MECHANICAL CHARACTERIZATION OF TISSUE-LIKE MATERIALS USING INFORMATION BASED MACHINE LEARNING

BY

CAMERON HOERIG

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Bioengineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Advisers:

Professor Michael F. Insana
Professor Emeritus Jamshid Ghaboussi

Abstract

Changes in the mechanical properties of soft tissues may be indicative of disease processes. Medical elastography techniques are an attempt to create images of the mechanical behavior to increase the sensitivity and specificity of existing imaging modalities. Current quantitative elasticity imaging methods rely on *a priori* assumptions of the tissue biomechanics in order to simply the forward problem, from which an inverse problem is developed. Erroneous assumptions and noisy image data result in incorrect estimates of the mechanical parameters.

This thesis presents a new method of characterizing the mechanical response of soft tissues. Machine-learning techniques and measured force-displacement data are used to create empirical models of the constitutive behavior. Informational models are developed without enforcing simplyfing assumptions of the true underlying mechanics, allowing the mechanical properties of the tissue to be investigated after the model is developed. Knowledge of the true behavior allows the appropriate consitutive model to be chosen to create a parametric summary of the tissue suitable for imaging.

The informational modeling process is demonstrated on gelatin phantoms comprised of a soft background material with one or three stiffer inclusions. An ultrasound probe was used to uniaxially compress the phantoms while acquiring surface force and displacement data, as well as ultrasound images. A speckle-tracking algorithm estimated motion of the phantoms within the imaged region. Force-displacement data and the Autoprogressive training algorithm was then used to build informational models describing the constitutive behavior of the gelatin materials.

It will be shown that estimates of the full stress and strain vectors throughout an entire model can be computed with the use of informational models, a feat not previously possible in ultrasound elastography. These vectors can then be used to create a parametric summary of mechanical properties of the gelatin materials - in this case, estimates of the Young's modulus. The resuling images of the Young's modulus distribution clearly differentiate the stiff inclusion(s) from the soft background.

Results from this investigation are just the starting point for developing informational models of soft tissues. Sampling requirements and training methods to improve the ability of the models to characterize the linear-elastic properties of the gelatin are discussed. Future work will involve extending this method to 3D and characterizing more complex mechanical behaviors, including nonlinear, time-dependent, path-dependent properties.

Acknowledgments

This work would not have been possible without the support of many people. First, I would like to thank my adviser, Dr. Michael F. Insana, for his motivating rants and raves. His insight has kept me on track and focusing on the bigger picture. I must also thank Dr. Jamshid Ghaboussi for his patience while I started my foray into the world of machine learning. Without his experience and expertise, this project would not have made it this far. I am also thankful of my lab mates, Yue Wang, Minwoo Kim, Sara Bahramian, and Yang Zhu, for their input during my experiment design and data analysis. Lastly, I must thank my parents, who put up with me being a weird kid and always encouraged my curiousity. Without them, I would not be in my current position.

This work was supported in part by the National Cancer Institute (Award R01 CA168575).

Table of Contents

List of Abbreviations

ANN	Artifical Neural Network
AutoP	Autoprogressive Algorithm
B-mode	Brightness Mode
BP	Backpropagation
FE	Finite Element
FEA	Finite Element Analysis
F/T	Force/Torque
NNCM	Neural Network Constitutive Model
NR	Newton-Raphson
RF	Radio Frequency
RPROP	Resilient Propagation
US	Ultrasound

Chapter 1

Introduction

Ultrasound elastography is a non-invasive method of assessing the mechanical properties of soft tissues. Many pathologies, particularly cancer, induce a change in the local cellular structure that alters the biomechanical response of the diseased tissue [38] [6]. Similar to a physician performing manual palpation to feel for areas expressing abnormal mechanical response (usually areas of increased stiffness), elasticity imaging provides a view of the mechanical behavior.

Small lesions, or those surrounded by dense tissue, are often non-palpable. Furthermore, the change in biomechanical properties does not correlate with increased contrast in conventional B-mode ultrasound (US) imaging [35]. For example, specific characteristics in US images of infiltrating ductal carcinomas could not be found, although the majority exhibited an irregular boundary and were slightly hypoechoic [10]. Contrast between healthy and diseased tissue may be increased by instead imaging the mechanical properties. US elastography, then, can provide a radiologist with more information about imaged tissue than B-mode images alone.

Several US elastography techniques exist based on either quasi-static or dynamic application of a stimulus. Of interest in this study is the former. In the following sections, the principles of quasi-static elastography are described, followed by a discussion of inverse methods for mechanical parameter estimation.

1.1 Quasi-static Ultrasound Elastography

Quasi-static ultrasound elastography was first introduced by Ophir and colleagues [27] as a method to generate images of elastic parameters from US images. A series of RF images acquired during compression of the tissue with an US probe are used to estimate the motion of the tissue in the imaged region. Unlike dynamic methods, which often involved an internal or external stimulus to generate shear waves, quasi-static loading allows points throughout the entire object to sense and respond to the applied force. This means the motion of tissue throughout the imaged region can be tracked after applying a single stimulus.

Displacement estimation is primarily done by tracking the change in the speckle pattern resulting from the compression. One often used speckle-tracking technique is to calculate the cross-correlation of individual scan lines in the pre- and post-deformation RF echo frames. Speckle is created through coherent detection of echoes generated as an US pulse propagates through the tissue. As the tissue deforms, the movement of scatterers in a small region is governed by the same overlying mechanical behavior and is thus correlated. Small changes in the speckle field caused by scatterer motion can therefore be tracked through cross-correlation techniques.

Consider the situation where a single scatterer is embedded in an acoustically transparent medium. In the left plot of Figure 1.1, the RF echo recorded from the scatterer is illustrated by the solid black line. Once the US probe compresses the medium, the scatterer appears to move toward the face of the probe, resulting in a time-lag, Δt, of the RF echo signal (dashed line). Computing the cross-correlation between the two received signals gives an estimate of the time-lag (right plot, Figure 1.1). Using the relationship $\Delta d = \Delta t * c$, where Δd is the change in position and c is the speed of sound, the motion of the scatterer can be found in terms of distance.

Applying the cross-correlation across all scan lines produces a displacement map of the imaged region, like that displayed in the left plot of Figure 1.2. Strain is calculated from the displacement field by simply computing the gradient along the image axis (right plot,

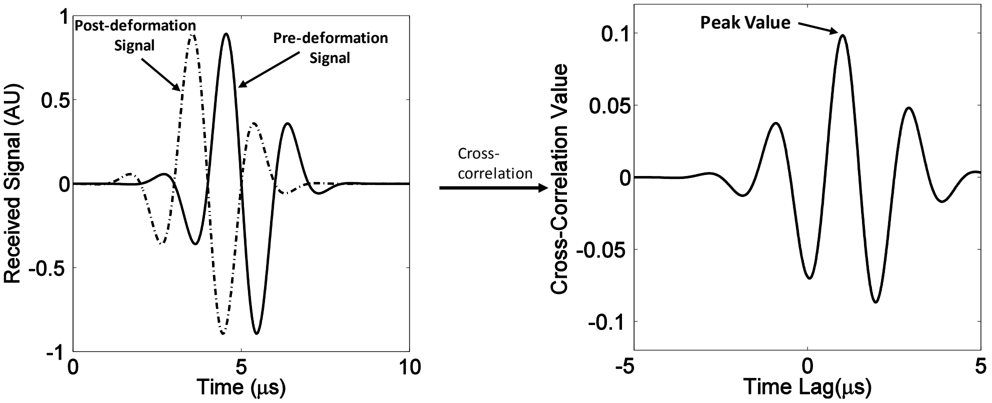


Figure 1.1: (Left) Examples of an RF line before (solid line) and after (dashed line) the tissue is compressed by the US probe. (Right) The result of the cross-correlation between the two RF lines gives an estimate to the time-lag between the two pulses.

Figure 1.2). Darker regions in the strain image indicate areas of less strain, implying the material is stiffer at that location.

The caveat to speckle-tracking methods is the necessity of small strains between images to avoid phase ambiguity and pulse decorrelation. Motions due to large strain cause are significant on the scale of the pulse length, resuling in decorrelation between the pre- and post-deformation RF echo frames. Decorrelation effects may be circumvented by acquiring many images of small deformations up to the desired strain amount. For example, the strain image in Figure 1.2 is the cumulation of 31 RF echo frames. A total of 3% strain was applied to the phantom in 30 steps, resulting in a total of 31 frames. From these, 30 strain images were computed between successive RF echo frames. Summing all of the images produced the single map of a large strain. This type of multi-image strain computation comes at the cost of increased computation time and may accumulate or compound errors - errors in each of the images get summed during the creation of the final image.

Another short-coming of strain estimation based on speckle-tracking is the limitation on accuracy of computed displacements. Motion of the material in the same direction as the propagated pulse can be accurately estimated, but displacements orthogonal to the beam are more difficult to track. This is due to the phase information contained within the US pulse along the axis of propagation. Methods have been introduced that attempt to track motion

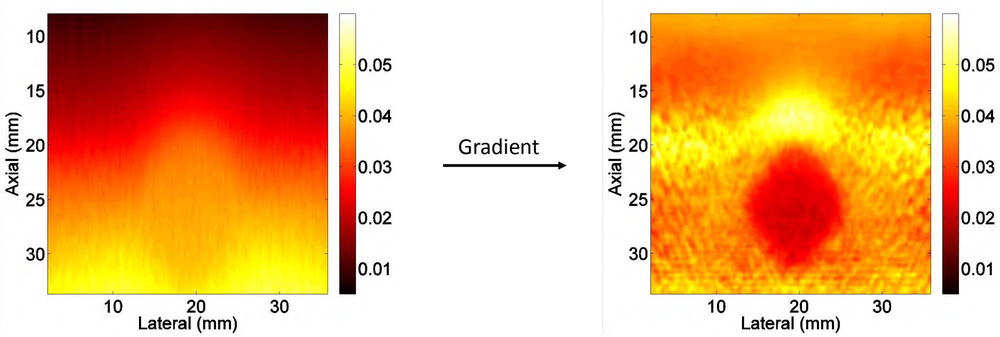


Figure 1.2: (Left) Axial displacement estimate (in mm) produced from application of a speckle-tracking algorithm on RF images. (Right) Calculating the gradient of the displacement field produces an estimate of the axial strain.

orthogonal to the pulse, including a weighted interpolation [25] and computing a synthetic lateral phase [9], but the methods have limited success and low resolution. Because speckle-tracking is relatively insensitive at estimating lateral motion, strains other than those in the direction of the pulse propagation are prone to errors and suffer from very poor spatial resolution.

Regardless of how many components of displacement and strain can be estimated, strain images only offer a qualitative view of the mechanical properties. Strain distributions highlight areas of increased or decreased stiffness, but do not provide enough information to distinguish between benign and malignant lesions at a satisfactory level of specificity [7],[36]. The goal, then, is to quantify the mechanical properties of the structures in the imaged region. Different types of soft tissues exhibit different values of shear modulus [32],[13]. Furthermore, it may be possible to differentiate benign and malignant lesions based on the estimated shear modulus [2]. By constructing an inverse problem to estimate mechanical parameters, the diagnostic capabilities of US elastography can be enhanced.

1.2 Inverse Problems in Elastography

Developing the inverse problem begins with the equilibrium equation of an elastic continuum:

$$\sum_{j=1}^{3} \frac{\partial \sigma_{ij}}{\partial x_j} + f_i = 0 \tag{1.1}$$

where σ_{ij} are the components of internal stress and f_i are the external forces. Note that this equilibrium condition is true for both linear and nonlinear elastostatics. In the nonlinear case, the internal stress is dependent on the current strain (i.e., $\sigma_{ij} = \sigma_{ij}(\varepsilon_{kl})$). Stress is related to strain through a constitutive matrix, $\mathbb{C}_{ijkl}$:

$$\sigma_{ij} = \sum_{k=1}^{3} \sum_{l=1}^{3} \mathbb{C}_{ijkl} \varepsilon_{kl} \tag{1.2}$$

Equation (1.2) also remains true in the nonlinear case where the constitutive matrix becomes dependent on the current strain (i.e., $\mathbb{C}_{ijkl} = \mathbb{C}_{ijkl}(\varepsilon_{kl})$).

Inverse methods for quantitative elastography are an attempt to determine the values of the constitutive matrix based on measured data. Up to 81 independent components exist within the matrix, making the inverse problem exceptionally difficult. To combat this, assumptions are made about the stress-strain behavior, such as linearity and isotropy, to reduce the number of independent parameters to just a few. Most often, the assumption is of a linear-elastic, isotropic solid under small (infinitesimal) strain. Some formulations assume the material is compressible ($\nu < 0.5$) while others assume incompressibility ($\nu = 0.5$). Assuming incompressibility leaves only the shear modulus, μ, to be determined; otherwise, the inverse problem includes the other Lamé parameter λ. Further approximations are often made to reduce the 3D problem to a 2D formulation. US images scan a plane through a three-dimenional object, allowing only the two normal displacement components to be estimated. Therefore, a plane-stress or plane-strain approximation is most commonly adopted.

An example of an inverse formulation for a two-dimensional problem is given here. Applying the assumption of a linear-elastic, incompressible material under plane-strain loading, a partial differential equation can be derived from (1.1) for the unknown shear modulus [33]:

$$\frac{\partial^2 \mu \varepsilon_{xy}}{\partial x^2} - \frac{\partial^2 \mu \varepsilon_{xy}}{\partial y^2} + 2\frac{\partial^2 \mu \varepsilon_{yy}}{\partial x \partial y} = 0 \tag{1.3}$$

The direct approach to shear modulus reconstruction is to solve equation (1.3) for every location in the imaged region using strain fields computed from speckle-tracking methods. Two important factors give rise to errors in the modulus estimation: 1) the inclusion of lateral and shear strains in the inverse problem and 2) the existence ofsecond-order derivatives. Recall that speckle-tracking methods poorly estimate displacements orthogonal to the axis of propagation for the US pulse. Inaccuracies in the displacements lead to inaccurate strain estimates, causing errors in the modulus estimation with (1.3). Regarding the second-order derivatives, the first problem arises due to noise. Differentiation is a poorly conditioned operation which amplifies noise, meaning noise in the strain estimates is magnified after taking the derivative. Furthermore, (1.3) implies that the shear modulus is twice differentiable, making it impossible to clearly delineate instantaneous changes in the stiffness.

Not as apparent from equation (1.3) is the need for sufficient boundary values. Solutions to the ODE with second-order derivatives requires values along the boundary to be defined. Without knowledge of the boundary values, the inverse problem is ill-conditioned and produces non-unique solutions [5] [4]. Investigations have also shown that displacement information alone is insufficient and a large amount of information related to the shear modulus must be supplied or assumed *a priori* [3] [4]. Barbone and Bamber [3] demonstrated that errors in the assumed value of μ on the boundary has significant effects on the reconstructed modulus image. They specifically show how the same set of strain data produced three different modulus reconstructions.

Other approaches to solving the inverse problem using computational techniques, referred to as iterative approaches, do not suffer from the same issues of non-uniqueness due to insufficient boundary data, such as those described in [23] and [37]. Often these methods employ finite element analysis techniques and create a computational model of the entire imaged object in 2D or 3D. After assuming a constitutive model for the object (or every distinct region in the object), the tractions and/or displacements on the surface provide enough information to solve the forward problem. Solutions to the inverse are found by performing several iterations. In each iteration, the forward problem is solved and the computed displacements are compared to the measured values. The error between computed and measured values are used to adjust the parameters of the constitutive model chosen for the material. Iterations continue until predetermined convergence criteria are satisfied.

Iterative techniques have the drawback of increased computational load due to a forward problem being solved in every iteration. Direct methods may be on the order of seconds to minutes whereas iterative methods may takes tens to hundreds of minutes. And, even though they do not suffer from the same problems with boundary conditions, solutions found with iterative methods aren't necessarily unique. It is possible that two different modulus distributions could give rise to the same measured displacements. Then, the solution found by the iterative method will depend on the initial guess of the shear modulus. Just like with the direct inversion, some information about the shear modulus must be known *a priori* in order to be sure the correct solution is found.

Inherent to the above inverse formulations is an assumption of small strain. Limiting the inverse problems to small deformation may miss important diagnostic information since biomechanical properties of healthy and diseased tissue may differ under small and large strain. For example, intraductal *in situ* carcinomas had a mechanical response similar to fat under small strains, but were much stiffer than glandular tissue under large strain [26]. Similarly, infiltrating ductal carcinomas were found to approximately 2.5 times as stiff as normal glandular tissue under compression at 1% strain, but nearly 5 times stiffer under 15% strain [38]. Work has been done to develop methods of solving the inverse problem for nonlinear elastostatics [33] and to determine the requirements for unique solutions [12].

It has been shown that assuming a linear model for material that exhibits more complex behavior, such as nonlinearity or anisotropy, leads to unpredictable errors in the modulus estimation [11].

To summarize, the mechanical properties of soft-tissues are not yet understood well enough to make assumptions of the underlying constitutive behavior required by current inverse methods. Restricting deformations to incremental displacements will likely cause important properties (e.g., strain stiffening) to be missed. A method is needed to estimate the stress and strain fields without assumptions to solve this problem.

1.3 The Innovation

Presented here is a method to build informational models of the constitutive behavior expressed by soft-tissues. Instead of enforcing an assumption of the constitutive matrix, C_{ijkl}, creating an inverse problem, and using measured data to estimate mechanical parameters, force-displacement data is acquired and used in conjunction with machine learning techniques to develop empirical models. These models obey the same mechanical laws as the interrogated tissue because the information used in their creation describes the true underlying constitutive behavior.

Perhaps the greatest advancement with the informational modeling approach is the ability to compute accurate images of the full stress and strain vectors using noisy data acquired from surface measurements and US imaging. Some existing methods, especially those which implement FEA to solve the inverse problem, may also produce estimates of the full vectors, but the estimates are based on simplifying *a priori* assumptions of the underlying constitutive behavior. Conversely, informational models are empirical and are not developed with an assumption of the biomechanical response of the tissue, thus the computed stress and strain vectors are based only on measurable data and more accurately portray the true response.

Once the informational model is developed, investigations into the underlying biomechanical properties of soft-tissues can be performed. Trained ANNs can be interrogated to determine if the tissue expresses isotropy, nonlinearity, viscoelasticity, etc. Information about the constitutive properties will guide the selection of an appropriate analytical model for mechanical parameter estimation. Unlike existing inverse methods, the behavior learned by the ANNs is exploited to determine which constitutive model best describes the tissue behavior instead of making the assumption *a priori*. Once a model is chosen, parameter estimation can be done using the computed stress and strain vectors. Moreover, informational models are not limited to a small-strain assumption and can describe the tissue motion under large strain. The ability to choose the best model for parameter estimation after the constitutive behavior is known and removing the small-strain limitation may further improve the specificity of elasticity imaging.

Chapter 2 contains a brief overview of artificial neural networks used as the machine learning method in informational modeling, followed by a derivation of the backpropagation learning algorithm. Chapter 2 continues with a concise derivation of the finite-element formulation for an elastic material to illustrate how a neural network is used in the solution. A description of the Autoprogressive algorithm concludes in the chapter. In Chapter 3 describes the informational modeling method applied to gelatin phantoms. A discussion of the important findings from Chapter 3 and future work concludes this document in Chapter 4.

Chapter 2

Neural Network Constitutive Modeling

Constitutive models describing elastic behaviors are all derived from the equilibrium condition defined in equation (1.1) and the stress-strain relationship in (1.2). Reducing the constitutive matrix by making assumptions of mechanical properties allows for a concise set of parameters to be defined. These parameters provide a simple way to describe the constitutive behavior of a material. The goal of elastography is to create images of mechanical parameter distributions to identify various tissues and structures in a medical image. Assumptions about tissue behavior are made not only to simplify the inverse problem, but to facilitate the parameterization process and provide as much information as possible in a single elastogram. If, for example, multiple parameters were estimated based on a more elaborate model, multiple elastograms would be needed to display the information, reducing the practicality of elastography.

Informational models avoid *a priori* assumptions and learn the mechanical behavior directly from measured data. Artificial neural networks (ANN), the chosen machine-learning tool, act in place of the constitutive matrix in (1.2) to relate strain inputs to stress outputs. An example of this is shown in Figure 2.1. The plane-stress approximation reduces the three-dimensional problem (with a certain geometry) to two dimensions using assumptions of linear-elasticity, isotropy, and compressibility (i.e., Poisson's ratio, $\nu < 0.5$). Mechanical parameters in the plane-stress formulation are Young's modulus, E, and Poisson's ratio. Assume that force-displacement data were acquired from a material capable of being described by the plane-stress approximation and used to train an ANN. After training, the connection weights in the ANN hold the same information as the constitutive matrix at the top of

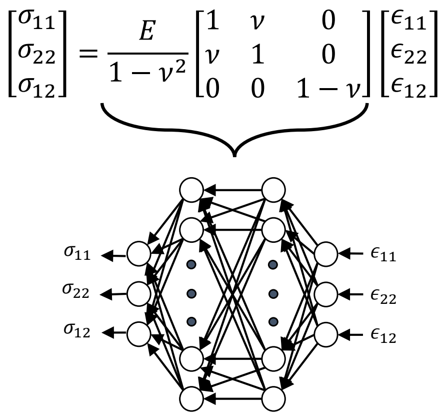

$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{12}\end{bmatrix} = \underbrace{\frac{E}{1-\nu^2}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & 1-\nu\end{bmatrix}}\begin{bmatrix}\epsilon_{11}\\ \epsilon_{22}\\ \epsilon_{12}\end{bmatrix}$
σ_{11}
σ_{22}
σ_{12}
ϵ_{11}
ϵ_{22}
ϵ_{12}

Figure 2.1: Under certain geometric constraints, linear-elastic, isotropic, compressible materials may be modeled with the plane-stress approximation shown at the top. Trained ANN material models act in place of the constitutive matrix to relate strain inputs to stress outputs. It is important to note that plane-stress behavior is not assumed during NNCM development. The relationship between the constitutive matrix and the ANN diagrammed here occurs because of the model geometry and aforementioned mechanical properties of the material.

Figure 2.1. Interrogating the trained ANN would reveal it describes the same behavior as the plane-stress approximation. It is at this point that the plane-stress model can be chosen and parameters estimated based on the stress-strain behavior characterized by the ANN in order to create a concise summary of material properties suitable for display.

Training ANNs on the constitutive behavior of a material requires a method to convert acquired force-displacement data to stresses and strains. Because no assumptions are made about the mechanical properties, an iterative process is needed to direct the ANNs to the true behavior. The method used in this approach to building neural network constitutive models (NNCM) is the Autoprogressive algorithm (AutoP). AutoP utilizes finite element analysis (FEA) to iteratively estimate stresses and strains from force-displacement data. Iteratively solving FEAs using ANNs produces increasingly accurate estimates of the true stresses and strains.

Section 2.1 provides a brief overview of ANNs and associated training methods. A concise derivation of finite element methods follows in Section 2.2 to demonstrate how ANNs can be used to solve FEAs. This chapter concludes with a more thorough description of the Autoprogressive algorithm in Section 2.3.

2.1 Artificial Neural Networks

2.1.1 Overview

Artificial neural networks are an information-based machine learning technique often used to solve complex problems. Loosely based on biological neural networks, ANNs consist of a network of "neurons", or nodes, interconnected to exchange information and compute an output based on a given input stimulus. Several characteristics inherent to ANNs make them desirable as a problem-solving technique, including their massive parallelism, ability to learn and generalize, inherent nonlinearity, adaptability, robustness, and ability to handle noisy data [15],[20]. Because of these properties, ANNs are able to produce accurate results when provided with "fuzzy" information and when modeling complex behaviors and relationships. Perhaps the most important ability of ANNs for the subject of informational modeling is their ability to model phenomena only understood through measurement or observation. From a set of noisy measurements, ANNs can generalize and learn the underlying relationships between chosen inputs and outputs.

Many different types of neural network topologies exists; for the purposes of informational modeling, our interest lies only in feed-forward networks: Figure 2.2 contains an example of a simple feed-forward, two-layer ANN. Nodes in the first layer produce outputs that feed the nodes in the second layer. Connection weights multiply the output of a node before it reaches the next node. The label w_{ji} is the weight of the connection from node i in the previous layer to node j in the next layer. Input to a node, referred to as the input signal, is computed by summing the output of all feeding nodes multiplied by the corresponding connection weights. Node output is found by supplying the input signal to an activation function defined for the node. For informational modeling, the activation function used is the hyperbolic tangent.

ANNs "learn" by adjusting the connection weights between nodes. Input-output relationships described by ANNs are the result of the chosen activation function and connection

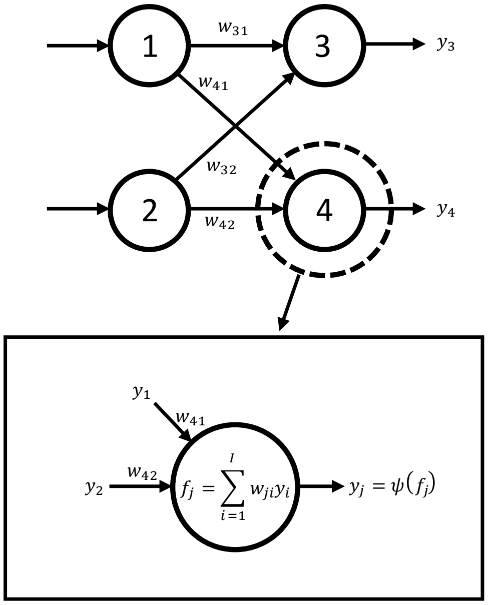


Figure 2.2: Example of a simple two-layer, feed-forward ANN. Output from nodes are weighted and summed to form the input to a node in the succeeding layer. The input signal is passed to an activation function, the output of which becomes the output of the node.

weights. The activation function(s) remain static, meaning the connection weights must change for the ANN to adjust its response. The type of learning used for building informational models is a form of supervised learning, meaning a known output is given for every training input. Many supervised learning techniques are based on the backpropagation (BP) algorithm. BP is a method used in conjunction with gradient descent optimization to adjust the connection weights based on the error between ANN output and expected response. Section 2.1.2 is the derivation of the BP algorithm to help the reader understand how an ANN can learn from given information.

Informational models - also referred to as NNCMs or ANN material models - are ANNs that have been trained on the constitutive behavior of a material. A strain vector is provided at the input of the network and a stress vector is computed. NNCMs act as the constitutive matrix between the strain and stress vectors. The empirical nature of the NNCMs means they accurately model the true mechanical behavior of a material without requiring any assumptions about the constitutive matrix.

Care must be taken when implementing developed NNCMs in computational models since trained ANNs produce valid outputs only over the range covered by the training data. They have the ability to generalize information and make accurate predictions within the bounds of training, but are unable to extrapolate. If presented with an input outside of their valid range, the response becomes nonlinear and unpredictable. Therefore, a data set must be provided for training that is comprehensive and spans the entire space of information that may be entered during later computations. In the case of NNCMs, the force-displacement data must cover the stress-strain range that may be seen in later FEAs. For example, attempting to apply a 5% strain to a NNCM that was trained up to 3% strain will produce inaccurate results.

2.1.2 Derivation of Backpropagation Algorithm

The purpose of this section is to illustrate how learning occurs in an ANN using BP. By deriving the weight update formula, it should become clear how the error at the output of the network trickles backward, or backpropagates, through the network and adjusts the nodal connection weights. The weight update formula will first be derived for a node at the output layer of the network. Then, it will be shown how the errors at the output can be used in a recursive fashion to define a weight update equation for a hidden node. The derivation shown here is based on [31] and [18].

An example of an ANN is provided in Figure 2.3. For the first part of the derivation, the hidden nodes that feed into the output nodes are indexed by i whereas output nodes are indexed by j. The connection weight from node i to j is denoted as w_{ji}. A training pair consists of an input vector and its corresponding output vector. The n^{th} training iteration consists of presenting the input vector for a training pair to the input nodes that propagates forward through the network and produces a computed output $y(n)$ at the output nodes.

Suppose the input vector for the training pair in the n^{th} training iteration has been presented to the input nodes of the ANN. The signal to output node j is the sum over

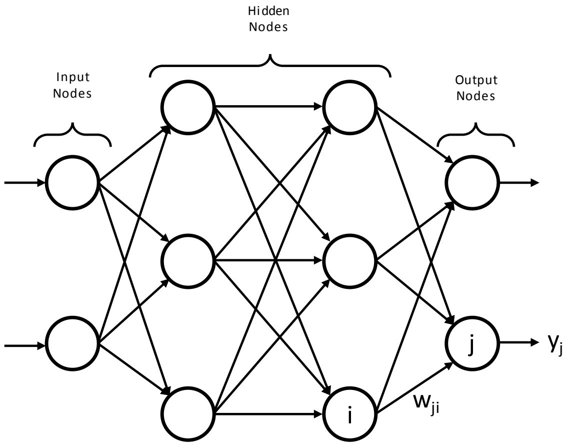


Figure 2.3: Example of an ANN with two hidden layers. The connection from node i to node j is weighted by w_{ji}.

the output of all I nodes in the previous layer multiplied by the corresponding connection weights:

$$f_j(n) = \sum_{i=0}^{I} w_{ji}(n) y_i(n) \tag{2.1}$$

After calculating the input signal, $f_j(n)$, the output of node j is found by passing the signal into the activation function $\psi_j()$:

$$y_j(n) = \psi_j(f_j(n)) \tag{2.2}$$

The error of output node j is the difference between the computed output and the given output vector of the training pair:

$$e_j(n) = d_j(n) - y_j(n) \tag{2.3}$$

where $d_j(n)$ is the expected output of node j (given by the training pair) for the n^{th} iteration and $y_j(n)$ is the actual output for node j. Error energy for node j can be defined as $\frac{1}{2}e_j^2(n)$.

With this, the total error energy is found by summing over all J output nodes:

$$\mathcal{E}(n) = \frac{1}{2}\sum_{j=1}^{J} e_j^2(n) \tag{2.4}$$

The back-propagation algorithm is used in conjunction with gradient descent methods. Connection weights are updated by a value Δw_{ji} that is proportional to the partial derivative of the error energy to connection weight w_{ji}. Applying the chain rule, the derivative can be written as:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial f_j(n)} \frac{\partial f_j(n)}{\partial w_{ji}(n)} \tag{2.5}$$

Each term on the right-hand side of (2.5) will be evaluated separately to make the solution process more clear. Solving for the first term:

$$\begin{aligned}\frac{\partial \mathcal{E}(n)}{\partial e_j(n)} &= \frac{\partial}{\partial e_j(n)} \frac{1}{2}\sum_{k=1}^{K} e_k^2(n) \\ &= \frac{1}{2}\sum_{k=1}^{K} \frac{\partial e_k^2(n)}{\partial e_j(n)} \\ &= \delta_{jk} e_k(n) \\ &= e_j(n) \end{aligned} \tag{2.6}$$

Differentiating the second term on the RHS of (2.5):

$$\begin{aligned}\frac{\partial e_j(n)}{\partial y_j(n)} &= \frac{\partial}{\partial y_j(n)}(d_j(n) - y_j(n)) \\ &= -1 \end{aligned} \tag{2.7}$$

Evaluating the third term:

$$\begin{aligned}\frac{\partial y_j(n)}{\partial f_j(n)} &= \frac{\partial}{\partial f_j(n)}\psi_j(f_j(n)) \\ &= \psi_j'(f_j(n)) \qquad (2.8)\end{aligned}$$

And finally, the fourth term evaluates to

$$\begin{aligned}\frac{\partial f_j(n)}{\partial w_{ji}(n)} &= \frac{\partial}{\partial w_{ji}(n)}\sum_{k=0}^{K} w_{jk}(n)y_k(n) \\ &= \sum_{k=0}^{K}\frac{\partial}{\partial w_{ji}(n)}w_{jk}(n)y_k(n) \\ &= \delta_{ik}y_k(n) \\ &= y_i(n) \qquad (2.9)\end{aligned}$$

Plugging (2.6), (2.7), (2.8), and (2.9) back into (2.5):

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = -e_j(n)\psi'(f_j(n))y_i(n) \qquad (2.10)$$

The correction factor $\Delta w_{ji}(n)$ is defined as

$$\begin{aligned}\Delta w_{ji}(n) &= -\eta\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} \\ &= \eta e_j(n)\psi'(f_j(n))y_i(n) \qquad (2.11)\end{aligned}$$

where η is a parameter that defines the learning rate. That is, increasing the value of η increases the amount that weight w_{ji} can change during an update step. Defining the local gradient as

$$\delta_j(n) = -\frac{\partial \mathcal{E}(n)}{\partial f_j(n)} = e_j(n)\psi'(f_j(n)) \qquad (2.12)$$

(2.11) can be written in a more succint form:

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \tag{2.13}$$

This correction factor applies to nodes in the output layer only where an error function can be defined. For nodes in the hidden layer(s), the node output is not definitively known, meaning an equation such as (2.3) cannot be directly defined for that node. Instead, the error for a hidden node must be determined recursively based on the output errors of all the nodes to which the hidden node feeds.

To begin the derivation of the local gradient for a hidden node, a change of indices is performed:

$$j \to k$$
$$i \to j$$

This means in Figure 2.3, the hidden node is now indexed by j and the output node indexed by k. The local gradient for a hidden node j is defined the same way as (2.12):

$$\begin{aligned} \delta_j(n) &= -\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial f_j(n)} \\ &= -\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \psi'(f_j(n)) \end{aligned} \tag{2.14}$$

Using the error energy defined in (2.4) with index k, the differentiation becomes

$$\begin{aligned} \frac{\partial \mathcal{E}(n)}{\partial y_j(n)} &= \sum_k e_k(n) \frac{\partial e_k(n)}{\partial y_j(n)} \\ &= \sum_k e_k(n) \frac{\partial e_k(n)}{\partial f_k(n)} \frac{\partial f_k(n)}{\partial y_j(n)} \end{aligned} \tag{2.15}$$

Recalling that the error for output node k takes the form $e_k(n) = d_k(n) - y_k(n) = d_k(n) -$

$\psi_k(f_k(n))$, the derivative of the error with respect to the sum of the input signals is

$$\frac{\partial e_k(n)}{\partial f_k(n)} = -\psi_k'(f_k(n)) \tag{2.16}$$

Continuing with the derivative on the last term of (2.15):

$$\begin{aligned}\frac{\partial f_k(n)}{\partial y_j(n)} &= \frac{\partial}{\partial y_j(n)} \sum_{l=0}^{L} w_{kl} y_l(n) \\ &= \sum_{l=0}^{L} \frac{\partial}{\partial y_j(n)} w_{kl} y_l(n) \\ &= \delta_{jl} w_{kl} \\ &= w_{kj} \end{aligned} \tag{2.17}$$

The index l is used as a dummy variable to index over all L hidden nodes connected to output node k. Putting (2.16) and (2.17) back into (2.15):

$$\begin{aligned}\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} &= -\sum_k e_k(n)\psi_k'(f_k(n)) w_{kj}(n) \\ &= -\sum_k \delta_k(n) w_{kj}(n)\end{aligned} \tag{2.18}$$

where the local gradient $\delta_k(n)$ for output node k is defined the same as (2.12) using k instead of j. To get the local gradient for hidden node j, (2.18) is substituted back into (2.14):

$$\delta_j(n) = \psi_j'(f_j(n)) \sum_k \delta_k(n) w_{kj}(n) \tag{2.19}$$

Summarizing the above derivation, the local gradient for node j in layer l is

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(L)}(n)\psi'(f_j^{(L)}(n)) & \text{node j is an output node in layer L} \\ \psi_j'(f_j^{(l)}(n) \sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n) & \text{node j is a hidden node in layer l} \end{cases} \tag{2.20}$$

and the weight adjustment becomes

$$\Delta w_{ji}^{(l)}(n) = \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n) \tag{2.21}$$

where the superscripts specify which layer the node resides. Finally, the weight update equation can be concisely defined:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \Delta w_{ji}^{(l)}(n) \tag{2.22}$$

Training can be performed in either “online” or “batch” mode. During “online” training, the connection weights are updated according to (2.22) after the presention of each training pair to the ANN. On the other hand, in “batch” mode all of the training pairs are presented and the error vectors collected. After collecting all of the error vectors for the given set of training data, the ANN connection weights are then updated. A discussion of benefits and drawbacks of each training method are beyond the scope of this paper. It should be noted that the “batch” method of training is used for NNCMs developed using AutoP.

Provided here is an example of ANN training using the standard backpropagation algorithm. A simple network with a single hidden layer shown in Figure 2.4 is considered. Connections weights are initialized to unity (i.e., $w_{ji} = 1$). Input to the ANN is “1” and the expected output is “2”. Training begins by calculating the ANN output after presenting the input. First, the output of node 1 is

$$y_1(1) = \psi(1) = tanh(1) = .7616 \tag{2.23}$$

Given that the connection weights from node 1 to nodes 2 and 3 are identical, the output from the latter nodes is calculated as

$$y_2(1) = y_3(1) = tanh(w_{21} * y_1(1)) = tanh(0.7616) = 0.6420 \tag{2.24}$$

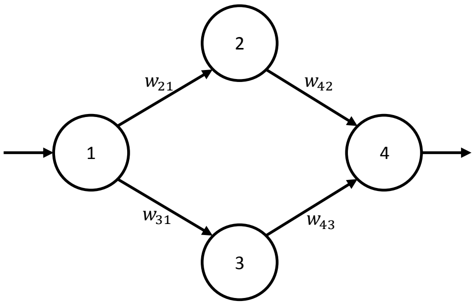


Figure 2.4: Simple ANN to demonstrate weight adjustment with the backpropagation algorithm. Each weight is initialized with a value of "1". The activation function for each node is a hyperbolic tangent.

Outputs from nodes 2 and 3 are weighted and summed to become the input signal for node 4:

$$f_4(1) = w_{42} * y_2(1) + w_{43} * y_3(1) = 1 * 0.6420 + 1 * 0.6420 = 1.2840 \tag{2.25}$$

Finally, output of the ANN is

$$y_4(1) = tanh(f_4) = tanh(1.2840) = 0.8576 \tag{2.26}$$

Output error is $e_4(1) = d(1) - y_4(1) = 2 - y_4(1) = 1.1424$. Using the definition in (2.20), the local gradient for nodes 2, 3, and 4 are as follows:

$$\begin{aligned} \delta_4(1) = e_4 * \frac{\mathrm{d}tanh(y_4(1))}{\mathrm{d}x} &= 0.3023 \\ \delta_2(1) = \delta_3(1) = \frac{\mathrm{d}tanh(y_2(1))}{\mathrm{d}x} * \delta_4(1)w_{42} &= 0.1777 \end{aligned} \tag{2.27}$$

Local gradients $\delta_2(1)$ and $\delta_3(1)$ are equivalent because both associated nodes connect to node 4, $w_{21} = w_{31}$, and $y_2(1) = y_3(1)$. For this example, the learning rate η will be unity. Using

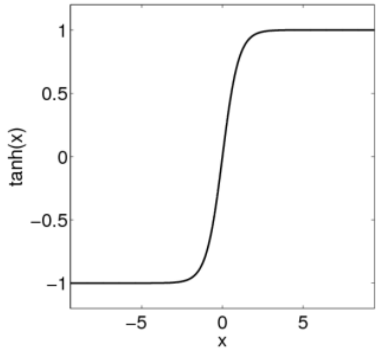


Figure 2.5: Activation function $\psi() = tanh()$ used for NNCMs.

(2.22), the weight updates become

$$\begin{aligned}\Delta w_{42} = \Delta w_{43} = \eta * \delta_4(1) * y_2(1) = 0.1941 \\ \Delta w_{21} = \Delta w_{31} = \eta * \delta_2(1) * y_1(1) = 0.1353\end{aligned} \tag{2.28}$$

After the update, the weights are $w_{21} = w_{31} = 1.1353$ and $w_{42} = w_{43} = 1.1941$. Performing another forward pass following the same process as (2.23) to (2.26), the ANN output is now $y_4(2) = .9314$. The output error after this second iteration is $e_4(2) = 0.9314$, which is smaller than $e_4(1) = 1.1424$. It is clear that the ANN has "learned" and has produced a more accurate output.

Calculation of the local gradient defined in (2.20) depends heavily on the derivative of the activation function. Based on the input signal, $y_j(n)$, the derivative of the function shown in Figure 2.5 may be very small (i.e., $y_j(n) >> 0$) or very large (i.e., $y_j(n) = 0$). Weight updates are prone to be very small or fluctuate wildly, possibly leading to the ANN to get stuck in a local minimum or overshooting minima. Many ANN training methods based on making adjustments to BP attempt to circumvent the problems associated with the dependence on the derivation of the activation function. One such algorithm, used in our method with AutoP, is the Resilient Propagation algorithm.

2.1.3 Resilient Propagation

Resilient Propagation (RPROP) [29] is an adaptive learning algorithm that acts directly on the weight update value based on the sign of the local gradient. Other adaptive techniques make adjustments to the learning rate parameter, η, that effectively increase of decrease the size of the weight adjustment. However, such learning schemes are still susceptible to the magnitude of the gradient.

To remove the effective of the magnitude of the gradient, RPROP works by defining a new weight update value, $\Delta_{ji}(n)$, based on the changing sign of the gradient defined in (2.20):

$$\Delta_{ji}(n) = \begin{cases} \alpha^{+}\Delta_{ji}(n-1) & \text{if } \frac{\partial \mathcal{E}(n-1)}{\partial w_{ji}(n-1)} * \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} > 0 \\ \alpha^{-}\Delta_{ji}(n-1) & \text{if } \frac{\partial \mathcal{E}(n-1)}{\partial w_{ji}(n-1)} * \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} < 0 \\ \Delta_{ji}(n-1) & \text{otherwise} \end{cases} \tag{2.29}$$

where $0 < \alpha^{-} < 1 < \alpha^{+}$. This learning rule states that every time the local gradient for a given connection weight changes sign, indicating that the last weight update caused a jump over a local minimum, the weight update value is decreased by a factor of α^{-}; otherwise, the weight update is increased by α^{+} to acclerate convergence in regions of a minimum. After determining the magnitude of the weight update value, the sign is determined as follows:

$$\Delta w_{ji}(n) = \begin{cases} -\Delta_{ji}(n) & \text{if } \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} > 0 \\ +\Delta_{ji}(n) & \text{if } \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} < 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.30}$$

Simply put, if the magnitude of the local gradient is positive, meaning the error is increasing, the connection weight is decreased by its update value; if the gradient is negative, indicating a decreasing error, the update value increases the connection weight, again in an attempt to accelerate convergence. The exception occurs if the gradient changes sign between training iterations:

$$\Delta w_{ji}(n) = -\Delta w_{ji}(n) \quad \text{if} \quad \frac{\partial \mathcal{E}(n-1)}{\partial w_{ji}(n-1)} * \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} < 0 \tag{2.31}$$

The change in sign indicates a jump over a local minimum. Changing the sign of the update value essentially causes a the connected weight to "backtrack" and undo the step. After determining the sign and magnitude of the weight update, calculation of the new connection weight proceeds the same as defined in (2.22).

2.1.4 Adaptive Node Creation

The information capacity of an ANN is determined by its size; that is, increasing the number of hidden nodes increases the amount of information that can be stored in the connection weights. Unfortunately, there currently exists no way to determine how large an ANN needs to be in order to solve a given problem, nor is there a good way to quantify the information capacity of an ANN.

A naive approach to implementing ANNs is to just use a large network with many hidden nodes and layers, thus ensuring there is enough capacity to hold all of the information contained within the training data. The drawback to this approach is illustrated in Figure 2.6. Suppose a set of noisy data is generated, marked by "$\circ$" in the plot, and we wish to develop an ANN that can generalize the data. In reality, the data follows a first-order polynomial function with added standard Gaussian white noise. With an ANN of the appropriate size, the linear function is correctly approximated despite the noise (solid line). Conversely, increasing the number of nodes by too much leads to an ANN that fits the noise (dashed line). The smaller ANN has successfully captured the true underlying behavior of the data whereas the larger ANN has "learned the noise".

A better way to implement ANNs may be to adaptively add nodes as the information capacity is reached. This idea is diagrammed in Figure 2.7. Training starts with an ANN with a small number of hidden nodes (Figure 2.7, left side). As training progresses, the error

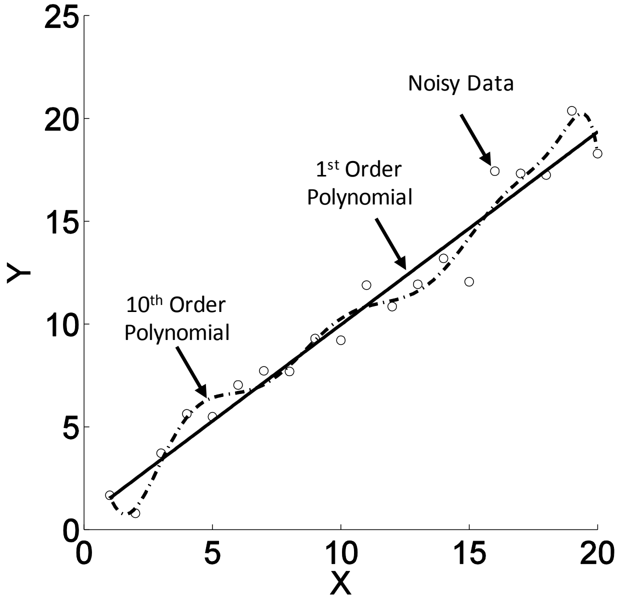


Figure 2.6: Example of an ANN overfitting to noisy data. The data points ("∘") were generated from a first order polynomial with standard Gaussian white noise added. An ANN of sufficient, but not too large of capacity, would ideally produce the solid line, which fits the underlying polynomial describing the data. The dashed line represents an ANN with too large a capacity that has started fitting the noise and thus represents a polynomial function of too high an order.

slope is monitored to determine if the information capacity of the ANN has been reached. If the error slope falls below a predefined threshold, the ANN is deemed to be saturated and cannot learn any new information (i.e., its capacity has been reached). When this occurs, one or more nodes are added to the hidden layers, increasing the capacity of the ANN (Figure 2.7, right side).

The purpose of adding new nodes is to capture information unable to learned by the ANN. Therefore, the training process is slightly altered after new nodes are added. Old connection weights are "frozen" and exempt from update for a number of training iterations. Instead, backpropagated errors work to adjust the connections of the new nodes only, thus allowing the new weights to capture information not contained within the old connection weights. After a certain number of training iterations, the old weights are unfrozen and all connection weights are able to be adjusted. With this method, an ANN grows to the size necessary to hold the information contained within the training data.

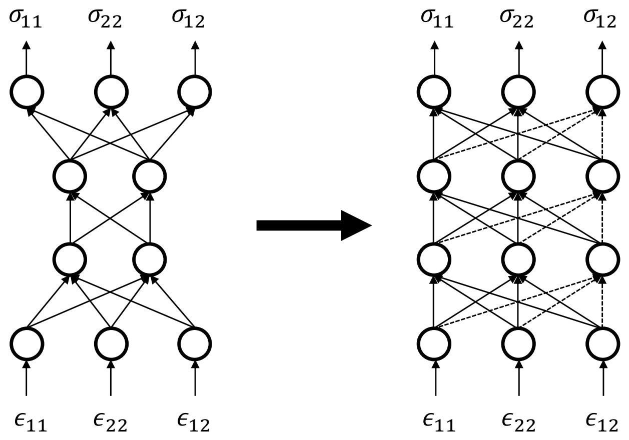


Figure 2.7: Example of adaptive node creation. During training, the error slope is monitored to determine if one or more nodes should be added to the hidden layer(s) of the ANN. If one or more nodes are added, the existing connection weights are frozen and training adjusts only the weights connecting the newly added nodes, indicated in the figure by the dashed lines.

2.2 Finite Element Formulation

A better understanding of how the ANN material models are utilized in the FEA can be obtained if the FE formulation is understood. Note that knowledge of the FE formulation is not necessary to understand AutoP. Therefore, this section is neither rigorous nor complete, it is only intended to help those interested in understanding how the ANN is used in the FEA solution. The following derivation is based on that found in [19] and [8]. Einstein notation is used throughout the derivation; e.g., $x_{ij}y_j = x_{i1}y_1 + x_{i2}y_2 + \dots x_{iN}y_N = \sum_{j=1}^{N} x_{ij}y_j$.

In Section 2.2.1, the weak form of the equilibrium equation for an elastic continuum is derived. The internal force term present in the weak form will then be used to derive the consistent tangent used in Newton-Raphson iterations. It will be shown how an ANN relating stress output to strain input can be used to solve for both of these terms. Section 2.2.2 proceeds with the FE formulation to illustrate how spatial information contained within the elements is passed indirectly to the ANN. First, process of mapping elements in global (x, y) coordinates to an isoparametric parent element in the (ξ, η) domain through the use of shape functions is explained. Then, it will be shown that displacements and strains within an element can be written in terms of nodal displacements and the shape functions. The section will conclude by presenting a method to solve the weak form of the equilibrium equation with the shape functions and Gauss quadrature.

2.2.1 Derivation of the Weak Form

The strong form of the governing equations for an elastic continuum relating internal stresses to external forces is as follows:

$$\begin{aligned} \sigma_{ij,j} + f_i &= 0 \text{ in } \Omega \\ u_i &= g_i \text{ on } \Gamma_{g_i} \\ \sigma_{ij} n_j &= h_i \text{ on } \Gamma_{h_i} \end{aligned} \tag{2.32}$$

where σ_{ij} are the internal stresses, f_i are the external forces, and Ω represents the interior domain of the region. In the second line, g_i are imposed displacement (u_i) on the boundary of the region indicated by Γ_{g_i}. The last line equates surfaces tractions, h_i to internal stresses in the direction of a boundary normal, n_j, on the boundary indicated by Γ_{h_i}. Note that the two different boundaries do not overlap and comprise the full boundary (i.e., $\Gamma_{total} = \Gamma_{g_i} + \Gamma_{h_i}$). It is assumed that the stress term is nonlinear and depends on the current value of strain (i.e., $\sigma_{ij} = \sigma_{ij}(\varepsilon_{kl})$). The consitutive can then be defined as

$$\mathbb{C}_{ijkl} = \frac{\partial \sigma_{ij}}{\partial \varepsilon_{kl}} \tag{2.33}$$

The weak, or variational, form of the equilibirum equation is obtained by premultiplying (2.32, top) by a weighting function and integrating over the domain:

$$\int_{\Omega} w_i(\sigma_{ij,j} + f_i) \mathrm{d}\Omega = 0 \tag{2.34}$$

Noting that

$$(w_i \sigma_{ij})_{,j} = w_{i,j}\sigma_{ij} + w_i \sigma_{ij,j} \tag{2.35}$$

equation 2.34 can be rewritten:

$$\int_{\Omega} (w_i \sigma_{ij,j})_{,j} \mathrm{d}\Omega - \int_{\Omega} w_{i,j}\sigma_{ij} \mathrm{d}\Omega = -\int_{\Omega} w_i f_i \mathrm{d}\Omega \tag{2.36}$$

Using the Divergence Theorem on the first term of (2.36) allows the weak form of the governing equations to be rewritten with the traction boundary conditions:

$$\int_{\Omega} w_{i,j}\sigma_{ij} \mathrm{d}\Omega = \int_{\Omega} w_i f_i + \int_{\Gamma} w_i \sigma_{ij} n_j \mathrm{d}\Gamma \tag{2.37}$$

Surface tractions enforced according the boundary conditions specified in the last line of (2.32) replace the $\sigma_{ij}n_j$ term in the last integral. The first term of (2.37) describes the internal force, which is dependent on the vector of displacements $\vec{\mathrm{d}}$:

$$n(\vec{\mathrm{d}}) = \int_{\Omega} w_{i,j}\sigma_{ij} \mathrm{d}\Omega \tag{2.38}$$

The σ_{ij} term is dependent on the current strain (which is dependent on $\vec{\mathrm{d}}$ and is governed by the ANN material model. Internal force, then, is calculated with the use of the ANN(s).

Solutions to the FEA are found using Newton-Raphson (NR) iterations to account for nonlinear behavior. NR methods require the derivation of the consistent tangent to guide the iterations to the solution. The consistent tangent is found by taking the variational derivative of the internal force term:

$$Dn(\vec{\mathrm{d}}) = \frac{\partial n(\vec{\mathrm{d}})}{\partial \vec{\mathrm{d}}} \tag{2.39}$$

Carrying this derivative into the integral in (2.38), only the stress term is affected:

$$Dn(\vec{\mathrm{d}}) = \int_\Omega w_{i,j} \frac{\partial \sigma_{ij}}{\partial \vec{\mathrm{d}}} \mathrm{d}\Omega = \int_\Omega w_{i,j} \frac{\partial \sigma_{ij}}{\partial \varepsilon_{kl}} \frac{\partial \varepsilon_{kl}}{\partial \vec{\mathrm{d}}} \mathrm{d}\Omega = \int_\Omega w_{i,j} \mathbb{C}_{ijkl} \frac{\partial \vec{\varepsilon}}{\partial \vec{\mathrm{d}}} \mathrm{d}\Omega \tag{2.40}$$

The material modulus, defined in (2.33), appears in the calculation of the consistent tangent and is able to be computed from an ANN material model. It is in the internal force and consistent tangent calculations, then, that the ANN is used in solving the FEA.

A FE formulation breaks up the domain of the problem into individual elements; i.e., $\Omega = \mathbb{A}_{k=1}^{n_{en}} \Omega_k^e$, where n_{en} is the number of elements and Ω_k^e is the domain of element k. Equation (2.37) must be solved for each element in order to generate a global solution. The details of how (2.37) is solved for each element, and how the solution process passes spatial information to the ANN, follows.

2.2.2 Solving for Forces Within an Element

Solving for the internal and external forces within each element involves evaluating (2.37). The solution process is facilitated by mapping each element onto an isoparametric parent element before evaluating the integrals. By doing so, the domain of the integration becomes identical for each element. Furthermore, the domain of the parent element can be specifically chosen to allow for Gauss quadrature, a particular type of numerical integration, to be implemented. An example of isoparametric element mapping is illustrated in Figure 2.8. Here, an arbitrary 4-node quadrilateral element is mapped onto a 4-node square element spanning $-1 \ \leq \xi \ \leq \ 1$ and $-1 \ \leq \eta \ \leq \ 1$.

For the parent element shown, a set of shape functions can be defined. The shape functions, denoted by N_i, are the same as the weighting functions, w_i, in the weak form above. They exhibit local support and Kronecker delta properties. The former means the shape functions are zero everywhere outside of the domain of the element and the latter implies the function N_i takes a value of "1" at node i and "0" at all other nodes. An example of the shape functions exhibiting these properties for a 3-node, 1D isoparametric

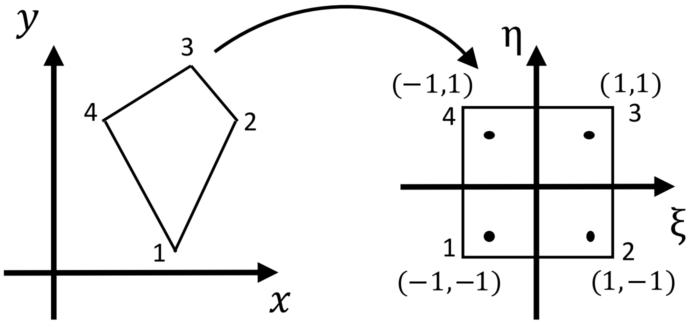


Figure 2.8: Mapping an arbitrary 4-node, quadrilateral element (left) to an isoparametric parent element (right). Using isoparametric elements allows Gauss quadrature to be more easily employed when solving the integrals for the consistent tangent, internal force, and external force vectors for an element.

element is shown in Figure 2.9.

The general form of the shape functions for a 4-node, quadrilateral element is

$$N_i(\xi,\eta) = \alpha_0 + \alpha_1\xi + \alpha_2\eta + \alpha_3\xi\eta \tag{2.41}$$

Using (2.41), local support, and the Kronecker delta property, a system of equations can be defined in order to derive the shape functions. For example, the set of equations to derive the shape function for node 1 of the parent element (Figure 2.8, right) are

$$\begin{aligned} N_1(-1,-1) &= 1 = \alpha_0 - \alpha_1 - \alpha_2 + \alpha_3 \\ N_1(1,-1) &= 0 = \alpha_0 + \alpha_1 - \alpha_2 - \alpha_3 \\ N_1(1,1) &= 0 = \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 \\ N_1(-1,1) &= 0 = \alpha_0 - \alpha_1 + \alpha_2 - \alpha_3 \end{aligned} \tag{2.42}$$

from which the form of $N_1(\xi,\eta)$ is found to be

$$N_1(\xi,\eta) = \frac{(1-\xi)(1-\eta)}{4} \tag{2.43}$$

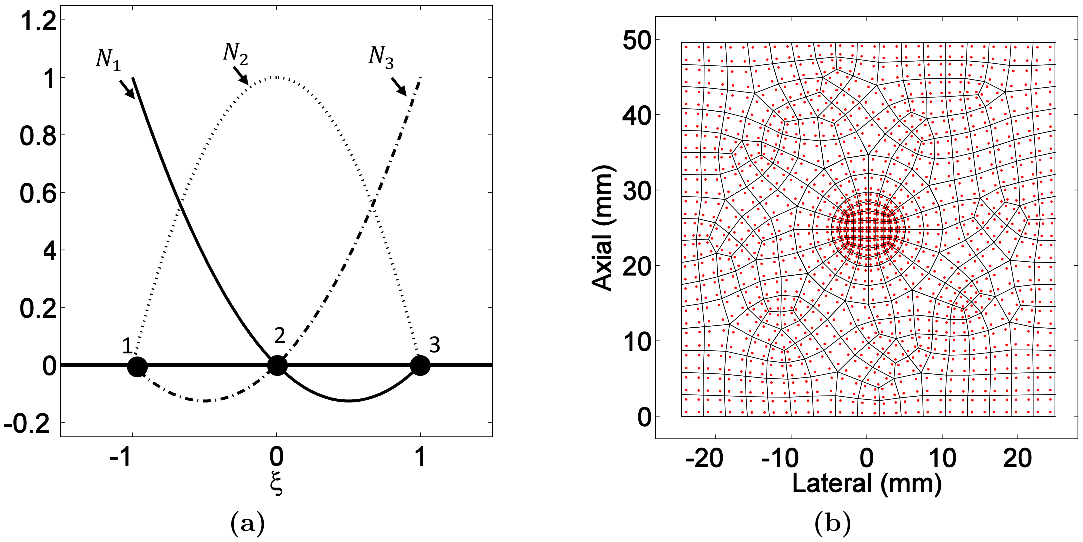


Figure 2.9: (a) Example of the shape functions for a 3-node, 1D isoparametric element. The shape function, N_i, takes a value of "1" at node i and is zero at all other nodes (Kronecker delta property). Each shape function only takes non-zero values over the domain of the element (local support). (b) Location of the integration points in a FE mesh. While displacements are calculated at the nodes, stresses and strains are computed at the integration points.

Following the same process, the shape functions for all four nodes can be derived:

$$
\begin{aligned}
N_1(\xi,\eta) &= \frac{(1-\xi)(1-\eta)}{4}\\
N_2(\xi,\eta) &= \frac{(1+\xi)(1-\eta)}{4}\\
N_3(\xi,\eta) &= \frac{(1+\xi)(1+\eta)}{4}\\
N_4(\xi,\eta) &= \frac{(1-\xi)(1+\eta)}{4}
\end{aligned}
\tag{2.44}
$$

Mapping is performed by relating the global coordinates of the nodes comprising an element to the shape functions. This is defined for a 4-node, quadrilateral element as

$$
\begin{aligned}
x^e(\xi,\eta) &= \sum_{i=1}^{4} x_i^e N_i\\
y^e(\xi,\eta) &= \sum_{i=1}^{4} y_i^e N_i
\end{aligned}
\tag{2.45}
$$

where x_i^e and y_i^e are the global coordinates of node i for element e. To complete the coordinate transformation, the Jacobian for the element is found:

$$\mathbf{J}^e = \begin{bmatrix} x^e(\xi,\eta)_{,\xi} & x^e(\xi,\eta)_{,\eta} \\ y^e(\xi,\eta)_{,\xi} & y^e(\xi,\eta)_{,\eta} \end{bmatrix} \tag{2.46}$$

and the Jacobian determinant is defined:

$$j^e = det(\mathbf{J}^e). \tag{2.47}$$

Displacements at location (ξ, η) within the domain of an element, $\vec{\mathbf{U}}^e(\xi,\eta)$, are found by interpolating the nodal displacements with the shape functions:

$$\vec{\mathbf{U}}^e(\xi,\eta) = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix} \begin{bmatrix} u^e_{\xi_1} \\ u^e_{\eta_1} \\ u^e_{\xi_2} \\ u^e_{\eta_2} \\ u^e_{\xi_3} \\ u^e_{\eta_3} \\ u^e_{\xi_4} \\ u^e_{\eta_4} \end{bmatrix} \tag{2.48}$$

where $u^e_{\xi_i}$ and $u^e_{\eta_i}$ are the displacements along the ξ- and η-coordinates of node i in element e. Element strains, $\vec{\varepsilon}^e(\xi,\eta)$, are defined as the gradient of the displacement and can therefore be written as

$$\vec{\varepsilon}^e(\xi,\eta) = \nabla\vec{\mathbf{U}}^e(\xi,\eta) = \begin{bmatrix} N_{1,\xi} & 0 & N_{2,\xi} & 0 & N_{3,\xi} & 0 & N_{4,\xi} & 0 \\ 0 & N_{1,\eta} & 0 & N_{2,\eta} & 0 & N_{3,\eta} & 0 & N_{4,\eta} \end{bmatrix} \begin{bmatrix} u^e_{\xi_1} \\ u^e_{\eta_1} \\ u^e_{\xi_2} \\ u^e_{\eta_2} \\ u^e_{\xi_3} \\ u^e_{\eta_3} \\ u^e_{\xi_4} \\ u^e_{\eta_4} \end{bmatrix} \tag{2.49}$$

Using (2.44), (2.47), (2.49), and Gauss quadrature, the weak form of the equilibrium equation in (2.37) for an element takes the form

$$\sum_{a=1}^{n_\xi}\sum_{b=1}^{n_\eta} N_{i,j}\sigma^e_{ij}(\vec{\varepsilon})j^e|_{\xi_a,\eta_b} = \sum_{a=1}^{n_\xi}\sum_{b=1}^{n_\eta} N_i f^e_i j^e|_{\xi_a,\eta_b} + \sum_{a=1}^{n_\xi}\sum_{b=1}^{n_\eta} N_i h^e_i j^e|_{\xi_a,\eta_b} \tag{2.50}$$

The limits of the sums, n_ξ and n_η, specify the number of points used for the numerical integration along the ξ- and η-axes, respectively. Integration points along each axes are denoted by ξ_a and η_b. An example of where the integration points lie in a 2D mesh comprised of 4-node, quadrilateral elements is shown in Figure 2.9b.

From this derivation for solving the weak form of the equilibrium equation for an element, it is hopefully clear how spatial information is conveyed to the ANN without having coordinates as explicit inputs. After defining an isoparametric parent element, shape functions are found that map the global (x, y) coordinates to the local (ξ, η) coordinates. Displacements throughout the element can be written in terms of the shape functions and nodal displacements. In turn, the element strains, computed as the spatial derivative of the displacements, can be defined from the derivatives of the shape functions and the nodal displacements. The element stress term, noted as σ^e_{ij} and defined by the ANN, is dependent on the value of the element strain and appears in the calculation of the element internal force. It is through this chain that spatial information propagates through to the ANN when solving the FEA.

2.3 Autoprogressive Training

2.3.1 Overview

With this understanding of ANNs, FEA, and how the former are used in the solution of the latter, the AutoP training process should become more understandable. The Autoprogessive Algorithm was developed by Ghaboussi *et al.* [14] as a method to obtain stress and strain field estimates from acquired force-displacement data. AutoP is an iterative algorithm that employs two FEAs to generate increasingly accurate estimates of the stress and strain vectors matching the dimensionality of the given problem. For example, the stress and strain vectors for a 2D model consists of three components: lateral, axial, and shear. Vector estimates computed during AutoP are used to train one or more ANN material models that describe the constitutive behavior of unique materials in the modeled object.

An overview of the AutoP algorithm is diagrammed in Figure 2.10. AutoP works by exploiting the fact that equilibrium and compatibility conditions are satisfied in the solution of a FEA. In FEA 1, measured forces are applied to a FE model of the US probe and object being imaged. Because equilibrium relates forces to stresses, the computed stress fields are more accurate estimates of the true stresses. Similarly, measured displacements are applied in FEA 2. Compatibility relates displacements to strains, resulting in computed strains that are more accurate estimates of the true strain fields. Stress and strain vectors are collected and used to update the ANN material models at the end of each iteration. A convergence check is performed by comparing displacements estimated in FEA 1 to measured displacements supplied in FEA 2. Iterations end if set convergence criteria are satisfied or if a maximum number of iterations is reached. A more in-depth discussion of the AutoP training process follows in Sections 2.3.2, 2.3.3, and 2.3.4.

Autoprogressive training of NNCMs has been used in several civil engineering applica-

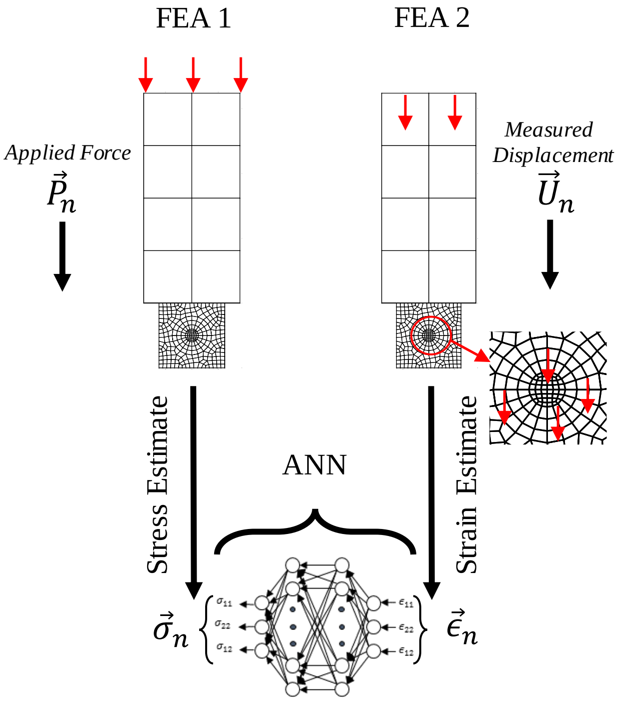


Figure 2.10: Overview of the AutoP training process. A FE model of the US probe and imaged object is created to replicate the data acquisition process. Forces applied in FEA 1 and displacements applied in FEA 2 - both indicated by the red arrows in FEA 1 and 2 - generate estimates of the stress and strain fields throughout the model. Stress-strain pairs are collected and used to train the ANN material model.

tions using only surface force-displacement measurements [39],[22],[21],[17]. Unlike previous applications of AutoP to develop NNCMs, the method being presented here uses interior displacements as well as surface measurements. The addition of ultrasound imaging allows for motion within the object to be monitored, providing far more information than surface measurements alone. With the interior displacements, it may be possible to develop NNCMs that capture the constitutive behavior of materials with many heterogeneities and complex geometries. Furthermore, information about interior motion will allow for better characterization of complex mechanical properties, including nonlinear, path-, and time-dependent behaviors. This report focuses on creating NNCMs that characterize linear-elastic of tissue-like gelatin materials.

Before AutoP training begins, quasi-static force-displacement data is acquired in a manner similar to what is shown in Figure 2.11a (Note: this diagram describes the data acquisi-

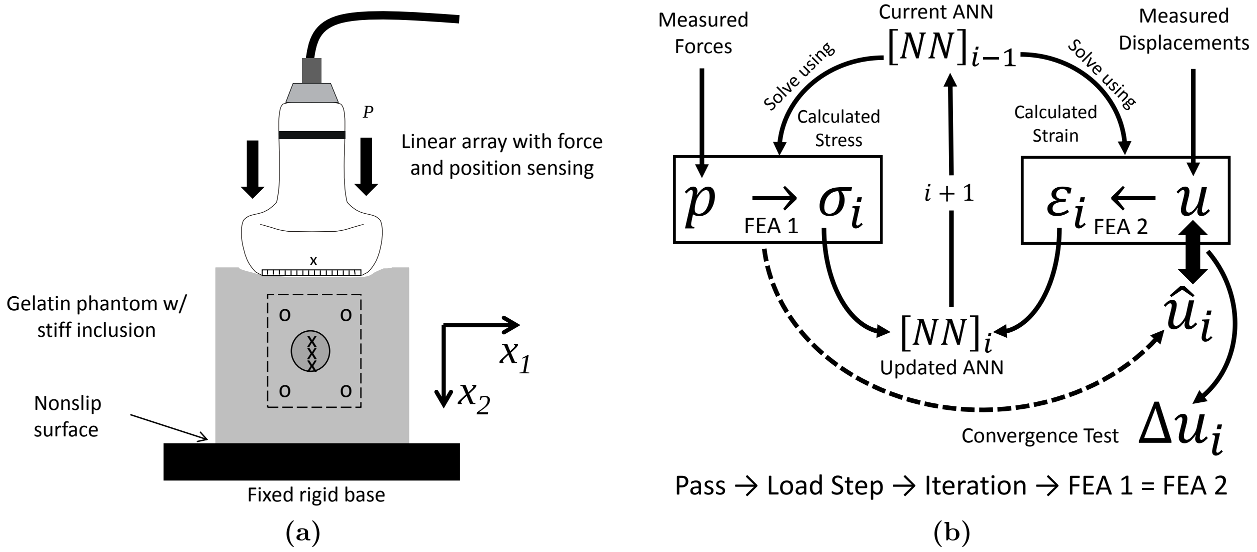


Figure 2.11: (a) Example of the data acquisition process on gelatin phantom models. A force applied by a US probe deforms the gelatin phantom and records images of the region of interest (dashed area). Application of a speckle-tracking algorithm would estimate displacements in the imaged region. A sparse set of the displacement data - locations marked "∘" and "×" - would be used in conjunction with the measured forces to develop NNCMs. (b) Flow diagram of the Autoprogressive algorithm. In the first pass, a load step is applied and measured forces and displacements are supplied to finite-element analyses FEA 1 and FEA 2. These are solved using the current version of the ANNs, $[NN]_{i-1}$, as the material models, producing estimates of the stress σ_i and strain ε_i vectors that update connection weights on the ANNs. The process iterates, and each time estimates of displacements $\hat{u}_i$ computed in FEA 1 are compared with measured displacements u to compute error vectors Δu_i. If errors are above set thresholds, the process continues to iterate using the same input data. Once errors fall below thresholds, the iteration has converged. The process is repeated for additional load steps to complete the first pass. The whole process may be repeated for the same or new data in a second pass.

tion process used in Chapter 3). An US probe fitted with a force sensor indents the surface of the object to be characterized. After each indentation step (also called a load step), forces, probe position, and an RF echo image are acquired. Measured probe positions are used to determine step-wise probe displacements while a speckle-tracking algorithm is employed to estimate displacements within the RF images. Locations in the imaged region (dashed area) in Figure 2.11a marked by "×" and "∘" indicate a possible sparse sampling of displacement information that can be used during training. More information about the sampling requirements is given in Section 2.3.5. After an adequate amount of force-displacement data is acquired, development of the ANN material models can begin.

2.3.2 Elastic Pretraining

Before starting the training iterations, the ANN material models are initialized with randomized connection weights. While it is possible to perform FEA 1 and 2 with these "random" ANN material models, the computed stress and strain fields are virtually meaningless and prolong the time to convergence of the ANNs. For example, the random connection weights may cause the ANN to compute tensile axial strains in response to a compressive axial force, a response that is impossible in reality. When such a case occurs the ANN must "unlearn" the unrealistic physical behavior in order to learn the true mechanical properties.

One way to circumvent non-physical behavior caused by randomized nodal connection weights is to perform elastic pretraining of the ANNs. For a 2D problem, the plane-stress formulation (top of Figure 2.1) of a linear-elastic model is used to generate a set of stress-strain training pairs. While the elastic behavior described by the ANN after initialization does not match the true behavior of the material to be characterized, the training process is more effective since the starting error is smaller, resulting in faster convergence to the true behavior. Note that the pretraining data is removed after the ANNs are initialized and AutoP training "overwrites" the ANNs with the underlying behavior captured by the measured force-displacement data.

2.3.3 Training Process

A FE model of the US probe and imaged object are created for use in FEA 1 and 2. Ultrasound B-mode images could be used to determine the internal geometry of the interrogated object while an external vision system captures the external shape. In the current application, though, the geometry of the object is known and modeled exactly. One ANN is used to model the constitutive behavior of every unique material in the object. In this study, two ANNs were used for each of the gelatin phantom models displayed in Figure 3.2, one to model the background and one to model the inclusion(s).

AutoP training is broken down into passes composed of load steps. During each load step, training iterations are performed until convergence is achieved or a maximum number of iterations is reached. The training process during one load step is diagrammed in Figure 2.11b.

In the first FEA, measured forces are applied to the model (FEA 1). FEA 1 is solved using the ANN(s) as the material model(s) for the target object. Because equilibrium relates forces to stresses, the computed stress fields in FEA 1 are taken to be a good approximation of the true stress field.

The second FEA is performed by applying measured displacements to the model (FEA 2). Like FEA 1, the ANN(s) are again used as the material model(s) for the target object during the solution of FEA 2. Compatibility requirements relate displacements to strains, so the computed strain fields are taken to be a good approximation of the true strain fields.

Once both FEAs are solved, stress vectors from FEA 1 are paired with their corresponding strain vectors from FEA 2. This set of stress-strain pairs comprises a training set that is used to update the connection weights of the ANN(s) using the RPROP algorithm described Section 2.1.3. A convergence test is done after the ANN update: if convergence has been achieved, or if the maximum number of iterations has been reached, the iterations exit and training continues with the force-displacement data for the next load step (if available); otherwise, the cycle continues with the same force-displacement data for the current load step.

Figure 2.12 illustrates the stress-strain response of an ANN during AutoP. Consider a 1D bar exhibiting linear-elastic behavior under uniaxial loading. For this illustration, two load steps were performed. Elastic pretraining initialized the ANN with a Young's modulus larger than the true true modulus value, indicated by the steeper slope. During training iterations in the first load step, the stress response computed in FEA 1 slowly converges to the true value for the strain applied in FEA 2. Upon completion of the training iterations, the ANN characterizes the true constitutive behavior of the bar over the stress-strain range covered by the training data. Outside of this range, the response deviates from the true behavior

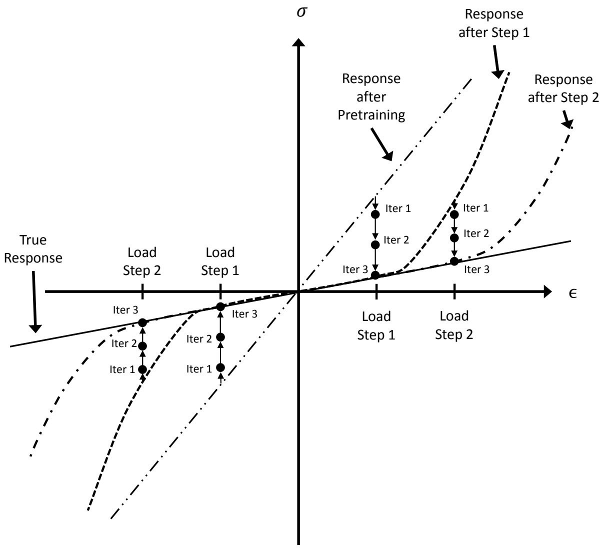


Figure 2.12: Illustration of the stress response computed by an ANN as training with AutoP progresses. The ANN was initialized using a linear-elastic constitutive model with a Young's modulus larger than the true value. During training, the stress response gets incrementally closer to the true response until convergence in achieved. Stress response outside of the training range becomes nonlinear and not representative of the true response.

and becomes nonlinear. Training during load step two extends the range over which the constitutive behavior described by the ANN is valid.

This example is for a very simple case, but the idea is the same for more complex and higher-dimensional problems. AutoP training adjusts the connection weights of the ANN(s) until the constitutive behavior described by the ANN(s) matches the true behavior of the material described by the force-displacement data. The diagram also illustrates how the response of the ANN is valid only over the stress-strain range covered by the force-displacement and deviates from the true response outside of this range.

2.3.4 Convergence Testing

A test for convergence is performed at the end of each training iteration by computing displacement errors. Displacements computed in FEA 1 ($\hat{u_i}$) are subtracted from the measured displacements applied in FEA 2 (u) to calculate a vector of displacement errors $\Delta u = u - \hat{u_i}$. Convergence is determined by comparing $\frac{max(\Delta u_i)}{max(u)}$ and $\frac{mean(\Delta u_i)}{max(u)}$ to predetermined thresholds.

An important observation about the convergence test is the number of nodes used in the displacement error calculation. Only displacements used in FEA 2 can enter into the error computation, meaning the fewer the nodes assigned displacements in FEA 2, the more lax the convergence test becomes due to fewer nodes having to satisfy convergence. The test for convergence can be constrained by either including more displacement information in FEA 2 or by comparing the displacements in FEA 1 to a different set of displacements not necessarily given in FEA 2. Consequences of the former are discussed in the next section. The latter case is explored in Chapter 3.

2.3.5 Node Selection

The selection of nodes that are supplied displacement information in FEA 2 will have a large impact on the ability of the ANN material models to learn the correct constitutive behavior of the target object. In preliminary studies where displacements were simulated and noise-free, displacement data for every node in the imaged region without causing negative effects during the informational modeling process. Displacements estimated from RF images are not noiseless, though, and care needs to be taken in order to avoid training errors.

To understand how noise in the displacements applied during FEA 2 impact training, consider the case where every node in the imaged region is assigned an axial displacement. Noise in the displacements can cause adjacent nodes to move in relation to each other in ways that are nonphysical. For example, two vertically adjacent nodes may move closer together, or pull apart, in the FEA because of the noise. Extending this to the entire model, the variance in the strain field may become very large. This “spreading” of the strain make it difficult for the ANNs to learn the true mechanical response.

This problem can be circumvented by providing displacement information to as few nodes as possible. It was found that supplying probe displacements - which are known with sub-micrometer precision - and displacements for a few nodes within each inclusion provided enough information for the informational models to learn the correct constitutive behavior.

Chapter 3

Characterization of Gelatin Phantoms

3.1 Phantom Models

Gelatin phantoms were created to mimic the linear-elastic response of soft tissues under small strain [16] using a formulation previously employed in the lab [28]. Each phantom was a 50x50x50 mm^3 cube consisting of a soft background with one or three stiffer inclusions. The stiffness of each gelatin material was controlled by varying the ratio of deionized water to gelatin: the higher the percentage of gelatin, the stiffer the phantom. Cornstarch was added to the mixture to act as a scattering field for US imaging. From a previous study, it was possible to estimate the Young's modulus of each material based on the percentage of gelatin [1]. This information was used to guide the creation process; the actual Young's modulus was estimated using an indentation technique described below.

Construction of the phantoms began by first creating the background gelatin material. While still in a liquid state, gelatin was poured into a specially made cubic acrylic mold. One face of the mold had interchangeable plates with holes at known locations. A pin with a known diameter could then be placed into the hole after pouring the background material. After the gelatin solidified, the pin(s) was removed, leaving a cylindrical hole through the length of the phantom. The stiffer gelatin material was then created and poured into the hole(s) in the background material and allowed to cool. For Phantom 3, which had three inclusions, all of the inclusions were composed of the same gelatin mixture, meaning they had identical mechanical properties.

During the creation of each gelatin material, a small amount was poured into a separate cylindrical mold to undergo macroindentation testing. Using a rigid spherical probe with a radius of 2.5mm, a force-displacement curve was generated by indenting the surface of the gelatin in a series of small steps up to 1.5mm. Space was left at the top of the cylindrical mold and filled with water before the indentation test began to remove static electric forces between the probe and the gelatin. Without the water, the surface of the gelatin may "jump" up to the probe before contact is made, resulting in a negative force reading. Adhesion between the probe and gelatin was limited with the use of a lubricant (Pol-Ease 2300 Release Agent, Polytek Development Corp., Easton, PA).

Under the assumption of no adhesion and elastic bodies, a Hertzian contact model between a sphere and a half-space was used to estimate the Young's modulus of the gelatin, as was done in [1]:

$$F = \frac{4}{3} E^{*} \sqrt{R} \sqrt{d^3} \tag{3.1}$$

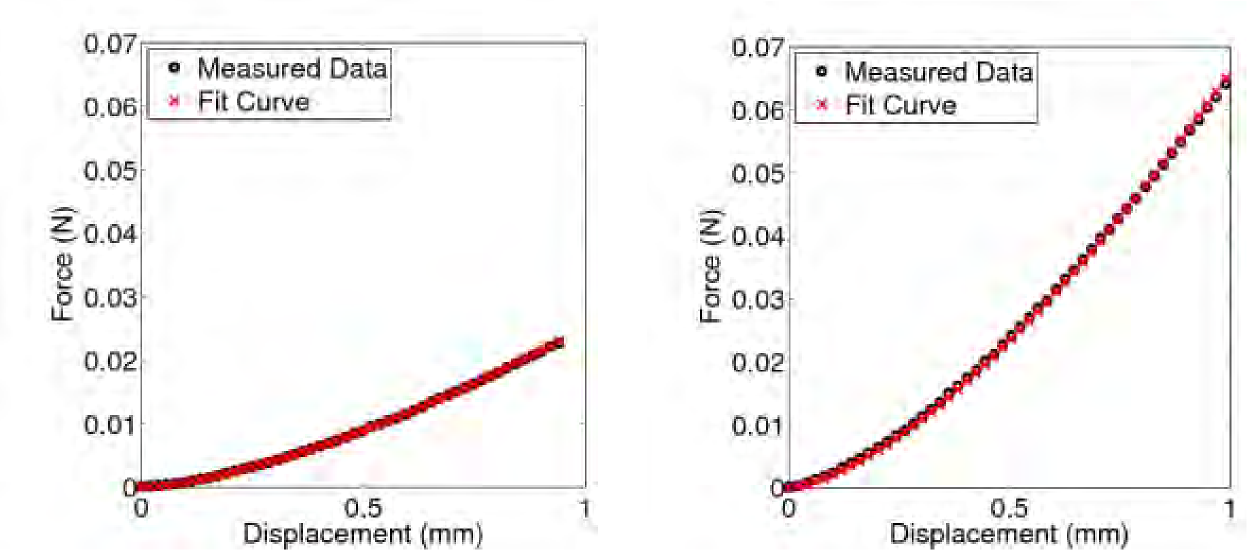
and E^* is defined as

$$\frac{1}{E^*} = \frac{1 - \nu_{sphere}^2}{E_{sphere}} + \frac{1 - \nu_{gelatin}^2}{E_{gelatin}} \tag{3.2}$$

In (3.1), R and d correspond to the radius and displacement of the indenter, respectively, and F is the measured force. E^* is a function of the Young's and Poisson's ratio of the spherical indenter (E_{sphere} and ν_{sphere}, respectively) and of the gelatin ($E_{gelatin}$ and $\nu_{gelatin}$). The large value of the Young's modulus for the spherical indenter allows the corresponding term in (3.2) to be neglected. Then, putting 3.2 into 3.1, the contact model reduces to

$$F = \frac{16}{9} E_{gelatin}^2 \sqrt{R} \sqrt{d^3} \tag{3.3}$$

An example of the force-displacement curve generated during indentation tests on the gelatin materials comprising Phantom 1 can be seen in Figure 3.1. The curve for the soft background is plotted in Figure 3.1a whereas Figure 3.1b is a plot of the curve for the



(a)

(b)

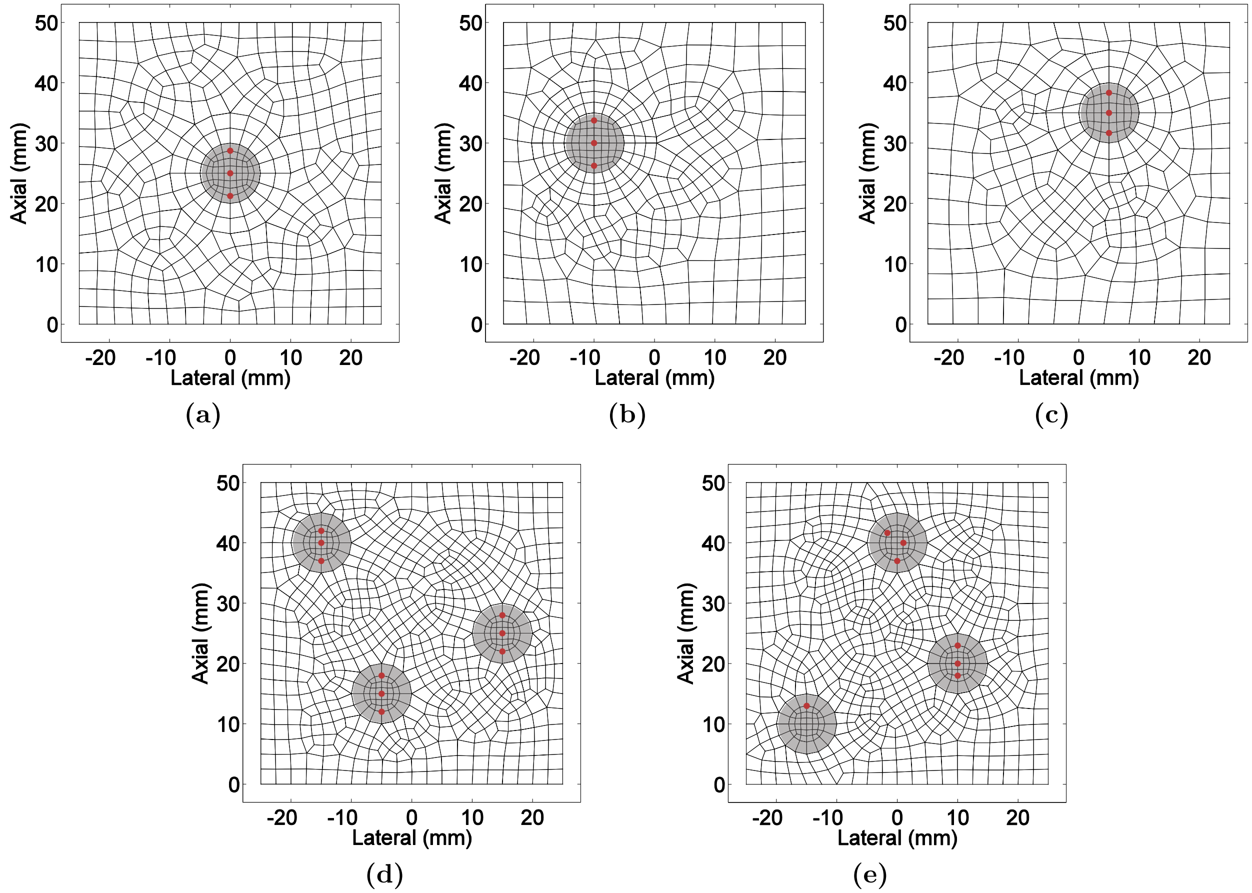
Figure 3.1: Examples of the force-displacement curve generated during a spherical indentation test. The solid black line in each figure is the measured result whereas the red dotted line is the data fit to the Hertzian contact model described in (3.3). Both measurements are from indentation tests on the gelatin materials that compose Phantom 1. (a) Spherical indentation on soft background gelatin (b) Spherical indentation on stiff inclusion gelatin

Young's Modulus Estimates		
Phantom #	Indentation Measurement (kPa)	
	Background	Inclusion
1	9.16	22.9
2	8.95	26.87
3	8.00	24.58

Table 3.1: Young's modulus values estimated from macroindentation tests. Force-displacement curves were fit to the model described in (3.3) to calculate the moduli.

stiff inclusion. Fitting each measured curve (solid black line) to the contact model in (3.3) produced a Young's modulus estimate for the material. Estimated modulus values could then be used to create estimated force-displacement curves for the indentation (dotted red line). Young's moduli estimated with this method for all three phantom models are listed in Table 3.1.

Three different gelatin phantom models were created. Phantom 1 contained a single inclusion centered on the background material. A small amount of complexity was added in Phantom 2 by moving the inclusion away from the center of the phantom, thus removing



(a) (b) (c) (d) (e)

Figure 3.2: FE meshes for each of the phantom models. (a) Phantom 1. This same mesh was used when the US probe was offset from the lateral center. The position of the FE model of the probe was adjusted. (b) Phantom 2 (c) Phantom 2, rotated 270° (d) Phantom 3 (e) Phantom 3, rotated 90°

symmetry from the imaging plane. The last phantom, Phantom 3, contained three inclusions placed such that no planes of symmetry were present in the imaging plane, adding even more complexity to the model.

The FE meshes used to model each of the phantoms are detailed in Figure 3.2. Figure 3.2a is the mesh for Phantom 1. Data was acquired on Phantoms 2 and 3 from two different sides, hence the two extra meshes in the figure. Figures 3.2b and 3.2c are the meshes corresponding to Phantom 2 being imaged from different sides, whereas Figures 3.2d and 3.2e are the meshes for Phantom 3. Highlighted regions in each of the meshes indicate the elements comprising the inclusions. An example of the full FE model used during training and forward analyses is shown in Figure 3.3b.

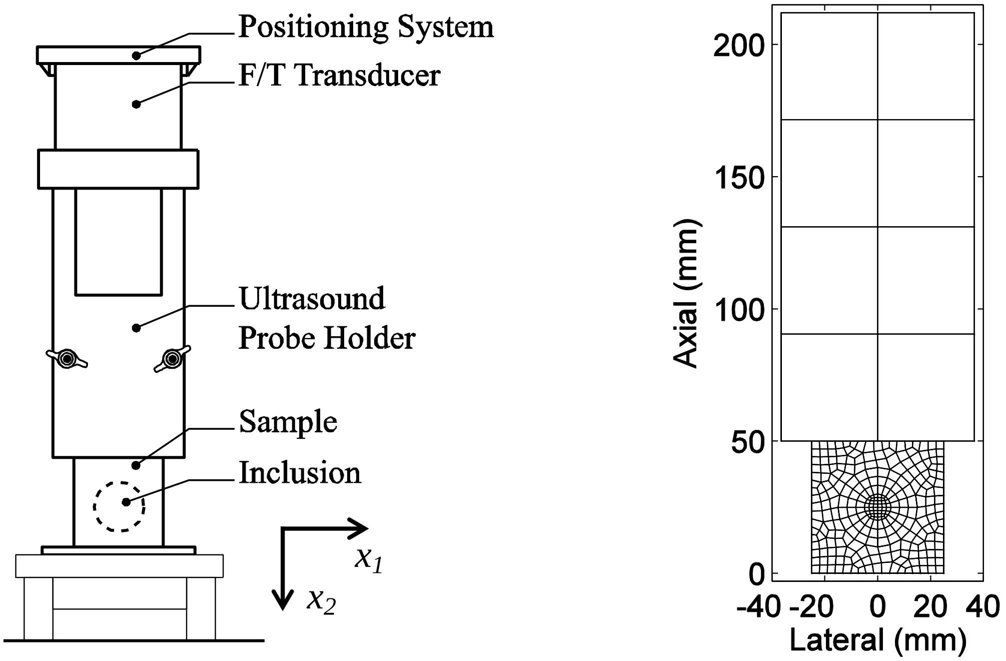
3.2 Data Acquisition

Measurements on the gelatin phantoms were performed through a series of indentations with an US probe. A plastic housing encased the US probe and was attached to a force/torque (F/T) transducer (ATI Industrial Automation, Apex, NC) capable of measuring three normal forces and torques. Position of the assembly was controlled to sub-millimeter accuracy using a motion controller. Figure 3.3a diagrams the experimental setup used in this study.

Data acquisition began by placing a gelatin phantom on a rigid base. Adhesion between the gelatin and the based fixed the bottom surface of the phantom, arresting any slippage of the phantom during compression. The US probe was positioned flush with the top surface of the phantom and centered in the lateral (x_1) and elevational (x_3) directions. US gel provided acoustic coupling between the probe and top surface of the phantom. The gel also acted as a lubricant and was modeled as frictionless contact in the FEAs.

A series of 15-30 indentations along the axial direction (x_2) displaced the US probe 1.5mm into the phantom, producing ≈3% global strain. The total strain was kept small to ensure a linear-elastic response of the gelatin and avoid effects of geometric nonlinearities. After each indentation, the US probe position, a F/T measurement, and RF image was acquired. For the applied strain, the resulting axial force ranged from 12-15 N among the measurement sets. US imaging was performed with a Siemens Sonoline Antares ultrasound system (Siemens Healthcare USA, Mountain View, CA) with a VF10-5 linear array probe at 8 MHz center frequency. Displacements within the RF images were estimated using speckle-tracking algorithm [30] after completion of the data acquisition.

Two different data sets were acquired for each phantom model. For Phantom 1, measurements were performed with the US probe laterally centered on the phantom (set 1) and with a 4mm lateral offset (set 2). Phantoms 2 and 3 were imaged from two orthogonal sides with the probe laterally centered during each data acquisition. Two additional meshes had to be created for Phantoms 2 and 3 to properly perform the FEAs, resulting in a total of five different FE models of the gelatin phantoms. All five FE meshes are displayed in Figure 3.2.



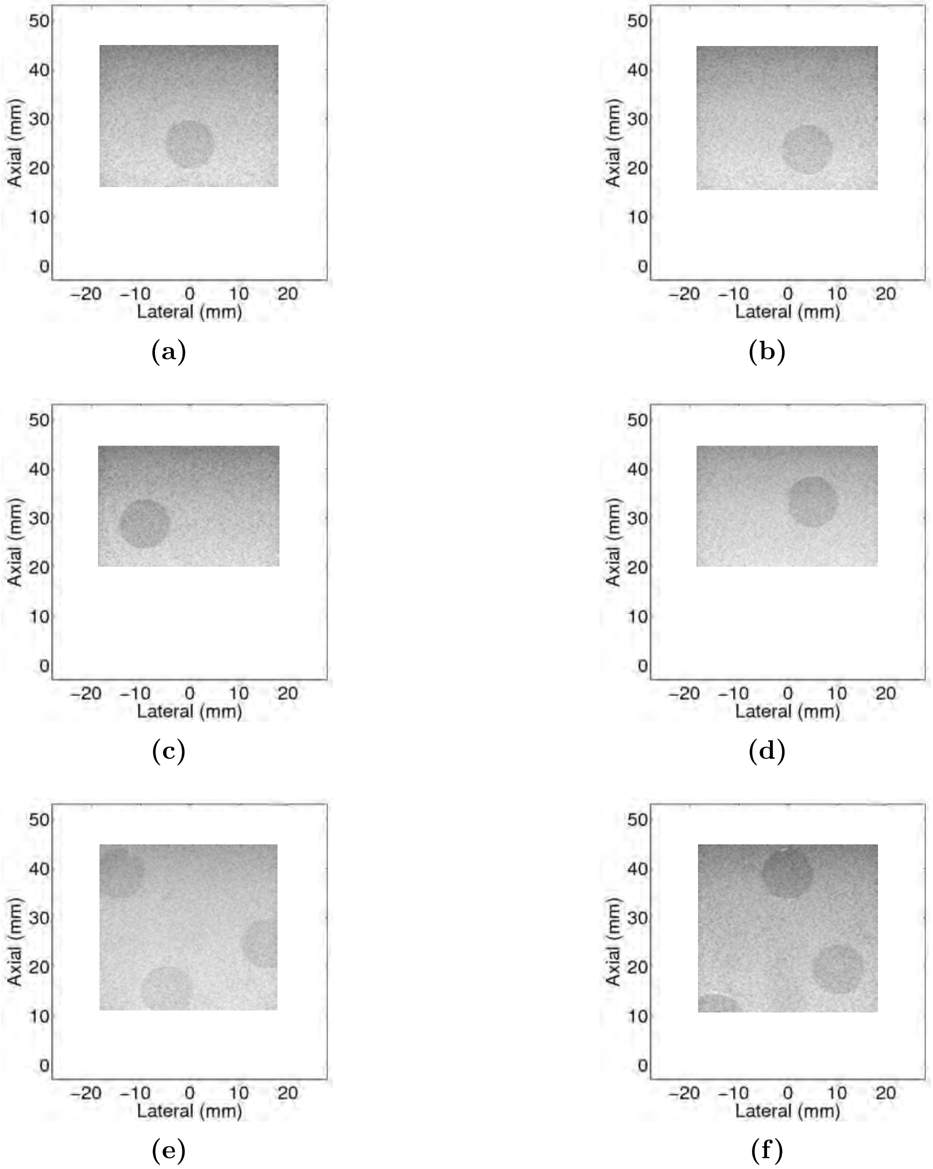
(a)

(b)

Figure 3.3: (a) Diagram of the experimental setup. An US probe was contained within a plastic housing and attached to a force/torque transducer. A 3D positioning system controlled the motion of the assembly with sub-micron accuracy. After the US probe was placed flush with the top surface of the gelatin phantom, a series of indentations were performed to produce ≈3% global strain. The probe position, a force reading, and an RF echo frame were acquired after each indentation. (b) An example of the full FE mesh used in Abaqus. Except for one set of measurements on Phantom 1, the position of the FE model of the US probe remained static while the mesh for the gelatin phantom was interchanged.

Example B-mode images for each set of measurements are displayed in Figure 3.4.

Adjusting the probe position or relative orientations of the probe and phantom changes the loading situation, producing different stresses and strains throughout the phantom. The stress and strain vectors produced under different loading conditions span different spaces, creating different information to be captured by the ANN material models. Even though the information may change, the same mechanical behavior is being described. Thus, informational models developed with different force-displacement data sets measured on the same gelatin phantom should model the same constitutive behavior, albeit over different ranges of the input stimuli.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 3.4: Example B-mode images acquired from each measurement set. Each image is set inside a box designating the boundaries of the phantom model to illustrate the size of the imaged region relative to the phantom. (a) Phantom 1 (b) Phantom 1, probe offset 4mm (c) Phantom 2 (d) Phantom 2, rotated 270° (e) Phantom 3 (f) Phantom 3, rotated 90°

3.3 Building the Informational Models

Training consisted of two passes over ten load steps. Force-displacement data used in each load step was taken from evenly spaced indexes over the range of the indentation measurements. For example, 30 indentations were performed during the measurement phase of Phantom 1, set 1. Data was then taken from steps 3, 6, ..., 27, 30 for the ten load steps in

AutoP. During FEA 2, displacements were applied to nodes along the top of the mesh modeling the probe and at the nodes highlighted in the meshes Figure 3.2. Training iterations were limited to five per load step and the convergence criteria was set to $\frac{max(\Delta u_i)}{max(u)} = 0.5$ and $\frac{mean(\Delta u_i)}{max(u)} = 0.1$ (see Section 2.3.4 for a discussion about convergence).

Two ANN material models were developed during training: one to describe the mechanical properties of the background gelatin and one to characterize the inclusion(s). During the elastic pretraining phase, the Young's modulus provided to generate the pretraining data was an incorrect value (i.e., significantly different than the value estimated from macroindentation). As an example, the background material for Phantom 1 had an estimated Young's modulus of 9.16 kPa, but a value of 4.0 kPa was given for pretraining. Initializing the ANNs with an incorrect modulus value allows for observing the rate of convergence to the "true" behavior.

Discussed in Section 2.3.5 was the effect of noise in the displacements applied in FEA 2 on the computed strain fields. While axial displacements were only applied to a few nodes in the phantom models, another step was taken to reduce the effect of strain "spreading". After collecting all of the stress-strain pairs from FEA 1 and FEA 2, the mean and standard deviation was computed for the axial strain component. Every strain vector with an axial strain outside of the mean ± 1 standard deviation range was removed along with its corresponding stress vector. This process was repeated with the lateral and shear strains.

NNCMs for the background and inclusion materials of Phantom 1 were built under four different scenarios. Recall that data was acquired with the US probe laterally centered on the inclusion (set 1) and with a 4mm offset (set 2). With these two data sets, AutoP training was done the following four ways:

case (a) Training using set 1 only

case (b) Training using set 2 only

case (c) Training using set 1 and retraining with set 2

case (d) Training using sets 1 and 2 simultaneously

Cases (a) and (b) represent the scenario where a single data set (i.e. single loading condition) is used to build informational models. Case (c) exemplifies developing ANN material models, acquiring new information, and then updating the existing ANNs with the new data. In case (d), all of the acquired information is available at the start. It is expected that the Young's modulus estimates resulting from ANN material models developed in cases (c) and (d) will be more accurate than the estimates coming from (a) and (b). This is because the information spanned by the combination of sets 1 and 2 is larger than either individual set. Therefore, the ANNs are able to learn more about the true constitutive behavior of the gelatin, thus producing more accurate modulus estimates.

A single data set was acquired for Phantoms 2 and 3 in each orientation. Training case (a) applies to these four situations. Table 3.2 lists all eight of the training scenarios for informational model building in this experiment.

The constitutive behavior of the gelatin phantoms used in this study can be described with a simple model. Because the gelatin is linear-elastic under the applied loads, and the inclusions are cylindrical and span the entire length of the phantom (i.e., they are 50mm long), the mechanical response can be modeled as plane-stress [34]. Under this assumption, the behavior described by the trained ANN material models mimics the constitutive matrix in the previously described plane-stress formula shown at the top of Figure 2.1. A comparison between the plane-stress model and a trained informational model is presented in Section 3.4.

3.4 Results

3.4.1 Images of the Stress and Strain Vectors

After completion of training with AutoP, a forward FEA was performed using the developed ANNs as the element material models. These forward analyses are given the appellation FE_{ANN} to identify them as FEAs solved with the use of ANN material models. Another

FEA was performed for each phantom using a linear-elastic material model with a Young's modulus corresponding to the macroindentation estimate. These FEAs are referred to as FE$_{Ideal}$ and represent the "true" mechanical response of the phantoms under axial loading. It is important to keep in mind that each FE$_{Ideal}$ is still an estimate of the actual constitutive behavior of the phantoms, but they do provide a basis in which to evaluate the accuracy of the informational models.

Figure 3.5 are images of the full stress and strain vectors for Phantom 1 estimated with trained ANN material models (i.e., calculated in FE$_{ANN}$). Lateral, axial, and shear stresses (σ_{11}, σ_{22}, and σ_{12}, top row) and strains (ε_{11}, ε_{22}, and ε_{12}, bottom row) are relative to the US beam direction along the vertical x_2 axis. Stress and strain values were calculated at the integration points – see Figure 2.9b for the locations of the integration points in the FE mesh for Phantom 1 – and interpolated to a uniform grid for plotting. The corresponding "true" images of stresses and strains for Phantom 1 computed in FE$_{Ideal}$ are displayed in Figure 3.6. Similar images are included for Phantom 3: Figure 3.7 are the stress and strain images estimated in FE$_{ANN}$ and Figure 3.8 were computed in FE$_{Ideal}$.

Comparing the stress and strain maps from FE$_{ANN}$ and FE$_{Ideal}$ for a given phantom model, the estimated axial and lateral fields closely match the corresponding "true" fields. Recall that during training only axial displacements were supplied in FEA 2. It is reasonable, then, that the estimated axial stresses and strains are very accurate. The accuracy in the lateral stress and strain estimates can be attributed to the incompressibility of the gelatin materials (i.e., Poisson's ration $\nu = 0.5$). During pretraining, the elastic model used to generate pretraining pairs assumed an incompressible material and force-displacement data used during AutoP did not contradict this assumption. Therefore, the accuracy in the lateral estimates is due to the accuracy in the axial fields. Shear stresses and strains, however, are not nearly as accurate. The topic of shear information is discussed later in Section 3.4.6.

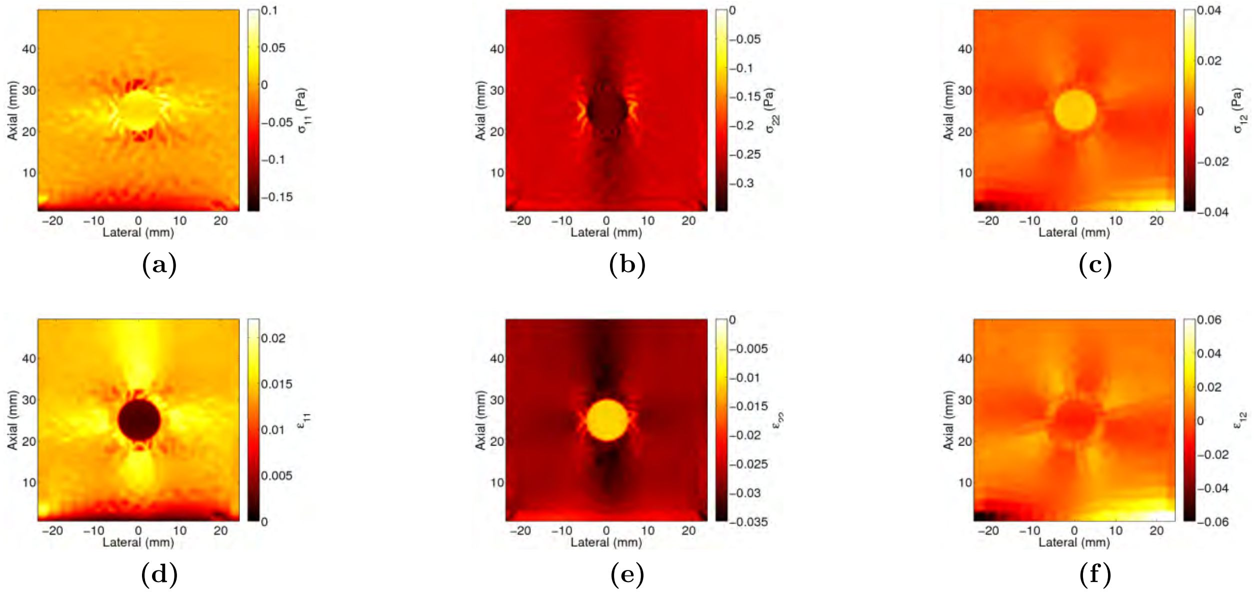


Figure 3.5: Images of the stress and strain fields computed during FE$_{ANN}$ for Phantom 1. (a) Lateral stress (b) Axial stress (c) shear stress (d) Lateral strain (e) Axial strain (f) shear strain

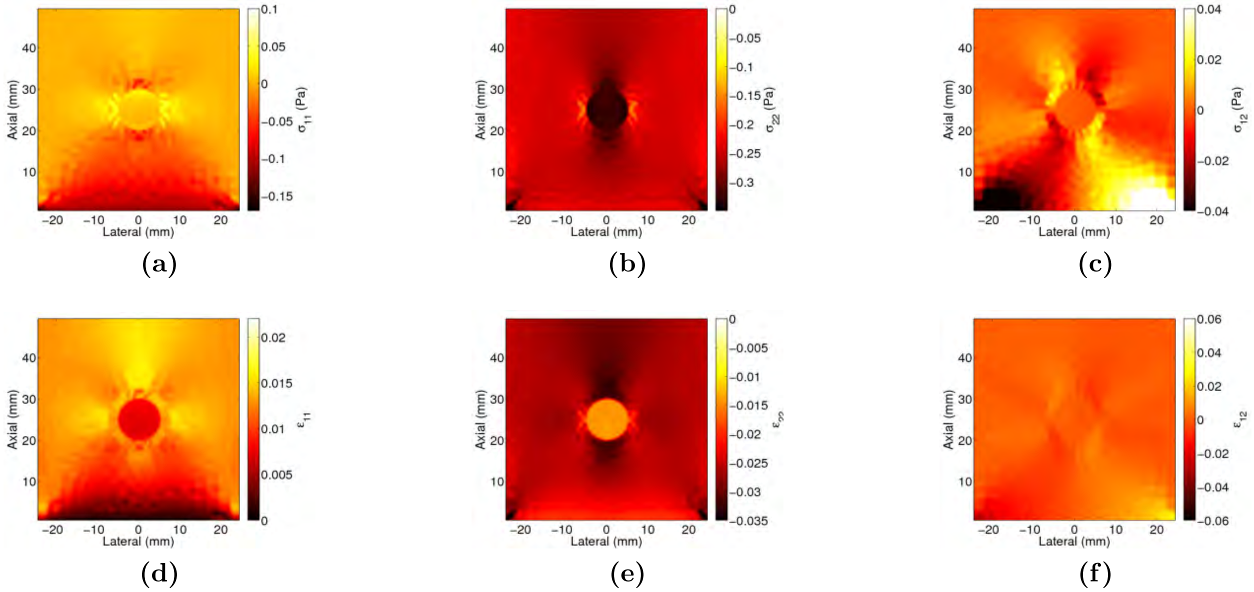
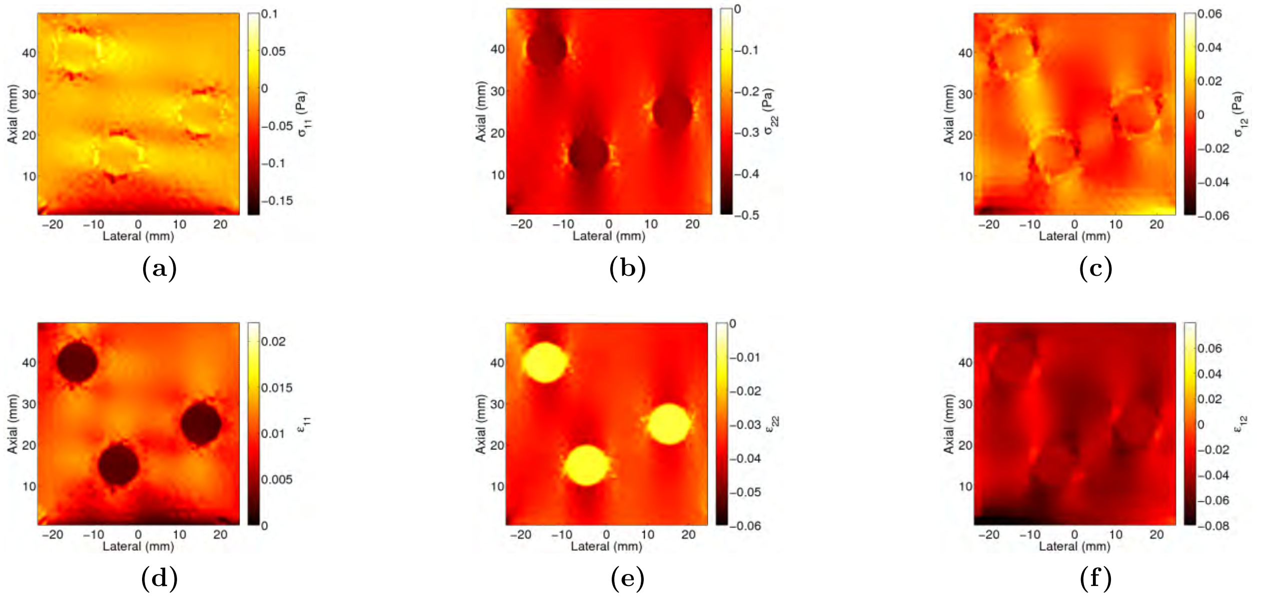


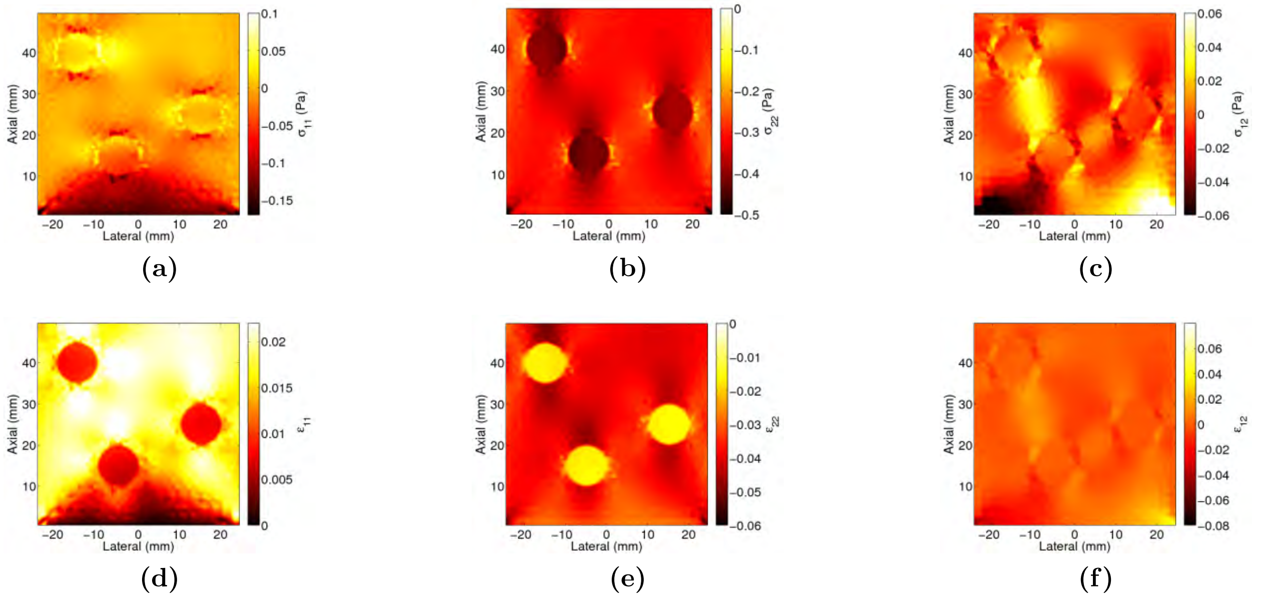
Figure 3.6: Images of the stress and strain fields computed during FE$_{Ideal}$ for Phantom 1. (a) Lateral stress (b) Axial stress (c) shear stress (d) Lateral strain (e) Axial strain (f) shear strain



(a) (b) (c)

(d) (e) (f)

Figure 3.7: Images of the stress and strain fields computed during FE$_{ANN}$ for Phantom 3. (a) Lateral stress (b) Axial stress (c) shear stress (d) Lateral strain (e) Axial strain (f) shear strain



(a) (b) (c)

(d) (e) (f)

Figure 3.8: Images of the stress and strain fields computed during FE$_{Ideal}$ for Phantom 3. (a) Lateral stress (b) Axial stress (c) shear stress (d) Lateral strain (e) Axial strain (f) shear strain

3.4.2 Young's Modulus Estimation

Mechanical parameter estimation can be easily performed with access to estimates of the full stress and strain vectors. It is at this point a constitutive relationship must be assumed. As mentioned before, the plane-stress approximation can be invoked to model the of the gelatin phantoms used in this study. A simple inverse problem can be created from the equation at the top of Figure 2.1:

$$E(x_1, x_2) = \frac{(1-\nu^2)\sigma_{22}(x_1, x_2)}{\nu\epsilon_{11}(x_1, x_2) + \epsilon_{22}(x_1, x_2)} \tag{3.4}$$

The center line of the plane-stress formula, which relates axial stresses to the strain vector, was used to derive (3.4). This was picked due to the accuracy of the axial stress and strain estimates. Using the incompressible result, a Young's modulus image was computed pointwise using the stress and strain fields estimated in FE$_{ANN}$. Values were computed at the integration points of the mesh and interpolated to a uniform grid. Young's modulus images for Phantoms 1, 2, and 3 are displayed in Figure 3.9. Modulus estimates for each gelatin material are computed as the mean of the estimated modulus values from every element governed by the corresponding NNCM. For example, a Young's modulus estimate is made at four points in each element comprising the inclusion. If there were 8 elements modeling the inclusion, there would be a total of 32 modulus estimates. The estimated Young's modulus for the inclusion would then be the mean of the 32 values.

Table 3.2 summarizes the modulus estimates for all eight training cases of the three phantom models. The column labeled "Estimated" corresponds to the modulus values computed from the ANN material models using (3.4) whereas the "Indentation Measurement" column corresponds to the values estimated from macroindentation measurements using (3.3). Standard deviations are given along with the estimated values to illustrate the variance of the computed modulus within each gelatin material.

While all of the modulus estimates closely match the expected values based on macroin-

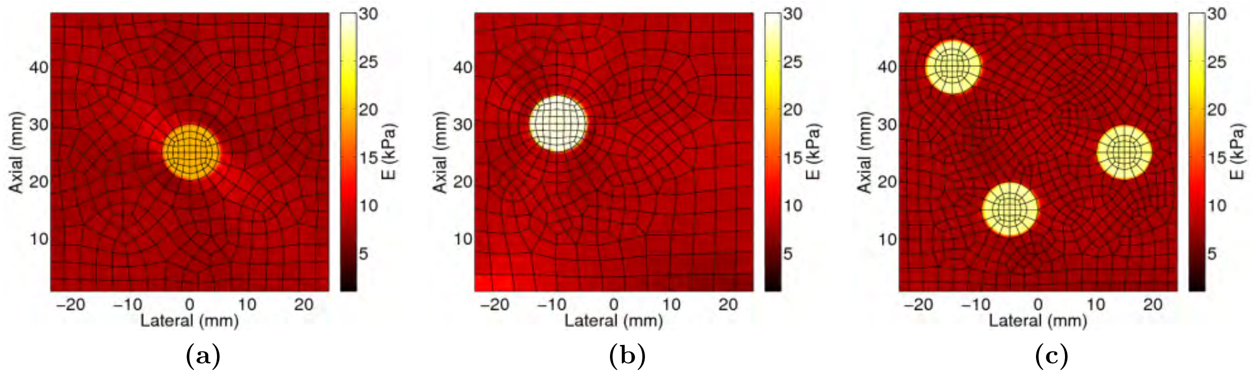


Figure 3.9: Computed Young's modulus images for the gelatin phantoms. The stress and strain fields, like those in Figure 3.5, were input into (3.4) to calculate the modulus values at the integration points. (a) Phantom 1 (b) Phantom 2 (c) Phantom 3.

dentation measurements, the most accurate estimates came from cases (c) and (d) of Phantom 1. Recall that two different sets of force-displacement data were used to train the ANN material models in these cases. The total information contained both sets of measurement data spans a larger space than either set alone, resulting in ANNs that learn more about the mechanical properties of the gelatin materials. Also to note is the increased accuracy in modulus estimates for Phantoms 2 and 3 versus cases (a) and (b) of Phantom 1. All six of these informational model building scenarios used one set of force-displacement data; however, the increased complexity in the geometry due to multiple inclusions creates more complex stress and strain fields. This can be seen by comparing the images in Figure 3.8 to those in Figure 3.6. With the increase in the amount of information contained in the more complex stress and strain fields, ANN material models are able to capture more detail about the underlying constitutive behavior.

3.4.3 Quantitative Test of Results

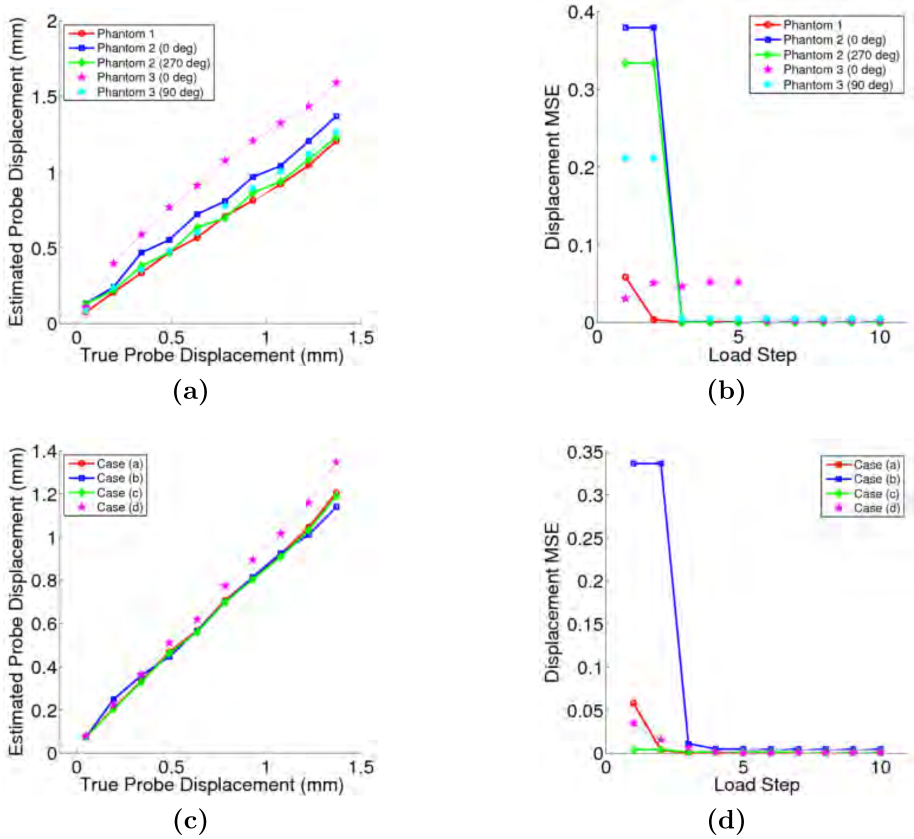
More quantitative tests of trained ANN material models can be found in Figure 3.10. During the data acquisition phase, the US probe position is known with sub-micrometer accuracy, allowing for a comparison between known and computed probe displacements after each load step. This comparison is done by noting that when performing the indentation tests,

Young's Modulus Measurements of Phantom Components				
{Phan. #} (Case) [Rot. Angle]	Estimated (kPa)		Indentation Measurement (kPa)	
	Background	Inclusion	Background	Inclusion
{1} (a) [0 deg]	8.01 ± .5599	19.35 ± .2022	9.16	22.9
{1} (b) [0 deg]	11.01 ± .1189	27.71 ± .5141	9.16	22.9
{1} (c) [0 deg]	10.18 ± .3408	22.15 ± .3711	9.16	22.9
{1} (d) [0 deg]	9.06 ± .1433	22.39 ± .3883	9.16	22.9
{2} (a) [0 deg]	8.57 ± .5596	28.84 ± .1770	8.95	26.87
{2} (a) [270 deg]	9.04 ± .0561	26.19 ± .1315	8.95	26.87
{3} (a) [0 deg]	7.99 ± .2755	26.65 ± .2105	8.00	24.58
{3} (a) [90 deg]	9.62 ± .1132	25.58 ± .3166	8.00	24.58

Table 3.2: Young's modulus values for the three phantoms studied. Estimated values were calculated from modulus images. Indentation measurements were made on samples for comparison. Cases refer to the training scenarios listed in Section 3.3. Rotation angles refer to the orientation of phantoms illustrated in Figure 3.2 with respect to the ultrasonic beam.

the US probe is displaced and the resulting force is measured; conversely, in FE$_{ANN}$, the measured force is applied to the FE model and the resulting displacement is computed. Perfect mechanical characterization by ANNs would mean the measured and computed US probe displacement values would match exactly. The results in Figure 3.10a are for the three phantoms at two scanning angles. These correspond to training case (a) of for each phantom orientation where only a single set of force-displacement data with the probe laterally centered on the phantom was used for training. A slope of "1" is the ideal result where the informational models have learned the constitutive behavior of the gelatin phantom exactly. In Figure 3.10b, the mean-squared error (MSE) between the displacements calculated in FE$_{Ideal}$ and those estimated in FE$_{ANN}$ at the same locations in the imaged region are plotted as a function of load step for the same phantoms. Notice in Figure 3.10b that after the first few load steps, AutoP training iterations have converged or reached a maximum number of iterations, suggesting the ANNs have learned as much as possible from the given force-displacement data. The large MSE suggests that the correct mechanical behavior has not yet been learned, though. After the sixth load step, however, convergence results in a displacement MSE that is less than 0.01 mm^2 for all phantoms studied.

Figures 3.10c and 3.10d contain the same information as 3.10a and 3.10b but now only



(a)

(b)

(c)

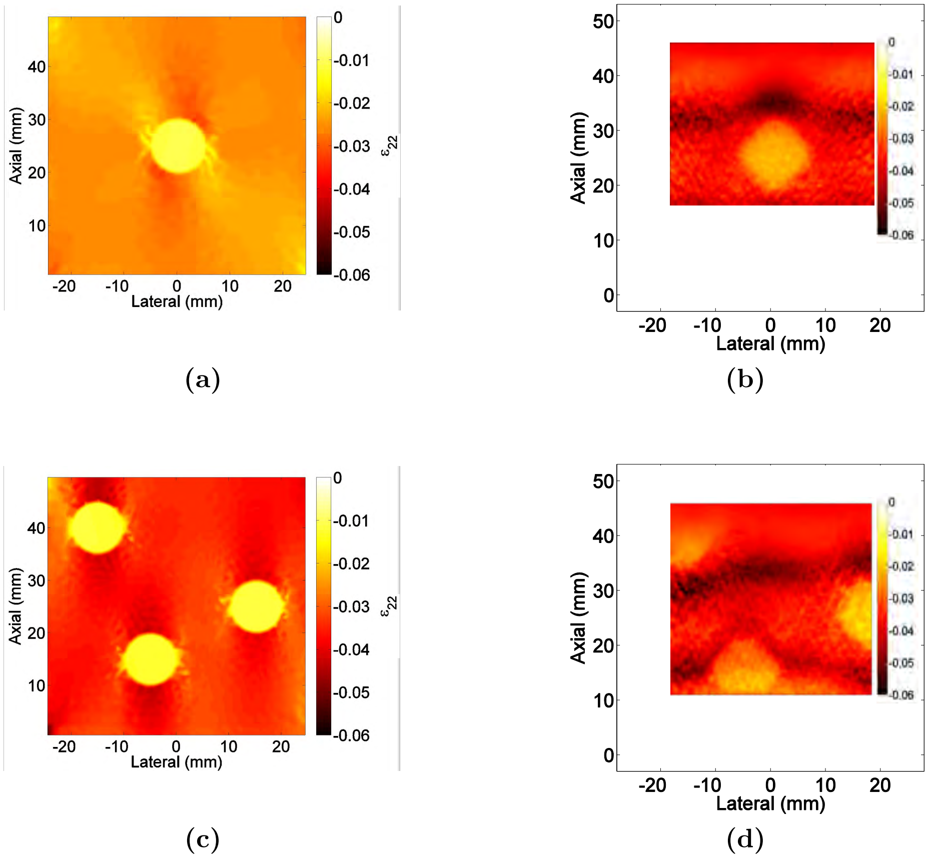
(d)

Figure 3.10: (a) Comparison of the probe displacements measured directly and predicted by trained ANNs for all phantoms (b) mean-squared displacement error for nodes in the imaged region for all of the phantom models (c) measured and estimated probe displacements for cases (a) - (d) of phantom 1 (d) mean-squared node displacement error for cases (a) - (d) of phantom 1

for Phantom 1 where the data used to train the ANNs was varied among the different curves plotted. In all cases, adding low-noise data from a different view reduced the errors.

3.4.4 Comparison of Informational Models to Current Methods

Discussed before was the more advanced abilities of this new method of mechanical characterization compared to current methods. A direct comparison is between the axial strain estimates using informational modeling and speckle-tracking is made in Figure 3.11. The left column consists of strain estimates made using our new method, computed in FE$_{ANN}$, whereas the right column are images of the axial strain estimated with speckle tracking. A least-squares strain estimator [24] was used to differentiate the displacement field estimated from the speckle-tracking algorithm [30] to produce Figures 3.11b and 3.11d.



(a)

(b)

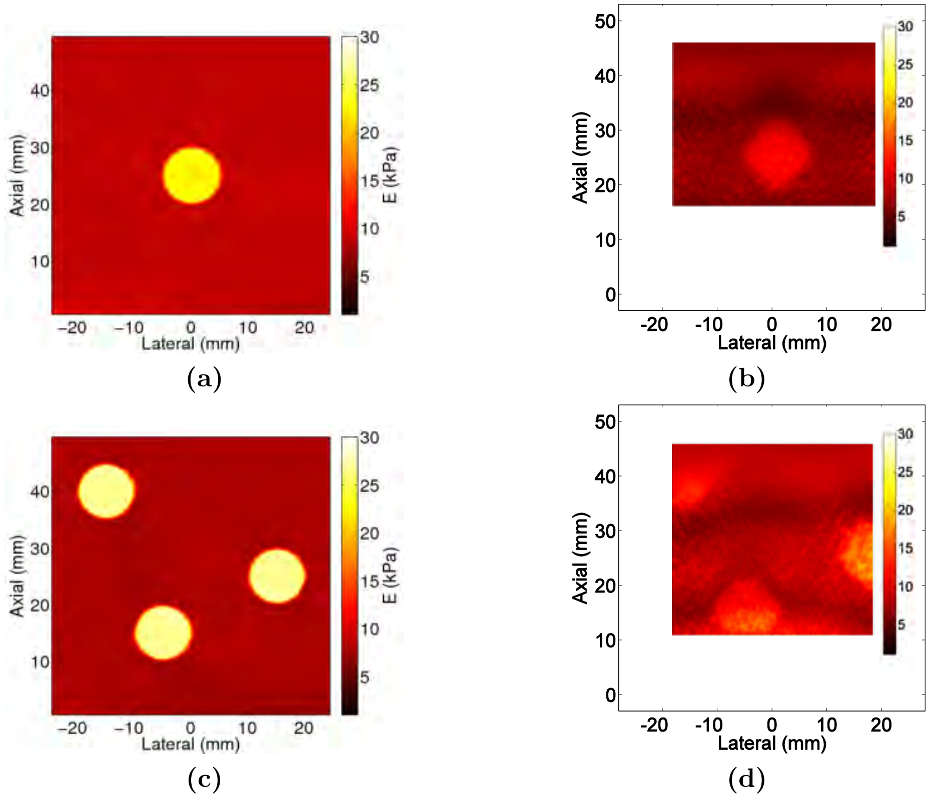
(c)

(d)

Figure 3.11: Comparison of the axial strain computed using trained ANN material models and a speckle-tracking algorithm. Axial strains computed using the ANNs come from FEA$_{ANN}$. A least-squares estimator was used to compute strain after displacement estimation from speckle-tracking. (a) Phantom 1, axial strain computed from FE$_{ANN}$ (b) Phantom 1, axial strain computed from speckle-tracking (c) Phantom 3, axial strain computed from FE$_{ANN}$ (d) Phantom 3, axial strain computed from speckle-tracking

A similar comparison is made in Figure 3.12 between the computed Young's modulus images. In the left column, modulus images were calculated point-wise with (3.4) using the stresses and strains computed in FE$_{ANN}$ (same as Figures 3.9a and 3.9c). The right column consists of Young's modulus images computed using the axial strain maps shown in Figures 3.11b and 3.11d. Inverting a simple Hookean model of linear-elasticity, the images were computed using $(\sigma_{axial})/(\varepsilon_{speckle})$, where σ_{axial} is the estimated stress field based on the applied force and $\varepsilon_{speckle}$ is the axial strain estimated from speckle-tracking.

The first advancement to notice is the ability to estimate the strain and Young's modulus throughout the entire model, not just within the imaged region. Strain computed based



(a) (b) (c) (d)

Figure 3.12: Comparison of the computed Young's modulus image using trained ANN material models and a speckle-tracking algorithm. Images computed using the ANNs come are the same as those in Figures 3.9a and 3.9c. Based on the axial strain estimated based on the speckle-tracking results, images were computed from $(\sigma_{axial})/(\varepsilon_{speckle})$ to exemplify a classical approach to image the Young's modulus. (a) Phantom 1, image computed from FE$_{ANN}$ (b) Phantom 1, image computed from speckle-tracking (c) Phantom 3, image computed from FE$_{ANN}$ (d) Phantom 3, image computed from speckle-tracking

on differentiating a displacement field calculated with speckle-tracking is confined to the imaged region. With the informational models, any geometry can be modeled and an FEA solved using the ANN material model(s), thus allowing the stress and strain fields to be estimated throughout an entire model. The second major advancement is the increase in resolution and reduction of artefacts. Adjusting the size of the elements in the FE model controls the resolution of the computed fields. Furthermore, a finer mesh will reveal more complex characteristics in the stresses and strains. With speckle-tracking, performing the strain calculation involves a derivative. Noise in the displacement estimates lowers the strain image resolution and produces artefacts which translate to the computed Young's modulus image.

3.4.5 Observing the Constitutive Behavior of Informational Models

It is possible to glimpse the constitutive behavior described by trained ANN material models by observing the “matrix” response. By probing an ANN with isolated strain probes and plotting the resulting stress responses, a constitutive matrix can be formed. For the plane-stress case, the isolated strain probes are $[\varepsilon_{11}\ 0\ 0]^T$, $[0\ \varepsilon_{22}\ 0]^T$, and $[0\ 0\ \varepsilon_{12}]^T$.

Each strain probe was used as an input to the trained ANN characterizing the background gelatin material for case (d) of Phantom 1 and the stress responses plotted, shown in Figure 3.13 as the black curves. Values spanned the ± 0.05 range for each strain probe, extending beyond the range covered during training with AutoP. After applying each strain probe to the ANN, the σ_{11}, σ_{22}, and σ_{12} responses were recorded. The slope of the black curves in Figure 3.13 correspond to the elements in the constitutive matrix in a plane-stress formulation.

Under the assumption of plane-stress response for these phantom models, the “true” response to the input strain probes can be predicted using the Young’s modulus estimate from macroindentation and a Poisson’s ratio of 0.5. Perfect mechanical characterization of the gelatin by the ANN material model would result in a set of “matrix” response curves identical to the true values. For Phantom 1, the background material had a Young’s modulus of 9.16 kPa as measured from macroindentation. A constitutive matrix was constructed following the formulation at the top of Figure 2.1 and interrogated with the same isolated strain probes as the ANN. True response curves are plotted as red curves along with the ANN response curves in Figure 3.13.

A close look at the center plot of Figure 3.13 displaying σ_{22} versus ε_{22} reveals a consequence of using trained ANNs to predict stress values approaching or extending beyond the range of training data. A linear stress response is observed in the strain range of training $-0.03 \leq \varepsilon_{22} \leq 0.03$, but the stress responses for larger input strain become increasingly nonlinear. The responses become linear when training is extended to the full range of testing.

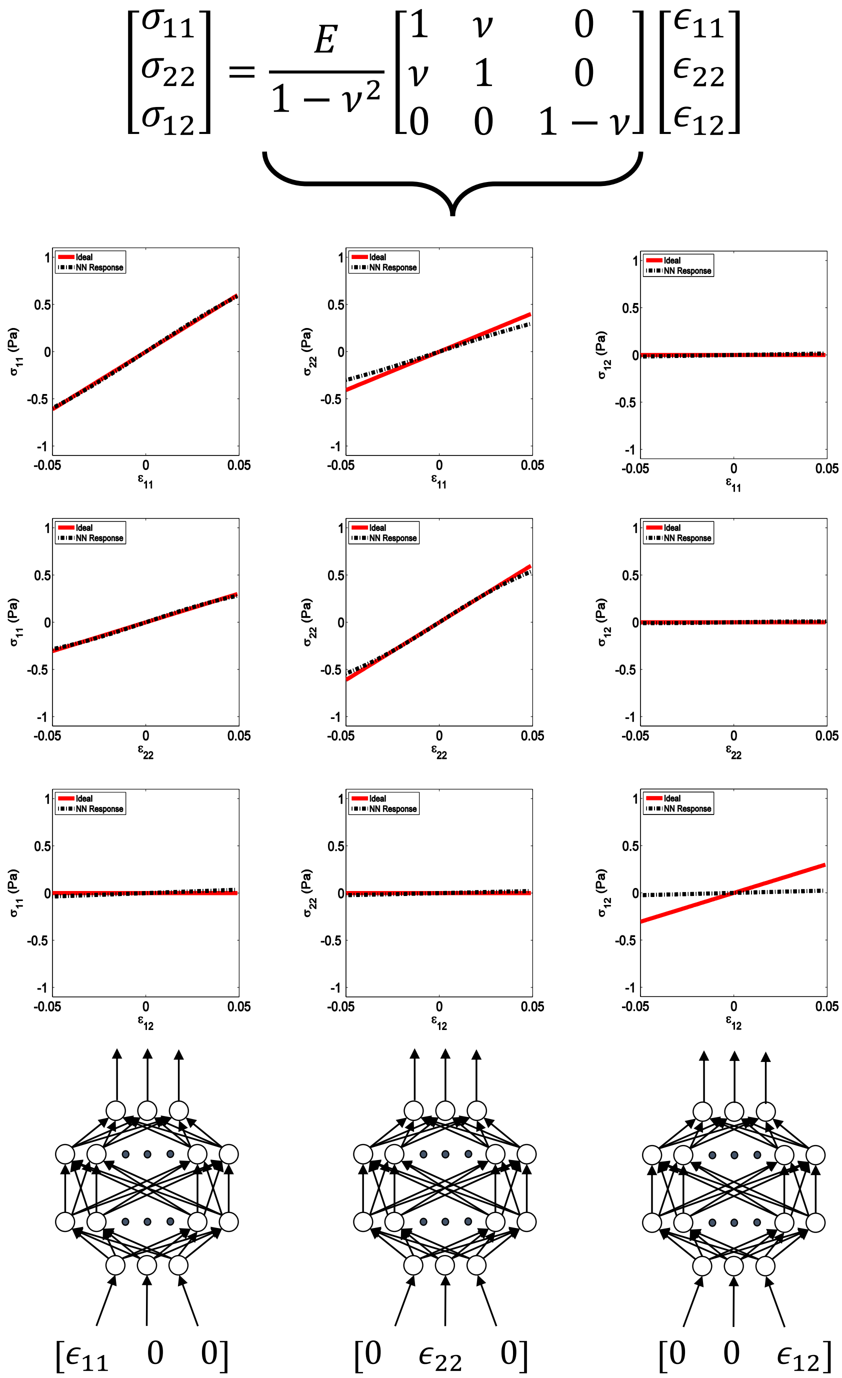


Figure 3.13: Stress response of an ANN to strain probes $[\varepsilon_{11}\ 0\ 0]^T$, $[0\ \varepsilon_{22}\ 0]^T$, $[0\ 0\ \varepsilon_{12}]^T$, Each strain vector was provided as an input and the full stress vector was recorded. Slopes of the stress-strain curves should match the values in the constitutive matrix of the plane-stress model for a Young's modulus of 9.16 kPa, estimated from indentation tests, and Poisson's ratio of 0.5. Black curves correspond to the stress values recorded from the ANN whereas the red curves are the expected response for the given Young's modulus and Poisson's ratio.

3.4.6 Effects of Varying the Amount of Information During Training

Figure 3.14 illustrates the influence of additional training data on informational model accuracy. Each plot contains the axial stress response σ_{22} to an isolated lateral strain ε_{11} for ANNs trained in different scenarios. The trained ANNs model the mechanical behavior of the background gelatin material for Phantom 1. Plot 3.14a is the stress response after training with a single data set - case (a) in Table 3.2. Plots 3.14b and 3.14c are the stress response after training with two sets of data either in succession (case c) or simultaneously (case d). These plots correspond to the top-middle plot in Figure 3.13. Red curves represent the expected behavior using the plane stress model with a Young's modulus of 9.16 kPa and Poisson's ration of 0.5.

Training with only a single set of data (case a) appears to be insufficient to fully capture the constitutive behavior of the gelatin, evidenced by the large difference between the red and black curves in Figure 3.14a. However, the addition of another set of force-displacement data acquired under slightly different loading situation produced ANNs that far more accurately model the constitutive behavior of the gelatin. Whether the additional data was used to update existing ANN material models (case c, Figure 3.14b) or incorporated with the first set of data into a single round of training (case d, Figure 3.14c), the response of the ANN more closely resembles the response predicted from a plane-stress model.

It was shown in Figures 3.5, 3.7, and 3.13 that the shear stress (σ_{12}) response to shear strain (ε_{12}) is difficult to capture using only a uniaxial load while measuring forces and displacements only along the load axis. Even though a shear stimulus is not applied, there is a shear response due to the pinned bottom boundary of the phantom and heterogeneity due to the stiff inclusion(s). The shear stress and strain images in Figures 3.6 and 3.8 illustrate these responses to a uniaxial load.

An investigation was done to find a way to tease out the information pertaining to the shear response with the force-displacement data that was acquired. Several versions of AutoP

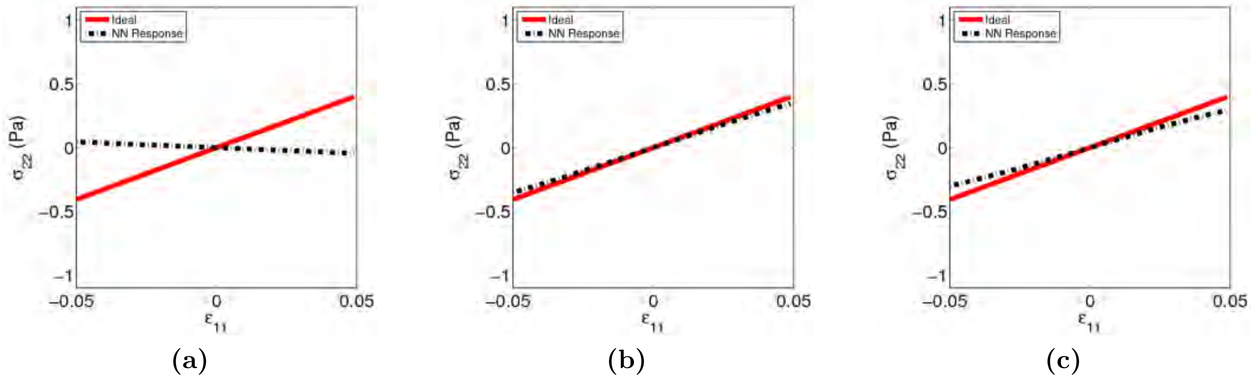


Figure 3.14: Comparing the axial stress response to a lateral strain input for different training scenarios. Black curves correspond to the trained ANN response. Red curves are the expected response based on the plane-stress formulation using $E = 9.16$ kPa and $\nu = 0.5$. All three plots correspond to training cases for Phantom 1 listed in Table 3.2. (a) Case a (b) Case c (c) Case d.

training were performed to evaluate the effects on the learned shear response: 1) training using a single set of axial force and displacement information (case (a) in Table 3.2); 2) training using two sets of axial force and displacement information simultaneously (case (d) in Table 3.2); 3) providing both axial and lateral displacements during training; and 4) checking axial and lateral displacement errors for nodes not included in training.

In tests (3) and (4), the nodes specified in Figure 3.2a remained the only ones to be given displacement information in FEA 2. No lateral displacements were provided during training in (4); lateral displacements were only used during the convergence check. Figure 3.15 details which nodes were used during the displacement error check in test (4) to constrain the test for convergence.

While none of the four described training scenarios produced ANN material models that captured the expected shear response predicted with the plane-stress model, improvement was seen when additional information was provided during training or the check for convergence. The improvement is seen as an increase in the slope of the shear stress response (black curves) toward the expected response (red curves) in Figure 3.16. ANNs developed in tests (1) and (2) had virtually no shear stress response to a shear strain input, evidenced by the a value of nearly zero of the slope. However, adding lateral displacement information during

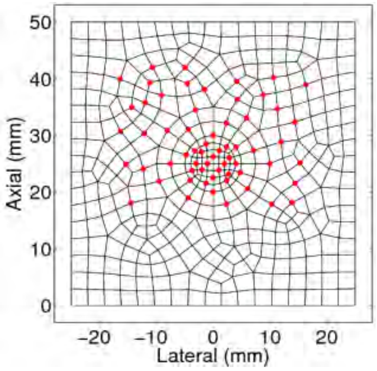
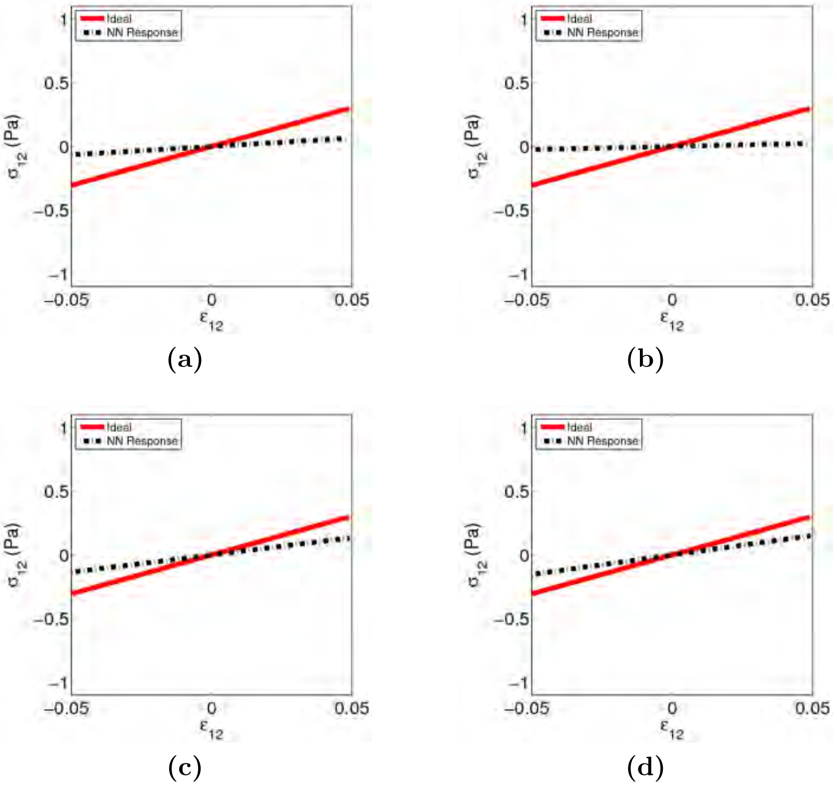


Figure 3.15: Example of the nodes used for more constrained convergence testing. Highlighted nodes are scattered throughout the imaged region. Including all of the highlighted nodes in the test for convergence, instead of just those assigned displacements in FEA 2, forces the ANNs to better learn the mechanical properties of the gelatin phantoms.

training (test (3), Figure 3.16c) or including axial and lateral displacement information for a larger number of nodes during the convergence check (test (4), Figure 3.16d) forced the ANNs to capture more of the shear behavior. Test (4) produced an ANN with the most accurate shear stress response.

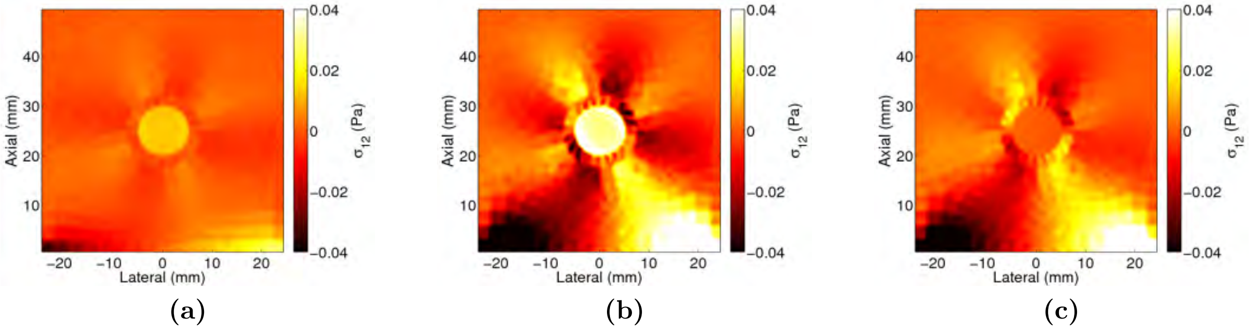
The improvement is clearly seen when comparing the shear stress fields. Figure 3.17 compares the shear stress images produced in FE_{ANN} for tests (1) and (4) to the expected shear stress field computed in FE_{Ideal} for Phantom 1. In Figure 3.17a, the nearly zero shear modulus (i.e., slope of nearly zero in Figure 3.16a) produces a suppressed shear stress field. Improved shear response characterization resulting from test (4) produced the image in Figure 3.17b which more closely resembles the expected shear stress field displayed in Figure 3.17c. The improvement resulting from test (4) suggest there is merit in adding nodes to the displacement error check for convergence without including said nodes in training. This strategy could possibly avoid the problem of including too much information during training, which risks the ANNs learning the noise, while still using more of the acquired data to increase the accuracy of the ANN material models.



(a) (b)

(c) (d)

Figure 3.16: Comparison of the σ_{12} response to an input of ϵ_{12} for Phantom 1 trained under different conditions. (a) No training adjustment (case a) (b) Training with two sets of data simultaneously (case d) (c) axial and lateral displacements provided to FEA 2 (d) convergence calculated using displacements at nodes throughout the imaged region, not limited to displacements applied in FEA 2.



(a) (b) (c)

Figure 3.17: Comparison of the estimated shear stress field σ_{12} of phantom 1 using training versions (1) and (4). (a) Training version (1) corresponds to the shear response in Figure 3.5c. (b) Training version (4) where displacement convergence is required for nodes not used for training; the results correspond to the shear response in Fig. 3.16d. (c) Shear stress estimated for an ideal linear-elastic plane-stress model. This is the same shear stress field displayed in Figure 3.6c

Chapter 4

Discussion

Results described in Section 3.4 demonstrate the ability of ANNs trained with AutoP to learn the linear-elastic constitutive behavior of gelatin materials using a sparse sampling of data. Using only a few estimated axial displacements within the inclusions, as well as axial forces and displacements at the surface, NNCMs were developed that were used to compute images of the stress and strain vectors throughout the entire phantom model. These stress and strain estimates were then used to compute accurate Young's modulus images for each model.

4.1 Sampling Requirements and Training Data Selection

Including the use of US imaging to track interior motion of the target object increases the amount of available information tremendously. Discussed in Section 2.3.5 was the drawback of providing too much information when noise is present. It was necessary, then, to determine the minimum amount of force-displacement data that had to be provided during training to build accurate ANN material models. Furthermore, the effect of the distribution of the displacement data throughout the model had to be accounted for.

Preliminary tests sought to answer or at least provide a heuristic for these issues. Three different analyses were run on the Phantom 1 model where displacements were given 1) for the probe only, 2) along a vertical line through the lateral center of the inclusion, and 3) along a horizontal line through the axial center of the inclusion. Note that the same number of nodes were given displacement information in cases (2) and (3). After training, the Young's modulus estimate for the background material was near the expected value in all three cases; however, only case (2) resulted in a modulus estimate close to the true value. Results of case (1) are not entirely unexpected because the background gelatin made up the bulk of the phantom model and would have the largest effect on the force-displacement behavior measured at the surface. Furthermore, the inclusion had no contact with any of the phantom boundaries, nor did the complexity in the stress or strain fields caused by the inclusion reach the boundaries for the applied loads. This means the inclusion was not sensed from surface measurements; interior motion had to be tracked in order to determine a heterogeneity existed within the phantom model.

Results from cases (2) and (3) are more interesting and begin the reveal what and where information needs to be supplied for training. Based on case (1), it is evident that some amount of information needs to be provided for each region in the model with unique mechanical properties. Cases (2) and (3) each provided the same amount of data, the difference lying in how the data was dispersed in the FE mesh. In case (2), the displacement data fell along a line parallel to the direction of the applied force whereas in case (3) the data fell on a line perpendicular to the loading direction. Because only axial displacement information was given, the motion of the nodes in case (3) reveals very little information about how the points move relative to each other under uniaxial loading. For example, if no noise was present in the displacement data, the nodes would have had similar - if not identical - axial motion. In case (2), though, the nodes compress in relation to each other, providing more information about how the material responds to the applied load and thus more about the constitutive behavior of the inclusion.

The results of cases (1)-(3) led to an empirically determined sampling strategy for the 2D models under plane-stress loading: displacement information should be provided for nodes oriented along the force gradient produced from the applied load. Furthermore, displacements must be provided within each region with unique mechanical properties. In this study where force was applied in the axial direction only, nodes supplied with displacement data in FEA 2 should be aligned along the axial direction and be within the boundaries of the inclusion(s). It was also found that providing displacements for three nodes within each region was enough to produce accurate ANN material models. Later studies involving 3D models and materials with more complex mechanical properties (e.g., nonlinearity, anisotropy, etc.) will likely require a different sampling strategy. It is expected that the basics of the proposed sampling strategy will transfer, with the largest difference being the amount of data that needs to be acquired and provided during training.

4.2 Effects of Constraining Convergence Testing

Stresses and strains resulting from uniaxial loading mostly span the axial and lateral spaces, as shown in the right columns of Figures 3.6 and 3.8. Pinning the bottom of the phantom and adding the inclusion produces some shear stresses and strains - mostly in the background material - that are much smaller in magnitude than the axial and lateral counterparts. This limited amount of shear information makes it more difficult for the ANNs to learn the shear response of the material. However, it was demonstrated that adjustments to the training process can be made that result in the ANNs better capturing the shear behavior. The most successful of these adjustments was the addition of displacement data during the convergence check not used in FEA 2.

Adding more displacement data during the displacement error calculation added more constraint on the convergence check. Even though the extra data was not given during FEA 2, the nodal displacements from FEA 1 still had to accurately predict the motion throughout a larger region. The extra constraint had the effect of forcing AutoP to perform more training iterations for each load step. An analogy of ANN training is learning information contained on flash cards. As a student flips through the stack cards, he or she checks the answer given to the correct one on the back. If the answer was correct, the student has successfully learned the information; otherwise a mental note is made to try and correct the error on the next pass. Enforcing the extra constraint during the convergence check is like forcing the student to make more passes through the deck of cards until a larger percentage of the information has been learned.

Another method was attempted to increasing the ANNs ability to learn the shear response, but was not discussed in Section 3.4. This test consisted of assuming the matrix of material moduli, $\mathbb{C}_{ijkl}$, was symmetric. Recall that this matrix is calculated from the ANN and is used to solve the FEA. With this assumption, there was no significant effect on the ability of the ANNs to learn the shear behavior. This may be due to the fact that displacements for Phantom 1 were given along a vertical line laterally centered within a symmetric model. A symmetry assumption would mostly effect the displacement symmetry within the FE model. Given that the nodes used during the displacement error calculation lie on the line of symmetry, virtually no extra constraint was enforced in the check for convergence. It is possible that the assumption of a symmetric matrix would have more of an effect on Phantoms 2 and 3 where symmetry was not present in the model or if nodal displacements used in FEA 2 were scattered asymmetrically. Regardless, it is an assumption about the underlying constitutive model and the lack of improvement when implementing this assumption justifies its neglect in future use of informational modeling.

4.3 Decreasing Computational Load

Informational models discussed in Section 3.4 took 102±14 minutes to train with AutoP on a standard quad-core processor operating at 3.4 GHz. The largest computational burden comes from solving FEA 1 and FEA 2. Training over two passes each comprised of ten load steps requires a minimum of 60 FEAs to be solved in the best-case, up to a maximum of 220 in the worst-case. Best-case scenario occurs when only one training iteration needs to be performed in each load step whereas the worst-case occurs when the maximum of five iterations is performed every load step. Computing the stress, strain, and Young's modulus images required time on the order of minutes, an insignificant amount of time compared to AutoP.

There are a few ways in which to training time can be reduced. Displacement MSE in Figures 3.10b and 3.10d appears to reach a minimum after a maximum of six load steps in the first pass. Removing the second pass of training would reduce the computation time by approximately half and not negatively impact the accuracy of the NNCMs. It may also be possible to monitor the magnitude of the displacement MSE and the slope of the curve. If both of these values fall below a predefined threshold, it may be valid to assume the models have converged and no more training is required, thus allowing the training process to terminate early. The drawback to early training termination is the information spanned by the later load steps would not be presented to the ANNs. For example, exiting AutoP after load step 7 would mean the extended stress-strain range covered in load steps 8-10 would not be learned by the ANNs. Care would need to be taken if such a method was implemented.

Other possible ways to reduce training time include reducing the number of load steps, but still covering the whole training range. Instead of training for all ten load steps, training could occur only on steps 2, 4, 6, 8, and 10. The full training range is covered, but the run time of AutoP would be reduced by approximately half. Because the ANNs can generalize within the training range, removing the intermediate steps would not likely be detrimental to the informational model building process on the 2D linear-elastic models produced in this study. However, when modeling more complex behavior, the intermediate steps may be required. Essential information may be lost if materials exhibit nonlinear, path-dependent, and/or time-dependent behavior and are inadequately sampled.

4.4 Future Studies

This report demonstrates the first use of incorporating informational modeling with AutoP to enhance the capabilities of ultrasound elastography. Validation was performed using gelatin material that exhibits linear-elastic behavior over the range of deformation applied in the previously described measurements. It was shown that images of the stress and strain vectors, as well as the Young's modulus, could be accurately computed after training with only force-displacement data. Gelatin was chosen based on the predictability of its response. Because this study is only the first step in using AutoP to develop NNCMs for biological tissues, models with simple mechanical characteristics were the appropriate choice.

Currently, the geometry of the imaged object is assumed to be known and used to create the FE mesh. This will not always be the case, especially when imaging biological tissues with complex geometries and many heterogeneities. Clinical applications will likely require US images to be segmented to create the mesh. A preliminary test was performed where a US image was manually segmented to create a mesh approximating the geometry of Phantom 1. Informational modeling results were similar to those for Phantom 1 described in Chapter 3. While not definitive or comprehensive, these preliminary results suggest the object geometry does not have to be perfectly modeled in the FEAs to produce accurate NNCMs. Later studies will be performed where an autonomous segmentation algorithm is implemented to generate the FE meshes.

It was mentioned in Section 2.3 the previous applications of AutoP. Several of those studies successfully built ANN material models that capture nonlinear, path-dependent, and time-dependent mechanical behaviors. Accurate characterization of biological tissues will require adjusting the training process described in Section 2.3.3 and developing a sampling strategy. Given the previous success of modeling more complex behaviors, it is expected that building NNCMs for soft-tissues will be met with success, especially with the extra information obtained with ultrasound imaging.

References

[1] K. N. Altahhan, Y. Wang, N. Sobh, and M. F. Insana. Indentation measurements to validate dynamic elasticity imaging methods. *Ultrasonic Imaging*, 2015.

[2] Alexandra Athanasiou, Anne Tardivon, Mickael Tanter, Brigitte Sigal-Zafrani, Jeremy Bercoff, Thomas Deffieux, Jean-Luc Gennisson, Mathias Fink, and Sylvia Neuenschwander. Breast lesions: quantitative elastography with supersonic shear imagingpreliminary results 1. *Radiology*, 256(1):297–303, 2010.

[3] P. E. Barbone and J. C. Bamber. Quantitative elasticity imaging: what can and cannot be inferred from strain images. *Physics in Medicine and Biology*, 47(12):310, jun 2002.

[4] P. E. Barbone and N. H. Gokhale. Elastic modulus imaging: on the uniqueness and nonuniqueness of the elastography inverse problem in two dimensions. *Inverse Problems*, 20(1):283–296, feb 2004.

[5] P. E. Barbone and A. A. Oberai. Elastic modulus imaging: some exact solutions of the compressible elastography inverse problem. *Physics in Medicine and Biology*, 52(6):1577–1593, mar 2007.

[6] R. G. Barr. Elastography in clinical practice. *Radiologic Clinics of North America*, 52(6):1145 – 1162, 2014. Update in Ultrasound.

[7] R. G. Barr, S. V. Destounis, L. B. Lackey, W. E. Svensson, C. Balleyguier, and C. Smith. Evaluation of Breast Lesions Using Sonographic Elasticity Imaging. *J Ultrasound Med*, 31:281–287, 2012.

[8] T. Belytschko, W.K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures*. Nonlinear Finite Elements for Continua and Structures. Wiley, 2000.

[9] X. Chen, M. J. Zohdy, S. Y. Emelianov, and M. O. Donnell. Lateral speckle tracking using synthetic lateral phase. 51(5):540–550, 2004.

[10] C Cole-Beuglet, RZ Soriano, AB Kurtz, and BB Goldberg. Ultrasound analysis of 104 primary breast carcinomas classified according to histopathologic type. *Radiology*, 147(1):191–196, 1983.

[11] M. M. Doyley. Model-based elastography: a survey of approaches to the inverse elasticity problem. *Physics in medicine and biology*, 57(3):R35–73, feb 2012.

[12] E. R. Ferreira, A. A. Oberai, and P. E. Barbone. Uniqueness of the elastography inverse problem for incompressible nonlinear planar hyperelasticity. *Inverse Problems*, 28(6):065008, 2012.

[13] Y-C Fung. *Biomechanics: mechanical properties of living tissues.* Springer Science & Business Media, 2013.

[14] J. Ghaboussi, D. A. Pecknold, M. Zhang, and R. M. Haj-ali. Autoprogressive Training of Neural Network Constitutive Models. *International Journal for Numerical Methods in Engineering*, 42(November 1996):105–126, 1998.

[15] J. Ghaboussi and D.E. Sidarta. New nested adaptive neural networks (nann) for constitutive modeling. *Computers and Geotechnics*, 22(1):29 – 52, 1998.

[16] T. J. Hall, M. Bilgen, M. F. Insana, T. Krouskop, et al. Phantom materials for elastography. *Ultrasonics, Ferroelectrics, and Frequency Control, IEEE Transactions on*, 44(6):1355–1365, 1997.

[17] Y. M. A. Hashash, S. Jung, and J. Ghaboussi. Numerical implementation of a neural network based material model in finite element analysis. *International Journal for Numerical Methods in Engineering*, 59(7):989–1005, 2004.

[18] S. Haykin. *Neural Networks: A Comprehensive Foundation.* Prentice Hall, 1998.

[19] T. J. R. Hughes. *The finite element method: linear static and dynamic finite element analysis.* Courier Corporation, 2012.

[20] A.K. Jain, J. Mao, and K.M. Mohiuddin. Artificial neural networks: a tutorial. *Computer*, 29(3):31–44, mar 1996.

[21] S. Jung and J. Ghaboussi. Characterizing rate-dependent material behaviors in self-learning simulation. *Computer Methods in Applied Mechanics and Engineering*, 196(13):608 – 619, 2006.

[22] S. Jung and J. Ghaboussi. Neural network constitutive model for rate-dependent materials. *Computers & Structures*, 84(1516):955 – 963, 2006.

[23] F. Kallel and M. Bertrand. Tissue elasticity reconstruction using linear perturbation method. *IEEE Transactions on Medical Imaging*, 15(3):299–313, June 1996.

[24] F. Kallel and J. Ophir. A Least-Squares Strain Estimator for Elastography. *Ultrasonic Imaging*, 19(3):195–208, July 1997.

[25] E. K. Konofagou and J. Ophir. A New Elastographic Method for Estimation and Imaging of Lateral Displacements, Lateral Strains, Corrected Axial Strains, and Poisson's Ratios in Tissues. 24(8):1183–1199, 1998.

[26] T. A. Krouskop, T. M. Wheeler, F. Kallel, B. S. Garra, and T. Hall. Elastic Moduli of Breast and Prostate Tissue under Compression. *Ultrasonic imaging*, 20(4):260–274, 1998.

[27] J. Ophir, I. Céspedes, H. Ponnekanti, Y. Yazdi, and X. Li. Elastography: a quantitative method for imaging the elasticity of biological tissues. *Ultrasonic imaging*, 13(2):111–134, 1991.

[28] M. Orescanin and M.F. Insana. Shear modulus estimation with vibrating needle stimulation. *Ultrasonics, Ferroelectrics, and Frequency Control, IEEE Transactions on*, 57(6):1358–1367, June 2010.

[29] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In *IEEE International Conference on Neural Networks*, volume 1, pages 586–591. IEEE, 1993.

[30] H. Rivaz, E. M. Boctor, M. A. Choti, and G. D. Hager. Real-time regularized ultrasound elastography. *IEEE transactions on medical imaging*, 30(4):928–45, April 2011.

[31] R. Rojas. *Neural Networks: A Systematic Introduction*. Springer Berlin Heidelberg, 1996.

[32] A. P. Sarvazyan, A. R. Skovoroda, S. Y. Emilianov, J. B. Fowlkes, J. G. Pipe, R. S. Adler, R. B. Buxton, and P. L. Carson. Biophysical Bases of Elasticity Imaging. *Acoustical Imaging*, 21:223–240, 1995.

[33] A.R. Skovoroda, L.A. Lubinski, S.Y. Emelianov, and M. O'Donnell. Reconstructive elasticity imaging for large deformations. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 46(3):523–535, may 1999.

[34] C. Sumi, A. Suzuki, and K. Nakayama. Estimation of shear modulus distribution in soft-tissue from strain distribution. *IEEE Transactions On Biomedical Engineering*, 42(2):193–202, 1995.

[35] E. Ueno, T. Shiina, M. Kuboto, and K. Sawai. *Research and Development in Breast Ultrasound*. Springer Japan, 2006.

[36] U. Ünlütürk, M. F. Erdoan, Ö. Demir, S. Güllü, and N. Başkal. Ultrasound Elastography Is Not Superior to Grayscale Ultrasound in Predicting Malignancy in Thyroid Nodules. *Thyroid*, 22(10):1031–1038, 2012.

[37] C. W. Washington and M. I. Miga. Modality independent elastography (MIE): a new approach to elasticity imaging. *IEEE transactions on medical imaging*, 23(9):1117–28, sep 2004.

[38] P. Wellman, R. Howe, E. Dalton, and K. Kern. Breast tissue stiffness in compression is correlated to histological diagnosis. *submitted to Journal of Biomechanics*, December 1999.

[39] G. J. Yun, J. Ghaboussi, and A. S. Elnashai. Self-learning simulation method for inverse nonlinear modeling of cyclic behavior of connections. *Computer Methods in Applied Mechanics and Engineering*, 197(3340):2836 – 2857, 2008.