APPLICATIONS OF NEURAL NETWORKS TO ULTRASOUND TOMOGRAPHY

BY

BARTLEY CLAY CONRATH

B.S., University of Illinois, 1988

THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering in the Graduate College of the University of Illinois at Urbana-Champaign, 1990

Urbana, Illinois

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

THE GRADUATE COLLEGE

JANUARY 1990

WE HEREBY RECOMMEND THAT THE THESIS BY

BARTLEY CLAY CONRATH

ENTITLED APPLICATIONS OF NEURAL NETWORKS TO ULTRASOUND TOMOGRAPHY

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF MASTER OF SCIENCE

CMW Taft
Director of Thesis Research

N. Narayana Rao
Head of Department

Committee on Final Examination†

Chairperson

† Required for doctor's degree but not for master's.

O-517

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

GRADUATE COLLEGE DEPARTMENTAL FORMAT APPROVAL

THIS IS TO CERTIFY THAT THE FORMAT AND QUALITY OF PRESENTATION OF THE THESIS SUBMITTED BY BARTLEY CLAY CONRATH AS ONE OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE ARE ACCEPTABLE TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING.

Full Name of Department, Division or Unit

4 December 1989

Date of Approval

Departmental Representative

ACKNOWLEDGEMENTS

The main acknowledgement goes, of course, to my advisor and friend, Dr. Christopher M. W. Daft. He has been the guiding force, leading my professional and academic development. With his help, I was able to accomplish my research goals, including the presentation of this work at the 1989 IEEE Ultrasonics Symposium in Montreal.

I would also like to thank Dr. Andrew Peterson for his help with the modelling of the field equations. We had several helpful discussions on field equations, finite element analysis, and Bessel functions.

Thanks also go to Dr. Y. Bresler. He enlightened me during our conversations on tomographic reconstruction.

The people in the Bioacoustics Laboratory helped me with many tasks. Therefore, I would like to thank Dr. William O'Brien, Jr., Dr. Leon Frizzell, Jay Alameda, Bob Cicone, and Wanda Elliot.

I would not have been able to earn this Master's Degree without the support of my parents and my sister. I dedicate this work to my family for all of the love they have provided throughout my life.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Ultrasound tomography holds promise in the area of medical diagnosis, offering benefits over the use of x-rays. For example, x-ray tomography, although helpful in many cases, poses health risks due to the ionizing radiation. Ultrasound is a non-ionizing form of energy and is safe at low levels. X-ray tomography images the distribution of density in the object. Ultrasound is sensitive to different tissue characteristics, such as changes in acoustic attenuation and speed, and therefore provides information that cannot be gained using x-ray absorption. Economic factors boost the value of ultrasound tomography, since ultrasound equipment is generally less expensive than x-ray equipment. These benefits warrant investigation into the utility of ultrasound tomography.

Although ultrasound tomography possesses many strengths, a major hindrance has prevented its widespread use. Ultrasound suffers from diffraction and refraction effects. In inhomogeneous objects, these effects create alterations in the ultrasound path that are difficult to represent geometrically. X-rays travel in straight lines, allowing the projections to be simple line integral functions. The ultrasound projection information is considerably more complicated.

Several approaches have been developed to correct for the effects of refraction and diffraction. Good introductions to these approaches and to ultrasound diffraction tomography were written by Greenleaf [1983] and Schueler *et al.* [1984]. Some of these methods, such as the Born and Rytov approximations, make assumptions about the size and refractive index of the object. The Fourier Diffraction Theorem, similar to the Fourier Slice Theorem used in x-ray tomography, assumes that the object is weakly diffracting. Frequency domain interpolation is prone to instability problems. The filtered backpropagation method, inspired by the filtered backprojection method used in x-ray tomography, possesses neither an efficient nor an accurate algorithm. These approaches will be discussed briefly in Chapter 2. A need exists to develop an alternative approach that may offer some situational advantages over the inadequate methods.

The alternative approach that will be explored involves the use of neural networks in the reconstruction of the object function from the projection information. Neural networks, discussed in more detail in Chapter 3, are highly parallel, distributed processing systems. Inspired by the topology of the human brain, neural networks can learn by example. This application of neural networks developed from the observation that the filtered backprojection method used in x-ray tomography can be implemented with a linear, parallel, distributed processing network. The theoretical development, tracing the connection from the x-ray case to the ultrasound case, can be found in Chapter 4.

Following the theoretical basis for the research, the experimental analysis is presented, starting with Chapter 5 and a discussion on the field equations needed to generate the training data for the neural network. Chapter 6 describes the results of tests that offer some insight into the performance of neural networks. After some successful, preliminary experiments, a neural network was trained based upon the theoretical approach using the x-ray analogy . The issues and results of this approach are discussed in Chapter 7. The use of neural networks in tomography is new, so future directions of research are envisioned in Chapter 8, along with a summary of the results.

CHAPTER 2

METHODS OF ULTRASOUND DIFFRACTION TOMOGRAPHY

2.1 The First Born Approximation

The presence of diffraction effects in ultrasound tomography prevents the use of straight ray models. Therefore, no direct methods exist that specify the propagation of acoustic waves. The first Born approximation is one method used to model the scattering process. The inhomogeneous wave equation representing the scattered field $u_s(r)$ is given by

$$u_s(\underline{r}) = \int g(\underline{r}-\underline{r}')\ f(\underline{r}')\ u(\underline{r}')\ d\underline{r}' \qquad (2.1.1)$$

where $u(r)$ represents the total field. The impulse response $g(r)$ and the object function $f(r)$ also appear. The problem with this equation is that the scattered field u_s is given in terms of the total field $u = u_0 + u_s$. The first Born approximation considers the scattered field in the integral above to be negligible compared to the incident field u_0, and is given by

$$u_s(\underline{r}) \cong u_B(\underline{r}) = \int g(\underline{r}-\underline{r}')\ f(\underline{r}')\ u_0(\underline{r}')\ d\underline{r}'. \qquad (2.1.2)$$

Higher-order Born approximations produce better results, but reconstruction algorithms do not exist. The first Born approximation holds only when the scattered field u_s is smaller than the incident field u_0. This condition limits the size of the object, since large objects scatter more than small objects if the other parameters are identical. This constraint can be stated in terms of the phase change, $\Delta\phi$, between the incident field and the field inside the object given by

$$\Delta\phi = 4\pi \, \Delta n \, \frac{a}{\lambda} \qquad (2.1.3)$$

where Δn is the change in the refractive index and a is the radius of the object cylinder. In other words, the first Born approximation is no longer valid if the phase change is greater than π.

2.2 The First Rytov Approximation

The Rytov approximation is an alternative to the Born approximation. Instead of considering the size of the scattered field, the Rytov approximation regards the total field as a complex phase, $u(\underline{r}) = e^{\phi(\underline{r})}$. Inserting the phase terms into Eq. (2.1.1), the inhomogeneous wave equation becomes

$$u_0\phi_s = \int g(\underline{r}-\underline{r}') \; u_0[(\nabla\phi_s)^2 + f(\underline{r}')] \; d\underline{r}'. \qquad (2.2.1)$$

The Rytov approximation can be made if the term $(\nabla\phi_s)^2$ is small in comparison to $f(\underline{r})$, yielding

$$u_0\phi_s = \int g(\underline{r}-\underline{r}')\ u_0(\underline{r}')\ f(\underline{r}')\ dr'. \tag{2.2.2}$$

This equation can be related to the Born approximation by

$$\phi_s(\underline{r}) = \frac{1}{u_0(\underline{r})} \int g(\underline{r}-\underline{r}')\ u_0(\underline{r}')\ f(\underline{r}')\ d\underline{r}' = \frac{u_B(\underline{r})}{u_0(\underline{r})} \tag{2.2.3}$$

In terms of the Born scattered field u_B, the Rytov approximation gives

$$u_B(\underline{r}) = u_0(\underline{r})\ \ln\left(\frac{u_s}{u_0} + 1\right). \tag{2.2.4}$$

However, the Rytov approximation is valid under different conditions than the Born approximation. Recall that $(\nabla\phi_s)^2$ must be much smaller than $f(\underline{r})$ to use the Rytov approximation. In terms of the change in the refractive index, Δn, this condition becomes

$$\Delta n >> \frac{(\nabla\phi_s)^2}{k_0^2}\ . \tag{2.2.5}$$

This condition differs from the Born approximation, since the change in the phase of the scattered field over one wavelength is important, and not the total phase change through the object.

2.3 Comparison of the Born and Rytov Approximations

Since both methods are approximations, they are not perfect in every situation. However, these approaches can produce good results, if certain requirements are met. The Born approximation requires the phase shift through the object to be less than π, even if the refractive index is large. The Rytov approximation holds even for large objects on the order of 100λ, but the change in refractive index must not be larger than a few percent [Kak and Slaney, 1988]. The Rytov approximation also depends upon phase unwrapping algorithms when discrete values of the field are used [Kaveh *et al.*, 1984]. When the object and the change in refractive index are small, the Born and Rytov approximations become similar, both giving a good estimation of u_B.

2.4 The Fourier Diffraction Theorem

The Fourier Diffraction Theorem is similar to the Fourier Slice Theorem of x-ray tomography in that it relates the 2-D Fourier transform of the object to the 1-D Fourier transform of a projection. The difference is that in the Fourier Slice Theorem, the Fourier transform of the projection is mapped onto a straight line in the Fourier domain of the object. Using the Fourier Diffraction Theorem, the mapping is onto a semicircular arc with a radius of $k_0 = 2\pi/\lambda$. Both theorems are related. At very small wavelengths, the Fourier Diffraction Theorem becomes the Fourier Slice Theorem, since the radius of the arc increases, causing the arc to straighten. A good discussion on this topic can be found in Mueller, Kaveh, and Iverson [1980]. For good reconstruction, many projections are needed to fill up the Fourier space. Then estimates of the object can be made up to a frequency of $k_0\sqrt{2}$ [Kak and Slaney, 1988]. Unfortunately, the Fourier Diffraction Theorem is limited to weakly scattering objects.

2.5 **Frequency Domain Interpolation**

The Fourier Diffraction Theorem provides the necessary data, but reconstruction methods using the scattered field information still need to be mentioned. Frequency domain interpolation is based on polynomial approximations of the samples of the scattered field in the Fourier domain. Kak and Slaney [1988] present a good introduction to this method. Frequency domain interpolation gives a bilinear interpolation that can be improved by increasing the sample density in the resultant frequency domain. Pan and Kak [1983] present a modification to this method by following the bilinear interpolation by direct two-dimensional Fourier inversion. Another method exists called unified frequency domain interpolation [Kaveh *et al.*, 1984]. This approach considers the spatial characteristics of the object. By considering the object's spatial limit, the interpolation can be carried out by convolving the Fourier transform of the object by a Bessel function. If only the main lobe of the Bessel function is used in the interpolation, the computation is reduced significantly. Frequency domain interpolation is the most frequently implemented method of reconstruction, due to its computational efficiency. However, since it is based upon data acquired using approximations and/or the Fourier Diffraction Theorem, the same limitations apply to this method. Also with interpolation, instabilities are possible.

2.6 **Filtered Backpropagation**

The filtered backpropagation method is an attempt to apply the computationally efficient and accurate filtered backprojection algorithm of x-ray tomography to diffraction tomography. Because the method begins with the Fourier Diffraction Theorem, the scattered field data lie on semicircular arcs in the frequency domain [Devaney, 1982]. For each projection, the data are filtered at different depths in the image. The projection is being smeared back over the image frame. Because this filtering is a discrete process, a pixel may not necessarily be situated on a known depth line. The nearest depth is then selected. This process is repeated for every projection, adding up all contributions at a certain pixel. Although this description is very brief, several shortcomings of this method can be seen. First, as in frequency domain interpolation, backpropagation is at the mercy of the restrictions of the Fourier Diffraction Theorem. Second, this method is not efficient. The implementation of the depth-dependent filter is computationally intensive. Third, since the depths are discrete, the algorithm is inexact for pixels not on depth lines. Interpolation may be necessary to improve the results.

CHAPTER 3

INTRODUCTION TO BACKPROPAGATION NEURAL NETWORKS

3.1 Introduction

The current state of ultrasound tomography is dependent upon the Born and Rytov approximations and the Fourier Diffraction Theorem, which are valid under certain conditions only. A new approach is needed that is not restricted by weak scattering assumptions. Neural networks have been used already in many different areas. They have been used in optimization and interpolation problems, as well as in image processing and pattern recognition. For example, Winters [1988] discusses a neural network that produces super-resolved images from ultrasound propagating in air. Other applications in image reconstruction include the neural network of Yoneyama *et al.* [1988] that recognizes objects from ultrasound reflection data. Neural networks have also been used to classify sonar backscatter from a metal cylinder and a stone [Gorman and Sejnowski, 1988].

These interesting applications differ from the work discussed here in several important aspects. First, the approach taken in this paper uses a neural network to extract quantitative information from the scattered field. We are not simply classifying objects. Second, the other techniques deal with acoustic reflection, instead of transmission tomography. Third, this work is based upon some theoretical premise, instead of being based on a hope that the neural network can solve the problem. In this manner, some understanding of the neural network's functions is gained. This theoretical foundation is presented in Chapter 4. Even though, some applications may seem similar to the one presented here, the above differences illustrate that the application of neural networks to ultrasound tomography is an original concept.

3.2 Fundamentals of Backpropagation Neural Networks

Many varieties of neural networks exist. A good introduction to them can be found in Lippmann [1987]. Following the lead presented in the theoretical development, Chapter 4, this work focuses on backpropagation neural networks (not derived from the filtered backpropagation algorithm in Chapter 2). Backpropagation networks are trained using a supervised learning technique called the delta rule which distributes the blame for the error back to all the elements in the network [Rumelhart, Hinton, and Williams, 1985]. The origin of the name comes from this method of learning. Lapedes and Farber [1987] have shown that backpropagation networks naturally implement the least mean square rule, allowing solutions of nonlinear system modelling problems to be discovered. They also predict the behavior of pseudo-random time series.

Before training can begin, input data must propagate up through the layers in the network to the output. The typical network architecture used in this research is shown in Figure 3.1. The network used in this research consisted of three layers and a bias. The middle layer is often called the hidden layer. The nodes are fully connected to the nodes in neighboring layers. The connections between the nodes or processing elements hold weights, values multiplied by the signal passing through the connection. When the weighted signals reach the processing element, they are added together internally before passing through a nonlinearity, typically a sigmoid function, but hyperbolic tangents and sinusoids are also possible. Figure 3.2 shows an individual processing element.

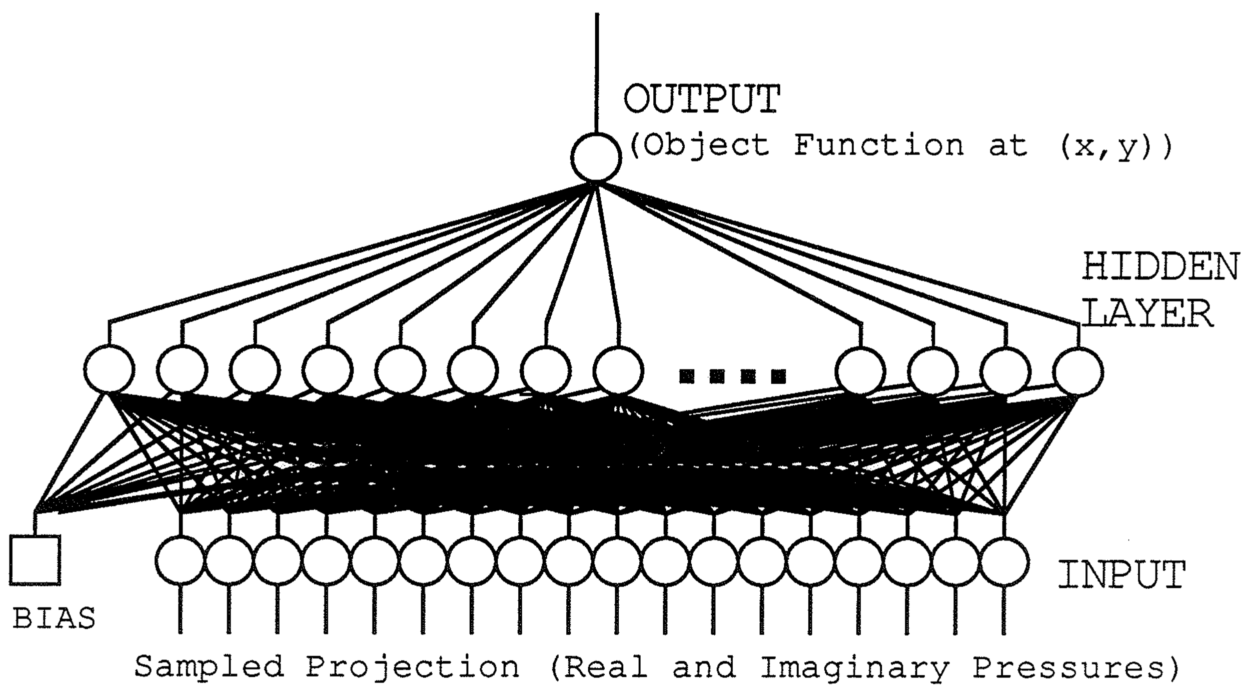

Figure 3.1. Backpropagation neural network.

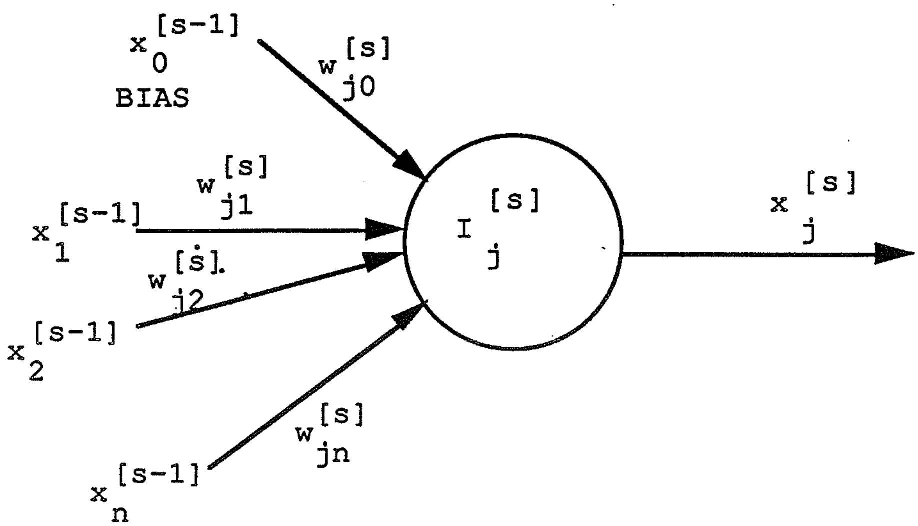

Figure 3.2. Processing element.

The process can be represented (using the notation from the NeuralWorks Professional II manual written by NeuralWare Incorporated) by

$$x_j^{[s]} = f\left(\sum_{i=0}^{n}\left(w_{ji}^{[s]}\ x_i^{[s-1]}\right)\right) = f\left(I_j^{[s]}\right) \qquad (3.2.1)$$

and

$$f\left(I_j^{[s]}\right) = \frac{1}{1+\exp\left(-I_j^{[s]}\right)} \quad \text{for a sigmoid function.} \qquad (3.2.2)$$

The layers are indexed by s, s-1, etc. The output of node j in layer s is $x_j^{[s]}$, and the connection weight from the ith node in layer [s-1] to the jth node in layer s is $w_{ji}^{[s]}$.

This process describes how data passes through the network. Training the network is a different process. The weights in the network are usually initialized to a random value. Training involves altering the weights to a correct value that gives the desired output. The training process begins by presenting an input i to each of the elements in the input layer. These values propagate through the network to the output layer according to Eqs. (3.2.1) and (3.2.2). At the output layer, the local error e_k is calculated by comparing the network output with the desired output d_k.

$$e_k = d_k - o_k \ . \tag{3.2.3}$$

The output processing elements are indexed by k. The connection weights are then altered according to

$$\Delta w_{ji}^{[s]}(t) = C_1 \, e_j^{[s]} + C_2 \, \Delta w_{ji}^{[s]}(t-1) . \tag{3.2.4}$$

The previous weight change is $\Delta w_{ji}{}^{[s]}(t-1)$. C_1 and C_2 represent learning coefficients. This weight change is calculated for each connection starting at the output layer and propagated back to the input. The first term on the right-hand side is dependent upon the local error. The second term, dependent upon the previous weight change, is known as the momentum term. The purpose of momentum is to speed up the learning process while preventing the convergence of the network into a local minimum. Since the delta rule learning method is a gradient descent algorithm searching for the global minimum in weight space, local minima are possible. The network may settle on a solution that is inexact or even nonsensical. Local minima are problems that the user should be aware of and should avoid. However, local minima have not prevented many successful applications of backpropagation neural networks. Chapter 6 discusses the lack of a local minima problem in this research. After the network converges to a correct solution, the weights are fixed. The neural network is now ready to perform the desired task.

CHAPTER 4

THEORETICAL DEVELOPMENTS

4.1 Statement of Problem

This chapter is devoted to the development of a theoretical basis for the use of neural networks to solve tomography problems. Ultrasound diffraction tomography at very high frequencies approaches the x-ray tomography case. Therefore, the tomography problem will be stated in terms of x-rays. Then, a transition to the ultrasound case will be made, bringing in the concept of neural networks. A good introduction to computerized tomography was written by Scudder [1978].

4.2 X-Ray Tomography

4.2.1 The Radon transform

Tomography, as considered in this work, is the cross-sectional imaging of an object by illumination at many different views and collecting projections using transmission information. The Radon transform is fundamental to the reconstruction of the image from the projections. The 2-D Radon transform, corresponding to the term *projection* in tomography literature, is given by

$$P_\theta(t) = \iint f(x,y)\ \delta(x \cos\theta + y \sin\theta - t)\ dx\ dy \qquad (4.2.1.1)$$

where $t = x \cos\theta + y \sin\theta$ (4.2.1.2)

and δ is the Dirac delta function. The Radon transform is the collection of one-dimensional line integrals through the object. The object function, the characteristic of interest, is represented by $f(x,y)$. The geometry used in this case is a parallel projection. Fan beam geometries are also possible, so the general equation for the projection or Radon transform is

$$P_\theta = \int_{line} f(x,y)\, ds \qquad (4.2.1.3)$$

where the integrals are measured along lines that form a fan. The fan beam geometry will be used throughout the remainder of this work (Figure 4.1).

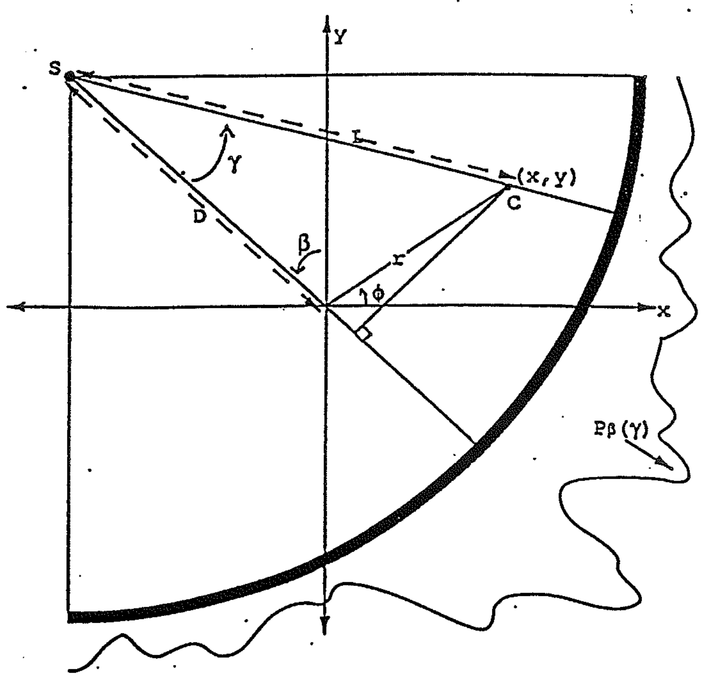

Figure 4.1. Fan beam geometry.

The typical ultrasound tomography experiment uses the fan beam geometry. The need to draw a direct comparison between the x-ray case and the ultrasound case led to the decision to use the fan beam geometry, although the reconstruction algorithms are more complex.

4.2.2 **The Fourier Slice Theorem**

The Radon transform gives a description of the projection. Unfortunately, the projection information does not give a direct reconstruction of this image. The Fourier Slice Theorem is an important relation stating that the 1-D Fourier transform of a projection maps to a line or slice in the 2-D Fourier transform of the object. The Fourier Slice Theorem is given by

$$S_\theta(w) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} P_\theta(t)\ \exp(-j2\pi wt)\ dt. \qquad (4.2.2.1)$$

If enough projections are gathered, the Fourier domain of the object fills up with lines containing slices of the Fourier transform of the object. By taking the inverse Fourier transform, the original image can be reconstructed. One problem arises, due to the higher concentration of information at the center of the Fourier domain of the object. Since all of the projections contribute slices through the origin, some high-pass filtering is needed to smooth out the extra contributions at the center. The filtered backprojection algorithm is designed to remedy this problem.

4.2.3 The filtered backprojection algorithm

X-ray tomography possesses a computationally efficient inversion algorithm known as filtered backprojection. The backprojection occurs by integrating the filtered backprojection data over all the views, according to

$$f(r,\phi) = \int_0^{2\pi} \frac{1}{L^2} Q_\beta(\gamma)\ d\beta \qquad (4.2.3.1)$$

where

$$L = L(r,\phi,\beta) = \sqrt{[D+r\sin(\beta-\phi)]^2 + [r\cos(\beta-\phi)]^2} \qquad (4.2.3.2)$$

and

$$\gamma = \tan^{-1} \frac{r\cos(\beta-\phi)}{D+r\sin(\beta-\phi)} . \qquad (4.2.3.3)$$

$Q_\beta(\gamma)$ is the filtered backprojection information at an angle of incidence β [Kak and Slaney, 1988]. $Q_\beta(\gamma)$ can be found by two distinct methods, both starting with projection data. Figure 4.2 shows these methods in chart form [Deans, 1983]. The Fourier method was mentioned earlier. The second route involves convolving the projection with a high-pass filter.

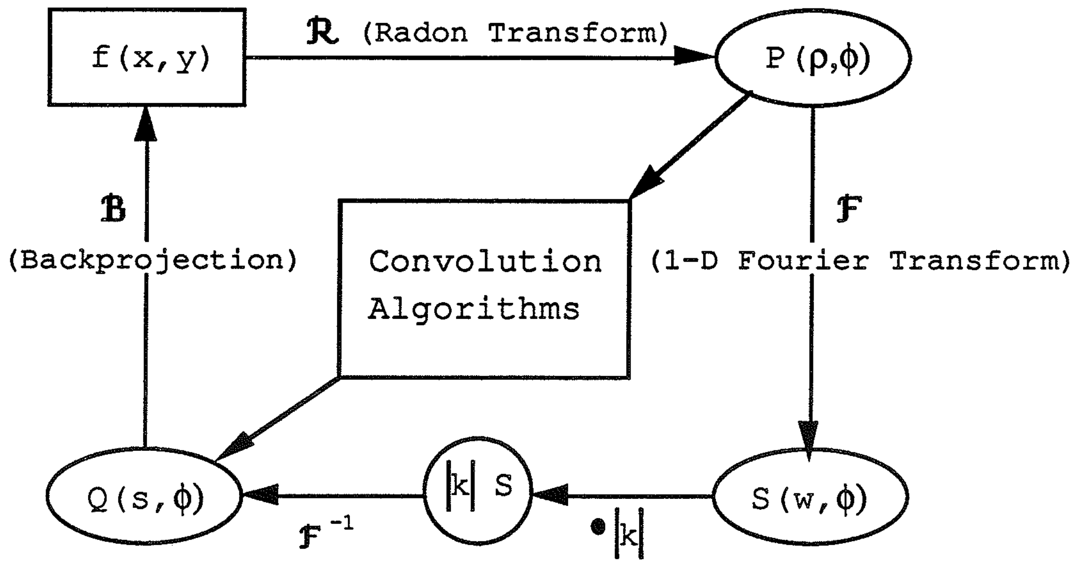

Figure 4.2. Methods of filtered backprojection.

The convolution method was used for several reasons. The purpose of this theoretical work is to design a network that can perform the filtered backprojection algorithm. The Fourier transform path requires substantial preprocessing, which is unjustified in the ultrasound case. Instead of simply entering the raw projection data, the Fourier transforms of the projections are needed. Second, a network that implements Fourier and inverse Fourier transforms must deal with complex weights. The simple implementation of the convolution route is not subject to these problems. The filtered backprojection data $Q_\beta(\gamma)$ can now be written as a convolution,

$$Q_\beta(\gamma) = P_\beta(\gamma) \otimes g(\gamma) , \tag{4.2.3.4}$$

where $g(\gamma)$ represents the high-pass filter. In order to implement the filtered backprojection algorithm on a network containing discrete elements, Eq. (4.2.3.4) must be discretized.

4.2.4 **The discrete filtered backprojection algorithm**

A practical x-ray tomography experiment consists of discrete transducers, and the projections are gathered at discrete angles. A network with discrete elements requires the input data to be sampled. Therefore, the filtered backprojection algorithm must be discretized. Equation (4.2.3.1) becomes

$$f(x,y) \cong \frac{2\pi}{M} \sum_{i=1}^{M} \frac{1}{L^2(x,y,\beta[i])} Q_{\beta[i]}(n\alpha) . \qquad (4.2.4.1)$$

The filtered backprojection data $Q_\beta(\gamma)$ become

$$Q_{\beta[i]} = \alpha \sum_{k=-(N-1)}^{N-1} g(n\alpha - k\alpha)\, P_{\beta[i]}(k\alpha) . \qquad (4.2.4.2)$$

Discretization poses a new problem. Equations (4.2.4.1) and (4.2.4.2) assume that the location of interest (x,y) is situated on one of the lines in the fan beam or, in other words, a line between the transmitter and a receiver. Therefore, the angle of incidence, γ, can be matched to $n\alpha$, where α denotes the angle increment between adjacent transducers. Interpolation is necessary between points that do not lie upon one of these lines. The discretized equations will be used in a network implementation of the filtered backprojection algorithm.

4.3 **Network Implementations**

4.3.1 **Carroll's implementation**

Carroll and Dickinson [1989] describe how a neural network can implement the filtered backprojection algorithm. This work is interesting for several reasons. It shows theoretically that an analogy can be derived between an algorithm and a neural network. This exercise illuminates the constraints, limitations, and practical bounds of the neural network, and provides a method to parameterize a multidimensional function. This approach allows the user to set the weights, so that training is not necessary.

Carroll begins with the discretized backprojection equation (4.2.4.2). Cybenko [1989] shows that any smooth function can be represented by a linear combination of sigmoids (σ). Using Cybenko's theorem, Eq. (4.2.4.2) can be replaced by

$$Q_{\beta[i]} = \sum_{j=1}^{M} a_{ij}\, \sigma(\underline{u}_i \bullet \underline{x} + b_{ij}) \qquad (4.3.1)$$

The convolved filter $g(n\alpha - k\alpha)$ and the projection data $P_\beta(k\alpha)$ have been replaced by a sum of sigmoids. The remaining information becomes a constant a_{ij}. The bias is given by b_{ij}.

Equation (4.3.1) closely resembles the output of a neural network (Eq. (3.2.1)) with one hidden layer, given by

$$Q_{\beta[i]} \cong O(\underline{x}) = \sum_{i=1}^{N} a_i \, \sigma(\underline{w}_i \bullet \underline{x} + b_i). \qquad (4.3.2)$$

The network that results from Eq. (4.3.1) is shown in Figure 4.3.

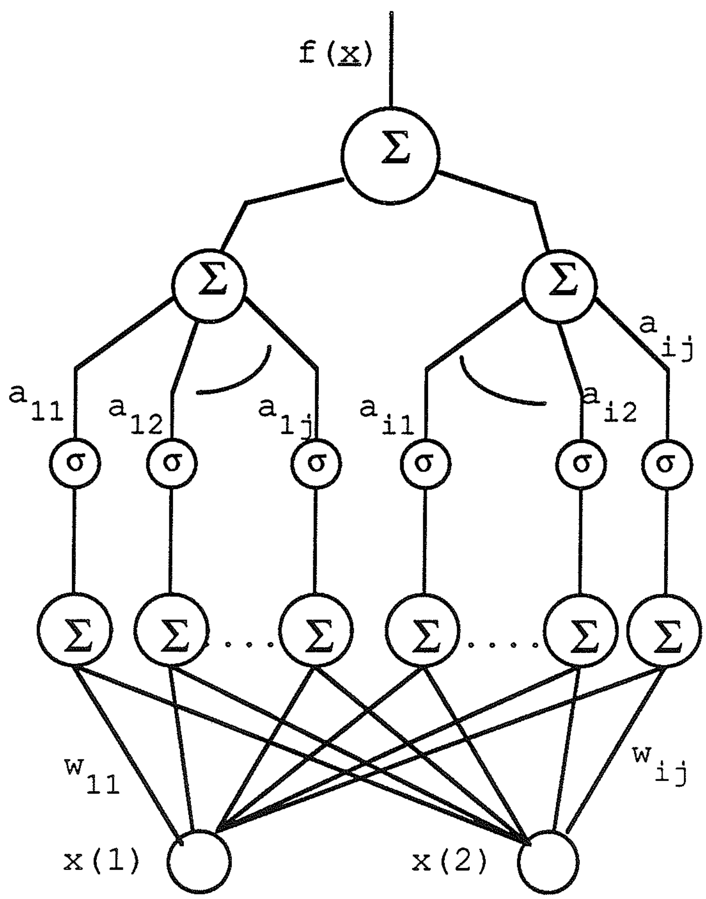

Figure 4.3. Carroll's neural network.

The input to the network is the location information $\underline{x}$. The weights on the connections to the hidden layer contain the angular information $\underline{u}_i$. Then the resultant signal, $\underline{u}_i \bullet \underline{x} + b_i$, is passed through sigmoids that are scaled by the factor a_i. The sigmoidal scaling factor contains the projection information. The hidden layer can be divided into groups of elements. Each group contains the linear sigmoidal combination for a particular irradiation view. The output of the network is the object function at the point of interest.

Several practical problems exist with this method. One problem is that the filtered backprojection information is decomposed into a linear combination of sigmoids by the user. This decomposition may prove very difficult, especially if the function contains large derivatives. However, since Q is two-dimensional, the sigmoidal decomposition can be determined using splines and interpolation. Another problem is the universality of this method. Only one image can be determined with this network. However, any location can be entered into the network and the object function is then computed at that location. The direction of this work requires that one network can handle any image. This implementation is discussed in Section 4.3.2.

4.3.2 **Linear network implementation**

The long-term goal of this research is to construct a neural network that can image objects, such as tumors, from the projection information. Therefore, one network must handle any image presented. The input to this network should be the projection information, and the output should be the object function at the pixel of interest. This network differs from Carroll's network. The input to that network is the location of interest.

The following discussion details the implementation of a linear network, instead of a neural network, that performs the filtered backprojection algorithm. This linear network will act as the base from which the extension to ultrasound tomography will be made. The linear network shown in Figure 4.4 implements Eqs. (4.2.4.1) and (4.2.4.2) in the following way:

1) Select an appropriate point of interest within the object (x,y).
2) Irradiate the object at M equally spaced angles $\beta[i]$, i = 1 to M.
3) Gather projection information $P_{\beta[i]}(\gamma)$.
4) Sample the projection at discrete angle intervals α to obtain $P_\beta(n\alpha)$.
5) Convolve the projection with the high-pass filter $g(n\alpha)$ where

$$g(n\alpha) = \begin{cases} \dfrac{1}{8\alpha^2} & n=0 \\ 0 & n \text{ even} \\ \left(\dfrac{\alpha}{\pi\alpha\sin(n\alpha)}\right)^2 & n \text{ odd} \end{cases} \qquad (4.3.2.1)$$

The sampled projection serves as the input to the network. The connection weights contain the high-pass filter information, shifted to perform the convolution. The network multiplies the input by the value of the connection weight, and the weighted signal is then passed on to the summation function. This process is repeated in the next layer. Therefore, the linear network (Figure 4.4) realizes the object function at a point (x,y) for any object.

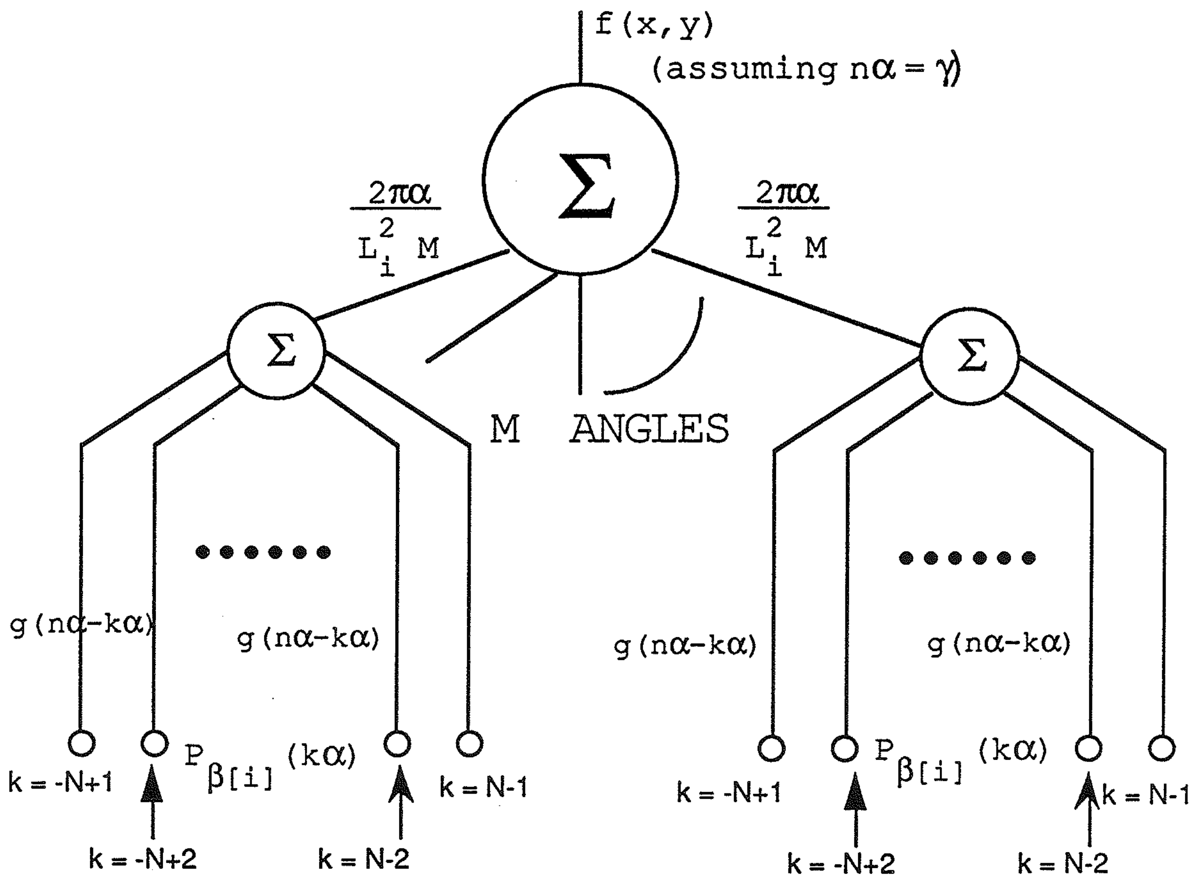

Figure 4.4. Linear network implementation of the filtered backprojection algorithm.

If other points of interest are desired, two methods are possible. Networks with weights set for particular pixels can be connected together to handle an entire image. This modular principle of neural networks was demonstrated by Waibel [1989]. Of course, this principle still needs to be evaluated for the ultrasound networks. Another approach uses only one network. This network can handle an entire image on a pixel by pixel basis, using a look-up table. The set of weights does not change, but they lie upon different connections, depending upon the view and the location of interest.

4.4 **Extension to Ultrasound Tomography**

4.4.1 **Introduction**

The network realization of the filtered backprojection algorithm can be extended to the ultrasound case. As stated in Chapter 1, the ultrasound case is considerably more complicated. Diffraction effects caused by acoustic attenuation and refraction effects caused by changes in the speed of sound result in waves that do not propagate in a straight line. However, in the limit of very high frequencies and in the absence of refraction, the acoustic and x-ray cases become analogous, with acoustic attenuation replacing density.

4.4.2 Consideration of diffraction effects

In the limit of very high frequencies, ultrasound that insonates a non-refracting, attenuating object produces shadows with well-defined edges, similar to x-rays. Therefore, starting with the linear network (Figure 4.4) at a very high frequency, a smooth transition can be made to the ultrasound diffraction case by gradually lowering the frequency and introducing a nonlinearity following all of the network's summation functions. This nonlinearity transforms the linear summation function into a neural network processing element (Figure 3.2). The purpose of a nonlinearity is to endow the network with the ability to map nonlinear functions. Applied to the ultrasound case, the neural network can map projection information to a characteristic that has a nonlinear effect on the scattered field, such as attenuation or size. This nonlinearity broadens the range of functions the neural network can learn. In this case, the most practical nonlinearity is the hyperbolic tangent function. This choice is made because the projection information can have negative values. The sigmoid function extends from 0 to 1, but the hyperbolic tangent ranges from -1 to +1. Of course, if the magnitude of the projection information is greater than 1, the input-output pairs presented to the network during training need to be scaled to within -1 and +1.

Every time the network is used to generalize to other values, the projection data need to be scaled in the same manner. In order to keep the analogy to the x-ray case valid, the input-output pairs should be linearly scaled, so that the initial connection weight values are not changed from the x-ray case values. After slightly reducing the frequency and adding a gentle nonlinearity, the network is trained using input-output pairs of data generated at the high frequency. The training will slightly alter the weights, compensating for the added small effect due to diffraction. This process is repeated until the network is trained at a suitable ultrasound frequency. As well as compensating for diffraction effects, the network avoids local minima in weight space. This desirable quality occurs since the network has been partially trained by the use of initial weights from the x-ray case. The network converges faster and along a gradient that offers the best solution, not along a random gradient.

Sometimes a neural network is not necessary to perform the inversion of tomography data. If the following three conditions hold, then a linear network can be used. First, the geometry must be known. Second, the geometry must be simple. Third, the object function must have a linear effect on the scattered field. If the first two conditions hold, the user can calculate the effects of diffraction and alter the connection weights accordingly. If the third condition does not hold, a neural network is needed to map the nonlinear relation. If any one of those conditions is not valid, then the neural network is necessary.

4.4.3 Consideration of refraction and diffraction effects

The previous theoretical discussion can be extended to acoustic speed reconstruction, by incorporating phase information. In this case, refraction, as well as diffraction, effects are now present. Again, starting from the x-ray case, assume that the phase information can be collected. The object function is now complex, if acoustic attenuation and change in acoustic speed are regarded. The real and imaginary parts of the object function can be calculated separately using two distinct linear networks. The ultrasound case requires that the networks for the real and imaginary parts be interconnected (Figure 4.5). The connections between the networks for the real and imaginary parts are necessary since both refraction and

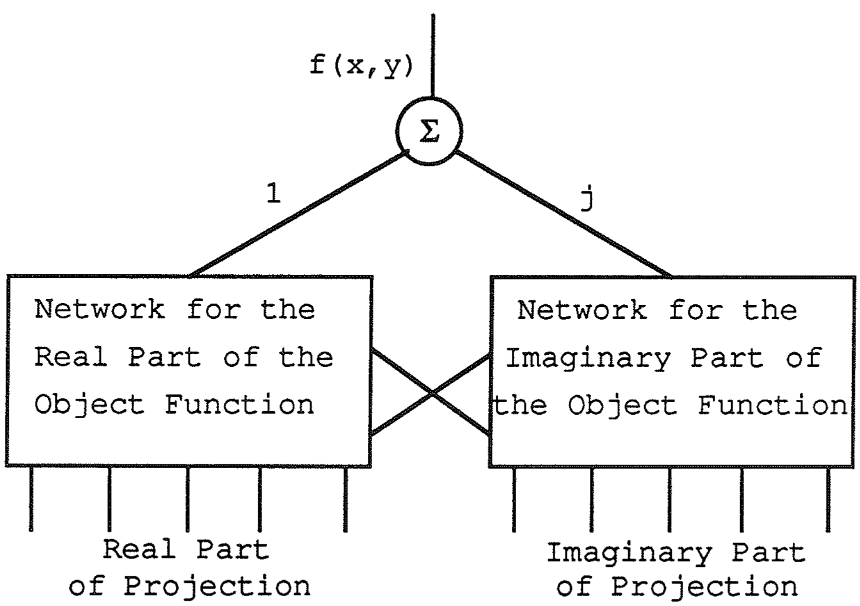

Figure 4.5. Interconnection scheme to handle refraction and diffraction effects.

diffraction effects contribute to the scattered field in such a manner that their individual effects cannot be separated. The initial weights in the interconnections must be set to small random values, since they have no relation to the x-ray case. The weights within each discrete network can be set initially to the x-ray case. Knowledge of these weights *a priori* can still decrease the training time and prevent against falling into local minima. The network must be trained more than in the case when only diffraction is present, due to the added interconnections. The interconnected scheme with these initial weights can be trained as in the case with only diffraction present. The resulting network can accurately invert projection data from refracting and diffracting objects.

CHAPTER 5

EXPERIMENTAL SYSTEM

5.1 Introduction

The theoretical development presented in Chapter 4 concluded that a neural network is naturally suited to perform the inversion of ultrasound tomography data. This idea now needs to be tested. This chapter details the experimental geometry and the data generation used in the neural network analysis. Questions that need to be answered include the positioning of the transducers, the kind of source, the kind of scatterer, and the generation of training data for the network.

5.2 Location of the Transducers

Several options exist for the placement of the transducers. The first situation resembles the typical tomography experiment. The transmitter is located on a ring of transducers surrounding the object. This geometry was used in preliminary experiments that tested the basic capabilities of neural networks. The second geometry, common in x-ray tomography, is called parallel beam. This method involves scanning the object along a line perpendicular to the direction of propagation. Then the entire set of transducers is rotated around the object. This technique is perhaps the simplest to express mathematically. The third method, called the fan beam, resembles the first method except that the transmitter is located in the center of the ring of receivers, instead of on the ring. The projection information can be gathered simultaneously without moving the transmitter, as in the parallel beam case. Although the fan beam geometry is faster in practice, it is also slightly more complicated mathematically. The fan beam geometry was used in the theoretical development with the x-ray analogy. Therefore, this geometry should be used when making the transition from the x-ray case to the ultrasound case.

5.3 **Programming Considerations for the Geometry**

Two programs were written to generate the transducer locations, one for each of the two different fan beam geometries. The first program, called *points.for*, deals with the geometry with the transmitter on the ring of transducers. The second program, called *fan.for*, generates transducer locations for the geometry with the transmitter in the center of the transducer ring. These programs were written in Microsoft Fortran. The computer used was a Tandy 4000 with 640 kbytes of memory.

5.4 **The Source Type**

Several kinds of sources are possible. The intention is to use one that is closest to the x-ray case as possible. One problem is that sound does not propagate in the same manner as x-rays. Because x-rays travel in straight lines, geometric ray tracing techniques can be used. One possible source would be a plane wave. This source is similar to the x-ray source, and is desirable. Another possibility, also well-suited for the fan beam geometry, is the line source. Unfortunately, this source does not replicate the x-ray case. However at high frequencies, rays perpendicular to the outgoing cylindrical wave can approximate the x-ray case. However, the density of the rays decreases as the energy spreads out over a larger area, causing the intensity to decrease as 1/r. The ray density issue does not cause a problem with the projection information. The projection contains a ratio of the total field over the incident field. In this situation, the 1/r term is cancelled out. The attenuation due to the scatterer is much larger than the decrease due to ray spreading. The projection information from the plane wave source was compared to the line source, and found to be identical. Therefore, the line source was used in the experiments.

5.5 **Experimental Target**

The object or target must possess several characteristics. The target should be a simple shape for the initial experiments. A benefit of using a simple shape is that the exact scattered field can be calculated for any location. These field equations are discussed in Section 5.8. If the object is a simple shape determined by the experimenter, then the first two conditions discussed in Section 4.4.2 are valid. An interesting experiment can be performed (Section 6.6.2) to test the appropriateness of a neural network by varying object functions that have a linear effect and those that have a nonlinear effect on the scattered field. For these reasons, an infinite, circular cylinder was chosen as the target. An additional benefit of the cylinder is derived by exploiting symmetry (Section 5.6). The cylinder will be used throughout the remainder of this work. Many characteristics can be varied, such as the radius, the acoustic speed, the attenuation, and the location. The neural network can be tested on its effectiveness at quantifying these parameters. Other targets can be used, including complicated shapes with many inhomogeneities. In this case, other methods are needed to generate the neural network training and test data.

Finite element methods can be used to generate training data when the target is highly heterogeneous or irregular. The finite element method, along with variants such as the finite difference method, provides a powerful tool for approximating solutions to differential equations [Zienkiewicz and Morgan, 1983]. The finite element method begins by dividing up the field into sections, such as triangles or hexagons, in order to generate a grid or mesh. The vertices of adjacent sections form nodes. The object function at the nodes of the grid yields a large set of simultaneous equations that solve the differential field equations. These equations can construct a very sparse matrix. The sparsity arises since each node belongs to only several bordering sections or elements. Exploiting the sparsity, the matrix can be inverted, providing a solution at the nodes. If the grid is constructed with the nodes at the transducer locations, then the field can be known to a high degree of accuracy.

The computational requirements of the finite element method prevent its use in this work. For good approximations, 10 elements are needed in each wavelength. For instance, the division of the area of a transducer ring with a radius of 75 mm requires on the order of 10^{12} elements when the frequency is 2 MHz. This very large matrix is difficult to invert. Therefore, hardware restrictions kept this work from using the finite element method to find the field surrounding the object. The area of the target is very small in comparison to the area of the transducer ring, so that changes in the field are small outside of the target. The finite element method becomes very wasteful in the semi-infinite area beyond the target. A method by Winkler [1986] uses infinite elements based upon a Green's function in that area.

When very complicated targets are needed, perhaps for medical diagnosis, the training data can be obtained experimentally. In some situations, however, acquiring large sets of data may be difficult.

5.6 **Symmetry Considerations with Cylindrical Object**

Perhaps, the best reason to use an infinite, circular cylinder is to take advantage of its circular symmetry. This symmetry allows the user to make initial experiments with a small neural network. In order to reconstruct any object, many views around the target are needed. If 1000 transducers surround the object, the projection is being sampled 1000 times. Then the transmitter is rotated around the object at 1000 locations, resulting in 1000 views. Using the x-ray approach, each view has its own set of input processing elements, so the resulting input layer contains 10^6 input elements. This neural network is too large for the current capabilities of the computer and software used. (Section 6.1 discusses the hardware and the neural network software.) In order to test the neural network, a different approach needs to be taken. If the point of interest is located at the origin and the homogeneous cylinder is centered at the origin, only one view is necessary with the fan beam geometry. This concept can be explained by considering the linear network (Figure 4.4). The projections are identical for each view around the object. The weights from the input to the hidden layer and those from the hidden layer to the output are the same for each view. Therefore, the signal leaving the hidden layer, Q, is the same for each view. Therefore, if there are M views, the output of the hidden layer would be M Q. Fortunately, the weights from the hidden layer to the output depend upon the reciprocal of M. This factor divides the output of the hidden layer by the number of views, causing the output of the network to be independent of the number of views. Only one view is needed to describe the object function at the origin, since there are, in essence, an infinite number of views. If the point of interest is not at the origin, the projection information is still the same due to the symmetry of the object. However, the weights containing the filter information vary for each view. The term ($n\alpha - k\alpha$) changes, because n varies with the view.

Another advantage of using the origin as the point of interest is that the lines of the fan beam intersect there for every view. Interpolation is not necessary to determine contributions from views where a line from the source to the transducer does not pass through the point of interest.

5.7 **Experimental Geometry**

Using the fan beam geometry, the line source, and the cylindrical target, the geometry approximates the x-ray, fan beam geometry. Figure 5.1 shows the geometry with the source on the transducer ring, and Figure 5.2 has the transmitter at the center of the ring. In both figures, the source is the open circle, and the cylinder is at the origin.

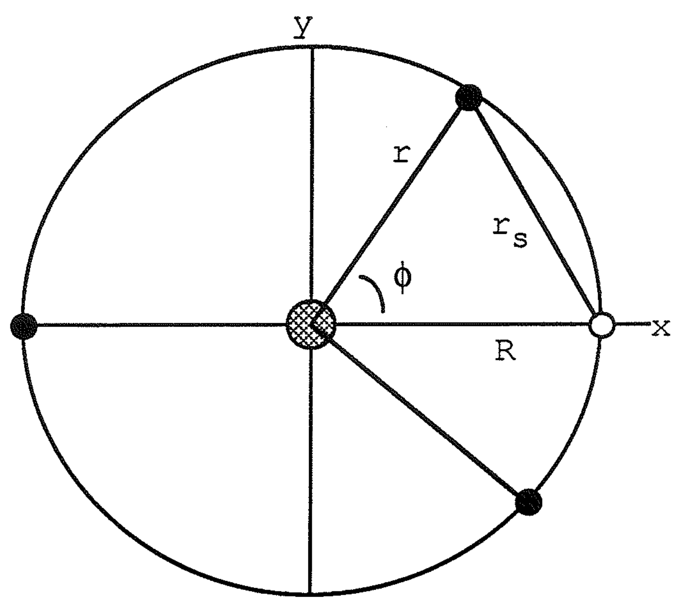

Figure 5.1. Fan beam with source on transducer ring.

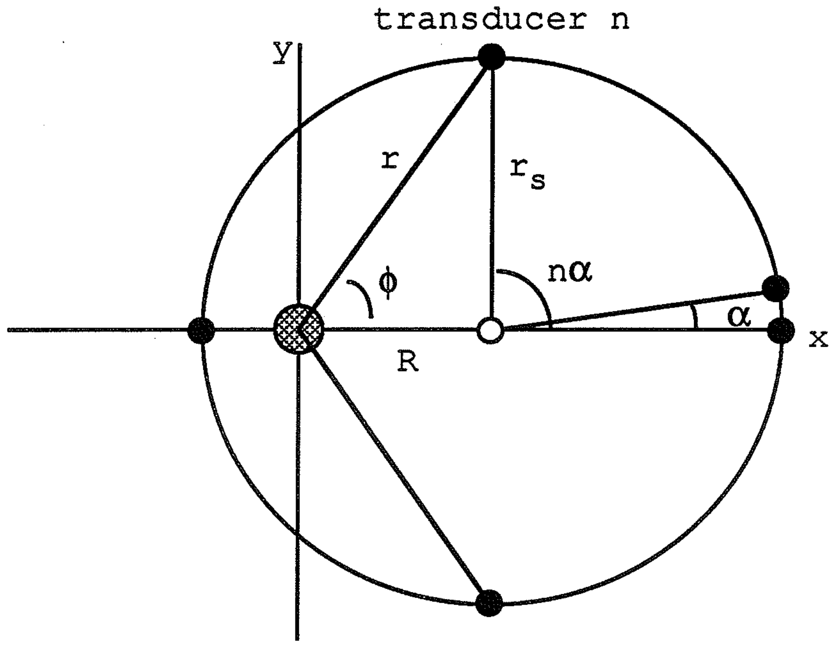

Figure 5.2. Fan beam with source at the center of the transducer ring.

5.8 **Exact Field Equations**

This section presents the exact field equations associated with the acoustic scattering of an incident cylindrical wave by an infinite circular cylinder. This situation is not commonly found in the literature. A similar problem is discussed in Morse and Ingard [1968], using a plane wave source instead. A cylindrical wave is used in a derivation by Shenderov [1962], but the cylinder is rigid. However, Cavicchi [1988] provides exact field equations for this situation.

The equations for the incident and scattered fields are basically combinations of Bessel and Hankel functions. $J_m(z)$ represents a Bessel function of the mth order with an argument z. $H_m^{(2)}(z)$ represents a Hankel function of the second kind with mth order and an argument z. A Hankel function of the second kind and mth order is related to a Bessel function by $H_m^{(2)}(z) = J_m(z) - jY_m(z)$.

Using the same notation used in Figures 5.1 and 5.2, the incident cylindrical wave is given by

$$f^{inc}(r,\phi) = J_0(k_0 r)\ H_0^{(2)}(k_0 R) + 2 \sum_{m=1}^{\infty} J_m(k_0 r)\ H_m^{(2)}(k_0 R)\cos(m\phi) . \qquad (5.8.1)$$

Equation (5.8.1) represents an expansion of the incident field equation. If the source is centered at the origin, then $f^{inc} = H_0^{(2)}(k_0 r_s)$. Since the source is not at the origin, the expansion must be used. Several equations are needed to fully specify the scattered field, given by

$$f^{sc}(r,\phi) = \sum_{m=0}^{\infty} S_m\ H_m^{(2)}(k_0 r)\ \cos(m\phi) \quad (r \geq a) . \qquad (5.8.2)$$

The symbol, a, denotes the radius of the cylinder. The scattered field weighting coefficients, S_m, need to be specified. First consider m = 0. The weighting coefficient S_0 is

$$S_0 = (-1/\Delta_0)\ [J_1(k_0 a)\ J_0(k_1 a) - J_0(k_0 a)\ J_1(k_1 a)\ Z_r]\ H_0^{(2)}(k_0 R) \qquad (5.8.3)$$

where

$$\Delta_0 = H_1^{(2)}(k_0a)\ J_0(k_1a) - H_0^{(2)}(k_0a)\ J_1(k_1a)\ Z_r. \quad (5.8.4)$$

Now consider the case where $m > 0$. S_m is given by

$$S_m = (-2/\Delta_m)\ H_m^{(2)}(k_0R)\ \{J_m(k_1a)\ [J_{m+1}(k_0a) - J_{m-1}(k_0a)] - J_m(k_0a)\ [J_{m+1}(k_1a) - J_{m-1}(k_1a)]\ Z_r\} \quad (5.8.5)$$

where

$$\Delta_m = J_m(k_1a)\ [H_{m+1}^{(2)}(k_0a) - H_{m-1}^{(2)}(k_0a)] - H_m^{(2)}(k_0a)\ [J_{m+1}(k_1a) - J_{m-1}(k_1a)]\ Z_r. \quad (5.8.6)$$

The complex impedance is given by

$$Z_r = \frac{\rho_0\ c_0}{\rho_1\ c_1}\left(1 - j\,\frac{\alpha_1\ c_1}{\omega}\right) \quad (5.8.7)$$

where α_1 is the attenuation in (Np/mm), and ω is the frequency in M radians s^{-1}. The acoustic speed in (km/s) and density of the cylinder are given by c_1 and ρ_1. The symbol k_0 denotes the wave constant of the surrounding medium. Then the total field is simply denoted by $f = f^{sc} + f^{inc}$ [Cavicchi, 1988]. These equations are used to generate neural network training and test data.

5.9 **Programming Considerations for the Field Equations**

The program *origcyl.for* generates the exact field at the transducer locations specified by *points.for* or *fan.for*. Figure 5.3 shows a block diagram explaining the various sections of *origcyl.for*.

When generating training and test data for the neural network, *origcyl*, *points*, and *fan* act as subroutines. Various master programs using these subroutines can provide large sets of data with varying parameters of the cylinder. These master programs, such as *alearn* and *atest*, control whether the object function is varied incrementally or randomly.

The program *origcyl.for* has restrictions on its use, due to the Bessel function library, *hankel.lib*. This library was written by Donald Amos of Sandia National Laboratories [1986]. Bessel and Hankel functions can be calculated easily when the frequency is on the order of 1 MHz. However, the issue of the Bessel function calculations needs to be considered whenever the acoustic frequency is on the order of x-ray frequencies. The Bessel function routines by Amos consider many different kinds of functions at large and small arguments and orders. The program also offers error flags. However, all calculations of Bessel and Hankel functions with large arguments must be verified.

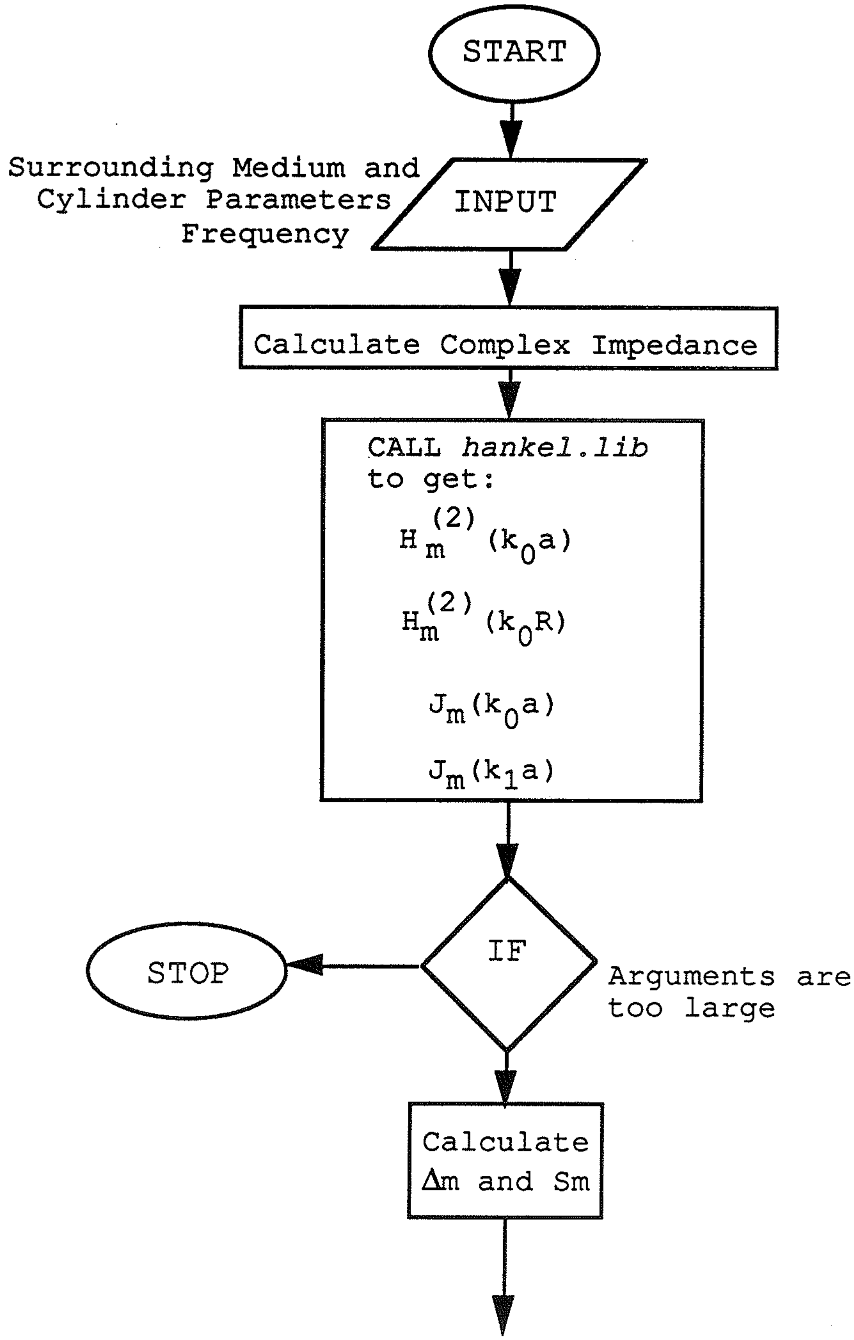

Figure 5.3. Block diagram of *origcyl.for*.

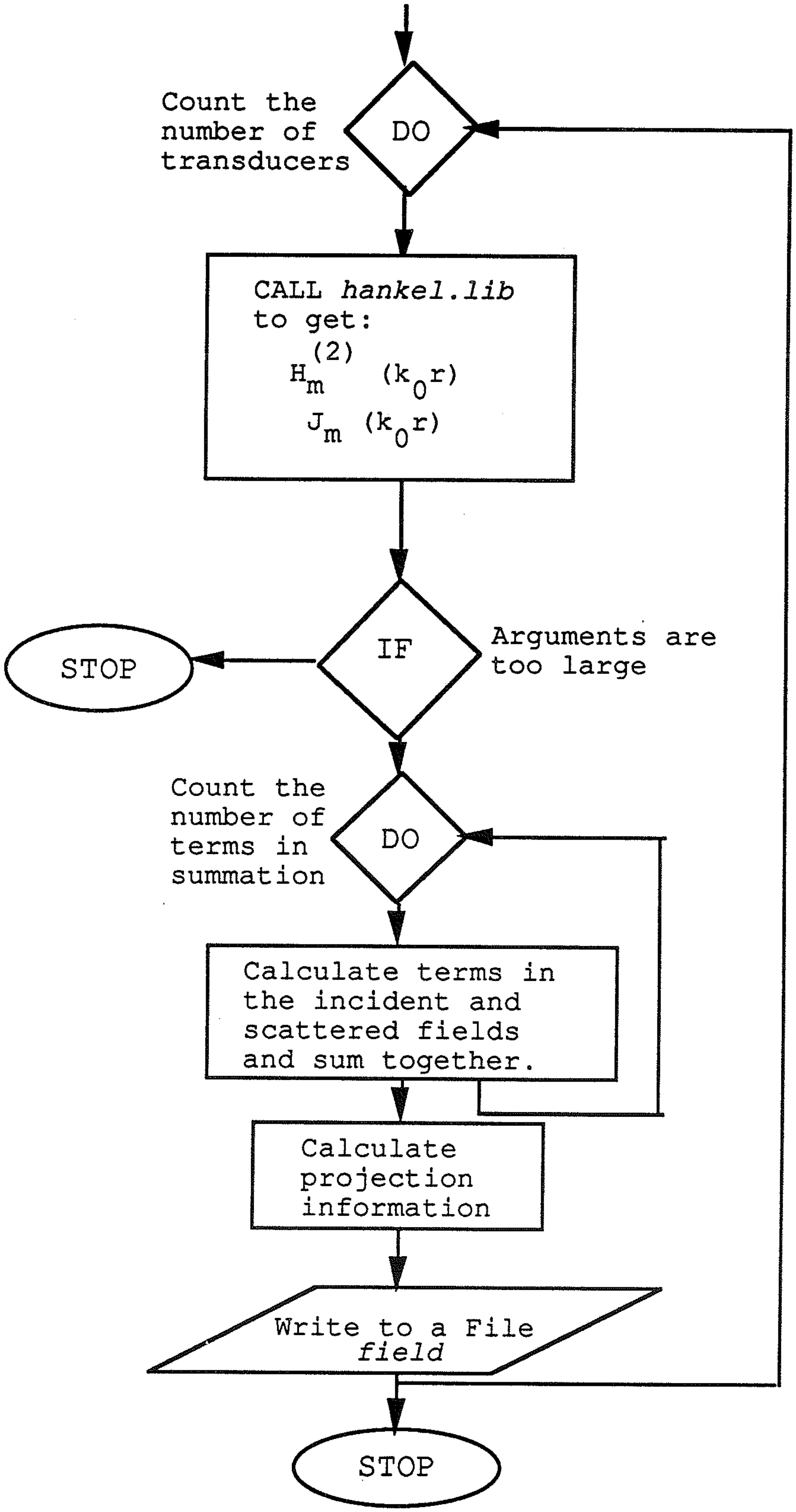

Figure 5.3. (Continued).

CHAPTER 6

PRELIMINARY EXPERIMENTS

6.1 Motivation

Several experiments were conducted to determine basic performance characteristics of neural networks on tomography data. These investigations were kept simple in order to avoid introducing extraneous parameters. The neural networks were simulated on a serial computer, the Tandy 4000, using the NeuralWorks Professional II package by NeuralWorks Incorporated. All of these preliminary experiments used the fan beam geometry with the source on the ring of transducers surrounding a circular cylindrical target.

6.2 The Effect of Learning Rates on Convergence

In order to keep the tests very simple, these first experiments used the acoustic speed in the cylinder as the object function. As stated in Section 4.4.2, a neural network is not necessary in this case. For this experiment, a problem is selected that the neural network can definitely solve. The first experiment involved varying the learning rate and the momentum term. The simple backpropagation neural network had 32 input elements that accepted the real and imaginary pressures, scaled between 0 and 1, from each of the 16 transducers surrounding the object. Due to the symmetry of the geometry, only 9 of the 16 transducers were necessary. However, all 16 were used at this time, in order to avoid unnecessary considerations. The hidden layer contained 8 elements. This selection was random. The issue of hidden layer size will be addressed in Section 6.6. The output layer contained 3 elements. This binary output classified the acoustic speed in the object into three ranges. Each test began by randomizing the connection weights. A convergence criterion caused the training process to end when the RMS error in one of the output elements fell below 0.002. This choice was not important, since the relative changes were of interest. The learning coefficient C_1 was held constant at 0.4, and the momentum term C_2 was varied between 0.0 and 0.9. Figure 6.1 shows the effect of changing the momentum term on the number of presentations of input-output training pairs needed to meet the convergence criterion. In general, when the momentum coefficient is made larger, the neural network needs fewer presentations to converge. The deviations shown in Figure 6.1 could be due to the different initial weights used in each test. When C_2 is equal to 0.9, the network falls into a local minimum. This occurrence can be seen from the graph. The network converged with very few presentations relative to the other tests. However, after testing, the network was found to have converged incorrectly.

For the next set of tests, the momentum term was fixed at 0.3, and the learning coefficient was varied between 0.1 and 0.9. Figure 6.2 shows the effect of varying C_1 on the convergence rate. This test was repeated with C_2 equal to 0.6, and again with 0.9. C_2 is equal to 0.3 and 0.6 for the top and bottom curves, respectively, in Figure 6.2.

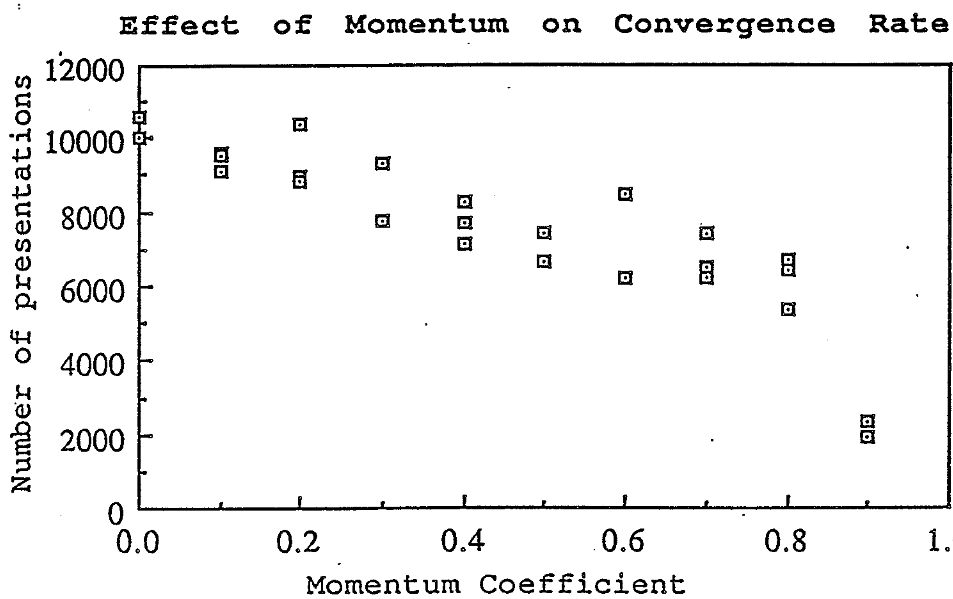

Figure 6.1. The effect of the momentum coefficient on the number of presentations needed for convergence.

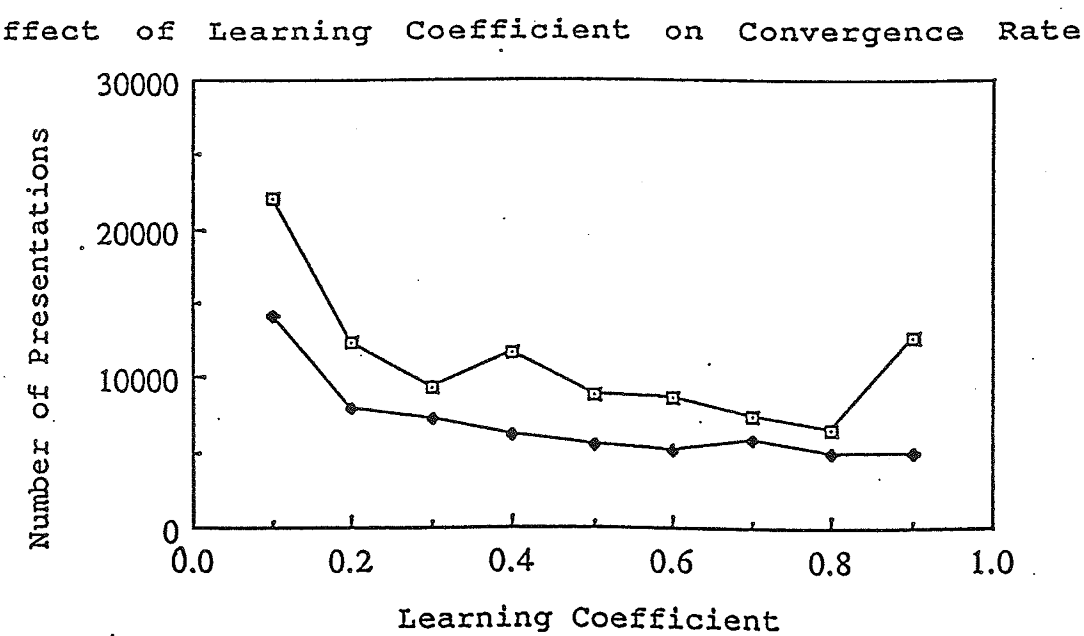

Figure 6.2. The effect of the learning coefficient on the number of presentations needed for convergence.

The conclusion of these tests is that whenever C_1 or C_2 is increased, the neural network needs fewer presentations to learn. However, when both of the coefficients are increased to a certain degree, the network converged quickly into a local minimum.

6.3 The Effect of Small Data Sets

The next set of experiments examined the effect of small training sets on the learning process. This experiment used neither the tomography data nor the fan beam geometry, because the neural network needed a very difficult problem this time. If the problem were too easy, the network may solve the problem so the relative difficulty of using small training sets may not be discernible. The input layer, consisting of 9 processing elements, received various tissue parameters, including the attenuation of sound at various frequencies in rat liver, the absorption, the mean speed in the liver, and the mean heterogeneity index. The output layer consisted of four units that gave the continuous-valued water, protein, lipid, and collagen information. Each input-output pair required the sacrifice of an animal, so the size of the training set was very restricted, on the order of 20 pairs, as opposed to 80 pairs used in the previous set of experiments. This backpropagation network started with two hidden layers each with 30 processing elements. Because the neural network was unable to learn with this morphology, the size of the hidden layers was reduced to 8 elements. One reason for the failure of this morphology was that the error, or blame, was spread among too many elements. Instead of trying to obtain four output functions simultaneously, the output layer was reduced to one element for the water content. With C_1 set at 0.2 and C_2 set at 0.05, the neural network converged after 184,500 presentations. The network recalled the training data perfectly, but it failed to generalize to parameters other than those used during training. Several other attempts were made by varying the learning rate and the size of the hidden layer. These details will not be mentioned since the network failed to generalize in every case.

Another simplification was attempted. Again, the water content was the function of interest, but five binary units were used to classify the output into ranges, instead of the one continuous-valued output. The network received 820,494 training presentations. Again, the result was the same. When the training set contained too few input-output pairs, the neural network could recall the same information used during training, but the network failed to generalize to parameters previously unseen.

6.4 **The Use of Continuous-Valued Outputs**

The previous experiments, except for those using the small training set, dealt with binary outputs. Continuous-valued outputs are more challenging for the neural network. In the binary case, the network classifies the object function into ranges of values. If one of the outputs is close to 1, above 0.8 for example, and the others are close to 0, below 0.2, then the output is conclusive. Continuous-valued outputs will give an approximation to the actual value. Therefore, the output needs to be more precise. The next set of experiments was identical to the first involving acoustic speed as the object function, but the output was analog. The network was trained until convergence. When presented with a recall set of data different from the training set, the network generalized correctly.

6.5 The Effect of Scaling

Scaling is the usually nonlinear transformation of the input or output data to within 0 and 1. The purpose of scaling is to move some or all input-output pairs to within the range of the sigmoidal nonlinearity. By scaling in a certain manner, selective pairs can be emphasized. This technique can be useful to determine the information that the neural network correlates. Scaling is not presented in much detail in this work, but further directions are likely to pursue this issue. The previous experiment, using continuous-valued outputs and acoustic speed as the object function, was repeated using scaling. The output data were scaled evenly between 0 and 1. This kind of scaling was needed to separate closely spaced output values. If the analog outputs are too close to each other, the neural network may not be able to distinguish them. The network was able to generalize better with scaling than without, confirming the theory.

6.6 **Network Morphology**

6.6.1 **Methods of morphological determination**

Morphological issues include the number of hidden layers and the number of processing elements in each hidden layer for optimal performance of the neural network. Cybenko [1989] proved that one hidden layer is theoretically sufficient to map any function, but more hidden layers may be more efficient in practice. However, the size of the hidden layer still needs to be discussed. If the hidden layer has too few elements, the network will not be able to learn the function. If the hidden layer has too many elements, the weight space becomes more complicated, and the possibility of the network converging into a local minimum increases. The training time increases when extra hidden layer elements are added. The theoretical approach of Chapter 4 gives an indication as to the network morphology, but the situation needs to be examined experimentally.

Several approaches have been suggested to discover the optimal hidden layer size. Sietsma and Dow [1988] offer a method called pruning. An estimate is made as to a hidden layer size that is larger than necessary for learning to occur. The output of each hidden layer is examined after presenting each input-output pair. The elements with outputs that do not change and the elements that function identically to another element are not considered to contribute to the solution. Then these extra units are deleted or pruned one by one. Then the weights are fine-tuned by presenting the input-output pairs several more times. This method offers good results. Unfortunately, pruning is not practical with the tomography problem. For example, if the initial hidden layer contains 10 elements and the training set contains 100 input-output pairs, then the user needs to compare 100 outputs from these 10 elements. This task is very difficult. Also, this method still requires guesswork when considering the initial hidden layer size.

A different approach called dynamic node creation is presented by Ash [1989]. This method starts out with a hidden layer that is too small for learning. When the output error is unchanged over many presentations of training data, the network is not able to converge to a solution. Then another hidden layer element is added. This process is repeated until the network converges. One issue with this method is the training time compared to the guesswork and to starting with an overly large hidden layer. Ash presents some experimental results suggesting that node creation is computationally efficient. However, this method creates many new issues. For example, the number of presentations needed before another element is added and the error criterion needs to be examined. This method is feasible with the tomography problem, but implementation of this algorithm is difficult.

A different method called weight analysis was created in the course of this work. This technique graphically displays the dynamics of the connection weights. Figure 6.3 shows the typical weight analysis diagram. Each bar, representing a connection weight, is made up of a series of thin bars that represent that the value of the weight after a certain number of training presentations. If the end of the bar is flat, then the weight did not change (element 8 in Figure 6.3). This constant weight signifies that either the connection is not contributing to the solution or that the initial, random weight was close to the correct value.

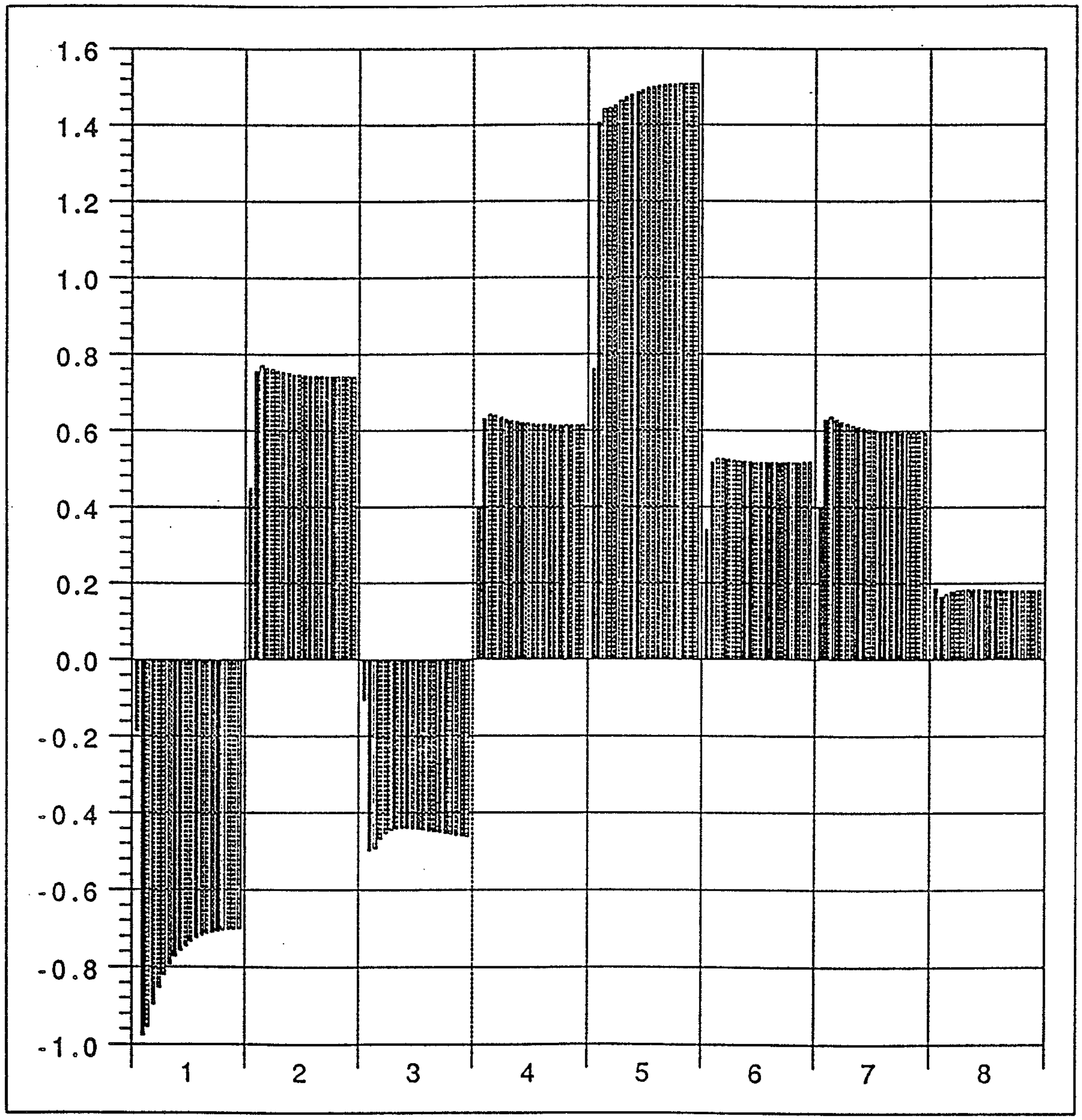

Figure 6.3. Weight analysis diagram.

In order to determine the significance of the unchanging weight, that weight should be randomized to a different value. After more training, if the weight still remains constant, then that connection can be deleted. This method offers a morphology that is more fine-tuned than the other methods, because the deletions are made connection by connection, instead of removing an entire element at one time. Of course, if all of the connections leading into an element are deleted, that element can be deleted. A negative aspect of this method is that the initial size of the hidden layer is still determined by guesswork. This analysis is an interesting direction to be pursued in the future.

Ideally, a mathematical analysis can be developed, allowing the prediction of a good network morphology for a given problem. This method would allow an optimal network to be created directly after specifying the problem and would eliminate the time-consuming guesswork.

6.6.2 Experimental determination of morphology

The optimal hidden layer size for the ultrasound tomography problem was found in several experiments. The experimental model of tissue was a circular cylinder insonated by a 2 MHz line source. A ring of 16 transducers, with one acting as the transmitter, surrounded the cylinder. The radius of the transducer ring was 75 mm. The medium surrounding the cylinder had the acoustic properties of water.

The first experiment used a non-attenuating cylinder with a radius of 0.5625 mm. The network was trained with acoustic speeds varying from 1526 ms^{-1} to 1625 ms^{-1}. Although a neural network was not necessary for this simple problem, a network with just one hidden layer element learned with the greatest accuracy, confirming the simplicity of this problem.

The second experiment involved varying the radius from 0.454 mm to 0.754 mm, while keeping the speed constant at 1576 ms^{-1}. The neural network was necessary for this kind of problem, since the radius has a nonlinear effect on the scattered field. The networks were trained with different numbers of elements in the hidden layer, using a training set of 80 input-output pairs. The entire training set was presented 20,000 times to each network. After training, the weights were fixed, and a test set was presented. The test set consisted of 80 pairs, using random radii within the training range that the network had not seen before. The input was the real and imaginary scattered field pressure information from the transducers. The output was the radius. The neural network with 13 hidden layer elements learned with the greatest accuracy with an RMS error of 0.00829. Figure 6.4 shows a plot of the results.

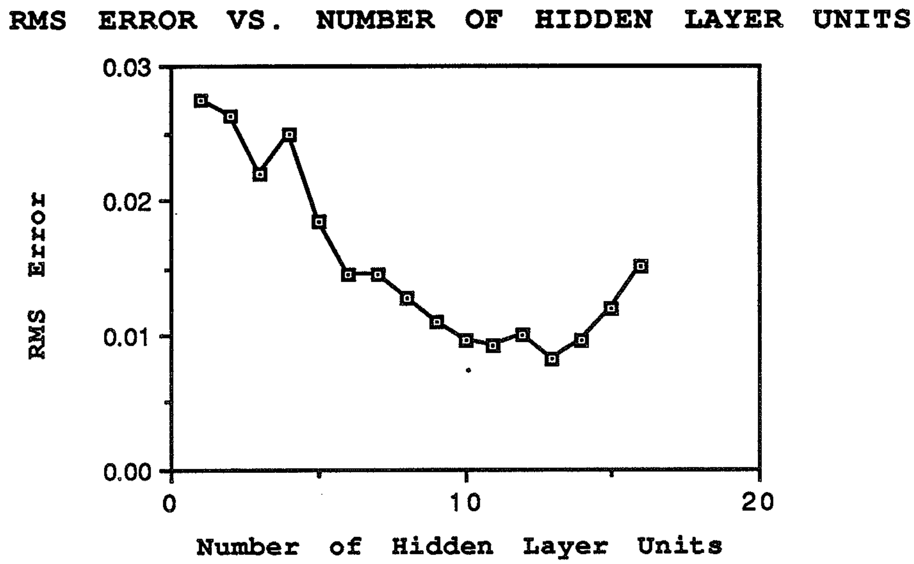

Figure 6.4. Results of experiments with radius object function.

Networks with more than 16 hidden layer elements failed to learn after 20,000 presentations, suggesting that larger networks require many more presentations to learn, or have a greater likelihood of falling into local minima in weight space. Neural networks can learn to calculate a difficult object function, such as the radius of a cylinder, from its scattered field data. The network could possibly achieve still greater accuracy by adjusting parameters, such as the learning rate, initial weight values, and scaling of the data, as well as the size of the hidden layer.

CHAPTER 7

EXPERIMENTS BASED UPON THEORETICAL DEVELOPMENT

7.1 Statement of Purpose

Chapter 4 shows theoretically that neural networks are naturally suited to perform the inversion of ultrasound tomography data. Chapter 6 shows that neural networks can indeed find solutions, using such object functions as acoustic speed and radius of the cylinder. However, an interesting process would be to confirm experimentally the theoretical conclusions of Chapter 4. This experiment involves starting with a network that performs the filtered backprojection algorithm for the x-ray case. Then this network is trained to compensate for diffraction effects introduced as the wavelength increases. Only diffraction effects will be considered in this chapter. Several issues must be addressed before performing this experiment, including the representation of the projection information and the experimental geometry at the x-ray frequency.

7.2 Representation of the Projection Information

At very high frequencies, the ultrasound case approaches the x-ray case, with acoustic attenuation replacing density. Therefore, the x-ray projection can easily be transformed to the ultrasound projection. For the x-ray case

$$I = I_0 \exp\left(-\int_{ray} \mu(x,y)\ ds\right) \tag{7.2.1}$$

where I is the received intensity, I_0 is the source intensity, and μ is the density. For the ultrasound case, Eq. (7.2.1) can be rewritten as

$$I = I_0 \exp\left(-\int_{ray} \alpha(x,y)\ ds\right) \tag{7.2.2}$$

where α is the acoustic attenuation. Since the ultrasound propagates along a straight ray at high frequencies,

$$\int_{ray} \alpha(x,y)\ ds = \alpha\ X \tag{7.2.3}$$

where X is the distance the wave travels through the object. Then evaluating the line integral gives

$$\alpha\ X = -\ln\left(\frac{I}{I_0}\right) . \tag{7.2.4}$$

In terms of the pressure field, f,

$$\alpha\ X = \frac{\left|f^{inc} + f^{sc}\right|^2}{\left|f^{inc}\right|^2} . \tag{7.2.5}$$

This information will serve as the input to the neural network conceived in Chapter 4.

7.3 **Very High Frequency Experiment**

In order to confirm the linear network conceived in Section 4.3.2, an experiment was conducted at a very high frequency. The actual frequency was not important. The relevant issue was that the ultrasound should propagate like x-rays. The transducers behind the object in relation to the source register a decrease in the intensity relative to the intensity at that location if no object was present. The transducers record the incident field magnitude. In terms of the projection, the samples are mostly zero, except behind the object where the projection is a positive value.

This experiment uses the fan beam geometry with the source at the center of the transducer ring with a radius of 75 mm (Figure 5.2). The object with a radius of 1 mm is located at the origin, causing the distance from the center of the object to the source to be 37.5 mm. In order to get fair reconstruction, 1001 transducers surround the object, but only the 17 transducers located in the shadow behind the object have non-zero projection values. Since the point of interest was the origin, only one view was needed (Section 5.6). Therefore, just 17 projection samples served as the input to the linear network. The network multiplied the filter value (Eq. (4.3.2.1)) to each of the input values. When n-k is an even number, the filter is equal to 0, eliminating some of the inputs. The remaining 9 filtered samples were then added together. Then, the resulting signal was weighted by αL^{-2}, where L was equal to 37.5 mm. The final weight modification was $2\pi M$, where M was equal to 1 for a single view.

The first experiment used a target with an attenuation of 0.4 Np mm^{-1}. The output of the network was 0.40569, an error of 1.42%. Of course, the error could be improved if more samples were used. The second test had an attenuation of 0.1 Np mm^{-1}. The network result was 0.10142, again an error of 1.42%. The error was identical in both situations since the network is linear.

The next experiment investigated the error when the location of interest was not on the axis connecting the source and the center of the object. Using an angle of 0.01883 radian from the axis and an attenuation of 0.4 Np mm^{-1}, the network gave an output of 0.37853, an error of 5.37%. This increase in error is not surprising, because more views are required for spatial resolution when the point of interest is off-axis. The point on the axis giving the exact value of 0.4 was located 36.48 mm from the source, a discretization error of -1.02 mm.

These experiments confirm the linear network developed from the theory. In combination with the experiments of Chapter 6, the successful groundwork has been laid for ultrasound tomography using neural networks.

CHAPTER 8

CONCLUSIONS

8.1 Summary

Ultrasound tomography is a safe, inexpensive diagnostic tool. Unfortunately, the inadequacies of current reconstruction algorithms hinder its effective implementation. This work summarized these current algorithms and their faults. A theoretical conception concluded that neural networks are naturally suited to perform the reconstruction. This idea was created by showing that a linear, parallel distributed processing (PDP) network can implement the filtered backprojection algorithm for the x-ray case. Then this theory was extended to the ultrasound case by realizing that x-rays and ultrasound at x-ray frequencies propagate in an analogous manner. Then a neural network was conceived by introducing nonlinear elements into the network. This neural network can then be trained to compensate for diffraction effects. The theory was extended to include refraction as well as diffraction effects by combining two neural networks in a modular fashion. Then several preliminary experiments were conducted, proving that neural networks are capable of solving tomographic reconstructions from projection information. These experiments also looked into several important areas, such as learning rates, training data generation, data representation, local minima, and morphological issues. These experiments introduced several important considerations, and provided the support needed to begin the next task: training a neural network based upon the theory. The linear network conceived theoretically was tested at very high frequencies. The success of this network at high frequencies lends credence to the success of the neural network at lower frequencies.

8.2 Future Directions

The ultimate goal of this work is to reconstruct high quality, ultrasound tomographic images using a neural network. The next stage of this research is to confirm the theory, using the linear network tested successfully in Section 7.3, lowering the frequency and training for diffraction effects. The importance of using the theoretical network is that a morphology and initial weights are known *a priori*. Starting with the weights from the x-ray analogy, instead of random weights, can decrease the training time and the possibility of convergence into a local minimum. Then the refraction effects need to accounted for by training the network conceived in Section 4.4.3. After confirming the theory, basic research is required, since the behavior of neural networks is not well understood. Several areas need to be explored such as the training, data representation, and morphological issues introduced in previous chapters.

After this work is completed, the scale of the problem should be increased to handle an entire image with inhomogeneities. A neural network to reconstruct a 100 x 100 pixel image requires 10,000 interconnected smaller networks that can input both real and imaginary pressures. Assuming that 100 samples are taken from every one of 100 projections, the input layer of each component network contains 10,000 processing elements. The entire input layer contains on the order of 10^6 elements, and the output layer contains 10,000 elements. Although this network is very large, neural network hardware, using optical or VLSI implementations, will be able to handle a problem of this magnitude.

Other advantages of neural networks include the possibility of greater accuracy than current reconstruction algorithms, because the neural network does not make any weak scattering assumptions. Neural networks are limited only by the accuracy of the training data and the effectiveness of the training, making the understanding of training and morphological issues very important. Another benefit is that neural networks can be fine-tuned, according to machine-dependent variables.

Although much research in this area is still needed, the foundation has now been built to perform accurate reconstruction using ultrasound. Neural networks can learn characteristics about a scanned object from its projection data.

REFERENCES

Amos, D.E. "Algorithm 644: A Portable Package for Bessel Functions of a Complex Argument and Nonnegative Order," ACM Trans. on Math. Software, vol. 12, no. 3, pp. 265-273, 1986.

Ash, T. "Dynamic Node Creation in Backpropagation Networks," ICS Tech. Rep. 8901, University of California, San Diego, 1989.

Carroll, S.M. and Dickinson, B.W. "Construction of Neural Nets Using the Radon Transform," Proc. IJCNN, vol. 1, pp. 607-611, 1989.

Cavicchi, T.J. "Diffraction Tomography and the Sinc Basis Moment Method," Ph.D dissertation, University of Illinois, Urbana, IL, Ch. 3, 1988.

Cybenko, G. "Approximations by Superpositions of a Sigmoidal Function," to appear in Math. of Control, Signals, and Systems, 1989.

Deans, S.R. The Radon Transform and Some of Its Applications. John Wiley & Sons, New York, pp. 1-65, 1983.

Devaney, A.J. "A Filtered Backpropagation Algorithm for Diffraction Tomography," Ultrason. Imaging, vol. 4, pp. 377-386, 1982.

Gorman, R.P. and Sejnowski, T.J. "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets," Neural Networks, vol. 1, pp. 75-89, 1988.

Greenleaf, J.F. "Computerized Tomography with Ultrasound," Proc. IEEE, vol. 71, pp. 330-337, 1983.

Kak, A.C. and Slaney, M. Principles of Computerized Tomographic Imaging. IEEE Press, New York, pp. 49-86, 1988.

Kaveh, M., Soumekh, M., and Greenleaf, J.F. "Signal Processing for Diffraction Tomography," IEEE Trans. Sonics and Ultrasonics, SU-31, no. 4, pp. 230-239, 1984.

Klimasauskas, C.C. "NeuralWorks User's Guide," NeuralWare Inc., Pittsburgh, pp. 198-202, 1988.

Lapedes, A. and Farber, R. "Nonlinear Signal Processing Neural Networks: Prediction and System Modelling," Los Alamos Tech. Rep. LA-UR-87-2662, 1987.

Lippmann, R.P. "An Introduction to Computing with Neural Nets," IEEE ASSP Magazine, pp. 4-22, 1987.

Morse, P.M. and Ingard, K.U. Theoretical Acoustics. McGraw-Hill, New York, Ch. 8, 1968.

Mueller, R.K., Kaveh, M., and Iverson, R.D. "A New Approach to Acoustic Tomography Using Diffraction Techniques," in Acoustical Imaging. Plenum Press, New York, pp. 615-628, 1980.

Pan, S.X. and Kak, A.C. "A Computational Study of Reconstruction Algorithms for Diffraction Tomography: Interpolation Versus Filtered Backpropagation," IEEE Trans. Acoustics, Speech, and Signal Processing, ASSP-31, no. 5, pp. 1262-1275, 1983.

Rumelhart, D.E., Hinton, G.E., and Williams, R.J. "Learning Internal Representations by Error Propagation," in Parallel Distributed Processing, vol. 1. MIT Press, Cambridge, MA, pp. 318-330, 1985.

Schueler, C.F., Lee, H., and Wade, G. "Funadamentals of Digital Ultrasonic Imaging," IEEE Trans. Sonics and Ultrasonics, SU-31, no. 4, pp. 195-217, 1984.

Scudder, H.J. "Introduction to Computer Aided Tomography," Proc. IEEE, vol. 66, pp. 628-637, 1978.

Shenderov, E.L. "Diffraction of a Cylindrical Wave by a Cylinder," Soviet Physics-Acoustics, vol. 7, no. 3, pp. 293-296, 1962.

Sietsma, J. and Dow, R.J.F. "Neural Network Pruning-Why and How," Proc. Second Int'l. Conference on Neural Networks ICNN-88, vol. I, pp. 325-333, 1988.

Waibel, A. "Consonant Recognition by Modular Construction of Large Phonemic Time-Delay Neural Networks," Neural Information Processing Systems, vol. 1, pp.215-223, 1989.

Winkler, J.R. and Davies, J.B. "Accurate Field Analysis of the Propagation Through an Acoustic Microscope, Part I: Theory," IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control, UFFC-33, no. 6, pp.644-656, 1986.

Winters, J.H. "Superresolution for Ultrasonic Imaging in Air Using Neural Networks," Proc. Second Int'l. Conf. on Neural Networks ICNN-1988, vol. I, pp.609-615, 1988.

Yoneyama, M., Watanabe, S., Kitagawa, H., Okamoto, T., and Morita, T. "Neural Network Recognizing 3-Dimensional Object Through Ultrasonic Scattering Waves," Proc. IEEE Ultrasonics Symposium, 1988.

Zienkiewicz, O.C. and Morgan K. _Finite Elements and Approximation_. John Wiley & Sons, New York, Chs. 1-3, 1983.