION SYSTEM

7.1 Introduction

As stated in the introduction to this thesis in chapter 1, the aim of the research was to see whether an ANN-based system could take data from simple non-destructive tests and give more information regarding the component's condition than would otherwise be available to an operator. Of particular benefit would be a system that could analyse data in a very short time, thus providing a method of online inspection that gave more information than just passing or failing components. Following the work completed using the ultrasonic method in chapters 4, 5 and 6, an investigation was carried out into creating a complete system that would take an ultrasonic A-scan signal and automatically analyse it to provide as much information regarding the component's condition as possible. This information was to include an indication of the type of defect found (in this case either no defect, one defect or two defects), and an estimation of all defects' positions and sizes. The complete system was trained using numerical data generated using the finite element method, and assessed using previously unseen numerical data to which varying amounts of artificial noise had been added.

7.2 Description of system

Following Rytter's four levels of damage identification (Rytter 1993) the first stage of a non-destructive evaluation system should be the detection of damage. In this case, the detection of damage was also considered to include the type of condition present, i.e. the number of separate defects in the component. The initial stage of the system, therefore, was an ANN classifier that assigned input data into three discrete categories: no defect, one defect or two defects.

Once the case had been assigned a category, the data were passed to the previously trained ANN for that category. Cases of single defects were passed to an ANN trained with data generated from FE simulations of cases with single defects, and cases of two defects were passed to an ANN trained with data generated from FE simulations of cases with two defects. The outputs from these ANNs were the normalised depth and size of each defect present, which were subsequently scaled to return them to values in mm.

The overall system, therefore, operated as a 'black box' whose input was the A-scan signal from an ultrasonic test, and whose output was the number of defects present along with their size(s) and location(s).

7.2.1 Preprocessing and feature extraction

In order to present the data from the ultrasonic tests to the ANNs, a small amount of preprocessing was required, as was the case in chapters 4-6. This preprocessing took several stages. Initially, the signal from a reference component with no defect was subtracted from the ultrasonic signal, so that the resultant signal was the difference between the two (Yagawa and Okuda 1996). The times and heights of all peaks in this signal were used as features to enable the classifier ANN to correctly classify each case, and were found using the algorithm previously developed in MATLAB 7.1, included as appendix D. For the case of no defect, a perfect signal would be a flat line at zero amplitude, representing a signal with no defect from which an identical signal had been subtracted. In order to make the classifier realistic, the case of no defect was taken to be this flat line with some added noise at 100kHz centre frequency. A regular sinusoidal function at a frequency of 100kHz was assumed, with an amplitude of between 5 and 15% of the maximum amplitude in the data set. The amplitude

of each peak in this signal was chosen at random within the range specified, and the time of each peak was varied at random by up to 10% of its value. By adding some noisy non-defected signals, the classifier ANN could be trained to distinguish between a feature (some kind of defect) and background noise where no defect was present.

Once the classification had taken place, features from the signals (in the case of one or more defects' being present) needed to be presented to the appropriate ANNs for the single defect or two defect case. In the case of a single defect, the time and height of the first peak were used as features to describe the case to the single defect ANN. In the case of two defects, the signal from which the reference signal had been subtracted was cross-correlated with the ultrasonic pulse shape, as described in section 6.2.1.3. The times and heights of all peaks in this cross-correlated signal were used as features to describe the case to the two defect ANN. All peak heights and times were normalised by dividing by the largest absolute value in the entire data set, thus the overall range of values for all cases was [-1, 1].

The complete defect identification system is shown as a flowchart in figure 7.1.

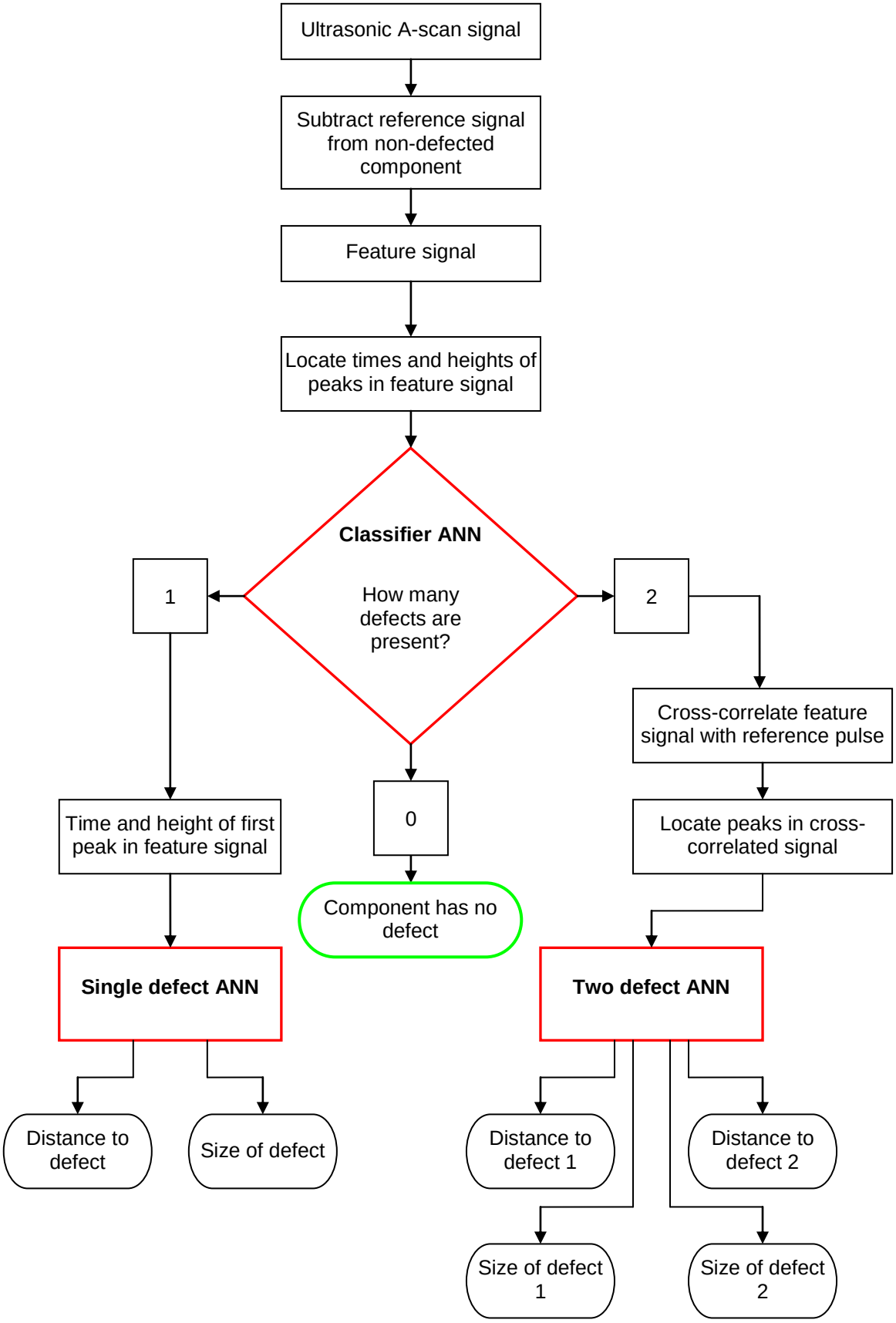


Figure 7.1 Flowchart diagram of the general defect identification system

7.2.2 Development and optimisation of ANNs

The first ANN in the system was used to classify cases into three discrete categories: no defect, one defect or two defects. The output from this classifier would then inform the system as to which ANN to use to assess the data presented. Two ANNs were used in the second stage as quantifiers, which were based on those used in chapter 4 (for the case of a single defect) and chapter 6 (for the case of two defects). The data set used for this study comprised 2000 cases of no defect, 1750 cases of a single defect and 2705 cases of two defects. The defects' sizes ranged from 1mm to 4mm in the 2 defect cases, and 1mm to 6mm in the single defect case. The architecture of all ANNs was optimised to reduce their size as much as possible whilst still maintaining reasonable accuracy, especially when presented with data to which random noise had been added. All ANNs in the complete system were trained with 90% of the available data, and were assessed with the remaining 10%. The layout and optimisation of these ANNs is described in this section.

7.2.2.1 Description and optimisation of classifier ANN

The classifier ANN was designed to have three discrete outputs representing 0, 1 and 2 defects, with the constraint that only one of these outputs could be positive, i.e. if there was no defect, there could not also be 1 defect. In order to achieve this, a 'competitive' layer of three neurons was used as the final layer of the classifier ANN. The neurons in this competitive layer used the step activation function to give an output of either 0 or 1. In the case of two neurons attempting to output a value of 1, the input values would be compared, and the neuron with the larger input value would output 1 whilst the other would output 0. In

this way, classification was constrained to have no 'unsure' regions. The schematic of this ANN is shown in figure 7.2.

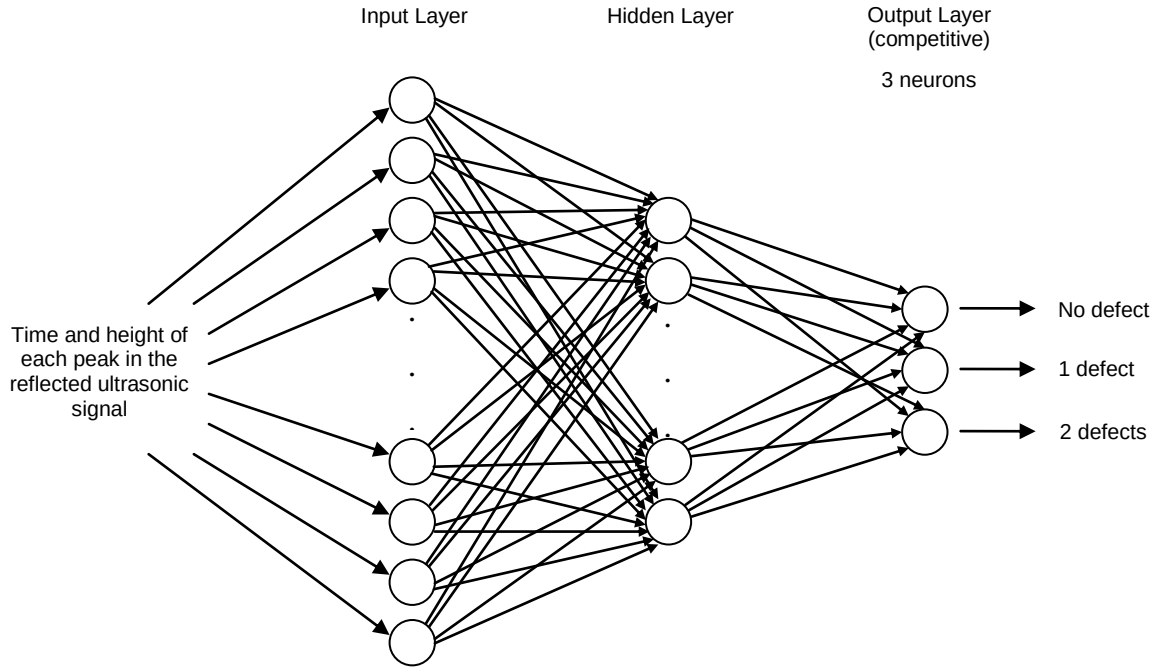


Figure 7.2 General layout of classifier ANN

The initial network architecture had 26 input neurons, representing the time and height of up to 13 peaks in the signal, 14 neurons in the hidden layer and 3 outputs. Tan-sigmoid activation functions were used in the input and hidden layers, and step functions used in the output layer. The classifier ANNs were trained using the scaled conjugate gradient method for 2,000 epochs. This number was significantly lower than the 20,000 epochs used to train the quantifier ANNs, as it was found that the classifier was able to draw meaningful links between input vectors and output class much more quickly than the function approximators.

A parametric process was used to reduce the number of neurons in the input layer from 26 to 2 in multiples of 2, whilst reducing the number of neurons in the hidden layer from 14 to 2, also in multiples of 2. The accuracy of the ANN was measured by how many cases were incorrectly classified when the trained ANN was presented with the assessment data set, both in the case of no noise and with artificial noise added following the method described in section 4.4.2. Following this parametric process, it was found that the ANN performed best with a layout of 6-2-3, thus this was the architecture chosen for the remainder of the study. This layout is shown graphically in figure 7.3.

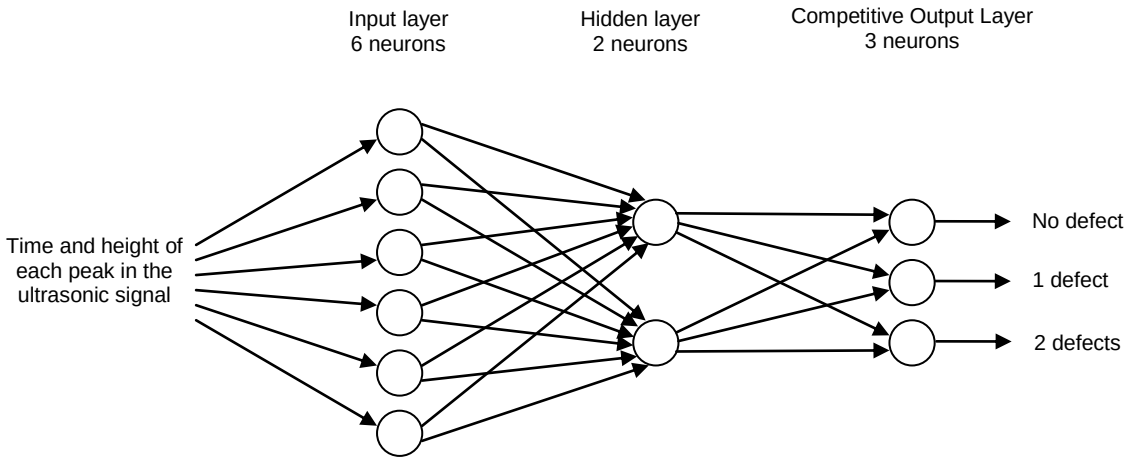


Figure 7.3 Optimised layout of classifier ANN

7.2.2.2 Description and optimisation of quantifier ANNs

Two separate quantifier ANNs were employed; one for the case of a single defect and one for the case of two defects. Each ANN was similar to those described in chapters 4 and 6, although the architecture of the single defect ANN was adjusted to optimise the layout, following the procedure described above where each layer was reduced in size and the

accuracy monitored using assessment data with and without added noise. The goal of optimisation of the ANN architecture in this case was to ensure that the network was as accurate as possible when presented with data with no noise, but also sufficiently flexible to be able to return reasonable outputs when presented with noisy data. The optimisation process in this case was not so straightforward, as with two output values an architecture that may minimise error for one variable would not necessarily do so for the other, especially when noise was added to the assessment data set. For the single defect ANN, the optimum architecture was found to be 14-12-2, whereas for the two defect ANN the 26-6-4 layout as used in chapter 6 was found to perform better than any smaller networks during optimisation, and not significantly worse than networks with considerably larger architecture, such as the 40-50-4 network described in section 6.3.3. The ANN architecture had to be larger for the two defect situation as the relationship between inputs and targets was considerably more complex than in the single defect situation, thus requiring more weighted connections between neurons to fully describe this relationship. Both ANNs used tan-sigmoid activation functions in the input and hidden layers, and linear activation functions in the output layer. Training was conducted using the scaled conjugate gradient method for 20,000 epochs.

7.3 Results

In this section, the results of each network are first considered individually, then the performance of the complete system as a whole is presented. All defects lay in the region specified in section 4.2.1 and shown in figure 4.1. The training data set comprised 90% of the cases, and the assessment data the remaining 10%.

7.3.1 Number of defects

Using the optimised ANN layout of 6-2-3, noise of between 0% and 40% was added to the assessment data set using the method described in section 4.4.2, and the number of incorrectly classified cases monitored as the amount of noise increased. The assessment data set contained 646 cases. Figure 7.4 shows the numbers of incorrectly classified cases expressed as a percentage of all cases in the assessment data set against the amount of noise added. It can be seen from figure 7.4 that the classifier ANN was robust when presented with data with up to 4% noise, and then the number of incorrectly classified cases increased linearly with the amount of noise added. For the purpose of this thesis, the main concern was the point at which cases started to be incorrectly classified, which can be taken as 5% added noise. That said, even with added noise of 40%, 93% of cases were correctly classified, which is comparable with many previous studies using data without added noise (Simone, Morabito *et al.* 2001; Gang, Takahashi *et al.* 2002).

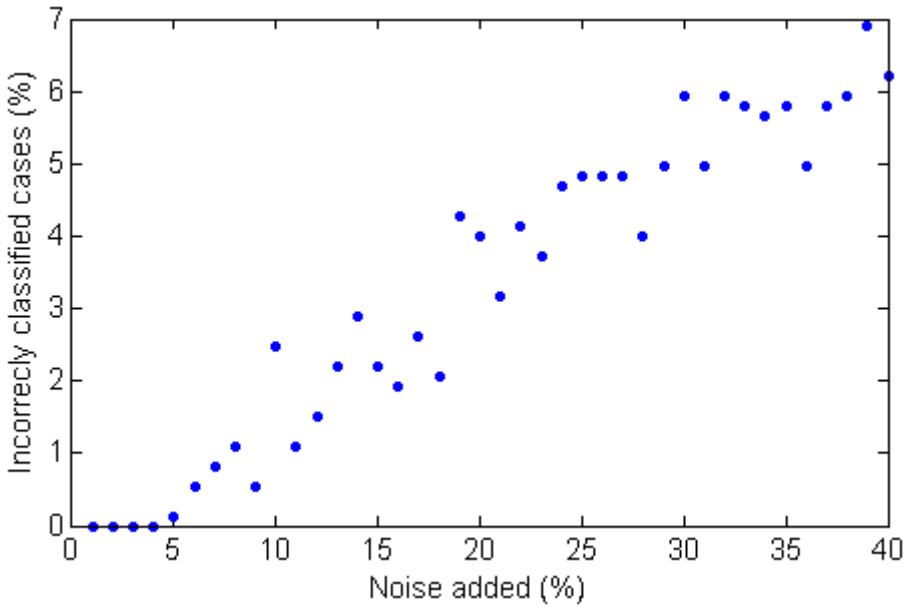


Figure 7.4 % of incorrectly classified cases against added noise for 6-2-3 classifier ANN

An important point to note regarding the incorrect classification of cases is that the classifier ANN did not make any erroneous classifications that suggested the component was free from defects; all the errors in classification were between single and two defect cases, or in incorrectly classifying a no defect case into one of the other two categories.

7.3.2 Location of defects

Two separate networks were used to locate and size defects, depending on the number of defects present. In the case of a single defect, the ANN returned very accurate results with the noise-free data set, and the accuracy reduced proportionally as noise was increased. Figure 7.5 shows the results from the single defect ANN with noise free data and with data to which 10% of random noise was added. As can be seen, the results are similar to those observed in chapter 4, where the ANN was reasonably robust when presented with data to which noise had been added.

A useful metric to evaluate the performance of the ANNs in locating defects was to determine the number of cases where the ANN's estimation was correct to within 5, 10 or 15% of the target value. The results from the single defect ANN are shown in table 7.1 along with the corresponding amount of noise added to the assessment data set. The total number of cases for the single defect ANN's assessment was 219.

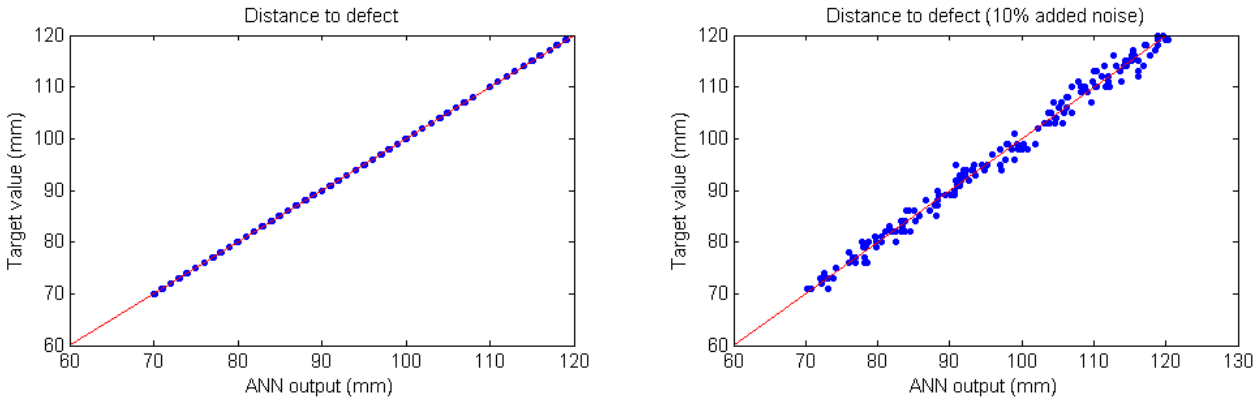


Figure 7.5 Accuracy of single defect ANN in locating a defect, with noise-free data (left) and data with 10% added noise (right)

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	219	219	219
2	219	219	219
5	219	219	219
10	219	219	219
15	212	219	219
20	197	219	219

Table 7.1 Number of cases within certain range of target for distance to defect (single defect ANN, 219 cases in total)

As can be seen from figure 7.5 and table 7.1, the ANN's estimation of the distance to the defect in the case of a single defect was generally very good. Even with 20% of added noise, which represents a severely corrupted signal, all estimations were within 10% of the target value.

In the case of two defects, the accuracy in estimating the position of the first defect was generally very good, whereas the location of the second defect was much less accurate, as observed in chapter 6. Figure 7.6 shows typical results from the two defect ANN with noise-free data and with 5% added noise. As can be seen, the accuracy of the two defect ANN was much lower than the single defect ANN when no noise was added to the input data, but the accuracy did not suffer greatly when 5% noise was added. Although the estimations of the distance to each defect were not as accurate as was initially hoped, they were still considered to be useful information, especially given that this information would not be available to an inspector looking at the reflected wave pattern.

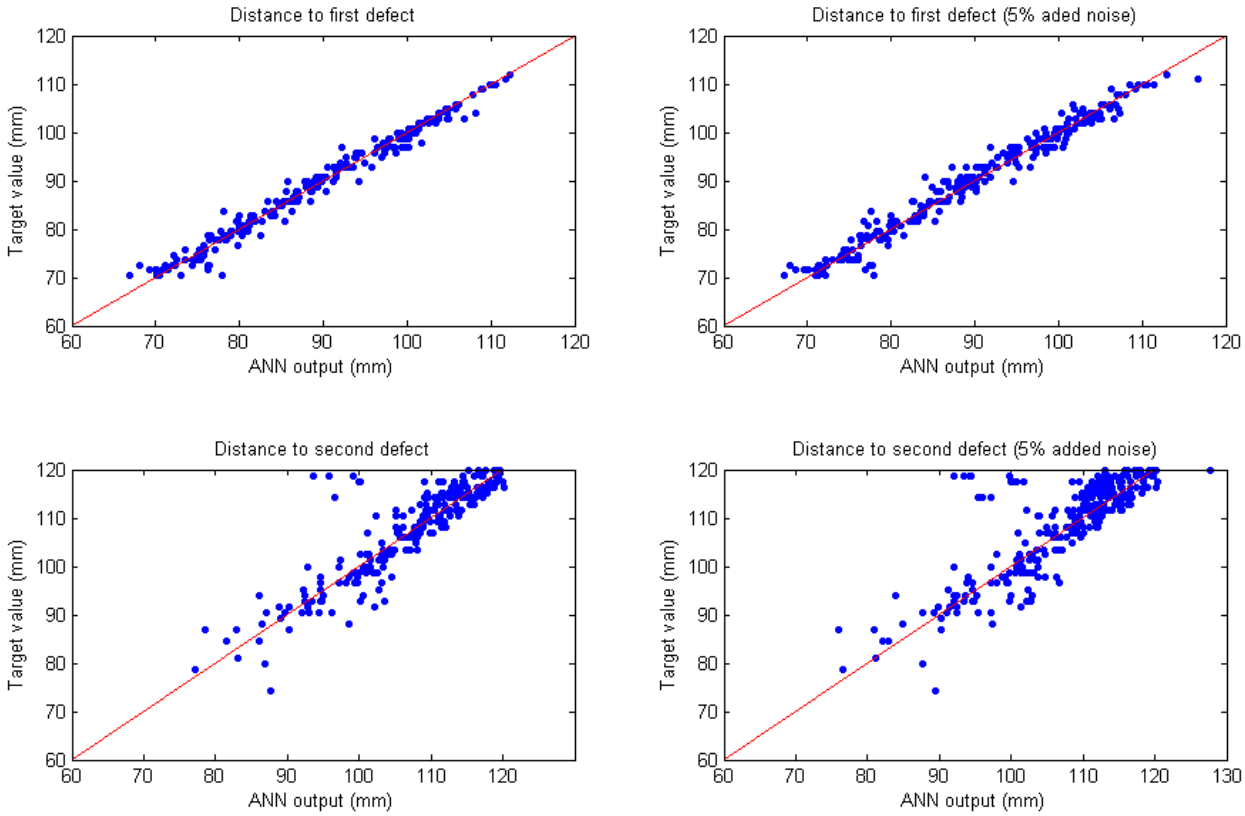


Figure 7.6 Accuracy of two defect ANN in locating defects, with noise-free data (left) and data with 5% added noise (right)

The accuracy of the two defect ANN was again measured by looking at the number of cases within certain ranges of the target value. Tables 7.2 and 7.3 show the accuracy of the ANN's estimation of distance to each defect along with the amount of noise added to the input data. The total number of cases for the two defect ANN's assessment was 266.

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	264	266	266
2	264	266	266
5	263	266	266
10	258	266	266
15	254	264	266
20	246	263	265

Table 7.2 Number of cases within certain range of target for distance to first defect (2 defect ANN, 266 cases in total)

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	237	258	259
2	235	255	259
5	229	250	259
10	216	247	256
15	198	239	250
20	184	225	244

Table 7.3 Number of cases within certain range of target for distance to second defect (2 defect ANN, 266 cases in total)

As can be seen from tables 7.2 and 7.3, although the accuracy of the ANN's estimation was reduced by the addition of noise, the estimation of the distance to each defect was within 10% of its target for 100% of cases (first defect) and 93% of cases (second defect), even with 10% noise added. With no noise, 99% of the estimations of the first defect's position were within 5% of the target, and 97% of the estimations for the second defect's position were within 5% of the target. This value of 5% corresponds to 6mm in distance.

7.3.3 Size of defects

The same ANNs for single defect and two defect cases were used to estimate the size of the defect. Again, artificial noise was added to the data and the performance of the ANNs observed. In the case of a single defect, a high level of accuracy was observed, even when noise was added to the input data, as was found in chapter 4. Figure 7.7 shows the effect of adding 10% noise to the assessment data set for the single defect ANN, and table 7.4 shows the number of cases within certain bounds of accuracy as the amount of noise increased.

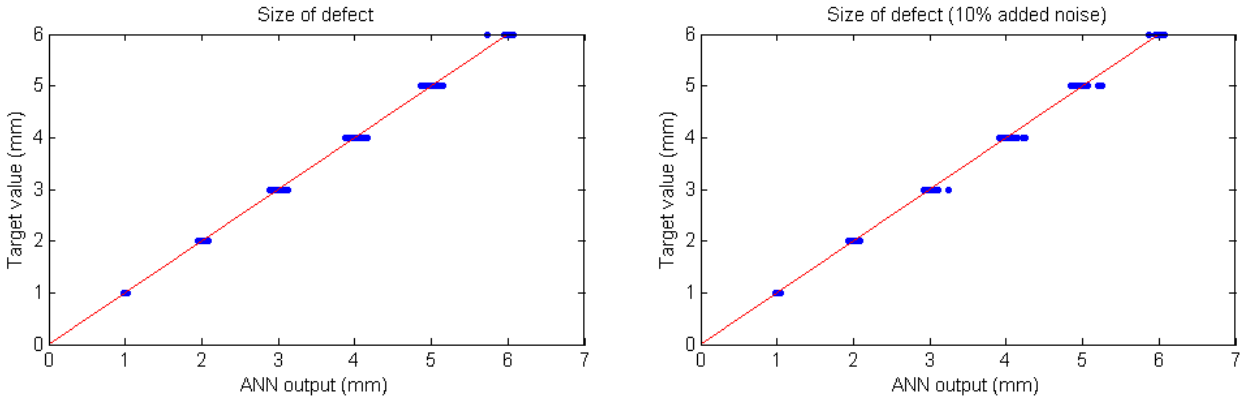


Figure 7.7 Accuracy of single defect ANN in sizing a defect, with noise-free data (left) and data with 10% added noise (right)

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	219	219	219
2	219	219	219
5	218	219	219
10	218	219	219
15	213	216	219
20	213	218	219

Table 7.4 Number of cases within certain range of target for size of defect (single defect ANN, 219 cases in total)

As is apparent from figure 7.7 and table 7.4, the accuracy of the ANN's estimation of a single defect's size was very good, even when large amounts of noise were added to the data set.

The performance of the two defect ANN was not as accurate as the single defect ANN, as was observed in chapter 6, but nonetheless the estimations of defect size were considered very useful information which would otherwise not be available from an ultrasonic A-scan. The performance of the two defect ANN in sizing defects is shown in figure 7.8, and tabulated in tables 7.5 and 7.6.

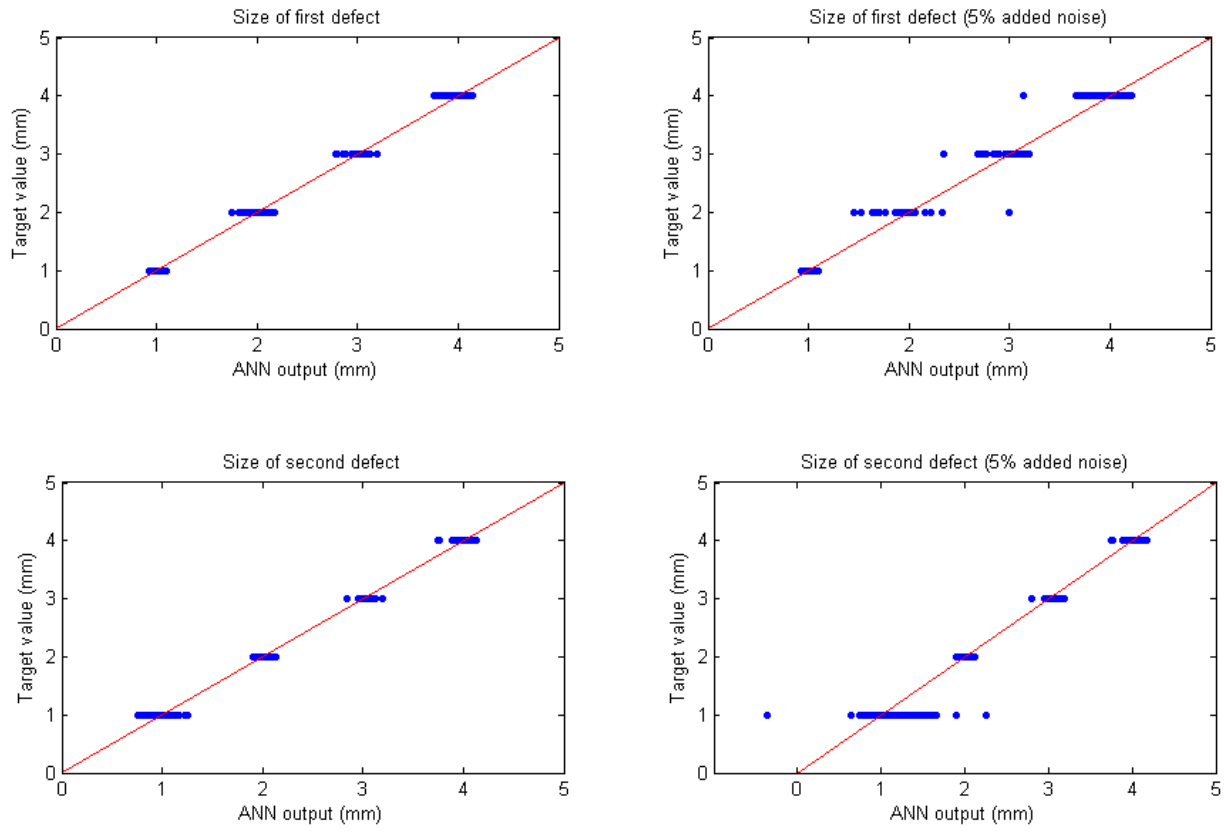


Figure 7.8 Accuracy of two defect ANN in sizing defects, with noise-free data (left) and data with 5% added noise (right)

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	234	264	266
2	229	256	263
5	212	248	253
10	193	230	237
15	175	215	227
20	163	202	215

Table 7.5 Number of cases within certain range of target for size of first defect (two defect ANN, 266 cases in total)

Amount of noise (%)	Number of cases within 5% of target	Number of cases within 10% of target	Number of cases within 15% of target
0	175	235	249
2	174	226	243
5	160	204	224
10	148	180	203
15	139	163	186
20	138	156	174

Table 7.6 Number of cases within certain range of target for size of second defect (two defect ANN, 266 cases in total)

As can be seen from tables 7.5 and 7.6, even with no noise, the accuracy in estimating each defect's size was fairly low. With no noise, 99% of the first defects and 88% of the second defects were sized within 10% of their correct value. Whilst this is not as accurate as would be ideal, the results are still considered to be highly valuable to an operator or in an online system. In the case of two defects in a component it may occasionally be necessary to conduct a more thorough manual inspection. In such cases the location of the second defect,

even if this is given as an approximate value, is a very useful output from the system. As with the distance to each defect, the sizes of each defect would not be able to be inferred from a normal A-scan display, thus the ANN-based system provides considerably more information regarding the defects than would otherwise be available.

7.3.4 Overall system results

The overall system followed the procedure flowchart, shown in figure 7.1, and was created in MATLAB 7.1. Input data were preprocessed to find the times and heights of each peak in the signal, as described in section 7.2.1, then presented to the classifier ANN. Depending on the outcome, data were then passed to the correct ANN for that class and the outputs recorded. The results of the overall system are presented in this section. In order to assess how well the system worked, cases were tracked from input to output and the accuracy of each parameter measured. Thus, if a case was incorrectly classified, it would be passed to the wrong ANN, generally producing a highly erroneous result. To measure the system's performance, if the outputs of the system for the distance to each defect and size of each defect were within 10% of the target value, the case was considered to be a correct output. The results are presented in tabular form in table 7.7. Distance 1 relates to the distance to the only defect in the case of a single defect, or the distance to the first defect in the case of two defects. Distance 2 relates only to the case of two defects. Sizes 1 and 2 follow the same format.

Added noise (%)	Correctly classified	Distance 1 within 10%	Distance 2 within 10%	Size 1 within 10%	Size 2 within 10%
0	100%	100%	97.0%	99.6%	88.3%
2	100%	100%	95.9%	97.9%	85.0%
5	99.8%	99.8%	92.9%	96.1%	76.5%
10	97.3%	95.1%	90.3%	90.1%	65.8%

Table 7.7 Performance of the complete system

As can be seen, with no noise added to the signal, the system was capable of classifying all cases correctly, and returning good estimations for the location of the first and second defects and the size of the first defect. The accuracy of the estimation of the size of the second defect was not so good, although it is considered that the ANN-based system has already performed a highly valuable task in identifying that there are two defects rather than one. In practice, it may be necessary to assign a level of confidence to each estimation to account for the spread in the ANN's estimations of the size of the second defect.

7.4 Summary of the general defect identification system

Following the results from chapters 4, 5 and 6, a complete ANN-based 'expert system' has been presented to characterise defects. The system takes ultrasonic A-scan data and returns information regarding the presence, location and size of defects that would not otherwise be available to an operator looking at the same data. Particular importance is placed on the fact that all cases of two defects were within one wavelength of each other, thus the ANN-based system effectively decouples two overlapping signals. Although the system was assessed with numerical data, it is anticipated that with further development of the FE model, and with a means of achieving uniform coupling conditions for the ultrasonic transducer, the system

will be able to return reasonable results when presented with experimental data. In terms of accuracy, estimates for the class of the component (no defect, one defect, two defects) were generally very good, even in cases where large amounts of noise were added. For the case of a single defect, the estimation of the position and size of the defect were generally very accurate, with 100% of cases typically within 5% of the target value. For the case of two defects, estimations for the first defect's position were generally very good, although the estimations for the location of the second defect and estimations for both defects' sizes were less accurate.

It is considered that the additional information provided by the ANNs by the 'expert system' presented in this chapter is very valuable, as an operator looking at an A-scan screen would not be able to discern the position or size of two defects in a particular case where the ultrasonic signals overlapped. Additionally, in the case of automated inspection, which tends to use an amplitude threshold to identify defects, two signals overlapping could easily have a maximum amplitude below the threshold at which the component would fail, thus passing a component that should fail the inspection process. The ANN-based system presented in this chapter avoids that possibility by analysing every signal and returning a result, regardless of the maximum amplitude.

CHAPTER 8

DISCUSSION

8.1 Introduction

This chapter contains a discussion on the material covered in this thesis, and assesses the viability of the novel FEM/ANN/Experimental 'expert system' presented in chapter 7. The accuracy of the individual ANNs used in this thesis is compared to that of similar systems in previously published work, thus their performance is objectively evaluated. Similarly, the correlation between numerical and experimental results is discussed.

8.2 General limitations on defect detection

In any form of non-destructive testing, it is not necessarily possible to detect every single defect, as there will always be the possibility of defects existing below detectable size, or the possibility that the chosen inspection method is not suitable for detecting a particular type of defect. However, as stated in the introduction to this thesis in chapter 1, the goal of NDT/NDE is not to pronounce a component perfectly fault-free, but to pronounce it fit for service, even if some defects may be present. Three simple methods of non-destructive testing have been presented in this thesis, as a means of generating data to train ANNs which can subsequently be assessed using numerical or experimental data. All of these methods were shown to be capable of detecting defects of 1mm in size using the numerical methods, although some issues may exist when trying to accurately locate small defects experimentally. If an experimental measuring device is to be used, detectable defects must be large enough to create a measurable difference in the parameter being measured. In the case of the impact method, the measurable displacement to detect a 1mm defect would need to be in the order of 0.1μm, which many devices can manage without difficulty. In the case of the modal analysis method, the natural frequencies would need to be measurable to within 0.5Hz in the range of 200Hz to 27kHz, in order to locate a 1mm defect. This may not be possible,

and although the method worked very well using numerical data, the probability of detecting defects of 1mm in practice using this method may be fairly low. For the ultrasonic method, defects of 1mm were measurable both numerically and experimentally. Traditional theory states that ultrasonic waves will only be reflected by features equal to or greater than the wavelength (Hull and John 1989), although modern techniques have demonstrated that features much smaller than one wavelength can be detected (Batra and Chaskelis 1985).

Inspection methods, therefore, must be appropriate both for the size of defect they are trying to detect, and for the particular application or component geometry.

8.3 Correlation of FEM and experimental results

Three methods of non-destructive testing have been investigated in this thesis, all of which have been shown to be able to produce outputs that contain enough information for an ANN to locate defects as small as 1mm in size. This section discusses the issues surrounding validation of the numerical models with experimental data.

8.3.1 Impact method

In chapter 5 the issues surrounding experimental testing using the impact method were mentioned, and the difficulties in generating repeatable and reliable data were highlighted. Although the impact method seemed a suitable, simple non-destructive test, experimental validation of the FE model proved very hard to achieve. If defects as small as 1mm are to be detected and assessed, the testing method must be extremely repeatable in order to overcome the effects of mounting a specimen slightly differently, or even the effects of manufacturing tolerances in the specimen itself. The main issue with the impact method was in the

measurement of displacements at precise locations within a very short timeframe. Published work using similar methods tends to use laser vibrometry for measuring displacement (Ishak and Liu 2002) although it is feasible that other forms of non-contact techniques such as digital image correlation may also be able to provide a means of accurately measuring displacements, assuming a very high speed camera is used. In the experimental testing carried out for this thesis, the only available method of measuring displacement was through the use of surface strain gauges, and it was found that the results generated from the tests were not sufficiently accurate using this technique. It is possible that by incorporating some kind of trigger device and load cell between the pendulum and the bar that the precise time of impact and force of impact could be measured. The trigger could be used to start the recording software, thus enabling the strains to be recorded at the desired time after impact. However, there remains the issue of simplicity of the test, and using strain gauges to test a specimen for defects is certainly not an acceptable method for online inspection due to the time taken to mount the gauges and also the cost of the gauges themselves. Although some results using the impact method showed the effects of the dilatational wave passing through the specimen, the test did not produce sufficiently repeatable results to gather any data to present to the ANNs. For this reason, and because of the immediate availability of ultrasonic measuring equipment, the impact method was not pursued any further in the course of this research.

8.3.2 Modal analysis method

As stated in chapter 4, the modal analysis method produced extremely good results from the numerical model, but these results relied on very small differences in the natural frequencies which would be very difficult to measure in practice. Typical differences in natural

frequency between a defected and a non-defected component were of the order of around 0.5% in the range of 200Hz to 27kHz, which would mean that the testing method would need to be extremely controlled in order to avoid changes to natural frequency due to mounting conditions, manufacturing tolerances or material properties of the component. When presented with numerical data to which very small amounts of noise had been added (typically around 1%), the ANNs' estimations of defect position were extremely poor, indicating the very high sensitivity of the method. Although the method produced good results from the numerical simulation with no noise added, it was not considered possible to measure natural frequencies to a sufficient level of accuracy in practice, thus the method was not explored experimentally in the course of this research.

8.3.3 Ultrasonic method

The ultrasonic method was attractive as it involved a very simple test, and experimental equipment was available. However, correlating data between numerical simulations and experiment was not straightforward. The first issue in matching the numerical and experimental results was the effect of ultrasonic frequency. In the parametric simulations conducted, a frequency of 100kHz was used, typical of Lamb wave inspections of plates, bars and pipes (Rose 1999). In the case of the steel bar analysed, this meant that only the fundamental symmetric mode, S_0, and the fundamental antisymmetric mode, A_0, could be present, thus avoiding some of the unwanted effects of dispersive higher modes' being present. The experimental equipment available had an operating frequency of 5MHz, which created some unexpected results such as the head wave described in section 5.3.2. Whilst this was not a major barrier to gathering data to present to the ANNs, it did mean that some human input was required to extract features from the experimental data, which was not

desirable. Additionally, the effect of the higher frequency was to reduce the wavelength of the ultrasonic wave from around 60mm to around 1.2mm, meaning that it was not possible experimentally to place two defects longitudinally within one wavelength of each other. Each defect was represented as a circular hole with a diameter of at least 1mm, thus two defects in close longitudinal proximity would still occupy a distance greater than one wavelength, even if they were both of the smallest size. In this thesis, the times and height of the peaks in the reflected wave were used as features to train and assess the ANNs, and although these parameters will vary with the frequency of operation, the distance between peaks will be inversely proportional to the frequency of inspection. When considering the time and height of the first peak, as was the case for the single defect characterisation in chapters 4 and 5, the results in section 5.3.4 show a good correlation between the ANNs' estimates of distance to the defect and the correct defect locations, once the experimental data were calibrated with the numerical data. The effects that the frequency of inspection may have on the speed of the ultrasonic wave are therefore easy to compensate for in the time domain using a scale and shift calibration method as presented in section 5.3.3. The shift term in this method also made it possible to compensate for any delay that is incorporated into the software driving the ultrasonic transducer, thus the measured start time when recording data can be exactly matched between experiment and simulation.

The next issue causing a discrepancy between the experimental and FE results was the effect of variable coupling conditions. The transducer used was a hand held piezoelectric unit, and was coupled to the specimen using a water-based gel. Although this is a standard coupling agent, small variations in the thickness of the coupling layer, and in the surface finish of the test specimen, can affect the way in which the ultrasonic pulse is transmitted into the material. Compensation techniques for coupling conditions exist if multiple probes are used

in a 'pitch-catch' or time of flight diffraction (TOFD) situation (Achenbach, Komsky *et al.* 1992), but when using a single probe the effect of coupling conditions is likely to cause differences in the amplitude of the received signal, even if these differences can be controlled to be very slight. For the purpose of this thesis, the underlying goal was to use simple non-destructive testing techniques to generate data that could be analysed by ANNs, thus contact transducers such as the one used were attractive. Additionally, although it would be possible to use two contact probes with the available equipment, the software does not have the capability of allowing each probe to act as both a transmitter and receiver, thus compensation techniques such as that proposed by Achenbach, Komsky *et al.* (1992) cannot be applied. It is possible that non-contact transducers, EMATs, or laser generated ultrasound may produce more repeatable results, but the use of these may mean that the testing becomes more complex, thus would defeat the object of the thesis of using simple test methods.

The effects of attenuation were evident from the experimental testing, as the amplitude of the reflected pulse from a defect decreased as the squared distance to the defect increased. The numerical model was created without any specified attenuation parameters or damping coefficients, although it was found that by modelling the full width of the plate, rather than the plate thickness, the numerical results exhibited a similar 'inverse square law' attenuation to that observed during experimental testing. As it was, it was possible to compensate for the effects of attenuation by using a correction procedure, which reduced the variation in amplitude between signals from similar sized defects, as described in section 5.3.3.

To summarise the discussion on the ultrasonic method, it is clear that the effects of frequency and attenuation are significant when creating a numerical model of an ultrasonic test. Although it was possible to compensate for the variation of the speed of sound between the numerical model and the experimental results, allowing the ANNs to accurately estimate the

distance to a defect, the small variation in transducer coupling conditions meant that estimation of the defect's size was slightly less accurate. Ideally, experimental equipment that operates at 100kHz would be used, although that was not available for the purpose of this thesis. Alternatively, a numerical model operating at 5MHz could be created, although as this would require a very fine mesh size and very small timesteps, completing a parametric study of several thousand cases at this frequency may be very time-consuming. As was observed in chapter 5, the effects of head waves were seen in the 2D finite element model of the thickness of the plate, but this model represented the ultrasonic pulse as a plane wave that could not dissipate in the z-direction due to the plane strain condition. A trial model was created to represent the front face of the plate in two dimensions, and it was seen that the reduction in intensity of the wavefront was roughly proportional to the square of the distance travelled (Halmshaw 1991). However, in this model, the effects of head waves were not observed. It is likely, therefore, that a highly accurate simulation of a 5MHz inspection of the steel plate would need to be modelled in 3D. Given that the file size for the 2D models at 5MHz was around 8GB, 3D simulations were considered to be beyond the scope of this research.

8.4 Accuracy of defect classification using ANNs

From the results of the ANN complete system presented in chapter 7, it can be seen that the classifier ANN was able to distinguish between cases where either no, one or two defects were present with 100% accuracy, even when small amounts of noise were added to the input data. Many studies have been published using ANNs as classifiers, although most of these use ANNs to distinguish between types of defect rather than the number of defects present in a signal. Margrave, Rigas *et al.* (1999) quote typical accuracy of their ANNs in classifying

manufactured defects in plates as being between 80-100%, using data from the time and frequency domains. Five different types of defect were included in the study, and testing was carried out using the ultrasonic pulse-echo method. Baker and Windsor (1989) showed that an ANN could correctly classify 100% of weld defects into four categories when six features were used to represent each case. In this study a very large amount of preprocessing was required, however, in order for the ANN to correctly distinguish between cases. Testing in this case was carried out using two ultrasonic probes (a transmitter and a receiver) at three separate angles of incidence. Other authors using ANNs to classify weld defects from ultrasonic signals show correct classification of 83% (Chen and Lee 1993), 80-95% (Simone, Morabito *et al.* 2001), and 93% (Gang, Takahashi *et al.* 2002), all using a range of feature extraction techniques.

The results from the classifier used in this thesis, presented in section 7.3.1, show that with minimal feature extraction from the raw signal, ultrasonic A-scan data can be classified into discrete cases with a high level of confidence. Typical accuracy of the classifier ANN presented in this thesis was 100%, even with small amounts of artificial noise added to the input data. When the amount of added noise increased to 40% (using the method presented in section 4.4.2), the accuracy fell to 93% of cases correctly classified; still on a par with published work where no noise was added to the data. As the classifier made use of a simple test with minimal feature extraction, the technique and results are considered significant. It is interesting to note that in the case of the published papers cited above, no further investigation was carried out into the severity of each flaw, that is, the goal of the ANN systems was solely to classify defects into discrete groups. The novel 'expert system' presented in chapter 7 of this thesis goes a stage further, by classifying defects and then characterising them by estimating their size(s) and locations(s).

8.5 Accuracy of defect quantification using ANNs

In the published literature, work involving ANNs in NDE generally falls into two categories: classification of defects and quantification of defects, with classification of defects receiving much more attention. However, defect quantification using ANNs, both in locating and in sizing defects, has been carried out by several researchers. This section aims to compare the accuracy of previous researchers' results with that of the results in this thesis.

Using an impact method similar to that used in section 4.2, Ishak, Liu *et al.* (2002) trained an ANN using a combination of experimental and numerical data, whose output was the size and location of a longitudinal crack in a beam. Features were extracted from the time and frequency domains in order to train the ANNs. Typical errors between the ANN's estimation and the actual value of the location, length and depth of the crack were of the order of 1-20%, with the majority of error found in the estimation of the depth of the crack from the edge of the beam. It should be noted that in this study the defects modelled were significantly larger in comparison to the overall component than those investigated in section 4.2 of this thesis. However, the results from section 4.2.3 show a comparable level of accuracy, albeit using data only from a numerical model. Typical errors from the impact study are in the order of 2% (the difference between estimated value and target value), both for the x and y co-ordinates of the defect's position. Given that these results compare favourably to those of Ishak, Liu *et al.* (2002), it can be considered that the impact method presented in section 4.2 is a viable method for locating defects, assuming that a suitable means of measuring small displacements can be found.

Using the method of extracting natural frequencies from a numerical model, Sahin and Shenoi (2003) trained ANNs to locate regions of damage, modelled as reduced stiffness, in a composite beam. The regions of damage in this case were the entire thickness of the beam,

and ANNs were employed to estimate the severity of the damage (the reduction in stiffness) and the location of the damage (the distance along the beam at which the damage occurred). Results from this numerical study show an accuracy in the ANNs' predictions far lower than was observed in this thesis from a similar study in section 4.3. Although no data were available to measure accuracy in percentage terms, the spread of values from Sahin and Shenoi's study suggests a typical error in the ANNs' estimations of around 5 - 15%. Based on this level of accuracy, it can be stated that the results found in this thesis are considerably more accurate, with a typical error of around 0.5 - 1%, and that the performance of the ANNs used exceeds that of the ANNs used by Sahin and Shenoi. However, the performance of the ANNs relies on having exceptionally accurate input data. Whilst Sahin and Shenoi did not experimentally validate their findings, it is likely that, because they used much larger defects in relation to the size of the component, their ANNs may perform in a similar way when presented with experimental data. In the case of the modal analysis method used in this thesis, it is clear that the natural frequencies contain enough information to locate a defect within a component, but experimental validation of the method would be necessary to determine the minimum size of defect detectable.

Using various ultrasonic methods, several authors have trained ANNs to estimate the location or size of defects, although in some cases the ANNs have a number of output neurons, each representing a discrete value, rather than a single output neuron whose value can vary continually. Zgonc and Achenbach (1996) trained an ANN to estimate the length of cracks emanating from a rivet hole using ultrasonic data from two transducers. In the final model, 10 output neurons were used, each relating to a particular length of crack. By enabling the output from each neuron to be continually variable, it was possible to interpolate between the discrete values to obtain a precise estimation of the crack length from the ANN. Typical

errors between the ANN's estimation of crack length and the actual value were in the region of 3-7%. The accuracy observed in the single defect ultrasonic study in this thesis was significantly higher than this (errors were typically 0 – 1%), and the accuracy in estimating the distance to the first defect in the two defect study also compares favourably (errors were typically 0 – 5%).

Thavasimuthu, Rajagopalan *et al.* (1996) used the experimental ultrasonic pulse-echo method to train ANNs to estimate the diameter of flat bottom holes, again using separate output neurons to represent discrete values of hole diameter. Holes with diameters between 1mm and 7mm were considered, all with a depth of 5mm. A large amount of feature extraction was required, but the ANNs returned results with very little error; typically around 0-5%. Again, this method used ANNs more as classifiers than as quantifiers, but the results show that even if signals are classed into discrete categories, good estimates of defect size can be obtained. The accuracy is comparable to that of the single defect ultrasonic study presented in section 4.4, and also comparable to that of the distance to the defect observed in the experimental ultrasonic testing in section 5.3. It should be noted that in the course of the research into characterising multiple defects, ANNs using a series of discrete output neurons, each to represent a particular value for distance or size, were briefly investigated. However, the results from these ANNs were considerably less accurate than those found using continually variable output neurons, as presented in this thesis.

Yagawa and Okuda (1996) demonstrated the use of an ANN in locating crack tips, using a numerical model of an ultrasonic test. In this case, the ANN was used as a function approximator with two outputs, relating to the x and y co-ordinate of the crack tip. Each output from the ANN was continually variable. Although the accuracy of the ANN's estimations is only shown graphically, it can be seen that it is comparable to that observed in

the location and sizing of a single defect in section 4.4 of this thesis, and also to the distance to the defect in the experimental ultrasonic testing in section 5.3.

When comparing results to published work, it is clear that for the case of a single defect, the accuracy of the ANNs' estimations in this thesis is comparable to, and in many cases significantly better than, the majority of published work. Although experimental validation at 100kHz was not possible with the available equipment, it is envisaged that with a means of creating more repeatable coupling conditions, or with the means to use two transducers to compensate for the effects of variable coupling conditions, a good degree of correlation could be found between experimental and numerical results, as is reported by many researchers (Zgonc and Achenbach 1996; Lowe, Cawley *et al.* 2002).

In the case of two defects lying in close proximity, no published work was found by the author that had used ANNs to decouple the overlapping signals, thus the methods presented in chapters 6 and 7 are novel. The accuracy of the ANNs used to estimate the position and size of each defect was generally comparable to previously published work, where only one parameter was investigated.

8.6 ANNs as an expert system

This thesis has investigated several methods of non-destructive testing, and has demonstrated that, using simple feature extraction techniques, the results from ultrasonic pulse-echo tests can be used to train and assess an ANN-based 'expert system'. Once trained, this system can output the number of defects in the component, along with the location and size of any defects present, particularly in the case where the reflected signals from two defects overlap. Very good correlation between experimental and numerical results was observed for the case

of estimating the distance to a defect (typical error was around 1-2%), and good estimations of the size of a defect (typical error was around 3-5%). It is anticipated that with a more uniform and repeatable method of coupling the transducer to the test specimen, estimations of the size of any defects could be made with even higher accuracy.

As discussed in section 1.3, fully automated inspection of components, as has been shown in this thesis to be possible with an ANN-based system, is desirable as it removes the possibility of human subjectivity, fatigue, boredom, etc. However, there needs to be a very high level of confidence in any automated system in order for it to be relied upon to make decisions regarding safety-critical components. Windsor (1995) suggests that one use for ANNs in NDE may be as a 'novelty detector', calling for human assistance if any unusual or previously unseen result is detected. The system presented in this thesis takes this application a step further, by using ANNs to provide more detail from an ultrasonic A-scan than would otherwise be available to an operator, thus making the non-destructive evaluation process faster and more reliable. As has been shown in chapter 6, in the case where two defects are within one wavelength of each other, their combined ultrasonic reflections can partially cancel each other, thus reducing the overall amplitude of the reflected pulse. In a traditional 'threshold-based' inspection, such a case would pass if the peak amplitude was below the threshold, even if the case contained two defects whose individual signals each had an amplitude above the threshold. The novel ANN-based system presented in this thesis has been shown to reliably identify cases where two signals overlap, and use features from the signal to characterise each defect, thus providing much more information regarding the case than would otherwise be available to an operator. In terms of online inspection, a system of this type could be used to identify and characterise anomalies rapidly and reliably, and provided that the training data set was sufficiently representative of all expected defect types

and locations, could be relied upon to provide estimations of defect position and size within reasonable tolerances. In the event of a highly anomalous output, for example a normalised value below 0 or above 1, the system could call for human assistance for that particular case.

The novel ANN-based 'expert system' presented in this thesis is therefore suitable for use as an online inspection tool, and in the case of the inspection of high volume, mass produced components, is a very attractive alternative to a human inspector. Not only is the system able to assess cases much more rapidly than a human operator, it is also able to decouple overlapping signals and give a reliable estimation of the position and size of each defect present. In the case of two overlapping signals, it is possible that a human operator may be able to recognise that two signals are present, but he would not easily, if at all, be able to determine the characteristics of each signal.

The 'expert system' presented in this thesis is able to correctly identify cases where one or more defects are present, even if the maximum amplitude of the ultrasonic signal is below that which would be used to reject a component using a traditional 'threshold-based' approach. Following the identification that defects are present, the accuracy of characterising defects in terms of their position and size is comparable to, and in many cases better than, many published studies using ANNs to output a single parameter.

CHAPTER 9

CONCLUSIONS AND FUTURE WORK

9.1 Summary of research project

The main aim of this thesis was to investigate the ability of ANNs to provide more information than would otherwise be available to an operator when presented with data from simple non-destructive tests. Following the suggestion by Windsor, Anselme *et al.* (1993) that ANNs may be able to deal with data without extensive feature extraction, minimal feature extraction was used to express the results from each test to the ANNs, thus making the process of converting test data into a form that could be presented to the ANNs more rapid and less complicated.

Three different methods of non-destructive test were investigated: an impact method, where a stress wave is sent along a bar and its behaviour is modified by the presence of a defect; a modal analysis method, where the natural frequencies of a component are modified by the presence of a defect; and an ultrasonic method, where a defect partially reflects the ultrasonic wave back to the transducer. ANNs were trained using data from these three methods, generated using the finite element method, and were all found to return good results for the position and size of a single defect. In the case of the ultrasonic method, the system remained robust even when artificial noise was added to the assessment data set.

Experimental data were gathered for the impact and ultrasonic methods, although it was found that the data from the impact tests were not sufficiently repeatable to be used to assess the ANNs. The results from the ultrasonic tests were used to represent each test case to three different ANN layouts, and the results for distance to each defect were generally very accurate. Some difficulty was experienced in maintaining repeatable coupling conditions between the transducer and the test specimen, which meant that the amplitude of the reflected signals varied more than in the numerical simulations. This led to a reduction in the accuracy of the ANNs in estimating the size of the defects, although the results gathered were still

within acceptable bounds. It is considered that with non-contact transducers, or with two transducers following the compensation technique presented by Achenbach, Komsky *et al.* (1992) to eliminate the effects of variable coupling conditions, more repeatable results could be obtained, which would lead to greater accuracy in sizing defects using ANN methods.

The case of two defects lying within one wavelength of each other was investigated using the ultrasonic pulse-echo technique, and ANNs were employed to decouple the two overlapping signals from the ultrasonic A-scan data. In this case, it was necessary to cross correlate the resultant A-scan signal with the transmitted pulse. The times and heights of each peak in the cross-correlated signal were used as features to represent the case to the ANNs. Good accuracy was observed for the ANNs' estimations of distance to the first defect, and reasonable accuracy observed for the estimations of distance to the second defect and for the defect size. The work presented in this area is novel, as no previous work was found by the author in the area of decoupling two ultrasonic signals using ANNs.

The successful results from using ANNs to locate and size cases of single defects and of two defects were incorporated into a complete 'expert system', that classified cases according to the number of defects and returned estimates on each defect's position and size. The system was shown to be robust in classifying cases when artificial noise was added to the data, although estimations for the location and size of defects in the case of two defects were reduced when noise was added to the data set. With no noise, the accuracy of the ANNs' estimations of the location and size of each defect was comparable to, and often better than, that of many published studies using ANNs for similar applications.

9.2 Conclusions from this study

From the work presented in this thesis, it is clear that ANNs can be used in NDE to accurately characterise defects when presented with data from non-destructive tests. The novel application of ANNs in decoupling two overlapping ultrasonic signals demonstrates their ability to provide more data from an ultrasonic signal than would otherwise be available to an operator.

ANNs have been shown to return very good estimations for locating single defects in a component using impact, modal analysis and ultrasonic testing methods, and to also return good estimations of a defect's size in the case of the ultrasonic method. It was generally observed that a larger training data set produced better results when ANNs were presented with previously unseen data, and that the more detailed the ANN input data, the better the ANN's ability to draw meaningful relationships between inputs and targets during training. In terms of ANN architecture, it was found that it was necessary to optimise the number of neurons used for a particular application to ensure that the ANN would still perform well when presented with previously unseen or noisy data. A very large network was found to perform well in terms of memorisation, but could perform poorly when required to generalise. Conversely, a very small network may not have enough storage capacity in the weights between neurons to adequately describe the relationship between input and output data.

ANNs trained with numerical data from the finite element method were assessed with experimental data from the ultrasonic pulse-echo method, and were found to be able to return very accurate estimations of the distance to a defect (error typically <1%), and acceptably accurate estimations of the size of a defect (error typically <5%). Some difficulty was found in maintaining consistent and repeatable coupling conditions between the ultrasonic

transducer and the test specimen. It is expected that if uniform coupling conditions can be achieved, the accuracy of the ANNs in estimating defects' sizes would increase. Even so, the results demonstrate that an ANN trained with numerical data can return accurate results when presented with experimental data.

In the case of two defects being present in a component, a novel method has been presented that uses an ANN to decouple two overlapping signals. It was found that although the time and height of each peak in the raw signal was initially considered sufficient to describe the signal to an ANN, cross-correlation of the signal with the transmitted ultrasonic pulse was necessary to generate a smoother signal. The time and height of each peak in this cross-correlated signal were sufficient to describe each case to an ANN, although the accuracy of the ANN's output was affected quite adversely by added noise. A trial study using the full ultrasonic signal, discretised into 500 steps of 0.1μs, as an input to a large ANN showed that although this method didn't significantly improve accuracy in characterising defects when presented with the numerical data, the results from this large ANN were much more accurate when noise was added to the input data. Although it falls outside the scope of the thesis, this finding is considered worthy of further investigation.

ANNs have therefore been proven to be successful at decoupling two signals from defects of varying size, and characterising the defects in terms of their position and size. This information would not be possible to infer from an ultrasonic A-scan, thus the ANN provides information from the ultrasonic data that an operator could not otherwise obtain.

Based on the success of the single and multiple defect analyses, a novel ANN-based 'expert system' has been presented, which takes ultrasonic A-scan data as its input, and is able to classify a case in terms of the number of defects present, then quantify the case by providing an estimation of each defect's position and size. Accuracy for the location and sizing of

single defect and two defect cases is of a similar order to, or better than, that found by previous researchers when looking at single parameters, thus the system is considered to be sufficiently accurate and merits further investigation from a commercial point of view.

9.3 Recommendations for future work

Although the results presented in this thesis are considered complete as they stand, there are several areas with scope for further work, as detailed below.

- Create a 3D finite element ultrasonic model to operate at 5MHz and include attenuation, to more accurately match the experimental situation. This would mean that the file sizes would become extremely large, requiring a multiple processor 'cluster computer', but the results would be directly comparable between numerical and experimental methods, thus potentially enabling a completely automated analysis of raw experimental data.
- Conduct ultrasonic experiments at 100kHz, enabling the two defect study to be experimentally verified. The advantage of this would be that the case of two defects lying within one wavelength of each other could be assessed, and directly compared with the current numerical model.
- Apply the techniques developed to components of more complex geometry, ideally from a 'real world' application. A more complex component may require measurement at several different points (Baskaran, Lakshmana *et al.* 2007), but would assess the versatility of the ANN-based system in a more complex situation.
- Investigate other simple methods of feature extraction to improve ANNs' performance, such as using the entire 'raw' signal, discretised into a finite number of

points. It is possible that some forms of signal processing could be conducted in real time, thus could still be suited to online inspection.

- Investigate the effect of different types of defect, such as cracks. Much work has been published on the classification of defects, though very little on the subsequent quantification of these. It would be useful to expand the current ANN-based system to identify and quantify a range of defect types.

APPENDIX A

```
*Heading
 Correct geometry etc for 3 step model
** Job name: 3step Model name: Model-1
*Preprint, echo=NO, model=NO, history=NO, contact=NO
**
** PARTS
**
*Part, name=Part-1
*End Part
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Part-1-1, part=Part-1
*Node
      1,           0.,         -0.1
      2,        0.001,         -0.1
.
. (truncated for brevity)
.
   2209,        0.008,          0.1
   2210,  0.009000001,          0.1
   2211,         0.01,          0.1
*Element, type=CPE4R
   1,    1,    2,   13,   12
   2,    2,    3,   14,   13
.
. (truncated for brevity)
.
1999, 2198, 2199, 2210, 2209
2000, 2199, 2200, 2211, 2210
** Region: (Section-1:Picked)
*Elset, elset=_PickedSet2, internal, generate
    1,  2000,     1
** Section: Section-1
*Solid Section, elset=_PickedSet2, material=Material-1
1.,
** Region: (Section-1:Picked)
*PARAMETER
ELNUM = 807 # if nothing specified by psf file
ELNUM2 = 810 # from psf file
ELNUMP1=ELNUM2+1
ELNUMP2=ELNUM2+2
ELNUMP3=ELNUM2+3
ELNUMP10=ELNUM2+10
ELNUMP11=ELNUM2+11
ELNUMP12=ELNUM2+12
ELNUMP13=ELNUM2+13
ELNUMP20=ELNUM2+20
ELNUMP21=ELNUM2+21
ELNUMP22=ELNUM2+22
ELNUMP23=ELNUM2+23
ELNUMP30=ELNUM2+30
ELNUMP31=ELNUM2+31
ELNUMP32=ELNUM2+32
ELNUMP33=ELNUM2+33

**Elset, elset=_I1, internal, generate
**   1,  192,    1
*Elset, elset=_I1, generate
1, 2000, 1
*Elset, elset=_I2  # generates region of defects
<ELNUM>, <ELNUM2>, <ELNUMP1>, <ELNUMP2>, <ELNUMP3>, <ELNUMP10>, <ELNUMP11>,
<ELNUMP12>,  <ELNUMP13>,  <ELNUMP20>,  <ELNUMP21>,  <ELNUMP22>,  <ELNUMP23>,
<ELNUMP30>, <ELNUMP31>, <ELNUMP32>, <ELNUMP33>
```

```
** Section: Section-1
*Solid Section, elset=_I1, material=Material-1
1.,
*Solid Section, elset=_I2, material=Material-2
1.,
*End Instance
*Nset, nset=_PickedSet4, internal, instance=Part-1-1, generate
  1,  11,   1
*Elset, elset=_PickedSet4, internal, instance=Part-1-1, generate
 1, 10,  1
*Nset, nset=Sensors, instance=Part-1-1, generate
 2201,  2211,     1
*Elset, elset=Sensors, instance=Part-1-1, generate
 1991,  2000,     1
*Elset, elset=__PickedSurf5_S3, internal, instance=Part-1-1, generate
 1991,  2000,     1
*Surface, type=ELEMENT, name=_PickedSurf5, internal
__PickedSurf5_S3, S3
*End Assembly
*Amplitude, name=Amp-1
0., 2.42445e-17, 6.25e-07, 0.05105, 1.25e-06, 0.124747, 1.875e-06, 0.211436
2.5e-06,  0.291213,  3.125e-06,  0.335815,  3.75e-06,  0.314683,  4.375e-06,
0.204531
5e-06, -1.47518e-15, 5.625e-06, -0.278424, 6.25e-06, -0.583133, 6.875e-06,
-0.847116
7.5e-06,  -1.,  8.125e-06,  -0.988363,  8.75e-06,  -0.793808,  9.375e-06,  -
0.442209
1e-05, 0., 1.0625e-05, 0.442209, 1.125e-05, 0.793808, 1.1875e-05, 0.988363
1.25e-05,  1.,  1.3125e-05,  0.847116,  1.375e-05,  0.583133,  1.4375e-05,
0.278424
1.5e-05, 7.71379e-17, 1.5625e-05, -0.204531, 1.625e-05, -0.314683, 1.6875e-
05, -0.335815
1.75e-05, -0.291213, 1.8125e-05, -0.211436, 1.875e-05, -0.124747, 1.9375e-
05, -0.05105
2e-05, 1.51518e-16
**
** MATERIALS
**
*Material, name=Material-1
*Density
7800.,
*Elastic
 2.1e+11, 0.3
*Material, name=Material-2
*Density
5.,
*Elastic
2.1e+3, 0.3
**
** BOUNDARY CONDITIONS
**
** Name: BC-1 Type: Symmetry/Antisymmetry/Encastre
*Boundary
_PickedSet4, ENCASTRE
** ----------------------------------------------------------------
**
** STEP: Step-1
**
*Step, name=Step-1
US applied
*Dynamic, Explicit
, 2e-05
*Bulk Viscosity
0.06, 1.2
**
** LOADS
**
```

```
** Name: Load-1   Type: Pressure
*Dsload, amplitude=Amp-1
_PickedSurf5, P, 1e+07
**
** OUTPUT REQUESTS
**
*Restart, write, number interval=1, time marks=NO
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT, number intervals=10
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history
*Node Output, nset=Sensors
U1, U2
*End Step
** ----------------------------------------------------------------
**
** STEP: Step-2
**
*Step, name=Step-2
wave propagation
*Dynamic, Explicit
, 1e-05
*Bulk Viscosity
0.06, 1.2
**
** LOADS
**
** Name: Load-1   Type: Pressure
*Dsload, op=NEW
**
** OUTPUT REQUESTS
**
*Restart, write, number interval=1, time marks=NO
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT, number intervals=10
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history
*Node Output, nset=Sensors
U1, U2
*End Step
** ----------------------------------------------------------------
**
** STEP: Step-3
**
*Step, name=Step-3
wave propagation
*Dynamic, Explicit, Direct User Control
1e-07, 5e-05
*Bulk Viscosity
0.06, 1.2
**
** LOADS
**
** Name: Load-1   Type: Pressure
*Dsload, op=NEW
**
** OUTPUT REQUESTS
**
*Restart, write, number interval=1, time marks=NO
```

```
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT, number intervals=500
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, time interval=1e-07
*Node Output, nset=Sensors
U1, U2
*File output, number interval=500
*Node file, nset=Sensors
U
*End Step
```

APPENDIX B

```
DISPL=ParStudy(par=('ELNUM','ELNUM2'),name='2def1_4')

DISPL.define(DISCRETE,par='ELNUM',domain=(932,996,977,1229,1184,938,1232,92
4,1292,1054,1234,986,1298,1215,1207,945,1115,1134,1224,1168,1044,994,873,10
47,883,1275,1187,977,1205,855,1213,1176,1182,1238,1105,946,954,1252,1199,12
94,1013,1072,1205,918,944,1148,1088,903,1219,1287,1272,873,947,1042,1066,12
52,996,1057,1025,1249,858,859,912,1038,1098,1209,1027,1175,1046,1222,963,12
65,965,869,1179,863,974,1059,855,1054,1265,1222,1119,1293,854,1103,873,1194
,953,937,1075,857,982,853,1207,854,1274,957,1295,1088,952,1176,1037,1227,11
74,1213,1217,854,965,1207,878,875,1065,992,893,1135,1088,1297,1162,1046,949
,865,984,1033,1226,1266,964,1037,1258,1086,1056,895,856,1032,908,994,1077,1
172,952,1127,866,1183,1057,1128,1227,1294,1063,1215,1288,1183,1153,875,1256
,995,1223,972,1114,1057,935,1295,865,1087,1267,1126,1026,1048,1113,1058,100
5,994,1293,1153,1112,1186,992,1213,1034,1086,1282,1268,966,1072,1178,1276,1
298,1066,1194,1244,928,964,1044,1288,933,1026,918,1244,905,1158,1265,1052,9
87,858,985,1042,934,1086,1134,962,966,904,1086,1268,1229,949,988,854,1039,1
219,1025,1022,1148,1023,1003,923,1175,1116,1254,1286,1013,1133,1137,1138,11
26,884,1109,1289,1178,1222,977,1196,1048,1067,922,1275,1165,1294,1282,1287,
854,1002,1236,1133,1098,1272,1155,1238,1193,906,1125,885,1045,1046,879,1117
,1006,1147,1099,1017,914,1278,1075,984,1146,1072,1089,1243,879,1189,1289,12
85,1059,1138,1116,1154,977,973,1017,1295,1217,1052,1203,1214,1047,1149,1272
,1033,1278,1274,1074,1016,1235,1155,896,914,1002,1282,975,1163,1172,1013,11
06,1218,1108,1046,1163,1103,1144,1243,955,1012,938,1233,868,1287,932,986,12
54,1197,1199,898,1049,1299,1258,1282,1182,1103,1075,1125,959,1155,1187,886,
958,929,943,867,1005,878,966,1097,1194,1132,1017,1018,1034,1073,1217,1223,1
217,946,1234,982,1178,1034,1256,1044,1114,968,918,1002,1243,1286,862,1272,1
164,934,1078,956,1198,1238,1292,995,1235,982,1276,1274,1212,934,1095,1015,9
48,1182,1116,852,1196,1253,1216,985,1287,1294,1227,904,1227,1207,936,1285,1
198,1056,908,1196,1217,1069,955,969,1064,1208,1248,1046,1002,895,1162,958,1
237,1299,1147,1183,1109,942,964,1286,1297,1075,1232,1153,982,1022,1103,933,
975,1052,1275,984,1074,1184,1223,1267,863,968,1018,1133,1164,1003,864,1108,
1136,1009,1189,1008,909,1179,1234,853,1262,956,1135,928,958,1076,1057,1008,
1269,1275,853,1054,1124,1212,874,1038,955,1033,1135,1136,1146,938,923,1113,
1282,986,1148,1194,1222,1283,1046,1035,883,1019,1267,1212,909,1004))

DISPL.define(DISCRETE,par='ELNUM2',domain=(866,914,865,1106,816,904,955,805
,1263,842,815,935,894,1023,932,826,1052,1093,1094,1014,886,962,805,1005,803
,1206,1142,913,1144,823,1093,1115,974,912,844,822,915,1034,1034,1016,933,97
6,944,864,865,932,1016,876,1003,943,853,835,862,933,944,824,896,975,844,923
,816,834,865,882,862,1044,812,992,874,965,834,1204,895,825,1065,805,933,886
,815,952,944,1196,1036,1116,825,945,805,865,872,803,825,832,855,813,863,803
,1164,836,985,1005,903,1143,902,954,876,1045,912,806,803,863,856,834,1016,9
24,853,835,985,1154,1086,832,892,815,923,975,945,846,934,816,1213,933,956,8
35,813,886,834,915,963,984,813,873,825,816,826,984,875,1216,843,816,1162,10
63,802,853,1086,902,1176,915,1036,966,864,1086,813,965,1064,876,875,864,101
6,1026,852,864,1055,1013,942,845,856,1186,874,825,1226,895,865,802,862,1145
,923,966,873,802,824,814,943,802,855,815,802,1155,855,836,1074,816,902,814,
933,856,895,1042,1033,915,935,875,1016,1233,1056,824,823,802,974,1033,805,8
16,903,824,913,892,1123,822,865,886,972,944,842,814,925,833,993,912,814,892
,916,1095,892,913,814,806,1022,1023,1074,834,806,954,1133,1093,952,965,842,
1065,854,815,1046,842,1016,944,832,825,822,1096,863,985,874,894,964,924,823
,945,1054,872,802,956,1122,1105,1006,823,822,1065,815,854,934,1164,984,1015
,1112,1096,844,1103,836,812,825,1243,905,876,1196,1104,863,882,956,1083,884
,1085,1102,853,855,842,954,884,945,896,896,826,913,866,805,1085,815,1056,81
5,905,1106,973,1053,812,963,1236,1185,902,1102,855,1053,915,894,1095,813,81
4,866,833,803,802,866,835,804,814,1054,1076,932,884,854,925,936,814,875,906
,993,905,944,886,1216,845,1076,903,893,956,832,1066,835,1124,895,803,802,87
4,1005,1013,1105,916,973,905,883,1096,1093,812,1022,856,926,976,1035,825,10
76,1183,1093,962,1244,1235,845,834,1173,844,915,863,1046,872,815,913,936,10
22,926,814,1034,1173,1004,816,872,833,893,874,973,912,886,1146,864,835,855,
1233,935,862,805,1045,942,815,1022,902,954,902,925,935,922,1116,965,1055,82
3,853,923,993,1055,866,804,1055,1066,925,934,933,822,916,884,803,876,873,91
2,855,915,1024,824,902,1013,1086,832,953,1032,1023,836,993,866,945,823,906,
824,826,822,1045,995,832,1055,1092,1142,912,924,822,856,934,1092,1073,863,9
24))

DISPL.sample(INTERVAL,par='ELNUM',interval=1)
```

```
DISPL.sample(INTERVAL,par='ELNUM2',interval=1)
DISPL.combine(TUPLE)
DISPL.generate(template='2def1_4')
DISPL.execute(ALL)
```

APPENDIX C

```
      SUBROUTINE HKSMAIN
!     commands to compile and run
!    abaqus make job=datacollect500
!
!     Interrogate 500 .fil files and extract nodal displacement for nodes
2201-2211 at exact time intervals
!
      INCLUDE 'aba_param.inc'
      CHARACTER*80 FNAME
      character*80 fln(500)
      DIMENSION RESULTU(150000,11),TOTT(150000)
      DIMENSION ARRAY(513),JRRAY(NPRECD,513),LRUNIT(2,1)
      EQUIVALENCE (ARRAY(1),JRRAY(1,1))
      OPEN(UNIT=20,FILE='simono.dat',STATUS='NEW')
      OPEN(UNIT=21,FILE='flname.dat',STATUS='old')
!
!    File initialization
!
!     **************    We need to automate the filename generation by
referring to the array created above
!
      IR=0
      DO 300 i=1,500
      read(21,*) fln(i)
      FNAME=fln(i)
!      FNAME='job-u'
      NRU=1
      LRUNIT(1,1)=8
      LRUNIT(2,1)=2
      LOUTF=0
      CALL INITPF(FNAME,NRU,LRUNIT,LOUTF)
      JUNIT=8
      CALL DBRNU(JUNIT)
!
!     Loop on all records in results file
!
      DO 100 K1=1,99999
!
        CALL DBFILE(0,ARRAY,JRCD)
        IF(JRCD.NE.0)GO TO 110
        KEY=JRRAY(1,2)
      IF(KEY.EQ.2000)THEN
        IR=IR+1
        INCN=JRRAY(1,9)
        TINC=ARRAY(13)
        TOTT(IR)=ARRAY(3)
!     write(6,*) TOTT, INCN, TINC
      ENDIF
!
        IF(KEY.EQ.101) THEN
!
!          Node record:
!
           NNUM=JRRAY(1,3)
           IF((NNUM.GE.2201).AND.(NNUM.LE.2211)) THEN
!          U1=ARRAY(1,4)
           RESULTU(IR,(NNUM-2200))=ARRAY(5)
!**   Again, don't think write (6,*) is needed
!          WRITE(6,*)JRRAY(1,3),ARRAY(4),ARRAY(5)
!            GO TO 110
           ENDIF
           ENDIF
!
 100  CONTINUE
 110  CONTINUE
```

```
 300  CONTINUE
      DO 220 I=1,IR
      WRITE(20,120) TOTT(I),(RESULTU(I,J),J=1,11)
 120  FORMAT(12(E15.8,1X))
 220  CONTINUE
      STOP

      END
```

APPENDIX D

```
%program to find peaks in the convoluted signals from 2 defect cases
clear all;
time=dlmread('time.dat'); %500r by 1c
d=dlmread('all2def_1to4mm_diff.dat'); %3500r by 506c
data=d(:,1:500)'; %data is now 500r by 3500c - first peak is negative
targets=d(:,501:506)';  %targets has elnum1, elnum2, dist1, dist2, size1,
size2
refsig=dlmread('refsig.dat'); %234r by 2c - time and sig
ref=refsig(:,2);

% to create cross-correlated array
for n=1:3500
    xcdata(:,n)=-xcorr(data(:,n),ref);
end
timexc=[1:1:999];
timexc=timexc'; % creates a reference array to give a value for peak time

clear data; clear time;
data=xcdata;
time=timexc;

% to create dy/dt array
for n=1:3500
    for m=2:999
        dydt(m,n)=(data(m,n)-data(m-1,n))/(time(m,1)-time(m-1,1));
    end
end
% to create d2y/dt2 array
for n=1:3500
    for m=3:999
        d2y(m,n)=(dydt(m,n)-dydt(m-1,n))/(time(m,1)-time(m-1,1));
    end
end

thres=3e-15;
thres2=0.2;
threstime=5;
thresdydt=5e-16;
%variables are as follows:
%data (999r by 3500c) = difference between defected and non-defected signal
%time (999r by 1c) = arbitrary ref for time
%dydt (999r by 3500c) = 1st differential
%d2y (999r by 3500c) = 2nd differential

%thresholds for deltaU and for dy/dt

peakt=zeros(14,3500);
peakh=peakt;

%To find the peaks of the reflected wave:  Peak is identified as a point
%that is greater (in magnitude) than the previous point, greater than
%or equal to the subsequent point, and greater than the next point after
that.
%The peak must also occur at a time 0.5 microseconds or more after the
%previous peak, and be higher or lower in value than the previous peak (to
%ensure that the sequence is peak, trough, peak, trough etc.).

for c=1:3500
```

```
    for r=3:997
        if  peakt(1,c)==0 & data(r,c)>data(r-1,c) & data(r,c)>=data(r+1,c)
& data(r,c)>data(r+2,c) & data(r,c)>thres  %finds first positive peak
            peakt(1,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(1,c)=data(r,c)+dydt(r,c)*(peakt(1,c)-time(r,1));
        end
        if  peakt(2,c)==0 & peakt(1,c)~=0 & data(r,c)<data(r-1,c) &
data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) & data(r,c)<peakh(1,c) &
time(r,1)>(peakt(1,c)+threstime)  %finds second peak (-ve)
            peakt(2,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(2,c)=data(r,c)+dydt(r,c)*(peakt(2,c)-time(r,1));
        end
        if  peakt(3,c)==0 & peakt(2,c)~=0 & peakt(1,c)~=0 &
data(r,c)>data(r-1,c) & data(r,c)>=data(r+1,c) & data(r,c)>data(r+2,c) &
data(r,c)>peakh(2,c) & time(r,1)>(peakt(2,c)+threstime)  %finds third peak
(+ve)
            peakt(3,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(3,c)=data(r,c)+dydt(r,c)*(peakt(3,c)-time(r,1));
        end
        if  peakt(4,c)==0 & peakt(3,c)~=0 & peakt(2,c)~=0 & peakt(1,c)~=0 &
data(r,c)<data(r-1,c) & data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) &
data(r,c)<peakh(3,c) & time(r,1)>(peakt(3,c)+threstime)  %finds fourth peak
(-ve)
            peakt(4,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(4,c)=data(r,c)+dydt(r,c)*(peakt(4,c)-time(r,1));
        end
        if  peakt(5,c)==0 & peakt(4,c)~=0 & peakt(3,c)~=0 & peakt(2,c)~=0 &
peakt(1,c)~=0 & data(r,c)>data(r-1,c) & data(r,c)>=data(r+1,c) &
data(r,c)>data(r+2,c) & data(r,c)>(peakh(4,c)+thres2*thres) &
time(r,1)>(peakt(4,c)+threstime)  %finds fifth peak (+ve)
            peakt(5,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(5,c)=data(r,c)+dydt(r,c)*(peakt(5,c)-time(r,1));
        end
        if  peakt(6,c)==0 & peakt(5,c)~=0 & peakt(4,c)~=0 & peakt(3,c)~=0 &
peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)<data(r-1,c) &
data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) & data(r,c)<(peakh(5,c)-
thres2*thres) & time(r,1)>(peakt(5,c)+threstime)  %finds sixth peak (-ve)
            peakt(6,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(6,c)=data(r,c)+dydt(r,c)*(peakt(6,c)-time(r,1));
        end
        if  peakt(7,c)==0 & peakt(6,c)~=0 & peakt(5,c)~=0 & peakt(4,c)~=0 &
peakt(3,c)~=0 & peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)>data(r-1,c) &
data(r,c)>=data(r+1,c) & data(r,c)>data(r+2,c) &
data(r,c)>(peakh(6,c)+thres2*thres) & time(r,1)>(peakt(6,c)+threstime)
%finds seventh peak (+ve)
            peakt(7,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(7,c)=data(r,c)+dydt(r,c)*(peakt(7,c)-time(r,1));
        end
        if  peakt(8,c)==0 & peakt(7,c)~=0 & peakt(6,c)~=0 & peakt(5,c)~=0 &
peakt(4,c)~=0 & peakt(3,c)~=0 & peakt(2,c)~=0 & peakt(1,c)~=0 &
data(r,c)<data(r-1,c) & data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) &
```

```
data(r,c)<(peakh(7,c)-thres2*thres) & time(r,1)>(peakt(7,c)+threstime)
%finds eighth peak (-ve)
            peakt(8,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(8,c)=data(r,c)+dydt(r,c)*(peakt(8,c)-time(r,1));
        end
        if  peakt(9,c)==0 & peakt(8,c)~=0 & peakt(7,c)~=0 & peakt(6,c)~=0 &
peakt(5,c)~=0 & peakt(4,c)~=0 & peakt(3,c)~=0 & peakt(2,c)~=0 &
peakt(1,c)~=0 & data(r,c)>data(r-1,c) & data(r,c)>=data(r+1,c) &
data(r,c)>data(r+2,c) & data(r,c)>(peakh(8,c)+thres2*thres) &
time(r,1)>(peakt(8,c)+threstime)  %finds ninth peak (+ve)
            peakt(9,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(9,c)=data(r,c)+dydt(r,c)*(peakt(9,c)-time(r,1));
        end
        if  peakt(10,c)==0 & peakt(9,c)~=0 & peakt(8,c)~=0 & peakt(7,c)~=0
& peakt(6,c)~=0 & peakt(5,c)~=0 & peakt(4,c)~=0 & peakt(3,c)~=0 &
peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)<data(r-1,c) &
data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) & data(r,c)<(peakh(9,c)-
thres2*thres) & time(r,1)>(peakt(9,c)+threstime)  %finds tenth peak (-ve)
            peakt(10,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(10,c)=data(r,c)+dydt(r,c)*(peakt(10,c)-time(r,1));
        end
        if  peakt(11,c)==0 & peakt(10,c)~=0 & peakt(9,c)~=0 & peakt(8,c)~=0
& peakt(7,c)~=0 & peakt(6,c)~=0 & peakt(5,c)~=0 & peakt(4,c)~=0 &
peakt(3,c)~=0 & peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)>data(r-1,c) &
data(r,c)>=data(r+1,c) & data(r,c)>data(r+2,c) & data(r,c)>(peakh(10,c)-
thres2*thres) & time(r,1)>(peakt(10,c)+threstime)  %finds eleventh peak
(+ve)
            peakt(11,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(11,c)=data(r,c)+dydt(r,c)*(peakt(11,c)-time(r,1));
        end
        if  peakt(12,c)==0 & peakt(11,c)~=0 & peakt(10,c)~=0 &
peakt(9,c)~=0 & peakt(8,c)~=0 & peakt(7,c)~=0 & peakt(6,c)~=0 &
peakt(5,c)~=0 & peakt(4,c)~=0 & peakt(3,c)~=0 & peakt(2,c)~=0 &
peakt(1,c)~=0 & data(r,c)<data(r-1,c) & data(r,c)<=data(r+1,c) &
data(r,c)<data(r+2,c) & data(r,c)<(peakh(11,c)-thres2*thres) &
time(r,1)>(peakt(11,c)+threstime)  %finds twelfth peak (-ve)
            peakt(12,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(12,c)=data(r,c)+dydt(r,c)*(peakt(12,c)-time(r,1));
        end
        if  peakt(13,c)==0 & peakt(12,c)~=0 & peakt(11,c)~=0 &
peakt(10,c)~=0 & peakt(9,c)~=0 & peakt(8,c)~=0 & peakt(7,c)~=0 &
peakt(6,c)~=0 & peakt(5,c)~=0 & peakt(4,c)~=0 & peakt(3,c)~=0 &
peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)>data(r-1,c) &
data(r,c)>=data(r+1,c) & data(r,c)>data(r+2,c) & data(r,c)>(peakh(12,c)-
thres2*thres) & time(r,1)>(peakt(12,c)+threstime)  %finds thirteenth peak
(+ve)
            peakt(13,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(13,c)=data(r,c)+dydt(r,c)*(peakt(13,c)-time(r,1));
        end
        if  peakt(14,c)==0 & peakt(13,c)~=0 & peakt(12,c)~=0 &
peakt(11,c)~=0 & peakt(10,c)~=0 & peakt(9,c)~=0 & peakt(8,c)~=0 &
peakt(7,c)~=0 & peakt(6,c)~=0 & peakt(5,c)~=0 & peakt(4,c)~=0 &
peakt(3,c)~=0 & peakt(2,c)~=0 & peakt(1,c)~=0 & data(r,c)<data(r-1,c) &
```

```
data(r,c)<=data(r+1,c) & data(r,c)<data(r+2,c) & data(r,c)<(peakh(13,c)-
thres2*thres) & time(r,1)>(peakt(13,c)+threstime)  %finds fourteenth peak
(-ve)
            peakt(14,c)=((-dydt(r,c)/d2y(r,c))-
(dydt(r+1,c)/d2y(r+1,c)))/2+time(r,1);
            peakh(14,c)=data(r,c)+dydt(r,c)*(peakt(14,c)-time(r,1));
        end
    end
end

% peakt and peakh each have 14 rows by 3500 columns; data has 999r by 3500c
data=data';
peakt=peakt';
peakh=peakh';

out(:,1:999)=data;
out(:,1000:1013)=peakt;
out(:,1014:1027)=peakh;
out(:,1028:1033)=targets';

dlmwrite('alldata2def_1to4mm_withpeaks_xcorr_02.dat', out);
```

REFERENCES

Achenbach, J. D. (2000). "Quantitative nondestructive evaluation." *International Journal of Solids and Structures* **37**(1-2): 13-27.

Achenbach, J. D., Komsky, I. N., Lee, Y. C. and Angel, Y. C. (1992). "Self-calibrating ultrasonic technique for crack depth measurement." *Journal of Nondestructive Evaluation* **11**(2): 103-108.

Aldrin, J. C., Cheng, A., Achenbach, J. D., Andrew, G. A. and Mullis, R. T. (2000). "Detection of cracks in weep holes using neural networks." CP509, *Review of progress in Quantitative Nondestructive Evaluation*, AIP.

Alleyne, D. N. (1991). The non-destructive testing of plates using ultrasonic Lamb waves. Mechanical Engineering Department. London, Imperial College London. **PhD:** 252.

Anderson, J. A. and Rosenfeld, E., Eds. (1988). Neurocomputing: Foundations of Research. Cambridge, MA, MIT Press.

Bainton, K. F. (1977). "Characterizing defects by determining magnetic leakage fields." *NDT International* **10**(5): 253-257.

Baker, A. R. and Windsor, C. G. (1989). "Classification of defects from ultrasonic data using neural networks: The Hopfield method." *NDT International* **22**(2): 97-105.

Barauskas, R. and Daniulaitis, V. (2000). "Simulation of the ultrasonic wave propagation in solids." *Ultragarsas* **4**(37): 34-37.

Bartoli, I., Lanza Di Scalea, F., Fateh, M. and Viola, E. (2005). "Modeling guided wave propagation with application to the long-range defect detection in railroad tracks." *NDT and E International* **38**(5): 325-334.

Baskaran, G., Lakshmana Rao, C. and Balasubramaniam, K. (2007). "Simulation of the TOFD technique using the finite element method." *Insight: Non-Destructive Testing and Condition Monitoring* **49**(11): 641-646.

Batra, N. K. and Chaskelis, H. H. (1985). "Determination of minimum flaw size detectable by ultrasonics in titanium alloy plates." *NDT International* **18**(5): 261-264.

Bhagat, P. (2005). Pattern recognition in industry. Amsterdam ; London, Elsevier.

Bishop, C. M. (1995). Neural Networks for Pattern Recognition. Oxford, Clarendon Press.

Bloom, F. E. (1986). Handbook of physiology : a critical, comprehensive presentation of physiological knowledge and concepts. Section 1, Vol.4, The nervous system. Intrinsic regulatory systems of the brain. Bethesda, Md, American Physiological Society.

Burch, S. F. and Bealing, N. K. (1986). "Physical approach to the automated ultrasonic characterization of buried weld defects in ferritic steel." *NDT international* **19**(3): 145-153.

Castellini, P. and Revel, G. M. (2000). "Experimental technique for structural diagnostic based on laser vibrometry and neural networks." *Shock and Vibration* **7**(6): 381-397.

Chen, C. H. (1989). Applying and validating neural network technology for nondestructive evaluation of materials, Cambridge, MA, USA, Publ by IEEE.

Chen, C. H. and Lee, G. G. (1993). Neural networks for ultrasonic NDE signal classification using time-frequency analysis, Piscataway, NJ, USA, Publ by IEEE.

Christen, R. and Bergamini, A. (2006). "Automatic flaw detection in NDE signals using a panel of neural networks." *NDT and E International* **39**(7): 547-553.

Courant, R., Friedrichs, K. and Lewy, H. (1967). "On the partial difference equations of mathematical physics." *IBM Journal*: 215-234.

Darabi, A. and Maldague, X. (2002). "Neural network based defect detection and depth estimation in TNDE." *NDT and E International* **35**(3): 165-175.

De Carvalho, A. A., Veiga, J. L. B. C., Da Silva, I. C., Pereira, C. M. N. A. and Rebello, J. M. A. (2003). "Preliminary study of classification of defects by ultrasonic pulse-echo signals using an artificial neural network." *Insight: Non-Destructive Testing and Condition Monitoring* **45**(11): 754-757.

De Wilde, P. (1997). Neural network models : theory and projects. London, Springer.

Demuth, H. and Beale, M. (1998). Neural Network Toolbox User's Guide, The Mathworks.

Dewey, B. R. (1983). "Modelling of ultrasonic flaw detection using finite element methods", San Antonio, Tex, USA, Southwest Research Inst.

Doebling, S. W., Farrar, C. R. and Prime, M. B. (1998). "A Summary Review of Vibration-Based Damage Identification Methods." *Shock & Vibration Digest* **30**(2): 91.

Farley, S. J., Durodola, J. F., Fellows, N. A. and Hernández-Gómez, L. H. (2008). "A Neural Network Approach for Locating Multiple Defects." *BSSM 6th International Conference on Advances in Experimental Mechanics*, London, Trans Tech Publications, Switzerland.

Fan, Y. (2010). Private correspondence.

Fausett, L. V. (1994). Fundamentals of neural networks : architectures, algorithms, and applications. London, Prentice-Hall International.

Gang, T., Takahashi, Y. and Wu, L. (2002). "Intelligent pattern recognition and diagnosis of ultrasonic inspection of welding defects based on neural network and information fusion." *Science and Technology of Welding and Joining* **7**(5): 314-320.

Ghaboussi, J. and Banan, M. R. (1994). "Neural networks in engineering diagnostics." *SAE Special Publication 1048, 1994 Society of Automotive Engineers (SAE) off highway and powerplant congress and exposition*, Milwaukee, WI (United States): 71-82.

Gupta, S., Ray, A. and Keller, E. (2007). "Symbolic time series analysis of ultrasonic data for early detection of fatigue damage." *Mechanical Systems and Signal Processing* **21**(2): 866-884.

Guyton, A. C. (2000). Textbook of medical physiology. Philadelphia ; London, Saunders.

Halmshaw, R. (1991). Non-destructive testing. London, Edward Arnold.

Hands, G. (1997). "Resonant inspection for mass production industries." Insight: Non-Destructive Testing and Condition Monitoring 39(12): 871-873.

Harker, A. H. (1984). "Numerical modelling of the scattering of elastic waves in plates." *Journal of Nondestructive Evaluation* **4**(2): 89-106.

Hassan, W. and Veronesi, W. (2003). "Finite element analysis of Rayleigh wave interaction with finite-size, surface-breaking cracks." *Ultrasonics* **41**(1): 41-52.

Hebb, D. O. (1949). The Organization of Behavior. New York, Wiley.

Hernandez-Gomez, L. H., Durodola, J. F., Fellows, N. A. and Urriolagoitia-Calderon, G. (2005). "Locating defects using dynamic strain analysis and artificial neural networks." Clausthal-Zellerfeld, D-38670, Germany, Trans Tech Publications.

Hestenes, M. R. and Stiefel, E. (1952). "Methods of Conjugate Gradients for Solving Linear Systems." *Journal of Research of the National Bureau of Standards* **49**(6): 409-436.

Hill, R., Forsyth, S. A. and Macey, P. (2004). "Finite element modelling of ultrasound, with reference to transducers and AE waves", Granada, Spain, Elsevier.

Ho, K. S., Gan, T. H., Billson, D. R. and Hutchins, D. A. (2005). "Application of pulse compression signal processing techniques to electromagnetic acoustic transducers for noncontact thickness measurements and imaging." Review of Scientific Instruments 76(5): 054902.

Hopfield, J. J. (1982). "Neural networks and physical systems with emergent collective computational abilities." *Proc Natl Acad Sci USA, Biophysics* **79**: 2554-2558.

Hull, B. and John, V. (1989). Non-destructive testing. Basingstoke, English Language Book Society/Macmillan.

Ishak, S. I., Liu, G. R., Shang, H. M. and Lim, S. P. (2002). "Non-destructive evaluation of horizontal crack detection in beams using transverse impact." *Journal of Sound and Vibration* **252**(2): 343-360.

Kazys, R., Raisutis, R., Sliteris, R., Mazeika, L. and Jasiuniene, E. (2004). "Ultrasonic thickness measurement of the aluminum plate with wavy surface." *Ultragarsas* **50**(1): 11-16.

Kolmogorov, A. N. (1957). "On the Representation of Continuous Functions of Many Variables by Superposition of Continuous Functions of One Variable and Addition." *Dokl. Akad. Nauk USSR* **114**: 953-956.

Kolsky, H. (1963). Stress waves in solids. New York, Dover.

Kumar, A., Gupta, N., Gupta, R. and Kumar, Y. (2005). "Effect of coupling conditions on ultrasonic echo parameters." *Journal of pure and applied ultrasonics* **27**(2/3): 70-79.

Kuravsky, L. S. and Baranov, S. N. (2008). "Technical diagnostics and monitoring based on capabilities of wavelet transforms and relaxation neural networks." *Insight: Non-Destructive Testing and Condition Monitoring* **50**(3): 127-132.

Legendre, S., Massicotte, D., Goyette, J. and Bose, T. K. (2001). "Neural classification of lamb wave ultrasonic weld testing signals using wavelet coefficients." *IEEE Transactions on Instrumentation and Measurement* **50**(3): 672-678.

Liu, S. W., Huang, J. H., Sung, J. C. and Lee, C. C. (2002). "Detection of cracks using neural networks and computational mechanics." *Computer Methods in Applied Mechanics and Engineering* **191**(25-26): 2831-2845.

Lowe, M. J. S., Cawley, P., Kao, J. Y. and Diligent, O. (2002). "The low frequency reflection characteristics of the fundamental antisymmetric Lamb wave a_0 from a rectangular notch in a plate." *Journal of the Acoustical Society of America* **112**(6): 2612-2622.

Lowe, M. J. S., Challis, R. E. and Chan, C. W. (2000). "The transmission of lamb waves across adhesively bonded lap joints." *Journal of the Acoustical Society of America* **107**(3): 1333-1345.

Lowe, M. J. S. and Diligent, O. (2002). "Low-frequency reflection characteristics of the s_0 lamb wave from a rectangular notch in a plate." *Journal of the Acoustical Society of America* **111**(1 I): 64-74.

Ludwig, R. and Lord, W. (1988). "Finite-element formulation for the study of ultrasonic NDT systems." *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* **35**(6): 809-820.

Luna-Aviles, A., Hernandez-Gomez, L. H., Durodola, J. F., Urriolagoitia-Calderon, G. and Urriolagoitia-Sosa, G. (2008). “Locating and classifying defects with artificial neural networks.” London, United kingdom, Trans Tech Publications.

Mahmoud, M. A. and Abukiefa, M. A. (1999). "Neural network solution of the inverse vibration problem." *NDT & E International* **32**(2): 91-99.

Mahmud, K. M. and Ryoji, O. (2004). "Application of CLEAN algorithm in ultrasonic NDT for distinct detection of close peaks in a pulse compression system." *Ultragarsas* **50**(1): 7-10.

Margrave, F. W., Rigas, K., Bradley, D. A. and Barrowcliffe, P. (1999). "Use of neural networks in ultrasonic flaw detection." *Measurement: Journal of the International Measurement Confederation* **25**(2): 143-154.

McClelland, J. L. and Rumelhart, D. E. (1988). Explorations in Parallel Distributed Processing. Cambridge, MA, MIT Press.

McCulloch, W. S. and Pitts, W. H. (1943). "A logical calculus of the ideas immanent in nervous activity." *Bulletin of Mathematical Biophysics* **5**: 115-133.

Moller, M. F. (1993). "Scaled conjugate gradient algorithm for fast supervised learning." *Neural Networks* **6**(4): 525-533.

Moser, F., Jacobs, L. J. and Qu, J. (1999). "Modeling elastic wave propagation in waveguides with the finite element method." *NDT and E International* **32**(4): 225-234.

Mustapha, F., Manson, G., Pierce, S. G. and Worden, K. (2005). "Structural health monitoring of an annular component using a statistical approach." *Strain* **41**(3): 117-127.

Mustapha, F., Worden, K., Pierce, S. G. and Manson, G. (2007). "Damage detection using stress waves and multivariate statistics: An experimental case study of an aircraft component." *Strain* **43**(1): 47-53.

Perron, M.-C. (1988). "Feature extraction and learning decision rules from ultrasonic signals - applicability in non-destructive testing." *Ultrasonics Symposium Proceedings* **1**: 533-536.

Picton, P. (2000). Neural networks. Basingstoke, Palgrave.

Priddy, K. L. and Keller, P. E. (2005). Artificial Neural Networks - an introduction. Bellingham, Washington, SPIE.

Ramuhalli, P., Udpa, L. and Udpa, S. S. (2003). "Neural network-based inversion algorithms in magnetic flux leakage nondestructive evaluation." *Journal of Applied Physics* **93**(10 3): 8274-8276.

Rayleigh, J. W. S. (1945). The theory of sound. New York, Dover.

Ripley, B. D. (1996). Pattern recognition and neural networks. Cambridge, Cambridge University Press.

Rose, J. L. (1999). Ultrasonic waves in solid media. Cambridge +, Cambridge University Press.

Rosenblatt, F. (1958). "The Perceptron: a Probabilistic Model for Information Storage and Organization in the Brain." *Psychological Review* **65**: 386-408.

Rytter, A. (1993). Vibration based inspection of civil engineering structures. Department of Building Technology and Structural Engineering, Aalborg University, Denmark. **PhD**.

Sahin, M. and Shenoi, R. A. (2003). "Vibration-based damage identification in beam-like composite laminates by using artificial neural networks." *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* **217**(6): 661-676.

Salawu, O. S. and Williams, C. (1995). "Bridge Assessment Using Forced-Vibration Testing." *Journal of Structural Engineering* **121**(2): 161.

Silk, M. G., Stoneham, A. M. and Temple, J. A. G. (1987). The reliability of non-destructive inspection : assessing the assessment of structures under stress. Bristol, A. Hilger.

Simone, G., Morabito, F. C., Polikar, R., Ramuhalli, P., Udpa, L. and Udpa, S. (2001). "Feature extraction techniques for ultrasonic signal classification." *International Journal of Applied Electromagnetics & Mechanics* **15**(1-4): 291.

Sumpter, B. G. and Noid, D. W. (1996). "On the Design, Analysis, and Characterization of Materials Using Computational Neural Networks." *Annual Review of Materials Science* **26**: 223-278.

Thavasimuthu, M., Rajagopalan, C., Kalyanasundaram, P. and Raj, B. (1996). "Improving the evaluation sensitivity of an ultrasonic pulse echo technique using a neural network classifier." *NDT & E International* **29**(3): 175-179.

Thomas, R., Drinkwater, B. W. and Liaptsis, D. (2005). "The reflection of ultrasound from partially contacting rough surfaces." *Journal of the Acoustical Society of America* **117**(2): 638-645.

Udpa, L. and Udpa, S. S. (1989). Application of neural networks to nondestructive evaluation. Artificial Neural Networks, 1989., *First IEE International Conference on (Conf. Publ. No. 313).*

Udpa, L. and Udpa, S. S. (1997). "Application of signal processing and pattern recognition techniques to inverse problems in NDE." *International Journal of Applied Electromagnetics & Mechanics* **8**(1): 99.

Veiga, J. L. B. C., De Carvalho, A. A., Da Silva, I. C. and Rebello, J. M. A. (2005). "The use of artificial neural network in the classification of pulse-echo and TOFD ultra-sonic signals." *Journal of the Brazilian Society of Mechanical Sciences and Engineering* **27**(4): 394-398.

Widrow, B. and Hoff, M. E. J. (1960). "Adaptive Switching Circuits." *IRE WESCON Convention Record* **part 4**: 96-104.

Widrow, B. and Lehr, M. A. (1990). "30 years of adaptive neural networks: Perceptron, Madaline, and backpropagation." *Proceedings of the IEEE* **78**(9): 1415-1442.

Windsor, C. G. (1995). "Can we train a computer to be a skilled inspector?" *Insight -Non - Destructive Testing and Condition Monitoring* **37**(1): 36.

Windsor, C. G., Anselme, F., Capineri, L. and Mason, J. P. (1993). "Classification of weld defects from ultrasonic images. A neural network approach." *British Journal of Non-Destructive Testing* **35**(1): 15-22.

Winter, R. and Widrow, B. (1988). Madaline rule II: A training algorithm for neural networks, San Diego, CA, USA, Publ by IEEE.

Yagawa, G. and Okuda, H. (1996). "Neural networks in computational mechanics." *Archives of Computational Methods in Engineering* **3**(4): 435-512.

Zapico, J. L., Gonzalez, M. P. and Worden, K. (2003). "Damage assessment using neural networks." *Mechanical Systems and Signal Processing* **17**(1): 119-125.

Zgonc, K. and Achenbach, J. D. (1996). "Neural network for crack sizing trained by finite element calculations." *NDT & E International* **29**(3): 147-155.

ARTICLE PUBLISHED IN THE COURSE

OF THE RESEARCH