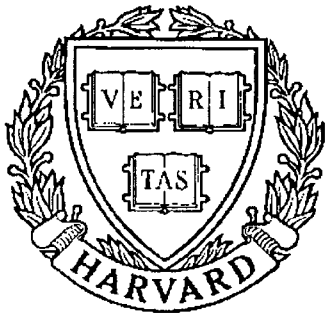
THESIS REPORT
Master's Degree

SYSTEMS
RESEARCH
CENTER



Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

Neural Networks for Low Level Processing of Tactile Sensory Data

by: Y.C. Pati
Advisor: P.S. Krishnaprasad

M.S. 89-2
Formerly TR 89-8

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1988		**00-00-1988 to 00-00-1988**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Neural Networks for Low Level Processing of Tactile Sensory Data	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Maryland,The Graduate School,2123 Lee Building,College Park,MD,20742**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**106**	

Neural Networks For Low Level Processing Of Tactile Sensory Data

by

Yagyensh C. Pati

Thesis submitted to the Faculty of The Graduate School
of The University Of Maryland in partial fulfillment
of the requirements for the degree of
Master of Science
1988

Advisory Committee:

Professor P. S. Krishnaprasad Chairman/Advisor
Professor Martin C. Peckerar
Associate Professor Shihab Shamma

ABSTRACT

Title of Thesis:	Neural Networks for Low-Level Processing of Tactile Sensory Data
Name of degree candidate:	Yagyensh C. Pati
Degree and Year:	Master of Science, 1988
Thesis directed by:	Dr. P. S. Krishnaprasad Professor Department of Electrical Engineering

As the field of robotics continues to strive forward, the need for artificial tactile sensing becomes increasingly evident. Real-time, local processing of tactile sensory data also becomes a crucial issue in most applications of tactile sensing. In this thesis it is shown that analog neural networks provide an elegant solution to some of the problems of low level tactile data processing. We consider the particular problem of 'deblurring' strain data from an array of tactile sensors. It is shown that the inverse problem of deblurring strain measurements to recover the surface stress over a region of contact is ill-posed in the sense defined by Hadamard. This problem is further complicated by the corruption of sensor data by noise. We show that the techniques of 'regularization' may be used to introduce prior knowledge of the solution space into the solutions in order to transform the problem to one which is well-posed and less sensitive to noise. The particular regularizer chosen for the recovery of normal stress distributions is of the functional form of Shannon entropy. Formulation of the inverse problem so as to regularize the solutions results

in a variational principle which must be solved in order to recover the surface stress. An analog neural network which provides the desired solutions to the variational principle as a course of natural time evolution of the circuit dynamics is proposed as a solution to the requirements for fast, local processing in tactile sensing. We discuss performance of the network in the presence of noise based upon computer simulations. We also demonstrate, by means of a breadboard prototype of the network, the speed of computation achievable by such a network. An integrated circuit implementation of the proposed network has been completed and the requirements of such implementations is discussed.

Acknowledgements

I would like to take this oppurtunity to express my gratitude to Dr. P.S. Krishnaprasad, Dr Martin Peckerar, and Dr. Shihab Shamma for their contributions to this work. I would like to thank Dr. Peckerar for introducing me to the subject of neural networks, for his insight into the problem of analog VLSI implementation of neural networks and for the continuous support and encouragement which he provided. I would like to thank my advisor, Dr. P. S. Krishnaprasad who suggested the application of neural networks to problems in tactile sensing and provided constant guidance, inspiration and motivation throughout the course of this research. I would like to express my gratitude to Dr. Shihab Shamma for the numerous references on the subject of neural networks and for being a member of my Advisory Committee. I would also like to thank Dr. Chung Ting Yao for his guidance and countless suggestions on the integrated circuit implementation of the neural network described in this thesis.

This research was supported in part by the National Science Foundation's Research Centers Program: NSFD CDR 8803012, the Air Force Office of Scientific Research under contract (AFOSR-88-0204) and the Naval Research Laboratory.

The experimental work in this thesis was carried out at the Intelligent Servosystems Laboratory of the Systems Research Center and at Microelectronics Processing Facility of the Naval Research Laboratory in Washington D.C..

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER
ONE

INTRODUCTION

Recent years have seen a significant increase in the complexity of tasks performed by robotic manipulators. As the complexity of these tasks continues to grow, the need for automated tactile sensing becomes increasingly evident. The term tactile sensing, as used here, refers to the continuous sensing of forces over regions of contact. Included in this definition of tactile sensing are the more rudimentary operations of binary contact sensing and pressure sensing.

Since the specific requirements of robot tactile sensing have not as yet been clearly defined, it is often useful to view tactile sensing in humans as a model for artificial tactile sensing. Tactile sensing in humans is a dynamic process in which dexterous hands are used in conjunction with dense arrays of subcutaneous sensors to extract information about the contact which is necessary for feature identification and formulation of manipulation strategies (see [1]). An 'ideal' artificial tactile sensing system should strive to perform a similar function.

The problem of tactile sensing can be hierarchically separated into three stages.

(a) At the bottom of the hierarchy, there are the device level problems of designing a tactile sensory device, and of designing a dextrous manipulator to

be equipped with such sensors. According to a survey of robotic researchers and manufacturers (see [2]), tactile sensors should be distributed in arrays on thin, flexible, compliant substrates. Also, since tactile sensing is based upon physical contact, it is required that the entire structure comprising the tactile sensors be mechanically durable and robust against environmental variations. It was also sugested in [2] that a fair amount of preprocessing of the sensory data be performed at the sensor level so as to limit the quantity of data to be transferred to the central processing unit.

(b) Given data from the tactile sensors, the second stage of the hierarchy is concerned with low-level processing and extraction of information. For example, often raw data provided by the sensors are not measurements of contact forces, but are related in some manner to the stress profile over the region of contact. As will be shown in Chapter 2, an inverse problem arises which must be solved to determine the stress profile. Detection of edges and identification of conditions such as slippage should also be performed at this level of the hierarchy.

(c) The top most level in this hierarchical picture of tactile sensing is the level at which decisions are made, based upon information provided by the two previous stages. Formulation of manipulation strategies is performed at this level. For example, if a condition of slippage has been recognized, a manipulation strategy should be formulated so as to alleviate this condition. Also at this level of processing, decisions are made regarding the identification of

a grasped object based upon information about texture, shape (which could have been determined from knowledge of edges), and material (which may possibly be determined from thermal conductivity).

It is among the goals of this thesis to take a step towards integrating the two lowest levels of this hierarchy.

In the case of human tactile sensing, the total number of sensor cells in a fingertip area of 20×30mm can exceed 60,000. In [2] it is suggested that a typical array of 10×10 sensor elements per square inch should suffice for many applications of tactile sensing. Considering only this reduced number of 100 sensors, the quantity of data that must be transferred to the central processing unit, if all processing is to be done there, is still prohibitive. It would not in general be acceptable to have a cable running from the hand to the CPU consisting of several hundred conductors. In addition to simply conveying the information to the central processing unit, there is also the actual processing task which would demand much of the available processing time. It is clear that a great deal of preprocessing should be performed at or near the sensor array itself. In doing so, a simpler representation of the data may be produced which could then be conveyed to the central processing unit. Demands upon the processing power of the CPU would then be limited to, at most, only the top level in the tactile sensing hierarchy. In humans, useful information is often acquired through active manipulation. A grasped object is often scraped, rolled, mutilated to determine its properties ([2]). Therefore it is reasonable to assume that in robots, as in humans, static contact analysis together

with dynamical analysis may prove to be useful. This dynamical aspect of tactile sensing, together with the fact that tactile sensing is generally to be applied in a 'real time' environment, implies the need for fast processing of tactile sensory data.

In this thesis we discuss a framework within which at least some of the second level of tactile information processing may be performed. The approach taken is designed to meet the requirement of local fingertip processing as well as the demand for fast computation. The problem considered in this thesis is the determination of surface stress from an array of sensors which provides measurements of strain induced in a elastic medium by contact at the boundary. Since it can be assumed that the data provided by the sensors is corrupted by noise it is necessary to consider this additional aspect of the problem as well. In order to meet the requirements of fast local fingertip processing, the paradigm of neural networks is considered.

Inspired by the work of a of neurophysiologists and psychologists and other researchers on the mechanisms of computation, learning and memory in biological systems, artificial neural networks are an attempt to reproduce the computational efficiency observed in the nervous system. An artificial neural network may be defined as a highly interconnected network of simple processing units. The processing units themselves are rarely more than simple amplifiers (usually nonlinear amplifiers such as those with sigmoidal characteristics are used), yet neural networks have in many instances demonstrated an ability to solve complex problems. The computational power of artificial neural networks is embedded in the nature of connectivities between the processing units (or *neurons*) of which they are composed.

Neural networks[1] are usually regarded as being comprised of layers of neurons. In general connections in a neural network are defined between neurons in different layers as well as between neurons of a single layer. Networks in which connections occur from the output to the input of a single neuron (feedback connections) are termed *recurrent networks.* Associated with a connection (also called a *synapse*) between two neurons, say neuron i and neuron j, is a number w_{ij} called the *weight* of the connection (sometimes referred to as the *synaptic weight*) between neuron i and neuron j, which determines the effect that the output of neuron i has upon the activity of neuron j. For example, if the output of neuron i is v_i then the input to neuron j due to the connection of neuron i to neuron j is given by $w_{ij}v_i$. It is the connectivity profile (distribution of connection weights) which determines the computational task performed by any given network. A layer of a neural network which accepts external inputs is called an input layer for the network. Similarly a layer whose outputs are considered as outputs of the network as a whole is termed an output layer. Since both the input and output layers can consist of many neurons, the inputs and outputs of a neural network are vectors (in finite dimensional vector spaces). Computation in a neural network is either instantaneous as in feedforward networks or involves the time evolution of the network as a dynamical system (as in recurrent networks). In either case the the network operates on vectors and the computation occurs in a parallel asynchronous fashion.

There is evidence that computation in the nervous system is mostly performed

[1]The term *neural networks* will be used in reference to artificial neural networks unless otherwise indicated

in analog. The paradigm of neural networks also provides a formalism for the analog hardware implementation of inherently parallel algorithms. Biological neurons are often modeled as integrators (with the sum of all inputs to the neuron as their inputs) followed by output functions. In terms of analog circuits this corresponds to a simple RC integrator circuit followed by an amplifier with the desired characteristics. Connections between neurons can be implemented as resistors. If the output of neuron i is a voltage which is connected to the input of neuron j through a resistor R_{ij} then Ohm's Law dictates that the current input to neuron j due to the output v_i of neuron i is $R_{ij}v_i$. Hence the synaptic weight w_{ij} is simply $1/R_{ij}$. Hence having arrived at a neural network model for computation, implementation of the network as an electrical circuit is a natural extension. Advances in microelectronics have provided the technology required for implementations of 'small' neural networks. VLSI implementations of networks consisting of hundreds, thousands, even tens of thousands of neurons, should be possible with further advances in microelectronic technologies, but as yet are unrealizable due mostly to limitations in integrated circuit density and physical size. Analog computation is limited in speed only by the propagation time of electrons in the conductors and by any dynamics induced by capacitors and inductors (parasitic or otherwise). Thus analog computation is inherently fast. Integrated circuit implementation of neural networks to perform the low level processing tasks of the second level of hierarchy in tactile sensing would provide fast computation which can be performed in close physical proximity to the sensors. The resultant of processing performed by such a network can then be transmitted to the central processing unit for higher

level interpretation (and decision making based upon the interpretation) using relatively few conductors. Such an approach would also result in lower computational demand upon the CPU.

In Chapter 2 a particular inverse problem which arises in the context of tactile sensing is introduced. It is shown that inverse problems of this nature are 'ill-posed' and hence solutions must be obtained using special techniques. The inverse problem is formulated as a variational principle and 'regularization' is used to compensate for the ill-posedness of the problem and provide for reliable computation in the presence of sensor noise.

In Chapter 3 some aspects of nonlinear RC electrical networks are discussed. It is shown that under certain conditions it is possible to guarantee stability and explicitly determine a strict Lyapunov function for such a network. Since such a Lyapunov function is minimized in the course of natural time evolution of the network, steady state outputs of the network correspond to the solutions of a minimization problem (defined by the Lyapunov function).

An analog neural network to solve the inverse problem of tactile sensing is described in Chapter 4. Using the results of Chapter 3, an energy function for the proposed network is determined and shown to be equivalent to the variational principle formulated in Chapter 2 for regularized solution of the inverse problem. Computer simulations of the proposed network are presented in order to evaluate performance of the network in the presence of noise.

Chapter 5 is concerned with the analog electrical circuit implementation of the network described in Chapter 4. A prototype breadboard model of the network as

well as a preliminary attempt at integrated circuit implementation of the network are discussed.

CHAPTER
TWO

THE INVERSE PROBLEM IN TACTILE SENSING

In this chapter we examine the fundamental inverse problem which arises in the context of robot tactile perception. Inverse problems of a similar nature also arise in the field of computational vision in early vision problems which have often been designated as inverse optics (see Poggio and Koch [3]). Hadamard [4] in 1923 defined a problem to be well-posed if,

1. The solution exists.

2. The solution is unique.

3. The solution depends continuously on the initial data.

If any of the above three criteria are not satisfied, the problem is said to be ill-posed. Most inverse problems which arise in physical settings (such as those in early vision) are ill-posed and thus require special techniques to solve.

In Section 2.1 and 2.2, models of the transduction process involved in tactile sensing are presented and the inverse problem is defined. In Section 2.4 it is shown that since the integral (convolution) operator describing the transduction process is compact, linear and with an infinite dimensional domain, the inverse operator is unbounded and thus the problem is ill-posed. Since data from an array of tactile

sensors is discrete, in Section 2.4 the process of approximating the integral operator by a finite matrix is justified.

2.1 Tactile Sensors and Compliant Contact

Numerous tactile sensors have been designed and fabricated for use with robotic end-effectors. All such sensors have been based on the deformation of materials by contact forces and the measurement and interpretation of the deformation to determine the forces inducing it. Examples of some approaches to tactile sensor design are to be found in [5], [6], [7], [8], and [9]. In typical approaches to tactile sensor design, the transduction process is effected by materials such as piezo-electric polymers [5] and crystals, conductive rubber [5], and piezoresistive materials such as silicon [10]. In [9] a tactile sensor is described which is designed to operate based upon changes in optical characteristics of a material at boundaries (total internal reflection). Such an approach has the advantage of very high spatial resolution since it is not necessary to construct arrays of discrete sensors. Another novel approach to tactile sensor design is described in [11] where transduction is based upon variations in magnetic fields which are measured by a VLSI array of Hall-effect sensors. In most approaches to tactile sensing, there arises an inverse problem, namely, given data from the sensors, determine the force profile at the contact region. For the purpose of providing a concrete example, we concern ourselves here with a silicon based piezoresistive triaxial tactile sensor which has been designed and fabricated at the Naval Research Laboratories in Washington D.C. and described in [10].

The sensor is constructed as a micromachined silicon mesa surrounded by a thin

diaphragm (see Figure 2.1). On the front side of the sensor, beneath the mesa, an array of resistors is formed in the silicon by a diffusion process. An array of these sensors is constructed on a single die and then bonded and packaged to be mounted on a robot fingertip. As force is applied to the mesa, the diaphragm is deformed which in turn deforms the diffused resistors causing a change in their resistances. The changes in resistances are measured to determine the magnitude and direction of the applied force. Sensitivity of the tactile sensor is determined by the magnitude of strain induced in the resistors by a given force. In order to increase sensitivity of the sensors, it is necessary to decrease the thickness of the diaphragm and thereby increase the likelihood of its rupture due to excessive force. To attenuate large forces and provide damping against large impulsive impact forces, the array of sensors is covered by a layer of compliant material such as rubber or polyimide.

The use of a compliant layer for fingertip contact serves three beneficial functions.

(1)As mentioned previously the compliant layer serves to protect the sensors from damage due to contact forces.

(2) In order to facilitate a stable grasp it is desirable to enhance friction at the contact surfaces. For given material at the contact surface, this can be achieved by maximizing contact area. A rigid material in contact with an irregular or bumpy surface, contacts the surface at a few discrete points only (see Figure 2.2(a)) whereas a compliant material can conform to the surface and thus maximize contact area (see Figure 2.2(b)). Since the tactile sensing array provides the desired contact surface on the robot finger, it is beneficial to provide compliant contact atop the

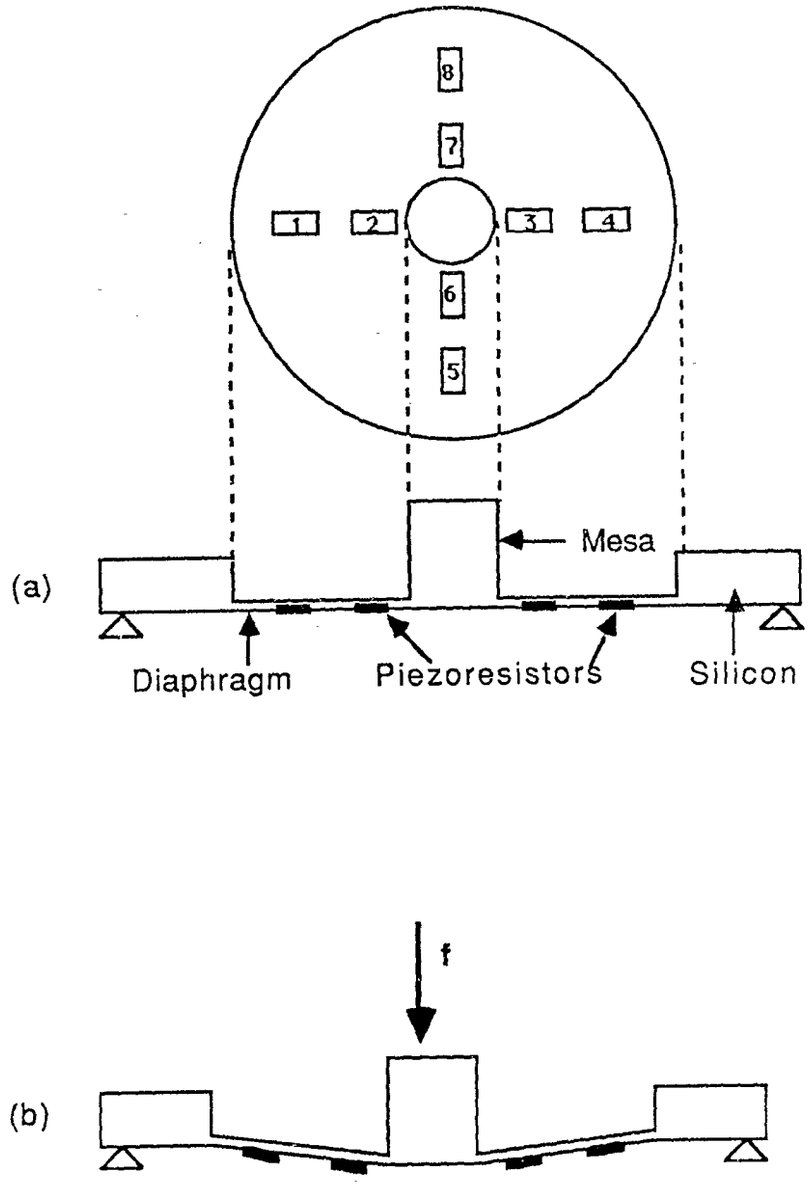


Figure 2.1: *(a) Schematic top and crossectional view of a single piezoresistive tactile sensing element. (b) Application of forces to the mesa deforms the diaphragm, causing changes in the resistance of the piezo resistors.*

(a) (b)

Figure 2.2: *(a) Rigid contact with an irregular surface results in contact at a number of discrete points only. (b) Compliant contact with an irregular surface*

array of sensors.

(3)The third function of the compliant layer is due to the resultant 'blurring' of the contact force distribution which causes information about the entire stress distribution to be passed to each individual sensor element. In the absence of such blurring, determination of the applied stress between sample points would not be possible.

We are now presented with the following scenario (see Figure 2.3(a)). Contact is established with an object resulting in a stress distribution at the surface of the compliant pad. Stress at the surface induces strain in the compliant material which is sampled by the sensors at a depth x beneath the surface. This leads to the following definition in words.

Definition 2.1 *Given samples of a strain distribution, measured by sensors at a given depth beneath the surface of a compliant layer, the inverse problem of tactile sensing refers to the problem of determining the surface stress distribution which induced the measured strain.*

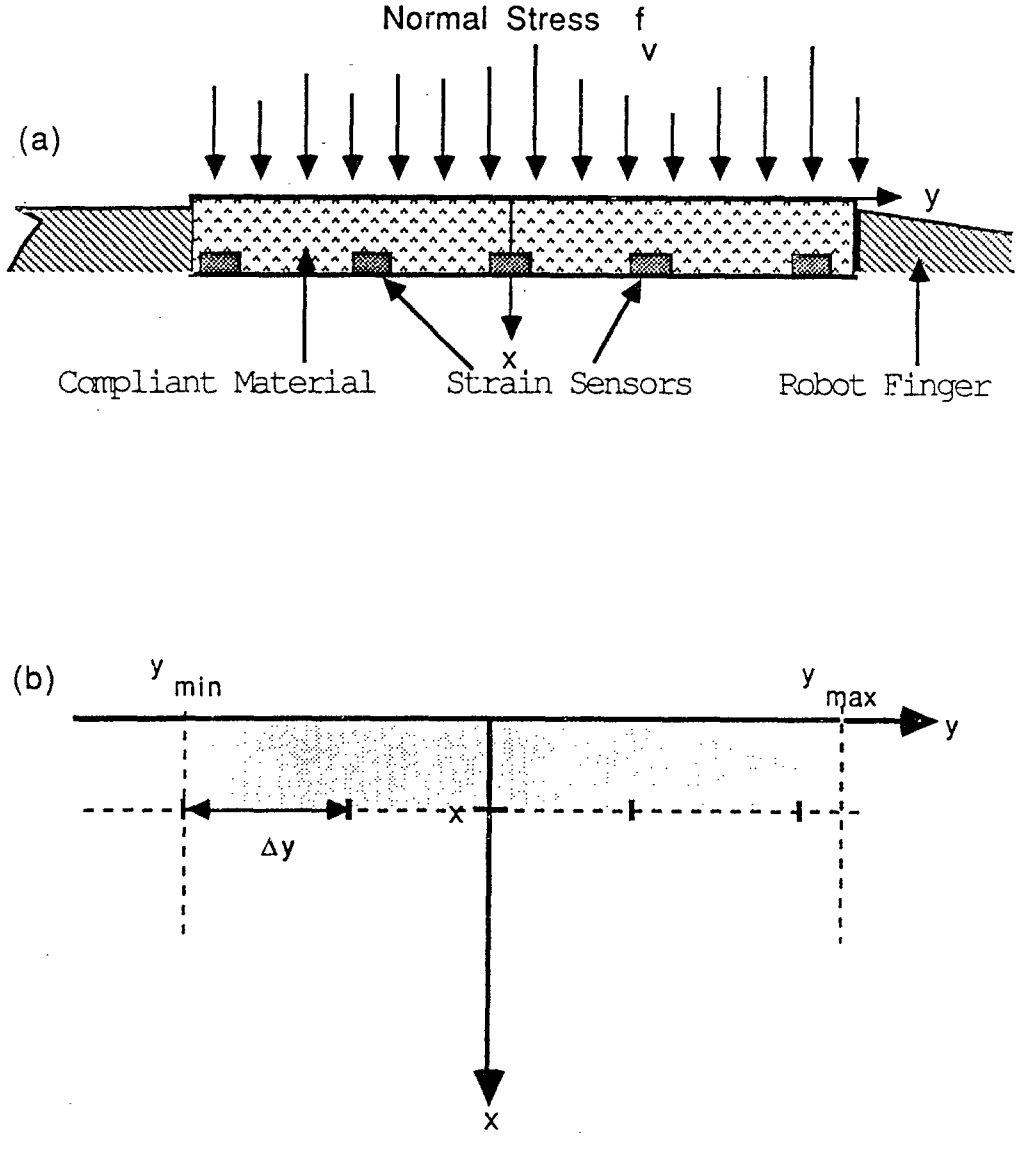


Figure 2.3: *(a) Stress applied at the boundary of elastic layer with strain sensors beneath surface. (b) Elastic layer extends from* y_{min} *to* y_{max} *with sensors at depth* x *beneath the surface at intervals of* Δy.

2.2 Modeling the Transduction Process

For sake of simplicity in the current discussion, two assumptions are made. (i) The general three dimensional problem is reduced to a two dimensional setting. i.e. we consider a linear array of sensors with planar stress applied to the pad. (ii) It is also assumed that the compliant layer is actually a homogeneous, isotropic, linear, elastic half-space.

To understand the inverse problem, a model for the forward transduction process is first developed i.e. a model describing the relationship between stress applied to the compliant layer and the strain induced at a depth x beneath the surface. From the theory of elasticity (see Timoshenko and Goodier [11]) two relationships (for detailed derivations see Yang [12]) can be derived..

Strain at a depth x beneath the surface of the compliant material due to normal stress at the surface is given by

$$p_x(y) = \int_{-\infty}^{\infty} k_x^v(y, y_0) q_v(y_0) dy_0 \tag{2.2.1}$$

where, $p_x(\cdot)$ is the strain at depth x, $q_v(\cdot)$ is the surface stress, and $k_x^v(\cdot,\cdot)$ is the convolution kernel relating the two given by,

$$k_x^v(y, y_0) = \frac{2x((1-\nu)^2 x^2 - \nu(\nu+1)(y-y_0)^2)}{\pi E(x^2 + (y-y_0)^2)^2} \tag{2.2.2}$$

where, ν is Poisson's ratio for the material and E is the modulus of elasticity. Strain due to tangential stress applied at the surface is given by,

$$p_x(y) = \int_{-\infty}^{\infty} k_x^t(y, y_0) q_t(y_0) dy_0, \tag{2.2.3}$$

where, $q_t(\cdot)$ is the tangential surface stress and

$$k_x^t(y, y_0) = \frac{2(y - y_0)((1-\nu)^2 x^2 - \nu(\nu+1)(y-y_0)^2)}{\pi E(x^2 + (y-y_0)^2)^2}. \tag{2.2.4}$$

For the sake of simplicity we only consider the case of normal stress throughout this thesis since the case with tangential stress is obviously analogous[1].

Since measurements of strain are made at a discrete number of points only, equation (2.2.1) must be discretized.

Assume that the sensors are distributed uniformly (equal spacing) beneath the surface of the compliant layer. Let $\Delta_p y$ be the distance between points at which strain is sampled. So,

$$\Delta_p y = \frac{y_{max} - y_{min}}{N-1}$$

(see Figure 2.3), where N is the total number of sensors. Let $\Delta_q y$ be the distance between points where the stress profile is to be reconstructed. Although it is not necessary for $\Delta_p y$ and $\Delta_q y$ to be equal, for the current discussion we let $\Delta_p y = \Delta_q y = \Delta y$. To obtain a discretized version of equation (2.2.1), let ϵ_x be the vector of strain samples i.e.,

$$\epsilon_x = (\epsilon_{x_1}, \ldots, \epsilon_{x_N})^T.$$

Therefore,

$$\epsilon_{x_i} = p_x(y_{min} + i\Delta y) \qquad i = 1, \ldots, N.$$

Similarly, let $f_v = (f_{v_1}, \ldots, f_{v_N})^T$ be the vector obtained from the stress distribution

[1]The analogy referred to here does not extend to the choice of regularizer for the case of tangential stress.

as,

$$f_{v_i} = q_v(y_{min} + i\Delta y) \qquad i = 1, \ldots, N.$$

The convolution kernel $k_x^v(\cdot,\cdot)$ can be discretized to form the matrix $T = \{T_{ij}\}$ by letting

$$T_{ij} = k_x^v(y_i, y_j),$$

where $y_i = y_{min} + i\Delta y$ for $i = 1, \ldots, N$. Hence the discretization of equation (2.2.1) results in,

$$\epsilon_x = T \cdot f_v. \tag{2.2.5}$$

The discretized inverse problem can now be stated more precisely as being the problem of determining f_v given ϵ_x and T. It will be shown in the next section that this inverse problem is ill-posed the sense defined by Hadamard and that in situations where the strain measurements ϵ_x are corrupted by noise, solutions to the problem are not 'well behaved'.

Returning to the third desirable feature of compliant contact, we observe that if we let T be a non-square matrix then in solving the inverse problem we are attempting to reconstruct the surface stress at points along the surface other than those directly above the sensors. This is only possible since the blurring of signals by the compliant layer results in information about most of the stress distribution at the surface being passed to every sensor. The exceptions to this are due to zeros in the convolution kernel.

2.3 Integral Equations of the First Kind

Equation (2.2.1) above is of a class more generally referred to as integral equations of the first kind. The general form of integral equations of the first kind is given by,

$$g(t) = \int_a^b k(t,s)f(s)\ ds \qquad c \leq t \leq d. \tag{2.3.6}$$

Equation (2.3.6) may be rewritten in operator notation as,

$$g(t) = (Kf)(t) \tag{2.3.7}$$

where K is the integral operator with kernel function k. In the particular case of equation (2.2.1) K is a convolution operator with the convolution kernel k. Since $g(\cdot)$ and $f(\cdot)$ in this case represent strain and stress respectively and are therefore signals with finite energy, we consider the case where $g, f \in L_2([a,b])$ with norm defined by,

$$||h||_2 = \left(\int_a^b |h(x)|^2\ dx\right)^{1/2} \tag{2.3.8}$$

and $a = y_{min}, b = y_{max}$. Thus we are interested in the case when, $k_x(y,y_0)$[2]$\in$ $L_2([a,b]\times[a,b])$ and

$$p_x(y) = (Kq_v)(y) = \int_a^b k_x(y,y_0)q_v(y_0)dy_0. \tag{2.3.9}$$

2.4 Ill-Posedness of the Inverse Problem

To demonstrate the ill-posed nature of the inverse problem, we would like to do the following,

[2]We will use $k_x(y,y_0)$ to denote $k_x^v(y,y_0)$ unless otherwise indicated

1. Show that in the infinite dimensional setting of equation (2.3.9) the inverse of the integral operator K is unbounded.

2. Verify that the finite matrix representation of K in equation (2.2.5) (obtained by discretization of the kernel function) is indeed a justifiable approximation of the operator K.

3. Show that the finite dimensional manifestation of the unboundedness of the inverse of K occurs in the ill-conditioning of the finite matrix representation of K.

In order to clarify latter discussion we present some definitions, notation, and theorems. Proofs of the theorems are to found in Gohberg and Goldberg [13] and/or Rudin[14].

Notation:

$H,\ H_i \quad i = 1, 2, \ldots$	Hilbert Spaces
$L(X)$	Bounded linear operators on X
$L(X, Y)$	Bounded linear operators from X to Y
$Sp\{\phi_1, \phi_2, \ldots\}$	Span of $\{\phi_1, \phi_2, \ldots\}$
$Im(A)$	Image of A
A^*	Adjoint of the operator A

Definition 2.2 *An operator $A \in L(H_1, H_2)$ is compact if for each sequence $\{x_n\}$ in H_1, such that $||x_n|| = 1$, the sequence $\{Ax_n\}$ has a subsequence which converges in H_2.*

Lemma 2.1 *If $A \in L(H_1, H_2)$ is of finite rank then A is compact.* ■

Theorem 2.1 *Let $\{A_n\}$ be a sequence of compact operators in $L(H_1, H_2)$ such that $||A_n - A|| \to 0$ as $n \to \infty$, where $A \in L(H_1, H_2)$. Then A is compact.* ■

To show that the integral operator K is bounded, we note that since $k \in L_2([a,b] \times [a,b])$,

$$\int_a^b \int_a^b |k(t,s)|^2 \, ds \, dt \; < \; \infty. \tag{2.4.10}$$

Now,

$$(Kf)(t) = \int_a^b k(t,s) f(s) \, ds. \tag{2.4.11}$$

So by Schwarz's inequality,

$$\int_a^b |k(t,s) f(s)| \, ds \leq (\int_a^b |k(t,s)|^2 \, ds)^{1/2} (\int_a^b |f(s)|^2 \, ds)^{1/2}. \tag{2.4.12}$$

Therefore,

$$||Kf||^2 \leq \int_a^b (\int_a^b |k(t,s) f(s)| \, ds)^2 \leq ||f||^2 \int_a^b \int_a^b |k(t,s)|^2 \, ds \, dt. \tag{2.4.13}$$

So,

$$||K||^2 \leq \int_a^b \int_a^b |k(t,s)|^2 \, ds \, dt \; < \infty. \tag{2.4.14}$$

Hence the operator K is bounded. K is clearly linear as well and so $K \in L(H)$ where $H = L_2([a,b])$. Since from equation (2.2.2) we see that $k(t,s) = k(s,t)$, the operator K is also self adjoint.

To show that K is a compact operator we construct a sequence of operators of finite rank and then use Theorem 2.1.

Let $\phi_1, \phi_2, \ldots$ be an orthonormal basis for $L_2([a,b])$. Then,

$$\Phi_{ij}(t,s) = \phi_i(t) \bar{\phi}_j(s) \qquad i,j = 1,2,\ldots$$

is an orthonormal basis for $L_2[a,b]\times[a,b]$) (see Gohberg and Goldberg [13]). Therefore,

$$k(t,s) = \sum_{i,j=1}^{\infty} < k, \Phi_{ij} > \Phi_{ij}(t,s). \tag{2.4.15}$$

Let

$$k_n(t,s) = \sum_{i,j=1}^{n} < k, \Phi_{ij} > \Phi_{ij}(t,s). \tag{2.4.16}$$

Then,

$$||k - k_n|| \to 0, \tag{2.4.17}$$

where $||\cdot||$ is the norm on $L_2([a,b]\times[a,b])$. Let K_n be the integral operator defined on $L_2([a,b])$ by,

$$(K_n f)(t) = \int_a^b k_n(t,s)f(s)\ ds. \tag{2.4.18}$$

So K_n is bounded, linear and of finite rank since $Im(K_n) \subset Sp\{\phi_1,\ldots,\phi_n\}$. Therefore by Lemma 1, K_n is also compact. Now,

$$||K|| \leq (\int_a^b \int_a^b |k(t,s)|^2\ ds\ dt)^{1/2} = ||k||. \tag{2.4.19}$$

Thus applying (2.4.17) and (2.4.19) to $K - K_n$, we get,

$$||K - K_n|| \leq ||k - k_n|| \to 0. \tag{2.4.20}$$

So by Theorem 2.1, K is compact.

To summarize, we have thus far shown that the integral operator K is bounded, linear, self adjoint and compact.

Although every linear operator on a finite dimensional Hilbert space over $\mathbb{C}$ has an eigenvalue, it is not true that even a self adjoint operator on an infinite

dimensional Hilbert space must have an eigenvalue. The next few theorems are concerned with the eigenvalues of such operators.

Theorem 2.2 *(a)Any eigenvalue of a self adjoint operator is real*

(b)If λ is an eigenvalue of $A \in L(H)$ then, $|\lambda| \leq ||A||$.

(c)If $A \in L(H)$ is compact and self adjoint then A has an eigenvalue and at least one of the numbers $||A||$ or $-||A||$ is an eigenvalue of A. ■

Theorem 2.3 *Let $A \in L(H)$ be a compact self adjoint operator, where H is an infinite dimensional Hilbert space. Then the spectrum $\sigma(A)$ of A consists of zero and the eigenvalues of A. $(0 \in \sigma(A))$* ■

Now, $(\lambda I - K)$ is invertible for $\lambda \notin \sigma(K)$. Since K is compact and self adjoint, by Theorem 2.3, $0 \in \sigma(K)$. Hence the inverse of $(\lambda I - K)$ is unbounded for $\lambda = 0$, which is equivalent to saying that the inverse of the integral operator K is unbounded.

From the above result, it is clear that in the infinite dimensional case the inverse problem is ill-posed in the sense of Hadamard since the inverse of the operator K is unbounded, solutions will not depend continuously upon the initial data.

2.5 Finite Matrix Approximation of the Convolution Kernel

In equation (2.2.5) the convolution operator K has been approximated by a finite rank operator by discretizing the kernel k and considering only samples of the stress and strain. In order to justify the use of such an approximation, we state the following theorem.

Theorem 2.4 *Every compact operator in $L(H_1, H_2)$ is the limit, in norm , of a sequence of operators of finite rank.* ∎

In fact, in Section 2.4 it was shown that K is the limit (in norm) of the sequence of finite rank operators $\{K_n\}$.

If $\{\phi_j\}$ is an orthonormal basis for $L_2([a,b])$ we know that

$$\Phi_{ij}(t,s) = \phi_i(t)\bar{\phi_j}(s) \qquad i,j = 1,2,\ldots \tag{2.5.21}$$

forms an orthonormal basis for $L_2([a,b]\times[a,b])$. The infinite matrix representation, $[a_{ij}]$, of the integral operator K with respect to the basis $\{\phi_j\}$ is given by,

$$a_{ij} =< K\phi_j, \phi_i >= \int_a^b \int_a^b k(t,s)\phi_j(s)\bar{\phi_i}(t)\, ds\, dt =< k, \Phi_{ji} > . \tag{2.5.22}$$

Since $0 \in \sigma(K)$ and $A = [a_{ij}]$ is unitarily equivalent to K, zero is also in the spectrum of A and thus the inverse of A is also not a bounded operator. Theorem 2.4 says that the operator K can be approximated arbritrarily well by an operator of finite rank, say $\widetilde{K_n}$ (of rank n) with finite matrix representation $\tilde{A}_n$. As $n \to \infty$ the approximation becomes better, however, since $0 \in \sigma(K)$, as $n \to \infty$ at least one of the eigenvalues of $\tilde{A}_n$ approaches zero. By Theorem 2(b), at least one of the eigenvalues of $\tilde{A}_n$ approaches either $||K||$ or $-||K||$.

The condition number $P(A)$ of a matrix A is defined as

$$P(A) = \frac{\max\limits_j |\lambda_j(A)|}{\min\limits_j |\lambda_j(A)|} \tag{2.5.23}$$

In solving a finite system of linear equations of the form $Ax = y$, the condition number is a measure of sensitivity of solutions to errors in the initial data y and

approximations made in the inversion of the matrix A (e.g. finite word length effects) on the solution x. Large values of $P(A)$ result in large errors in the solution. In the event that $P(A)$ is large ($P(A) = 1$ being the best case), we say that the matrix A is 'ill-conditioned'.

It is clear from the observations made earlier about the eigenvalues of $\tilde{A}_n$ that as $n \to \infty$, $\tilde{A}_n$ becomes increasingly ill-conditioned since the denominator in equation (2.5.23) approaches zero as the numerator approaches $||K||$.

To conclude this section, we summarize the results. We have shown that in the infinite dimensional case, the inverse problem is ill-posed in the sense of Hadamard. Ill-posedness of the problem in this case is due to the unboundedness of the inverse of the integral operator K which is in violation of Hadamard's third requirement that the solution depend continuously on the initial data for the problem to be well posed. We have justified approximating the operator K by a finite matrix since any compact operator in $L(H_1, H_2)$ can be approached (in norm) by a sequence of operators of finite rank. Lastly, we have made the observation that unboundedness of the inverse of K induces ill-conditioning of the finite matrix approximation to K.

2.6 Solving the Inverse Problem

It was shown in the last section that the problem of solving $Kx = y$, where K is an integral operator of the type defined in equation (2.3.9) is ill-posed. There exists a large body of literature devoted to approximating the solutions of ill-posed problems (see e.g. Tikhonov [15] and Tikhonov and Arsenin [16]). One successful

technique for solving ill-posed problem is regularization which was introduced by Tikhonov [15] in 1963. Ill-posed problems such as the one considered here are often insufficiently constrained and require the imposition of additional constraints for the solution to be well defined. Regularization is a technique in which the problem is formulated as a variational principle which is then used to impose physical constraints on the solution. A variational principle defines the solution to a problem as the function which minimizes an appropriate cost functional (Poggio and Koch [3]).

Regularization requires the choice of a norm $||\cdot||$ and of a stabilizing functional (typically of the form $||Px||$). The stabilizing functional embodies the physical constraints of the problem and thus must be chosen only after careful analysis of the physical setting in which the problem arises. Constraints such as smoothness and boundedness of solutions may be imposed by appropriate choice of the stabilizer.

The problem of solving $Kx = y$ can be formulated as a variational principle simply by choosing a norm $||\cdot||$ and the finding x which minimizes

$$||Kx - y||$$

To regularize the problem additional constraints are imposed through the stabilizing functional.

Standard regularization theory is composed primarily of three methods (see Poggio and Torre (1984)).

1. Among all x which satisfies the condition $||Px|| < c$, where c is a constant, find x which minimizes $||Kx - y||$.

2. Among all x which satisfies $||Kx - y|| < \epsilon$,where ϵ is chosen to represent estimated errors, find x which minimizes $||Px||$.

3. Find x which minimizes

$$||Kx - y||^2 + \lambda||Px||^2 \tag{2.6.24}$$

where λ is called the *regularization parameter.*

In standard regularization theory, the operator P is linear and the norm $||\cdot||$ is derived from an inner product. For such quadratic variational principles, of the form (2.6.24), it can be shown that under mild conditions the solution space is convex (which implies the existence, uniqueness and stability of solutions). In this thesis we will consider other forms of the stabilizing functional which we will denote by $M(x)$. Hence we will be considering variational principles of the from

$$||Kx - y||^2 + \lambda M(x) \tag{2.6.25}$$

The regularization parameter λ controls the degree to which a solution is regularized. Small values of λ compromise the degree of regularization in favor of accurately matching the initial data. Very large values of λ may result in very regular but unrealistic solutions.

2.7 Regularizing the Tactile Sensing Problem

To regularize the inverse problem of tactile sensing it is necessary to first identify the generic physical constraints that may be imposed upon the solution. In the case of normal stress applied to the the compliant pad (see Figure 2.3), it is clear

that the unisense nature of the compressive loading on the boundary can be captured by constraining solutions to lie in the positive orthant. To further suppress some of the deleterious effects of sensor noise, the solutions may be constrained to be smooth. Constraining solutions to be smooth may result in inaccurate solutions near physical edges, however, edges may be recovered in a second stage of regularization.

The constraints of nonnegativity and smoothness of the solutions can be embodied in the stabilizing functional by choosing

$$M(x) = \sum_i x_i \log x_i \tag{2.7.26}$$

which has the same functional form as Shannon entropy. Hence in the case of the inverse problem of tactile sensing, the problem is to find a vector $f \in \mathbb{R}^N$ which minimizes

$$||Tf_v - \epsilon_x||^2 + \lambda \sum_i f_{v_i} \log f_{v_i} \tag{2.7.27}$$

where $T \in \mathbb{R}^{N \times N}$ is a finite matrix approximation to the convolution operator, $\epsilon_x \in \mathbb{R}^N$ is the vector of measured strains and $f_v \in \mathbb{R}^N$ is the vector of stress components. (The norm $||\cdot||$ is the standard Euclidean norm on $\mathbb{R}^N$

Since the choice of $M(x)$ in (2.7.26) is not of a form for which there exists a large body of results regarding existence, uniqueness and stability of solutions, such properties must be verified.

2.7.1 Uniqueness of Solutions

Uniqueness of solutions to (2.7.27) is demonstrated in the following lemma.

Lemma 2.2 *Given $A \in \mathbb{R}^{n\times n}$ symmetric, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$,there exists at most one solution to the problem*

$$\min_{x \in \mathbb{R}^n} \quad ||Ax - y||^2 + \lambda \sum_i^n x_i \log x_i \tag{2.7.28}$$

Proof

Let

$$M(x) = ||Ax - y||^2 + \lambda \sum_i x_i \log x_i$$

Then,

$$\bigtriangledown M(x) = 2A^T(Ax - y) + \begin{pmatrix} \log x_1 \\ \vdots \\ \log x_n \end{pmatrix} + \mathbf{1} \tag{2.7.29}$$

Where $\mathbf{1} = (1, 1, \ldots, 1)^T$.

Assume there exists two solutions $\widehat{x}$ and $\widetilde{x}$ to (2.7.28). Then since at the solution $\bigtriangledown M(x) = 0$, we have

$$\bigtriangledown M(\widehat{x}) = 0 \quad \textit{and} \quad \bigtriangledown M(\widetilde{x}) = 0. \tag{2.7.30}$$

From (2.7.29) and (2.7.30),

$$2A^T(A\widehat{x} - y) + \begin{pmatrix} \log \widehat{x}_1 \\ \vdots \\ \log \widehat{x}_n \end{pmatrix} + \mathbf{1} = 0 \tag{2.7.31}$$

and

$$2A^T(A\tilde{x} - y) + \begin{pmatrix} \log \tilde{x}_1 \\ \vdots \\ \log \tilde{x}_n \end{pmatrix} + \mathbf{1} = 0 \tag{2.7.32}$$

Subtracting (2.7.32) from (2.7.31),

$$2A^T A(\hat{x} - \tilde{x}) = - \begin{pmatrix} \log \hat{x} - \log \tilde{x}_1 \\ \vdots \\ \log \hat{x}_n - \tilde{x}_n \end{pmatrix} \tag{2.7.33}$$

Taking the inner product of both sides of (2.7.33) with $(\hat{x} - \tilde{x})$,

$$(\hat{x} - \tilde{x}) \cdot 2A^T A(\hat{x} - \tilde{x}) = -(\hat{x} - \tilde{x}) \cdot \begin{pmatrix} \log \hat{x} - \log \tilde{x}_1 \\ \vdots \\ \log \hat{x}_n - \tilde{x}_n \end{pmatrix} \tag{2.7.34}$$

Since $A^T A$ *is positive semidefinite and symmetric and the* log *function is monotonically increasing, the left hand side of equation (2.7.34) is greater than zero while the right hand side is less than zero which is a contradiction.*

Hence it is necessary that $\hat{x} = \tilde{x}$ *i.e. there is at most one solution.* ■

CHAPTER
THREE

NONLINEAR ELECTRICAL NETWORKS

Nonlinear electrical networks have been a topic of active research for many years. Hence there exists a large body of results pertaining to such networks. As elaborated in Chapter 1, the transition from neural networks to electrical networks is not only trivial, but natural. Mathematical analysis of neural networks is as yet a developing field. It seems natural to look to the available results for nonlinear electrical networks for insight and understanding of the behavior of neural networks.

In this chapter an attempt is made to demonstrate one particular application of electrical network analysis to neural networks. We define an energy function for a dynamical system as a functional which is minimized (globally or locally) as a result of the natural time evolution of the system. In Chapter 4 the results of this chapter are applied to determine an energy function for a neural network designed to solve the inverse problem described in Chapter 2.

Some of the earliest work on electrical networks was done by James Clerk Maxwell in 1873 [18] and is concerned with the distribution of currents and voltages in linear resistive networks. Later work by Tellegen [19], Cherry (1951) [20], and Millar (1951) [21] helped to build a foundation for nonlinear network analysis. In 1964 Brayton and Moser [22] attempted to build a more general theory of

nonlinear networks by considering some geometric aspects of such networks. More recently, several researchers have adopted an even more general geometric view of nonlinear networks (see e.g. [22], [23], [24],[25], and [26]. In these latter works, network dynamics are viewed as flows (differential equations) on nontrivial manifolds (nonlinear spaces). It is clear that a great deal of powerful machinery has been developed for analysis of nonlinear networks. Applications of the same tools and body of results to neural networks should prove useful.

3.1 Basics of RC Electrical Networks

We consider a network composed of branches and nodes with the restriction that a branch connects exactly two nodes. Arbitrarily assigning a direction to the branch currents, we define i_μ as the current flowing from the initial node to the end node of the μth branch in the network. The branch voltage v_μ is defined as the voltage rise measured from the end node to the initial node of the μth branch in the network.

For any network, a complete set of generalized current or voltage coordinates can be chosen. Such a set of variables is complete in the sense that they can be assigned values independently without violation of Kirchoff's laws and that they determine in each branch of the network one of the two variables, branch current or branch voltage. In computing m, the number of defining current coordinates, constant current sources if any are not counted as branches and similarly in computing n, the number of defining voltage coordinates, constant voltage sources are not counted as branches. In the particular case of a RC network, a complete set of

variables is obtained by considering the voltages across all independent capacitive branches. Two capacitors in parallel are considered as a single capacitive branch with capacitance equal to the parallel combination of the two. We will denote the complete set of variables for a RC network by $v* = (v_1, \ldots, v_n)$.

In the theory of electrical circuits there are two fundamental laws, namely Kirchoff's laws which form the basis for analysis of any electrical circuit. Kirchoff's current or node law states that the sum of currents at any node in a network must be zero, i.e.,

$$\sum_{node} i = 0.$$

Kirchoff's voltage or loop law states that the sum of voltages around any loop in the network must be zero, i.e.

$$\sum_{loop} v = 0.$$

Among the first theorems for linear electrical networks is Maxwell's Minimum Heat Theorem (1837) which is stated below in modern terminology.

Theorem 3.1 (Maxwell's Minimum Heat Theorem) *For any linear, time-invariant, resistive network driven by voltage and/or current sources, of all the current distributions consistent with Kirchoff's current law, the only distribution which is also consistent with Kirchoff's voltage law and therefore the true distribution is the one which minimizes the quantity* $(W - 2P_v)$ *where* W *is the total power dissipated by the resistors and* P_v *is the total power supplied by the voltage sources.*

■

Having chosen a set of generalized current coordinates say for example $i_1, \ldots, i_m$,

Maxwell's theorem leads to the following set of m equations,

$$\frac{\partial}{\partial i_j}(W - 2P_v) = 0 \qquad j = 1, \ldots, m \tag{3.1.1}$$

which when solved for the current coordinates $i_1, \ldots, i_m$ determine all variables in the network.

The dual of Maxwell's theorem states that for the above network, among all voltage distributions which are consistent with Kirchoff's voltage law, the only one which is also consistent with Kirchoff's current law and therefore the true distribution is the one which minimizes $(W - 2P_I)$, where P_I is the total power supplied by the current sources. This dual theorem can be used to solve for all variable in the network by first solving for the generalized voltage coordinates.

Maxwell's theorem tells us that the quantities $(W - 2P_v)$ and $(W - 2P_I)$ are stationary with respect to the distributions of currents and voltages in the network respectively. It can be shown that these quantities are not in general stationary in the case of networks containing nonlinear elements. Hence to solve for voltages and currents in a nonlinear network in an analogous manner we must determine first the stationary quantities. Millar [20] determined such quantities and we present first some definitions relating to nonlinear networks before we state his results.

Definitions Pertaining To Nonlinear Networks

Elements: Two terminal elements can in general be described by a relationship of the form,

$$f(i, \frac{di}{dt}, \frac{d^2i}{dt^2}, \ldots, v, \frac{dv}{dt}, \frac{d^2v}{dt^2}, \ldots, t) = 0 \tag{3.1.2}$$

Where v is the voltage across the two terminals of the element and i is the current

through the element.

Nonreactive elements are those for which the dependence on time derivatives in equation (3.1.2) is absent, i.e. they can be described by $f(i, v, t) = 0$.

Time invariant elements are those for which the defining function is not explicitly time dependent

For time invariant nonreactive elements the locus of $f(i, v) = 0$ is called the *characteristic curve* of the element. For general time invariant elements, the i-v curve represents a trajectory in the phase space of the element viewed as a dynamical system.

A time invariant nonreactive element is said to be *passive* if the characteristic curve intersects the i-v axes at no point other than the origin. Otherwise the element is said to be *active.*

Sources: A current/voltage source is a time invariant active nonreactive element for which voltage/current is absent from the defining function $f(i, v)$.

3.2 Invariants of Motion in Nonlinear Electrical Networks

Two quantities defined by Millar [20] relating to elements of a nonlinear network are the 'content' and the 'co-content' of an element.

3.2.1 Content and Co-content

Let (i_1, v_1) be a point on the $i - v$ curve of a two terminal element. The 'content' of the element, denoted by G is defined by,

$$G = \int_0^{i_1} v \, di. \tag{3.2.3}$$

The 'co-content' of the element is denoted by J and is defined by,

$$J = \int_0^{v_1} i \, dv \tag{3.2.4}$$

We observe that for a passive element the total power dissipation is $W = J + G$ and that for a linear, passive element $J = G = W/2$

The total content (co-content) of a network is defined as the sum of contents (co-contents) of all constituent elements including current and voltage sources. We will use G and J to denote the total content and co-content respectively. We will use G_{jk} and J_{jk} to denote the content and co-content of the element connecting nodes j and k.

3.2.2 Stationary Quantities

The following theorem identifies G and J as stationary quantities.

Theorem 3.2 (Millar) *If in an active (possibly reactive) network, the sum J of the co-contents of all the constituent elements is expressed in terms of the defining number of voltage coordinates of the network subject only to the restrictions of Kirchoff's voltage law, then J is stationary for the actual distribution of voltages.*

■

The dual of this theorem is obtained by replacing co-content by content, J by G, and voltage by current everywhere.

Theorem 3.2 provides us with the equivalent of Maxwell's theorem for nonlinear networks of time invariant elements. To restate the above theorem in a manner analogous to Maxwell's theorem, we can say;

In any active (possibly reactive) network, of all distributions of current/voltage consistent with Kirchoff's current/voltage law, the ones for which G/J is stationary are the only ones that are also consistent with Kirchoff's voltage/current law and are thereby the true distributions.

Thus if we express G/J in terms of the defining number (m/n) of current/voltage coordinates, we can determine all currents and voltages in the network by solving either one of the following sets of simultaneous partial differential equations,

$$\frac{\partial G}{\partial i_r} = 0 \qquad r = 1, \ldots, m \ , \tag{3.2.5}$$

or

$$\frac{\partial J}{\partial v_q} = 0 \qquad q = 1, \ldots, n \ . \tag{3.2.6}$$

3.2.3 Invariants of Motion

The next theorem identifies G and J as invariants of motion. That is, as a network evolves in time following an impulsive change in one or more of the current or voltage sources, the total content and the total co-content of the network are conserved.

Theorem 3.3 (Millar) *In any network of time invariant elements (possibly including sources), the total content G and the total co-content J are invariants of motion. i.e.*

$$\frac{dG}{dt} = 0$$

and

$$\frac{dJ}{dt} = 0 \tag{3.2.7}$$

Where $\frac{dG}{dt}$ *and* $\frac{dJ}{dt}$ *are the total time derivatives of* G *and* J *given by,*

$$\frac{dG}{dt} = \frac{\partial G}{\partial t} + \frac{\partial G}{\partial i_1}\frac{di_1}{dt} + \cdots + \frac{\partial G}{\partial i_m}\frac{di_m}{dt}$$

and

$$\frac{dJ}{dt} = \frac{\partial J}{\partial t} + \frac{\partial J}{\partial v_1}\frac{dv_1}{dt} + \cdots + \frac{\partial J}{\partial v_n}\frac{dv_n}{dt} \tag{3.2.8}$$

where m *is the defining number of generalized current coordinates and* n *is the defining number of generalized voltage coordinates.* ■

3.3 Energy Functions for Nonlinear RC Electrical Networks

To derive an energy function for nonlinear networks of the type described we first restate and prove two theorems. The first of these theorems is known as Tellegen's Theorem [18], but is presented as in [21] so as to have a geometrical interpretation. The second theorem is stated by Brayton and Moser in [21].

In a directed network with b branches and m nodes, the set of branch currents $i = (i_1, ..., i_b)$ and the set of branch voltages $v = (v_1, ..., v_b)$ are vectors in a b-dimensional Euclidean vector space $\mathcal{E}^b$ with the inner product defined by $< x, y >= \sum_{\mu=1}^{b} x_\mu y_\mu$. Let $\mathcal{I}$ be the set of all vectors in $\mathcal{E}^b$ such that if $i \in \mathcal{I}$ then the constraint of Kirchoff's current law is satisfied for each node in the network, i.e. $\sum_{node} i_\mu = 0$. Similarly, let $\mathcal{V}$ be the set of all vectors in $\mathcal{E}^b$ such that if $v \in \mathcal{V}$ then Kirchoff's voltage law is satisfied, i.e $\sum_{loop} v_\mu = 0$. $\mathcal{I}$ and $\mathcal{V}$ are clearly subspaces of $\mathcal{E}^b$ since they are defined via linear relationships (Kirchoff's Laws). The following theorem shows that $\mathcal{I}$ and $\mathcal{V}$ are orthogonal subspaces of $\mathcal{E}^b$.

Theorem 3.4 (Tellegen's Theorem) *If $i \in \mathcal{I}$ and $v \in \mathcal{V}$ then*

$$< i, v >= 0. \tag{3.3.9}$$

Proof:

Let $(V_1, ..., V_n)$ be the set of node voltages such that v is the vector of voltage differences between the end nodes and the initial nodes of each branch. let i_{kl} be the current from node k to node l. Then if the branch connecting node k to node l is labeled as the μth branch,

$$v_\mu i_\mu = (V_k - V_l)i_{kl} = (V_l - V_k)i_{lk} \tag{3.3.10}$$

Due to the symmetry in k and l,

$$\begin{aligned} \sum_{\mu=1}^{b} v_\mu i_\mu &= \frac{1}{2}\sum_k \sum_l (V_k - V_l)i_{kl} \\ &= \frac{1}{2}(\sum_k V_k \sum_l i_{kl} - \sum_l V_l \sum_k i_{kl}) \end{aligned}$$

but,

$$\sum_l i_{kl} = \sum_{\text{node } k} \pm i_\mu$$

and

$$\sum_k i_{kl} = \sum_{\text{node } l} \pm i_\mu.$$

Since the branch currents in the network must satisfy Kirchoff's current law we know $\sum_{node} \pm i_\mu = 0$. Therefore $\sum_{\mu=1}^{b} v_\mu i_\mu = 0$ ■

Theorem 3.5 (Brayton–Moser) *Let Γ denote a one dimensional curve in $\mathcal{I} \times \mathcal{V}$ with coordinates denoted by i and v. Then,*

$$\int_\Gamma \sum_{\mu=1}^{b} v_\mu di_\mu = \int_\Gamma \sum_{\mu=1}^{b} i_\mu dv_\mu = 0 \tag{3.3.11}$$

Proof: Since $(di_1, \ldots, di_b) \in \mathcal{I}$,

$$< v, di >= \sum_{\mu=1}^{b} v_\mu di_\mu = 0 \quad \text{(By Tellegen's theorem).} \tag{3.3.12}$$

Integrating the above sum along Γ we get,

$$\int_\Gamma \sum_{\mu=1}^{b} v_\mu di_\mu = 0.$$

Integrating by parts we see that,

$$< v, i > \Big|_\Gamma - \int_\Gamma \sum_{\mu=1}^{b} i_\mu dv_\mu = 0. \tag{3.3.13}$$

By Tellegen's theorem we know $< v, i >= 0$, therefore,

$$\int_\Gamma \sum_{\mu=1}^{b} i_\mu dv_\mu = 0. \quad \blacksquare$$

Having stated the above two theorems and having defined the complete set of variables $v* = (v_1, \ldots, v_N)$ for the network, we proceed with the derivation of an energy function.

From Theorem 3.5 we know that $\int_\Gamma \sum_{\mu=1}^{b} v_\mu di_\mu = 0$. We choose Γ from a fixed initial point to a variable end point in $\mathcal{E}^b$ such that along Γ the characteristic relationships of the constituent elements of the network are satisfied. We can write equation (3.3.13) in the following form:

$$\int_\Gamma \sum_{\rho=1}^{N} v_\rho di_\rho + \int_\Gamma \sum_{\mu=N+1}^{b} v_\mu di_\mu = 0 \tag{3.3.14}$$

The first integral is over all capacitive branches and the second is over all other branches. Integrating the first integral by parts we get,

$$P + \int_{\Gamma} \sum_{\rho=1}^{N} i_\rho dv_\rho = 0, \tag{3.3.15}$$

where ,

$$P = -\left[\int_{\Gamma} \sum_{\mu=N+1}^{b} v_\mu di_\mu + \sum_{\rho=1}^{N} i_\rho v_\rho \Big|_{\Gamma} \right]. \tag{3.3.16}$$

Therefore from (3.3.15) we also have,

$$P = -\int_{\Gamma} \sum_{\rho=1}^{N} i_\rho dv_\rho. \tag{3.3.17}$$

Let $\xi = \sum_{\rho=1}^{N} i_\rho dv_\rho$. We now establish the conditions for ξ to be integrable i.e the conditions for the line integral (3.3.17) defining P to be independent of the path of integration Γ.

For (3.3.17) to be independent of Γ, it is necessary that ξ be a perfect differential. That is, we can write ξ as, $\xi = d\sigma$, where $\sigma = \sigma(v_1, \ldots, v_N)$ and,

$$d\sigma = \frac{\partial \sigma}{\partial v_1} dv_1 + \cdots + \frac{\partial \sigma}{\partial v_N} dv_N. \tag{3.3.18}$$

But since $\xi = d\sigma$ we have,

$$d\sigma = i_1 dv_1 + \cdots + i_N dv_N. \tag{3.3.19}$$

Comparing (3.3.18) and (3.3.19) we get,

$$i_\rho = \frac{\partial \sigma}{\partial v_\rho} \qquad \rho = 1, \ldots, N. \tag{3.3.20}$$

Therefore

$$\frac{\partial i_\rho}{\partial v_\eta} = \frac{\partial^2 \sigma}{\partial v_\rho \partial v_\eta} = \frac{\partial i_\eta}{\partial v_\rho} \quad \eta, \rho = 1, \ldots, N. \tag{3.3.21}$$

From the above equation (3.3.21) we see that in order for (3.3.17) to be independent of Γ, the following must hold:

$$\frac{\partial i_k}{\partial v_j} = \frac{\partial i_j}{\partial v_k} \qquad j,k = 1,\ldots,N \quad j \neq k. \tag{3.3.22}$$

If (3.3.22) holds, then P is a function of the endpoints of Γ alone.

Assuming that the line integral (3.3.17) is independent of path we can write for the currents in the capacitive branches,

$$i_\rho = -\frac{\partial P}{\partial v_\rho} \qquad \rho = 1,\ldots,N. \tag{3.3.23}$$

However, we know from the the dynamical law of capacitors that,

$$i_\rho = C_\rho \frac{dv_\rho}{dt} \qquad \rho = 1,\ldots,N. \tag{3.3.24}$$

Therefore,

$$C_\rho \frac{dv_\rho}{dt} = -\frac{\partial P(v*)}{\partial v_\rho} \quad \rho = 1,\ldots,N \tag{3.3.25}$$

where $v* = (v_1,\ldots,v_N)$. We now write the system of differential equations defining the dynamical behavior of the network in the following vector form,

$$-C\dot{x} = \frac{\partial P(x)}{\partial x} \tag{3.3.26}$$

where $x = v*$, $C = diag(C_1,...,C_N)$, and $\frac{\partial P(x)}{\partial x}$ is the gradient of $P(x)$. The following theorem relates $P(x)$ to the equilibrium states of the network.

Theorem 3.6 *Let*

$$\frac{dx}{dt} = -B\frac{\partial H(x)}{\partial x} \qquad x \in U \qquad H : U \to \mathbb{R} \tag{3.3.27}$$

where U is a normed vector space and B is a symmetric positive definite, $n \times n$ matrix function of x. Then $\frac{d}{dt}H(x(t)) \leq 0$ for all $x \in U$ and $\frac{d}{dt}H(x(t)) = 0$ if and only if x is an equilibrium of the gradient system (3.3.27).

Proof:

Taking the time derivative of H along trajectories,

$$\begin{aligned} \frac{dH(x)}{dt} &= (\frac{\partial H(x)}{\partial x}, \frac{dx}{dt}) \\ &= (\frac{\partial H(x)}{\partial x}, -B\frac{\partial H(x)}{\partial x}) \\ &= -\left\| B^{1/2}\frac{\partial H(x)}{\partial x} \right\|^2 \end{aligned}$$

Since B is positive definite $B^{1/2}$ is nonsingular and this completes the proof. ∎

Corollary. If $\tilde{x}$ is an isolated minimum of $H(x)$ then $\tilde{x}$ is an asymptotically stable equilibrium of the gradient system (3.3.27).

As the matrix C is positive definite, it is obvious that the function $P(x)$ which we shall call the energy function, decreases along solutions to (3.3.26) except at equilibrium points. Therefore the equilibrium states of the network correspond to stationary points of $P(x)$ and the local minima of $P(x)$ are the stable equilibria. If in addition to this $P(x) \rightarrow \infty$ as $||x|| \rightarrow \infty$ then it can be shown that all solutions to (3.3.26) approach one of the set of equilibrium solutions as $t \rightarrow \infty$.

We observe that $P(v*)$ is just the negative of the co-content of *all independent capacitive branches* in the network. The total co-content of the network, J, is an invariant of motion (Millar [20]) even for dissipative systems where the total energy is not conserved. The analogy to kinetic and potential energy of a nondissipative

CHAPTER
FOUR

NEURAL NETWORKS FOR SOLVING THE INVERSE PROBLEM

In Chapter 1 it was noted that, in all but the most rudimentary applications of tactile sensing, real-time processing of tactile sensory data is crucial. In Chapter 2 it was shown that the inverse problem of tactile sensing may be formulated as a variational principle. Furthermore, the inclusion of a regularizing penalty functional in the variational principle provides immunity to sensor noise and transforms the problem to one that is well-posed. Solutions to the inverse problem (obtained as solutions to a variational principle, require the minimization of the nonlinear cost functional in equation (2.7.27). Real-time solutions to such a minimization problem would require fairly powerful digital hardware using standard iterative algorithms. Since it is our objective to obtain fast solutions to the inverse problem as well as limit the hardware complexity to that which can be located in close proximity to individual sensor arrays, analog neural networks provide an attractive alternative.

In Section 4.1 an analog neural network is described which has been structured so as to solve the regularized inverse problem of tactile sensing. The network described also exploits any inherent parallelism in the problem since solutions are obtained as the result of asynchronous relaxation of all variables in the network.

In Section 4.2 it is shown that the network described in Section 4.1 satisfies the hypotheses of Theorem 3.7 and thus an energy function for the network is explicitly given by the theorem. It is shown that the resultant energy function is indeed the variational principle of equation (2.7.27).

Computer simulations of the network are used in Section 4.3 to evaluate the effect of the regularizing parameter λ on the resultant solutions and to evaluate performance of the network in the presence of sensor noise.

4.1 The Maximum Entropy Deconvolution Network

Inspired by the work of Tank and Hopfield (see [27]) on neural networks for solving optimization problems, Marrian and Peckerar [28] suggested a neural network for solving deconvolution problems with an entropy-like regularizer. Figure 4.1 shows a schematic of a two channel deconvolution network. The network proposed consists of two planes of amplifiers (i) the signal plane and (ii) the constraint plane (see Figure 4.1). Inputs to the constraint plane ($\epsilon = (\epsilon_1, \ldots, \epsilon_N)^T$) are currents proportional to sample strain measurements obtained from an array of tactile sensors. Outputs of the signal plane ($v = (v_1, \ldots, v_N)^T$) are voltages which, in equilibrium, represent regularized solutions to the inverse problem. The interconnections T_{ij} are conductances (resistors with values $1/T_{ij}$) corresponding to elements of the matrix representation T of the discretized convolution kernel. Amplifiers in the signal plane are exponential i.e. $g(x) = \exp(x)$ and constraint plane amplifiers are linear with gain s ($f(x) = sx$). To intuitively understand the manner in which this deconvolution network operates, dynamical evolution of the network can be

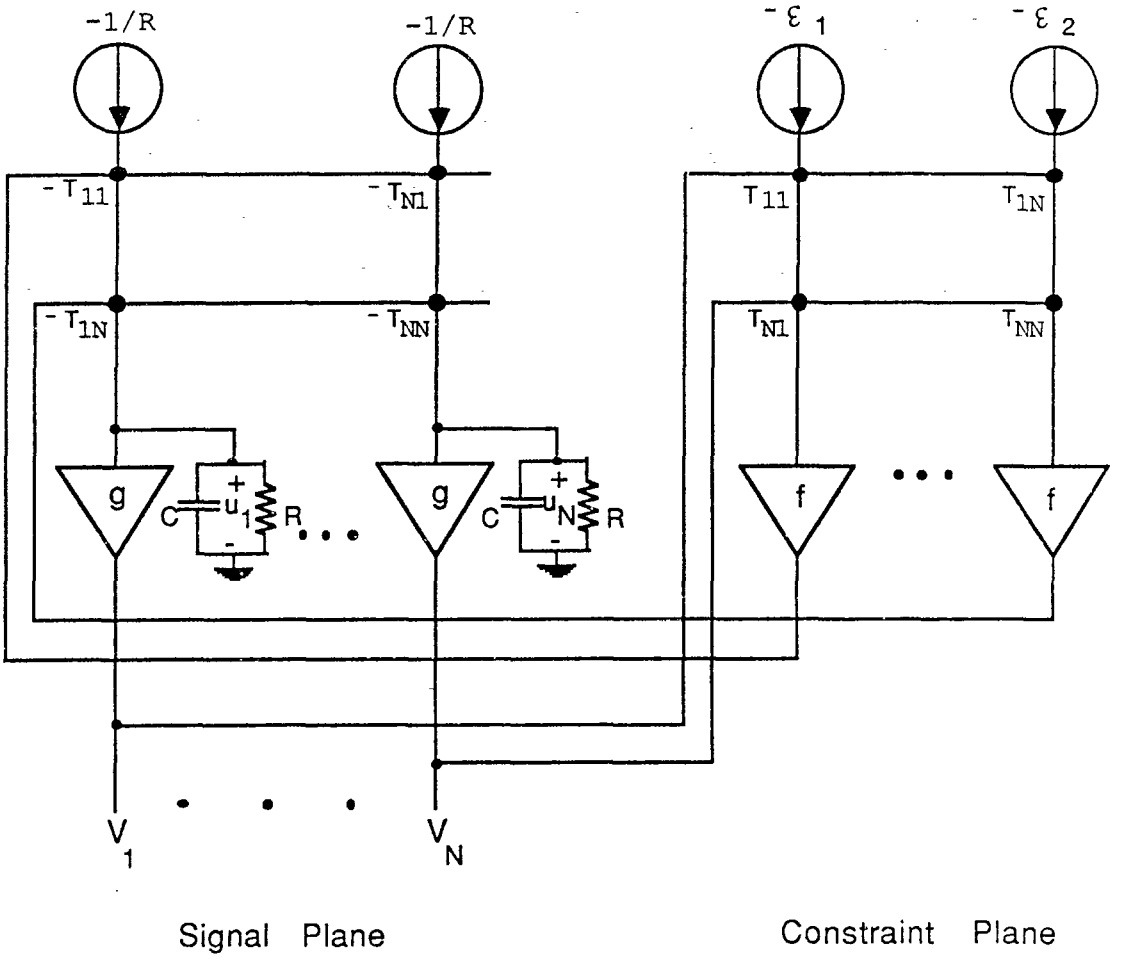


Figure 4.1: *N-Channel Maximum Entropy Deconvolution Network.*

viewed as occuring in a series of infinitesimal discrete time steps. If evolution of the network is viewed in this manner, the feedback loops generate a number of 'analog iterations'. Each analog iteration is approximately composed of the following steps:

1. Outputs of the signal plane are convolved with the discretized kernel T in the constraint plane, forming the vector $T \cdot v$.

2. Error in the current estimate of the solution (given by outputs of the signal plane) is evaluated in the constraint plane by subtracting the input strain vector ϵ from the results of the previous step i.e the vector $T \cdot v - \epsilon$ is formed and fed back to the signal plane through the constraint plane amplifiers with gain s.

3. Based upon feedback from the constraint plane, the outputs of the signal plane are updated so as to reduce the error.

4. If outputs of the signal plane have not yet settled repeat (1)–(3)

It remains to be shown that the above series of 'iterations' do indeed converge, i.e. that the network, as an analog electrical circuit, is stable. In the next section the results of Chapter 3 are used to demonstrate the stability of the network and to show that the outputs of the signal plane converge to regularized solutions to the inverse problem. Since (as will be shown in the next section) minimization of equation (2.7.27 also attempts to maximize the entropy of the solution, we will refer to the network in Figure 4.1 as the *maximum entropy deconvolution network* (or MaxEnt network for short).

4.2 Energy Function for Maximum Entropy Deconvolution Network

Stability of the MaxEnt deconvolution network can be established by applying Theorem 3.7 which also determines explicitly an energy function for the network. In the following it is assumed that for the MaxEnt network, any dynamics associated with the constraint plane amplifiers are negligible. This assumption is reasonable since the feedback capacitor associated with the signal plane amplifiers can be chosen so that the response of the signal plane amplifiers is sufficiently slower than those of the constraint plane. In order to apply Theorem 3.7 it is necessary to first verify that the network satisfies the two hypotheses. From Figure 4.1 it is clear that the voltages across the signal plane capacitors $u_1, \ldots, u_N$ form a complete set of variables for the network, i.e $u_1, \ldots, u_N$ can be assigned values arbitrarily and that they determine in every branch of the network one of the two variables, branch current or node voltage. It is intuitively obvious that a 'potential' function should depend only upon the current state of the network and the fixed 'zero' reference potential chosen. To establish path independence of the integral in equation (3.3.17) (i.e. to establish that (3.3.17) depends only upon the endpoints of Γ) the 'mixed-partials' condition of (3.3.22) is used.

From Figure 4.1, the current through the capacitor connected to the nth signal plane node is given by,

$$i_n = C\frac{du_n}{dt} = -\frac{u_n}{R} - \frac{1}{R} - \sum_k t_{kn} f(T_k \cdot v - \epsilon_k). \tag{4.2.1}$$

Here $T_k = (t_{k1}, t_{k2}, \ldots, t_{kN})^T$. Since $u_n = g^{-1}(v_n)$, the current i_n can also be

written as, [1],

$$i_n = C\frac{du_n}{dt} = -\frac{g^{-1}(v_n)}{R} - \frac{1}{R} - \sum_k t_{kn} f(T_k \cdot v - \epsilon_k). \tag{4.2.2}$$

The energy function P is thus a function of $v = (v_1, \ldots, v_N)^T$ i.e. v can be used as an alternate set of independent variables and

$$P(v) = -\int_\Gamma \sum_n i_n dv_n. \tag{4.2.3}$$

From (4.2.2) it is seen that,

$$\begin{aligned} \frac{\partial i_n}{\partial v_j} &= \frac{\partial}{\partial v_j}\left(-\textstyle\sum_k t_{kn} f(T_k \cdot v - \epsilon_k)\right) \\ &= -\textstyle\sum_k t_{kn} t_{kj} f'(T_k.v - \epsilon_k) \\ &\qquad k, j, n = 1, \ldots, N, \quad j \neq n \end{aligned} \tag{4.2.4}$$

and,

$$\begin{aligned} \frac{\partial i_j}{\partial v_n} &= \frac{\partial}{\partial v_n}\left(-\textstyle\sum_k t_{kj} f(T_k \cdot v - \epsilon_k)\right) \\ &= -\textstyle\sum_k t_{kj} t_{kn} f'(T_k.v - \epsilon_k) \\ &= \frac{\partial i_n}{\partial v_j} \qquad k, j, n = 1, \ldots, N, \quad j \neq n. \end{aligned} \tag{4.2.5}$$

Thus the mixed partials conditions of (3.3.22) are satisfied.

The dynamical equations of the network can now be written in the following form:

$$\frac{du}{dt} = -C^{-1}\frac{\partial P(v(u))}{\partial v(u)} \tag{4.2.6}$$

where $v = (v_1 \ldots, v_N)$ and $v_n = g^{-1}(u_n)$. Since $v_i = g(u_i)$,

$$\frac{dv}{dt} = -GC^{-1}\frac{\partial P(v)}{\partial v}, \tag{4.2.7}$$

[1]It has been assumed here that the amplifier function $g(\cdot)$ is bijective which is clearly true for $g(x) = \exp(x)$.

where $G = diag(g'(u_1), \ldots, g'(u_N))$. Because $g(u) = \exp(u)$, is a monotone increasing function G is always positive definite.

Applying Theorem 3.6 to (4.2.7), it is clear that the equilibrium states of the network must correspond to stationary points of $P(\cdot)$.

In order to understand the nature of the equilibrium states (if any exist) of the network we must evaluate the integral expression for P.

$$P(v) = \int_\Gamma \sum_n \left(\frac{g^{-1}(v_n)}{R} + \frac{1}{R} + \sum_k t_{kn} f(T_k \cdot v - \epsilon_k) \right) dv_n \tag{4.2.8}$$

Choosing 0 as the starting point of the path Γ, we can write (4.2.8) as,

$$P(v) = \sum_n \int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n + \sum_n \int_0^{v_n} \frac{dv_n}{R}$$
$$+ \int_\Gamma \sum_n \sum_k t_{kn} f(T_k \cdot v - \epsilon_k) dv_n. \tag{4.2.9}$$

If we let $F(z_k)$ be such that $dF(z_k)/dz_k = f(z_k)$ then,

$$P(v) = \sum_n \int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n + \sum_n \frac{v_n}{R} + \sum_k F(T_k \cdot v - \epsilon_k). \tag{4.2.10}$$

For the deconvolution network shown in Figure 4.1 the signal plane amplifiers are characterized by $g(u) = \exp(u)$. Therefore

$$\int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n = v_n \log v_n - v_n. \tag{4.2.11}$$

From Figure 4.1 we also see that the characteristics of the constraint plane amplifiers are given by $f(z) = sz$ where s is a constant defining the feedback gain. We can thus define the function $F(T_k \cdot v - \epsilon_k)$ by,

$$F(T_k \cdot v - \epsilon_k) = \frac{s}{2}(T_k \cdot v - \epsilon_k)^2. \tag{4.2.12}$$

where $G = diag(g'(u_1), \ldots, g'(u_N))$. Because $g(u) = \exp(u)$, is a monotone increasing function G is always positive definite.

Applying Theorem 3.6 to (4.2.7), it is clear that the equilibrium states of the network must correspond to stationary points of $P(\cdot)$.

In order to understand the nature of the equilibrium states (if any exist) of the network we must evaluate the integral expression for P.

$$P(v) = \int_\Gamma \sum_n \left(\frac{g^{-1}(v_n)}{R} + \frac{1}{R} + \sum_k t_{kn} f(T_k \cdot v - \epsilon_k) \right) dv_n \tag{4.2.8}$$

Choosing 0 as the starting point of the path Γ, we can write (4.2.8) as,

$$\begin{aligned} P(v) = & \sum_n \int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n + \sum_n \int_0^{v_n} \frac{dv_n}{R} \\ & + \int_\Gamma \sum_n \sum_k t_{kn} f(T_k \cdot v - \epsilon_k) dv_n. \end{aligned} \tag{4.2.9}$$

If we let $F(z_k)$ be such that $dF(z_k)/dz_k = f(z_k)$ then,

$$P(v) = \sum_n \int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n + \sum_n \frac{v_n}{R} + \sum_k F(T_k \cdot v - \epsilon_k). \tag{4.2.10}$$

For the deconvolution network shown in Figure 4.1 the signal plane amplifiers are characterized by $g(u) = \exp(u)$. Therefore

$$\int_0^{v_n} \frac{g^{-1}(v_n)}{R} dv_n = v_n \log v_n - v_n. \tag{4.2.11}$$

From Figure 4.1 we also see that the characteristics of the constraint plane amplifiers are given by $f(z) = sz$ where s is a constant defining the feedback gain. We can thus define the function $F(T_k \cdot v - \epsilon_k)$ by,

$$F(T_k \cdot v - \epsilon_k) = \frac{s}{2}(T_k \cdot v - \epsilon_k)^2. \tag{4.2.12}$$

From (4.2.10) and (4.2.11),

$$P(v) = \sum_k \frac{s}{2}(T_k \cdot v - \epsilon_k)^2 + \frac{1}{R}\sum_n v_n \log v_n. \tag{4.2.13}$$

It is clear from (4.2.13) that $P(v) \rightarrow \infty$ as $||v|| \rightarrow \infty$. Hence all solutions to (4.2.7) approach one of the set of equilibrium states as $t \rightarrow \infty$. In Chapter 2 it was shown that there exists a unique minimum of (4.2.13) which correspond to the regularized solution of the inverse problem. Therefore outputs of the network will converge to a regularized solution of the inverse problem.

Equation (4.2.13) gives us an explicit form for the energy function for the maximum entropy deconvolution network which has been derived using nothing more than Kirchoff's laws.

4.2.1 Introduction of other Regularizers

As discussed in Chapter 2, the choice of a regularizing principle must be based upon the physical constraints present in the problem. Also in Chapter 2 it was shown that the entropy regularizer is appropriate for the recovery of a stress distribution which is normal to the compliant sensing pad. In the case of tangential stress distributions, use of an entropy regularizer would be inappropriate since tangential stress is not unisense in nature. However, it can be seen from equation (4.2.10) that the network structure described in the last section is not restricted to use with an entropy regularizer only. It is clear that any regularizer which can be written in the form:

$$\sum_{i=1}^{N} \int_0^{v_i} g^{-1}(v)dv \tag{4.2.14}$$

can be introduced into the energy function of the network provided $g(\cdot)$ is a monotone increasing[2] function which can be implemented as the characteristic function of an analog amplifier.

It has been demonstrated in this chapter that analog neural networks can be structured so as to determine solutions to problems formulated as variational principles. Although the network discussed here is specifically structured to solve the inverse problem of tactile sensing, the results of Chapter 3 are more general and can be applied to a large class of problems.

4.3 Simulations

Computer simulations of the maximum entropy deconvolution network were performed in order to better understand behavior of the network in terms of speed of convergence, noise immunity and the effect of the regularizing parameter λ. SIMNON[3], an interactive simulation program for nonlinear dynamical systems was used to simulate the network.

For the purpose of simulation, the stress distribution at the surface of the compliant layer was assumed to be the result of applying pressure normal to the surface using a cylindrical object (see Figure 4.2). The resulting normal stress

[2]The monotonicity of the function $g(\cdot)$ is necessary to satisfy the requirement that the matrix G in equation (4.2.7) is positive definite.

[3]SIMNON was provided to us by Professor Astrom from the Lund Institute of Technology.

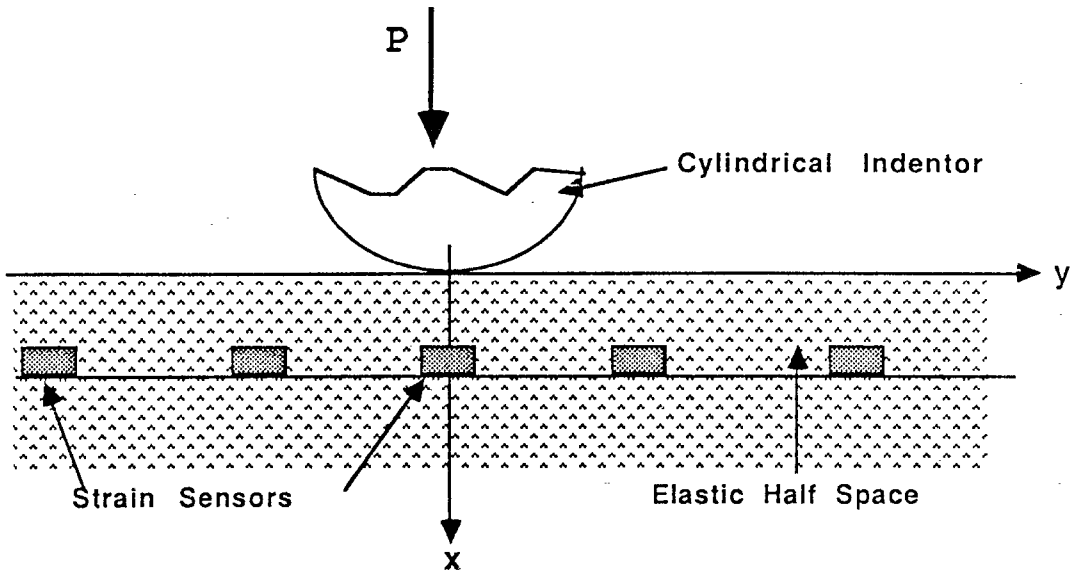


Figure 4.2: *Application of stress at boundary of elastic half-space using a cylindrical object*

distribution due to such a cylindrical indentor can be written as (see [30]),

$$f_v(y) = \begin{cases} \frac{p}{\pi a^2}\sqrt{a^2 - y^2} & \text{if } y \in [-a, a] \\ 0 & \text{elsewhere} \end{cases} \tag{4.3.15}$$

where p is the force per unit length and a is the halfwidth of the contact region. For $p = 3$ and $a = 1$ the resulting stress distribution is shown in Figure 4.3. This distribution is then convolved with the convolution kernel k_x relating normal stress to strain which is given by,

$$k_x^v(y - y_0) = \frac{3}{2\pi E}\frac{x(x^2 - (y - y_0)^2)}{(x^2 + (y - y_0)^2)^2}, \tag{4.3.16}$$

where the modulus of elasticity is E and Poisson's ratio for the material is assumed to be 0.5. Figure 4.4 is a plot of the convolution kernel for $E = 1$ and $x = 1$. The

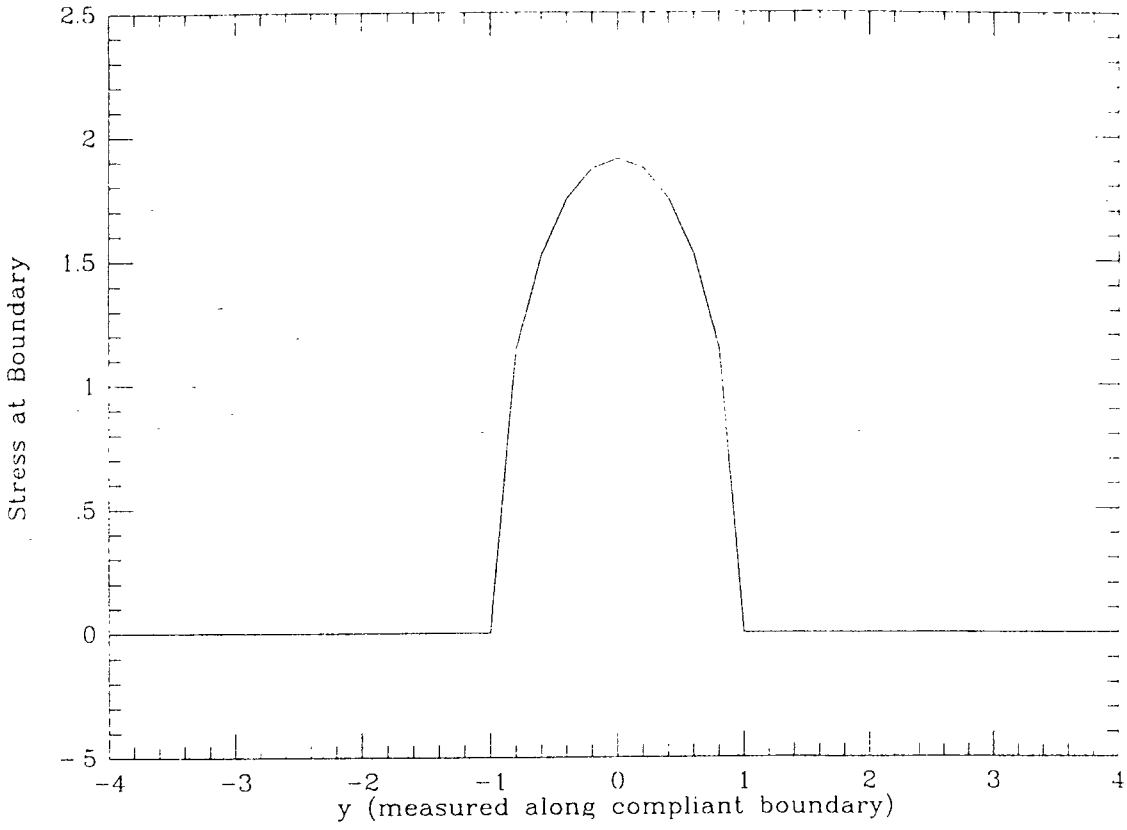


Figure 4.3: *Stress distribution due to cylindrical object applied to surface of compliant material ($P = 3$ and $a = 1$).*

resulting 'strain' (see Figure 4.5) is then provided as input to the network which then attempts to reconstruct the surface stress distribution of Figure 4.4.

The time evolution of three outputs of a 41 node network (using strain data which has not been corrupted with noise) is shown in Figure 4.6. It is observed that all three outputs rapidly converge to values which are very close to the known solutions (shown by the horizontal lines).

Accurate asessments of convergence time could not easily be made using digital computer simulations. If the signal plane capacitors were assigned values so as to reduce convergence time (by decreasing the RC time constant) numerical instability

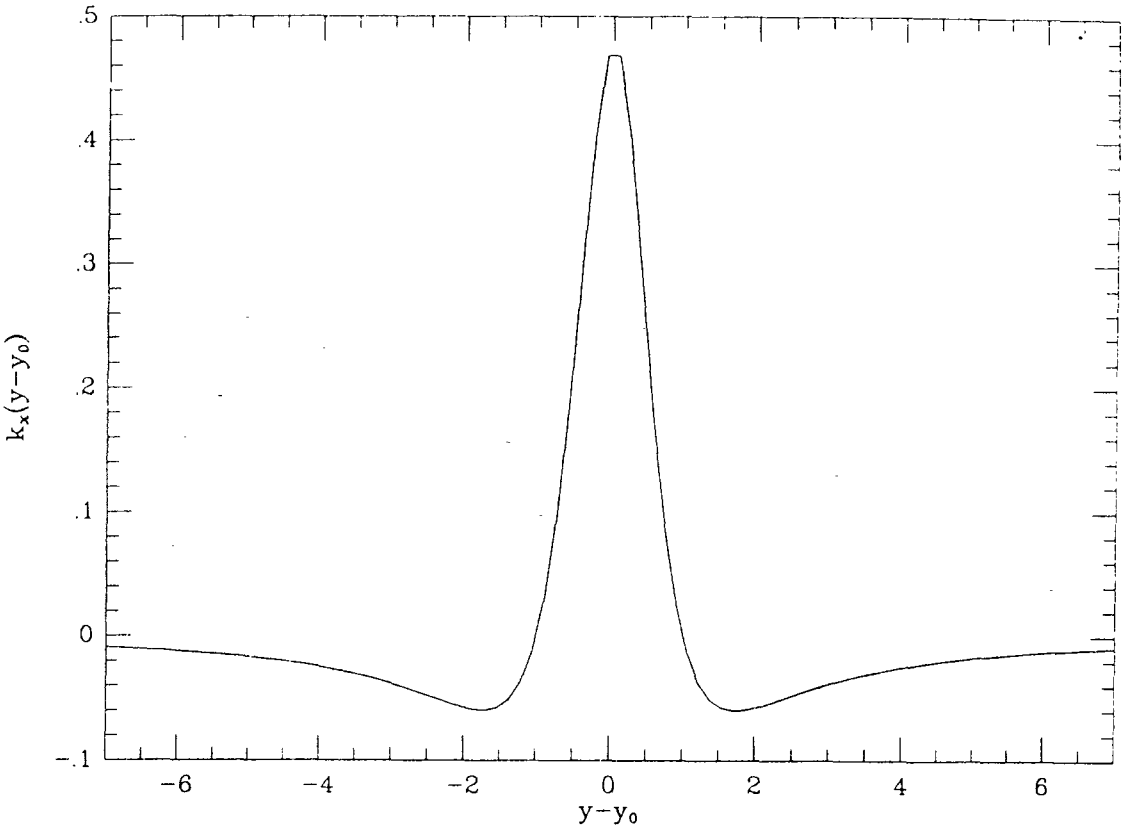


Figure 4.4: *Elastic convolution kernel (for $x = 1$ and $E = 1$).*

resulted. By decreasing the integration time step the problems with instability could be avoided at the cost of tremendous increases in the time necessary to perform a simulation; so it was merely observed at this stage that the network does converge.

To evaluate performance of the network in the presence of noise, Gaussian white noise with variance σ^2 was added to the strain from which surface stress was to be determined. Figure 4.7 shows the reconstruction of surface stress obtained by the network from noisy data ($\sigma^2 = 0.1$) with $\lambda = 0$ i.e. without any regularization. It can be seen that the reconstruction is very poor due to multiple peaks and negative solutions. However, in comparison to the reconstruction obtained under

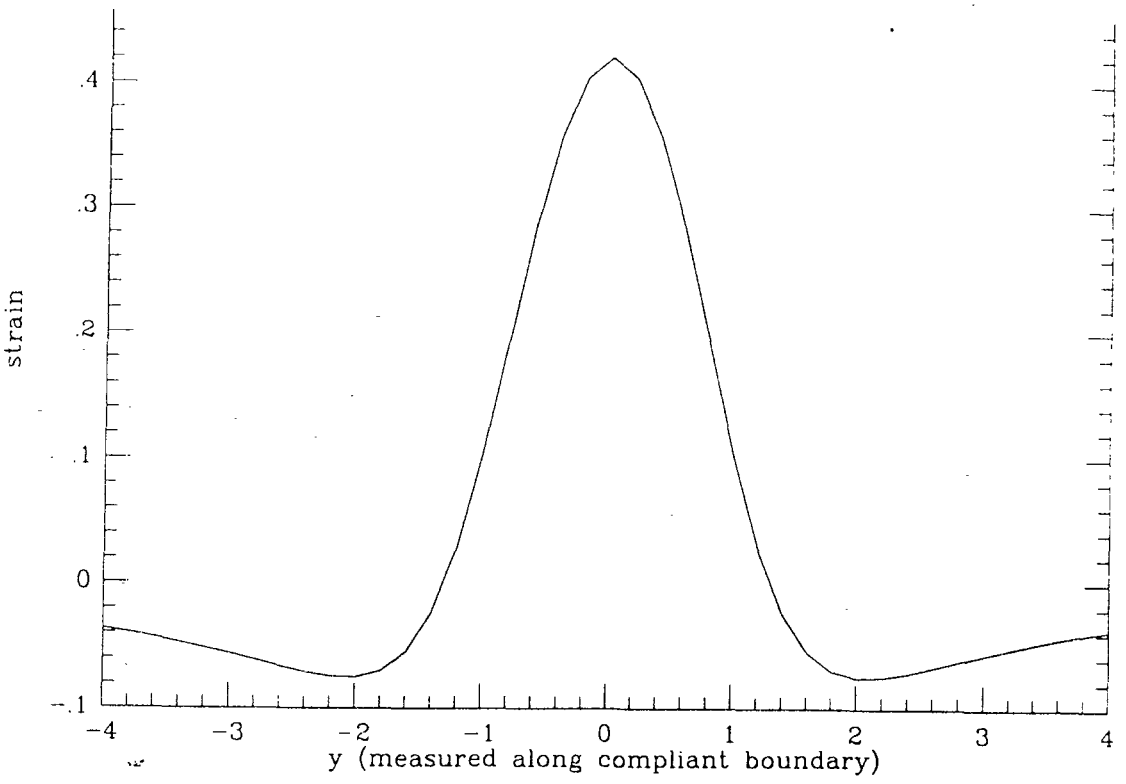


Figure 4.5: *Strain (at depth $x = 1$) resulting from cylindrical indentation of elastic half-space.*

identical conditions using the Discrete Fourier Transform (see Figure 4.8) the network solution is markedly superior. As the regularizing parameter λ is varied (see Figures 4.9–4.10), varying degrees of positivity and smoothness are imposed upon the solution. For $\lambda = 0.1$ (Figure 4.9)it is clear that although the solution has been constrained to the positive orthant, the degree of regularization is insufficient for the given noise conditions. In Figure 4.10 ($\lambda = 100$) the solution has been over-regularized since solutions which should have been close to zero have been pushed away from zero and the peak of the distribution has been greatly suppressed. Figure 4.11 shows reconstruction obtained using $\lambda = 10$ which is the 'best' of the three shown.

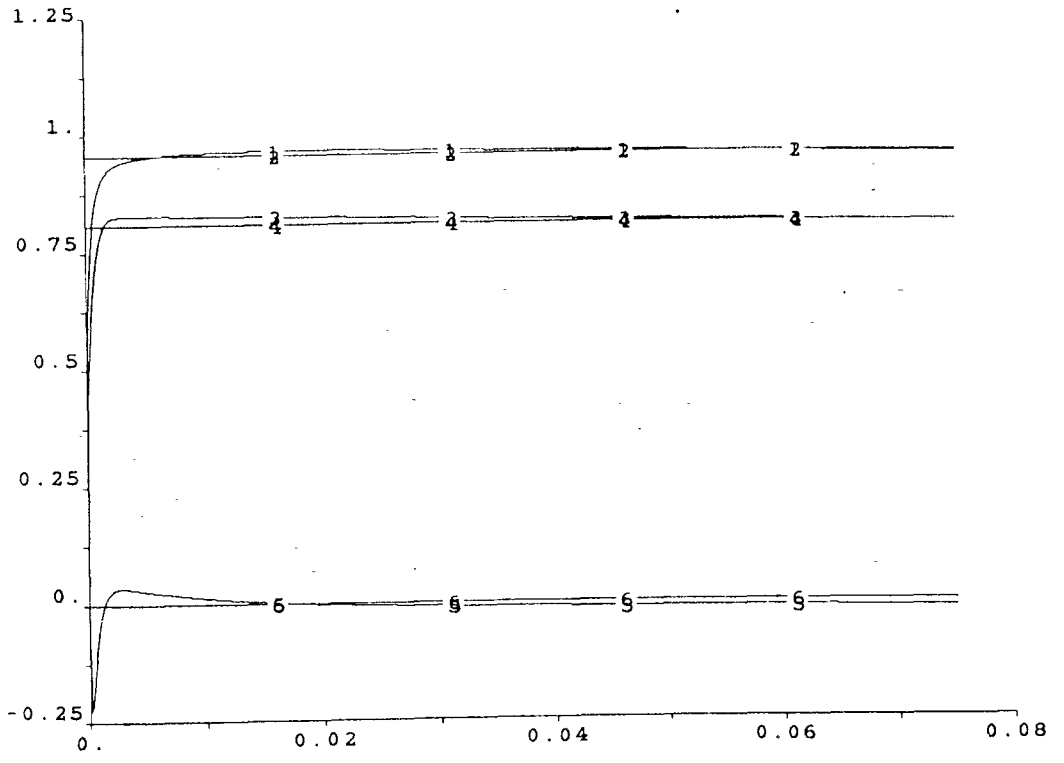


Figure 4.6: *Time evolution of three outputs ((1)V_{21}, (3)V_9 and (5)V_1) of a deconvolution network with 41 output nodes. Corresponding known solutions (f_{v21}, f_{v9} and f_{v1}) are shown by the horizontal lines (2), (4), and (6).*

Clearly, for given noise conditions, there exists a value of the regularizing parameter λ for which the reconstruction is optimal in some sense. Since the objective is to match the reconstruction to the original stress distribution as closely as possible, it is reasonable to choose a value of λ for which the discrepancy between the original stress distribution and the network reconstruction is minimized. One measure of the discrepancy between the known solution and the network solution is the mean squared error which can be written as,

$$MSE = \frac{1}{N}||f_v^0 - f_v||^2 \tag{4.3.17}$$

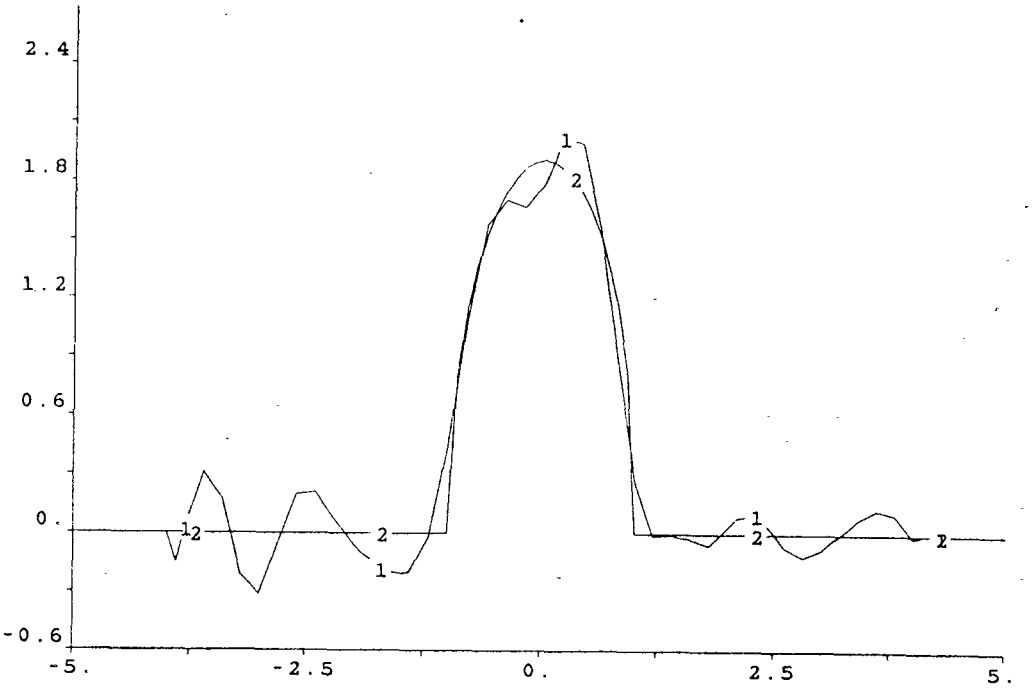


Figure 4.7: *Network reconstruction of surface stress from noisy strain data ($\sigma^2 = 0.1$) without regularization ($\lambda = 0$) (1)Reconstruction, (2)Designed surface stress.*

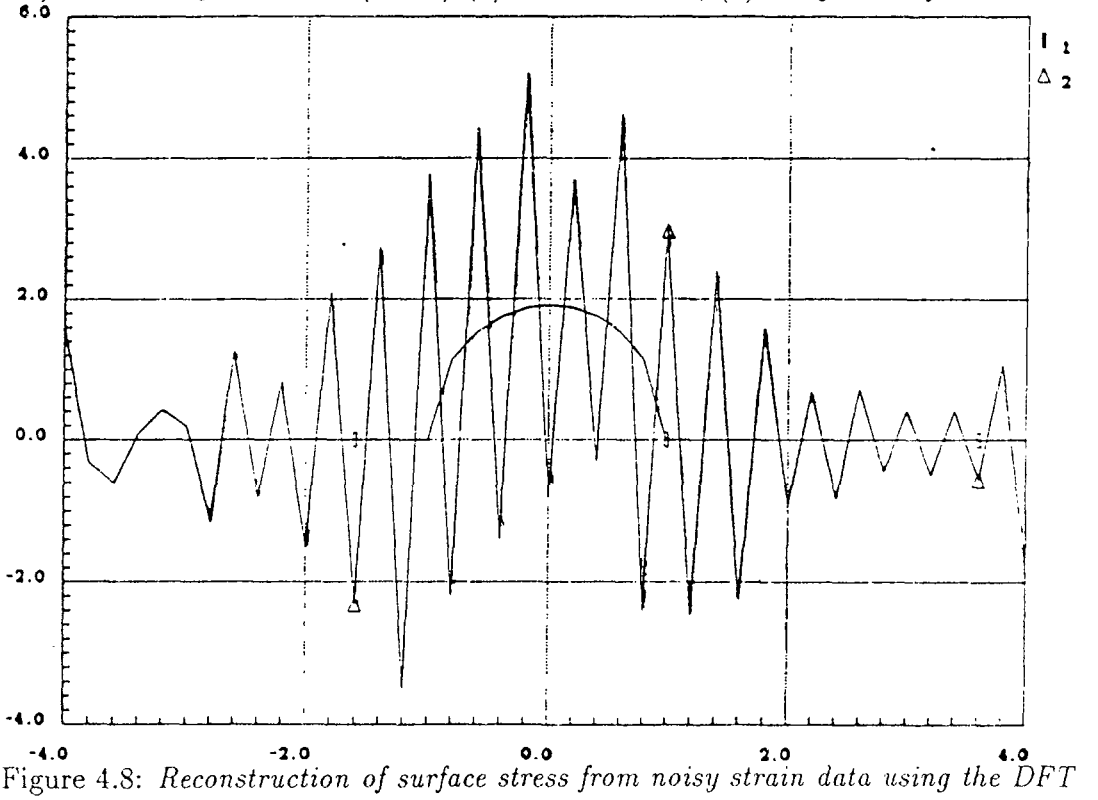


Figure 4.8: *Reconstruction of surface stress from noisy strain data using the DFT approach (1)Designed surface stress, (2)Reconstruction.*

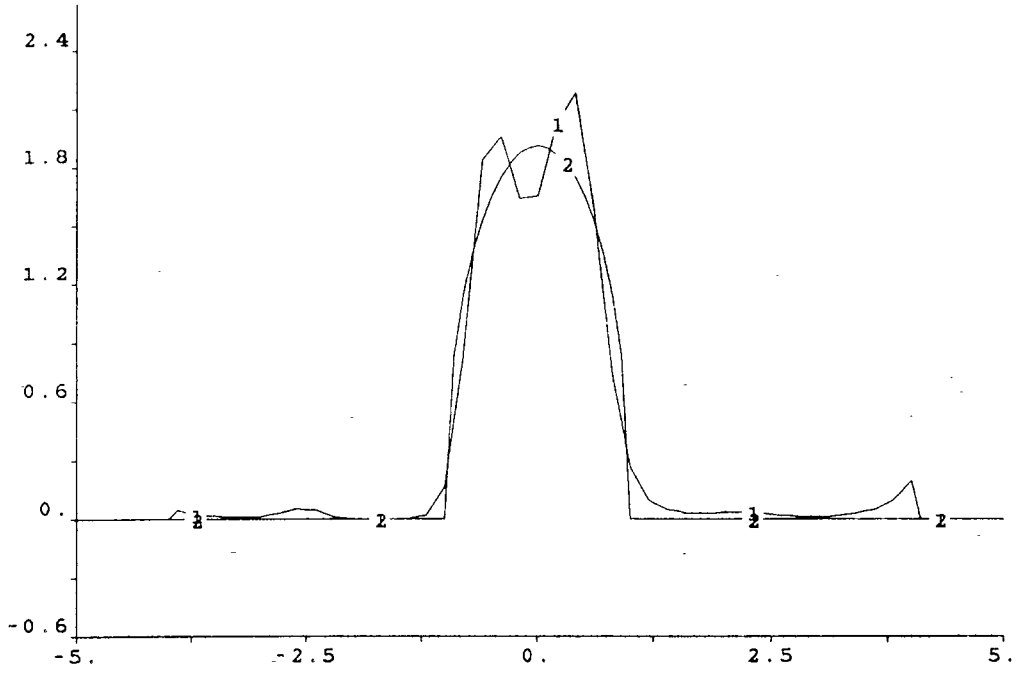


Figure 4.9: *Network reconstruction of surface stress from noisy strain data for* $\lambda = 0.1$ *(1)Reconstruction, (2)Designed surface stress.*

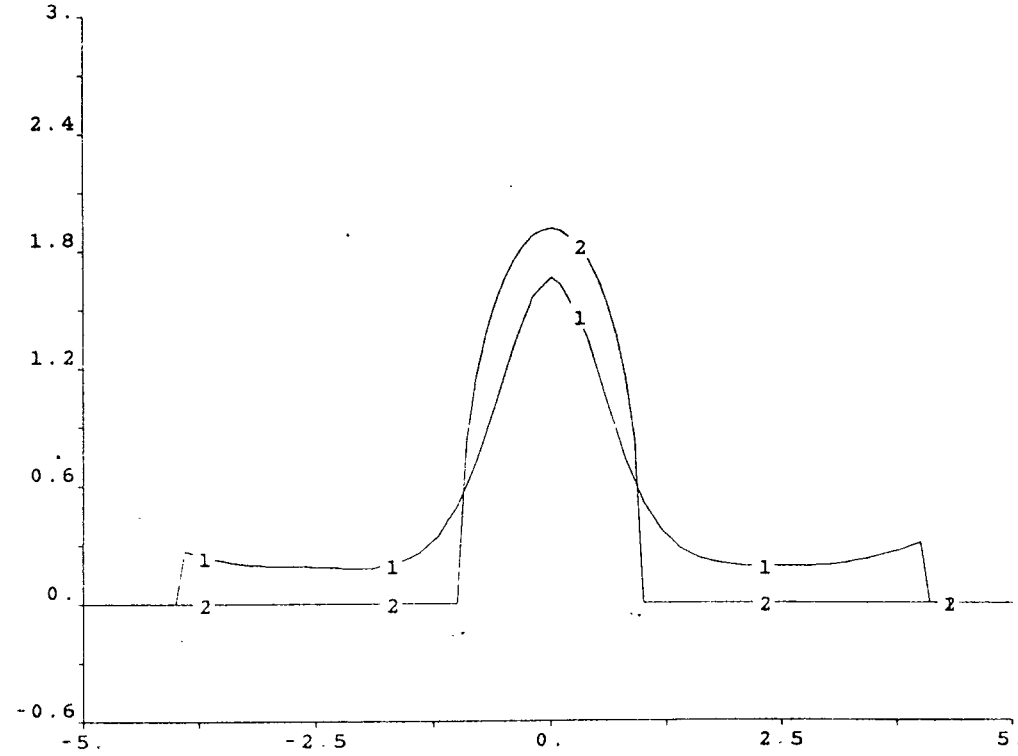


Figure 4.10: *Network reconstruction of surface stress from noisy strain data for* $\lambda = 100$ *(1)Reconstruction, (2)Designed surface stress.*

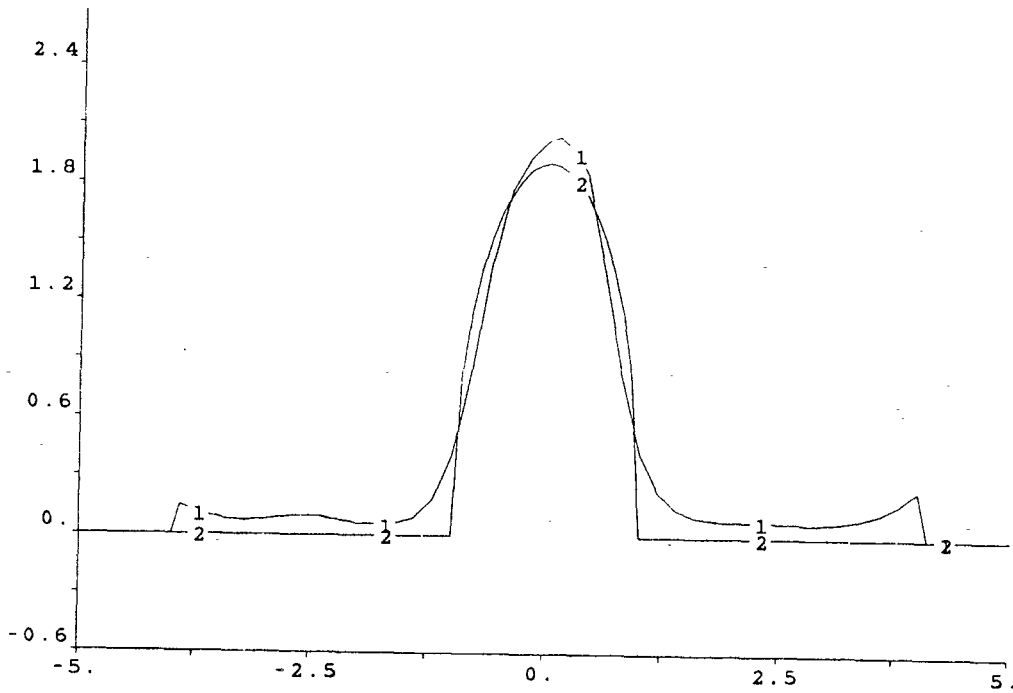


Figure 4.11: *Network reconstruction of surface stress from noisy strain data for* $\lambda = 10$ *(1)Reconstruction, (2)Designed surface stress.*

where $f_v^0 \in \mathbb{R}^N$ is the known solution to the inverse problem and $f_v \in \mathbb{R}^N$ is the solution obtained by the network. Such an approach to the problem of choosing λ suggests the use of design tools such as CONSOLE (see [30]) which is a computer aided design tool for parametric optimization of dynamical systems[4].

[4]CONSOLE was used to optimize parameters in design of the breadboard prototype network which was constructed

CHAPTER
FIVE

ANALOG HARDWARE IMPLEMENTATION

In order to realize the full computational power of neural networks it is important to consider implementation of neural networks as analog electrical circuits. The inverse problem considered here is among a class of problems which can in fact be solved using more traditional digital computation. However, traditional digital computation is likely to fall short of the needs of tactile sensing since the digital hardware necessary to achieve the desired speed of computation is not likely to lend itself to placement in close proximity to the sensors. It is empirically demonstrated in this chapter that analog neural networks can achieve the required speed while limiting hardware requirements to as little as a single integrated circuit chip. This also permits dedication of a single analog processor chip to each array of sensors thereby distributing the processing. Although the current state of VLSI technology does not allow for implementation of extremely large networks, there is still a class of useful networks that can be implemented. The network we consider here for the solution of the inverse problem of tactile sensing is among those networks for which currently implementable sizes are sufficient to prove useful.

Integrated circuit implementation of neural network models is an area of research which has recently inspired the interest of a significant community of re-

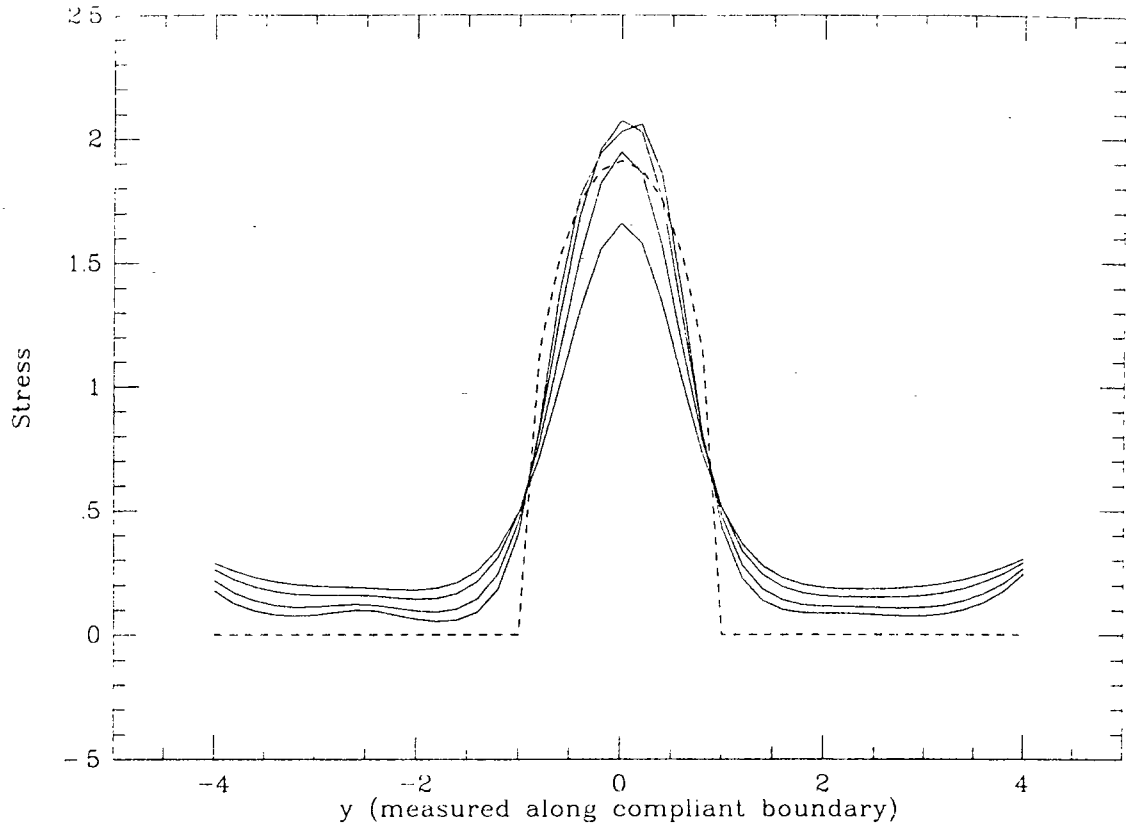


Figure 4.12: *Reconstructions of surface stress as λ is varied. The dashed curve represents the actual surface stress. As λ increases, the resulting reconstructions (solid curves) have increasingly suppressed peaks and move further into the positive orthant*

searchers. Large multilayered, fully connected networks require a large number of weighted connections. Even if these connections are implemented by simple resistors, physical size of the chip and achievable density of circuitry become limiting factors as the size of the network becomes large. Current microelectronic technology is essentially two dimensional and thus there arises the (graph theoretic) problem of laying out the conductive paths necessary to make the required connections. These types of problems have generated interest in optical implementations of neural networks (see [31]). Synaptic connections in an optical neural network can be defined in three dimensions and weights can be set by modulating the intensity of light over a region in space, and stored using holographic techniques. However, optical technology is still in its infancy although it may prove practical in the future. 'Learning' in neural networks requires the ability to change the weight of any connection in the network. In [32] a VLSI implementation a neural network is presented in which the synapses are programmable, but are restricted to be either excitatory (+1), inhibitory (-1) or inert (0). Although the implementation in [32] is a significant contribution to the development of neural network implementations, the limitations of current technology are evident. The network in [32] (among of the largest implemented networks to date) contains almost 3000 programmable connections and utilizes a 2.5 μm technology, but almost 90% of the 6.7 $\times$ 6.7 mm chip is occupied by the synapses. In general, it would be desirable to create synapses with weights which are continuously variable through zero and can extend significantly into both the positive and negative orthants. In [33], an approach to such continuously programmable synapses is taken in which MOS transistors are

employed as analog multipliers to effect the multiplication required of a synapse. A number of researchers are also exploring the development of new and different integrated circuit technologies which are better suited to neural network implementations. The primary focus of research into implementations of analog neural networks is the design of dense arrays of programmable synapses. Programmability of the connection weights, in the context of the inverse problem considered here, will permit the use of different convolution kernels in the deconvolution network. Hence variations due to changes in sensor response etc. can be accommodated if the new convolution kernel can be determined.

In Section 5.1 some experimental results from a breadboard prototype of the network described in Chapter 4 are discussed and Section 5.2 describes an initial approach to integrated circuit implementation.

5.1 Breadboard Prototype

As discussed in Chapter 4, accurate assesments of convergence time for the network are not easily made using digital computer simulations. Also, in the analysis of the deconvolution network in Chapter 3, it was assumed that any dynamics associated with the constraint plane could be ignored provided that the signal plane amplifiers are sufficiently slower in response. In practical implementations of such a network, it is necessary to understand what effects delays in the constraint plane response may have upon the network. It is ultimately the constraint plane dynamics which limit the speed of convergence which is achievable. A formal treatment of this subject is to be found in Marcus and Westervelt [34]. A prototype breadboard model

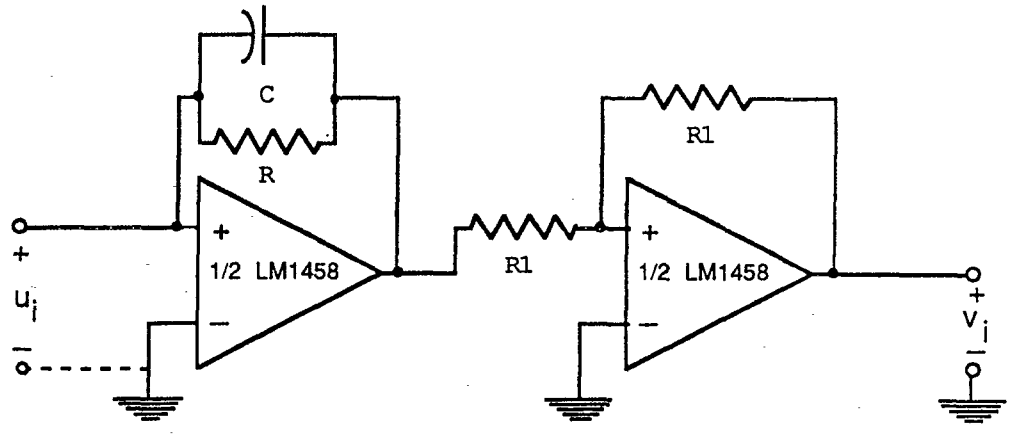


Figure 5.1: *Schematic circuit diagram of a single signal plane node*

of the deconvolution network was constructed using 'off-the-shelf' operational amplifiers, resistors and capacitors. The network was constructed with seven signal plane nodes and seven constraint plane nodes. Since the purpose of constructing the breadboard prototype was to estimate to some extent the speed achievable by such a network, exponential amplifiers in the signal plane were replaced by unity gain linear amplifiers to simplify the circuit [1]. Replacing the exponential amplifiers by linear amplifiers results in the entropy regularizer being replaced by a regularizer of the form,

$$\lambda M(v) = \frac{1}{R}\sum_i v_i^2. \tag{5.1.1}$$

In Figure 5.1 a circuit diagram of a single signal plane node is shown. Each signal plane node consists of two stages of amplification. Associated with the first stage is the feedback capacitor C, which introduces the relevant network dynamics,

[1] A second breadboard prototype was also constructed which contained the exponential amplifiers, but was used to solve a different problem (see [36]).

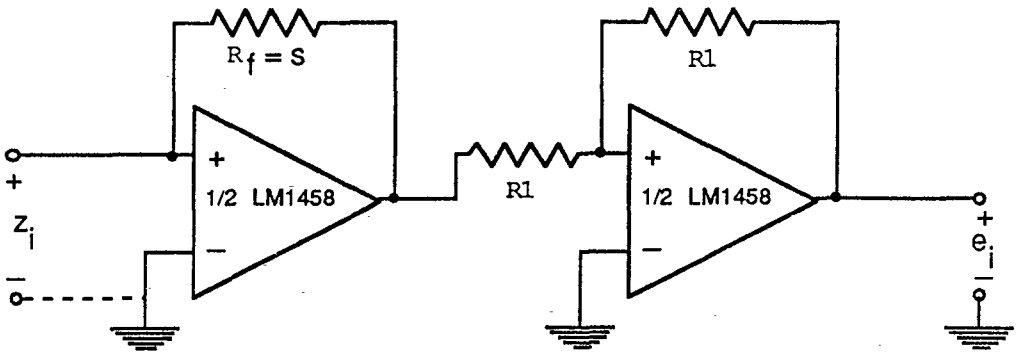


Figure 5.2: *Schematic circuit diagram of a single constraint plane node*

and the feedback resistor R, which introduces and weights ($\lambda = 1/R$) the regularizing term in the energy function. The second stage of each signal plane node is configured as an analog inverter. Negative connection weights are implemented simply by using the inverted output of the node. Figure 5.2 shows a single constraint plane node. Each constraint plane node also consists of two stages. The first stage is configured as a virtual ground transimpedance amplifier, which provides the feedback gain for signals fed back to the signal plane. As in the signal plane, the second stage of each constraint plane node is an analog inverter.

The interconnection matrix $[T_{ij}]$ was chosen as a seven point discretization of the elastic kernel $k_x^v(\cdot)$ in equation (4.3.16) and implemented using resistors with values $R_{ij} = 1/T_{ij}$. Thus if the voltage output v_k of node k is connected to the input of node j through a resistor R_{kj} then current input to node j due to node k

is given by Ohm's Law as,

$$i_{jk} = \frac{v_k}{R_{kj}} \tag{5.1.2}$$

which is as desired.

Inputs to the network (currents injected into the constraint plane) were chosen to represent samples of the strain distribution due to the compressive loading profile used for the simulations in Chapter 4. In Figure 5.3 photographs of the 7-Channel breadboard prototype of the deconvolution net work and the experimental setup used for testing it are shown. Input currents to the network are clocked using a 1 kHz relaxation oscillator so as observe transients (as the network evolves) using an oscilloscope. Outputs of the network are captured by a MetaResearch data acquisition board used in conjunction with a Macintosh Plus computer and the resultant reconstruction is plotted on the Macintosh display.

The rise time of the constraint plane amplifiers was measured to be approximately 1 μsec. Actual response time of the constraint plane would be longer than this since the parallel combination of all resistors connected to the input of any node contribute to the RC time constant. It was observed that for choices of the signal plane capacitors C for which the rise time of the outputs of the network would be below 10 μsec, the outputs would oscillate i.e. the network was unstable. For C=10 pF the rise time of the outputs of the network was measured to be 10 μsec (see Figure 5.4). It is clear that the use of faster operational amplifiers would result in an increase in achievable speed since this would decrease the constraint plane response time and thereby permit a decrease in the time constant of the

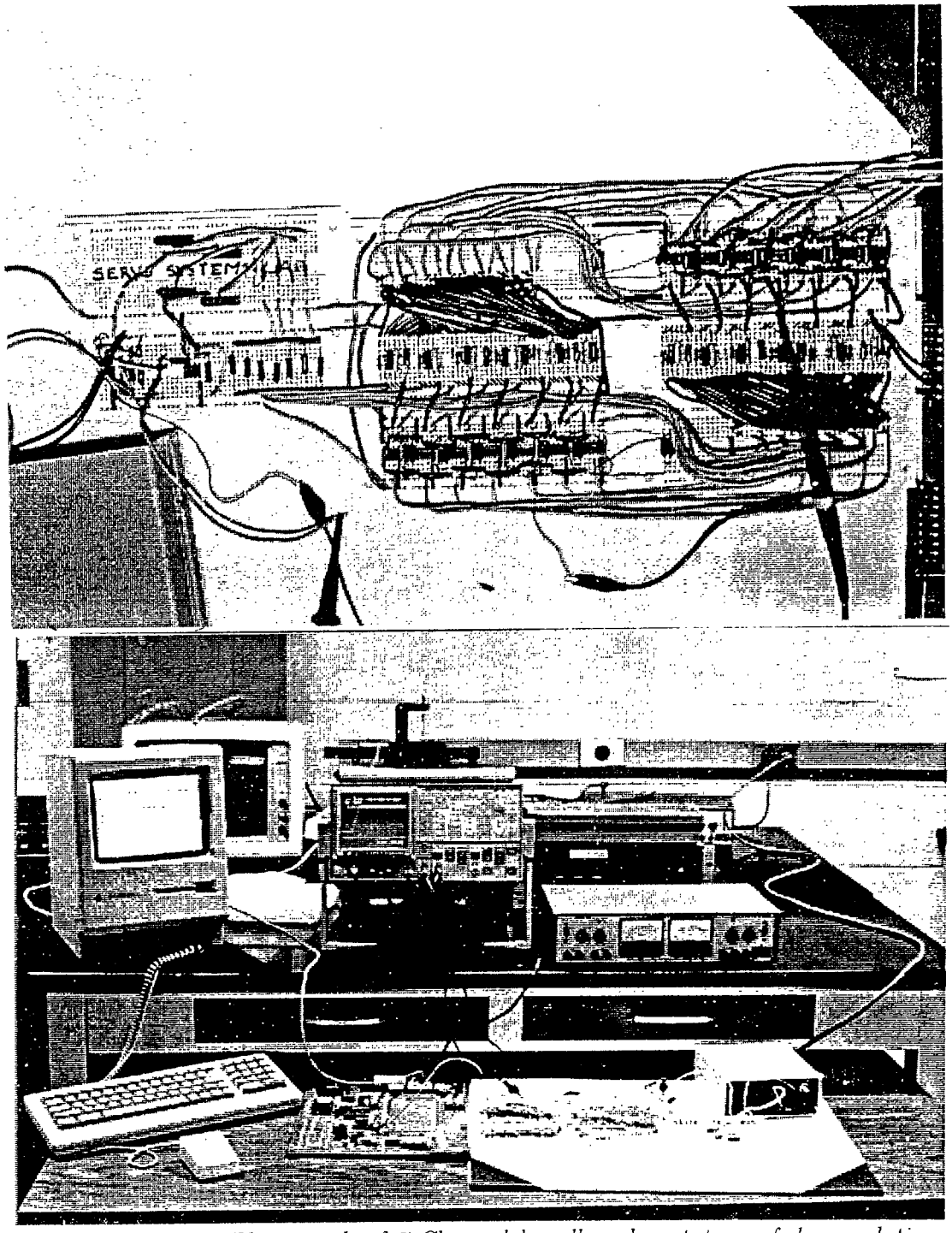


Figure 5.3: *Top: Photograph of 7-Channel breadboard prototype of deconvolution network Bottom: Photograph of experimental setup used to test the deconvolution network.*

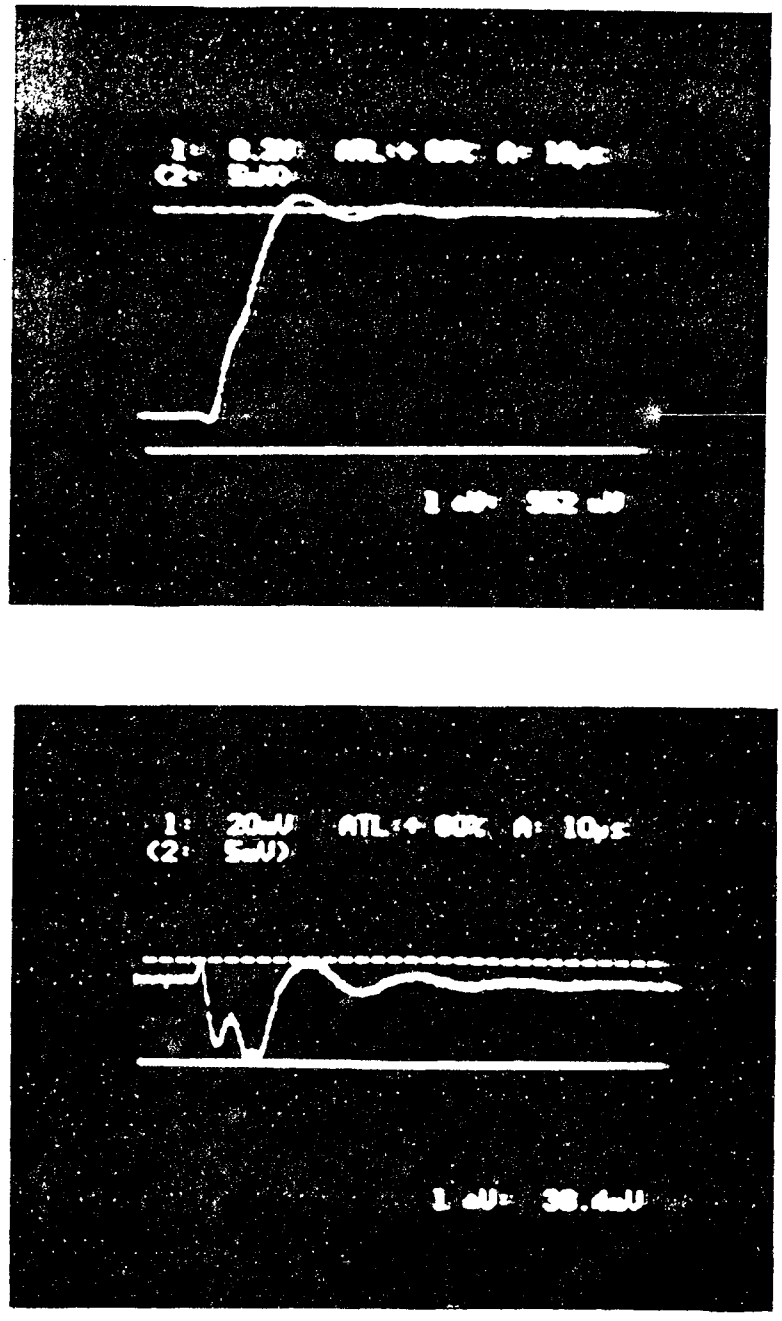


Figure 5.4: *Photographs of oscilloscope traces showing, Top: time evolution of a single output of the signal plane, and Bottom: time evolution of a single output of the constraint plane, for the 7-channel breadboard prototype deconvolution network.*

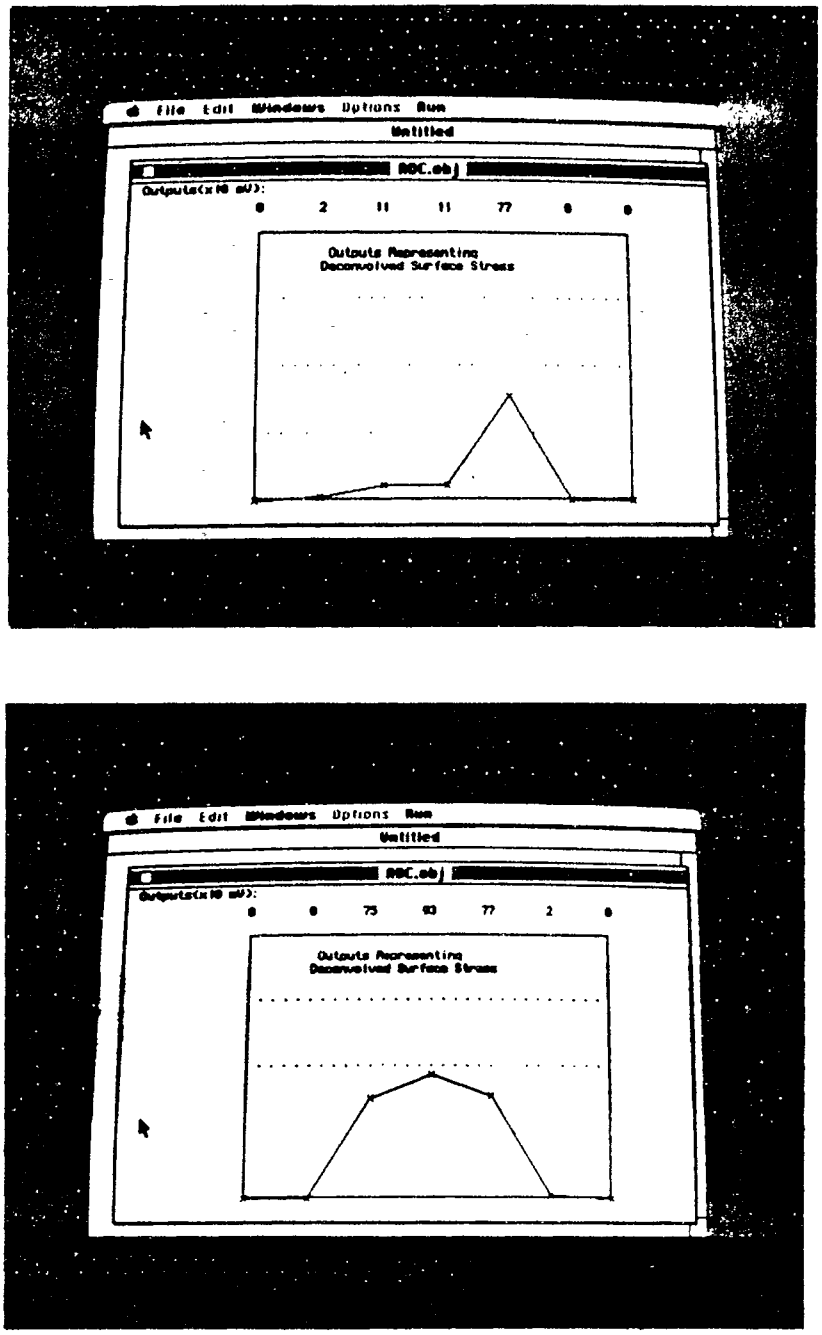


Figure 5.5: *Reconstruction of surface stress by breadboard prototype deconvolution network. Top: Reconstruction prior to settling of outputs, Bottom: Final reconstruction.*

signal plane.

Settling time and overshoot of the outputs of the network are controlled by the gain of the constraint plane nodes. CONSOLE (see [30]) was used to choose a value for the gain so as to minimize overshoot and settling time.

5.2 Integrated Circuit Prototype

A prototype analog integrated circuit implementation of the deconvolution network described here has been fabricated, but remains to be tested. A hierarchical design philosophy is practiced in this initial implementation. The deconvolution network may be thought of as composed of two sections: (i) Active components of the network including signal and constraint plane amplifiers and (ii) The functionally passive[2] resistive interconnection matrix. These two sections may also be thought of in the following manner. Once the size of the deconvolution network (number of inputs and outputs) has been decided, the amplifiers of the network are determined. However, the resistive matrix may be a variable entity. For instance, given two different elastic materials (or even two different thicknesses of a given elastic material), the convolution kernel $k_x^v(\cdot)$ and hence its discretization $[T_{ij}]$ are in general different. Thus the network may be thought of as being composed of a fixed part and a variable part.

If fixed resistors are to be used to implement the interconnect matrix, then

[2]The term 'functionally passive' here is used to describe the fact that in some situations it is desirable to use active circuit components configured to look like a passive resistor from an input/output standpoint.

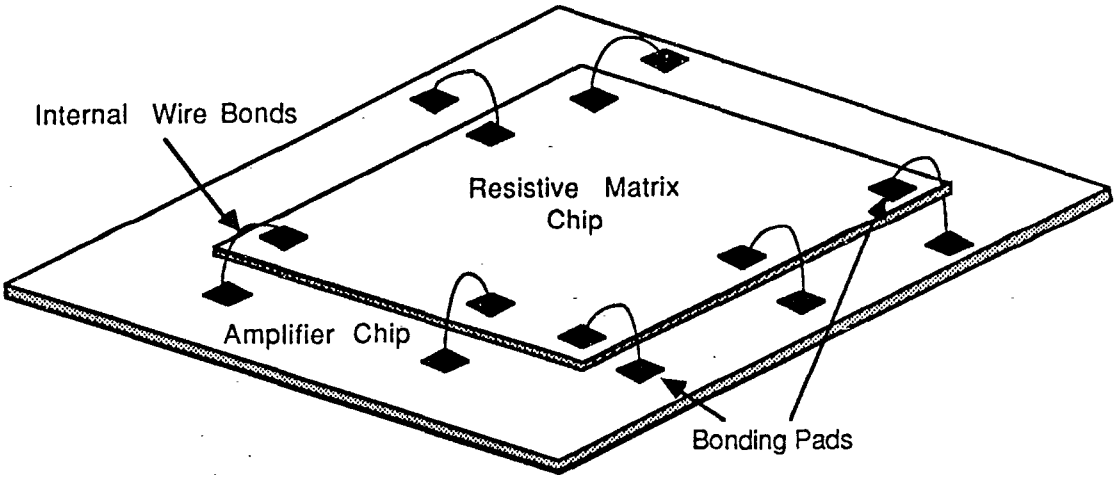


Figure 5.6: *Resistive matrix chip is placed directly atop the amplifier chip and connected to it using local bonding*

some provision should be made to change this matrix without having to refabricate the rest of the network. In order to provide some flexibility in the choice of the interconnect matrix and to permit the use of two different fabrication technologies (as discussed in Section 6.2.3), the deconvolution network was fabricated as two separate integrated circuits.

The amplifier chip, which is described in Section 6.2.2, is designed to serve as the 'motherboard' for the network on top of which the resistive connection matrix chip (which is described in section 6.2.3) is placed (see Figure 5.6). Connections between the two chips are made by local wire bonds between bonding pads provided for this purpose on both chips. This approach also facilitates experimentation with different types of connection matrices such as those with programmable connections.

5.2.1 The Amplifier Chip

Figure 5.7 shows the layout of the amplifier section of the deconvolution network. This chip provides the signal plane and constraint plane amplifiers for a deconvolution network with eleven input nodes and eleven output nodes. Fabrication of the amplifier section of the deconvolution network was undertaken through the MOSIS facility, using a two metal layer CMOS p-well technology with a minimum feature size of 3μm. The size of the amplifier chip is approximately $6300\mu m \times 8800\mu m$. In the center of the amplifier chip, a $4000\mu m \times 3200\mu m$ space has been provided for placement of the second chip containing the resistive connection matrix. It is actually not necessary to leave this space since the chip is passivated, but for an initial implementation, it was provided.

A total of forty four operational amplifiers are implemented on this chip; twenty two for the signal plane and twenty two for the constraint plane. Each signal plane node and constraint plane node is identical in structure to the signal and constraint plane nodes used for the breadboard prototype. Both the inverted and noninverted outputs of the signal and constraint plane nodes are connected to bonding pads adjacent to the area where the resistive matrix is to be placed. Thus provision is made for the implementation of positive and negative connection weights. Outputs and inputs of both the signal plane nodes and constraint plane nodes are connected to bonding pads located on the periphery of the die to permit access to these nodes after the chip has been packaged. Packaging of the chip requires a package with a minimum of fifty pins, if the inputs and outputs all nodes are to be accessible.

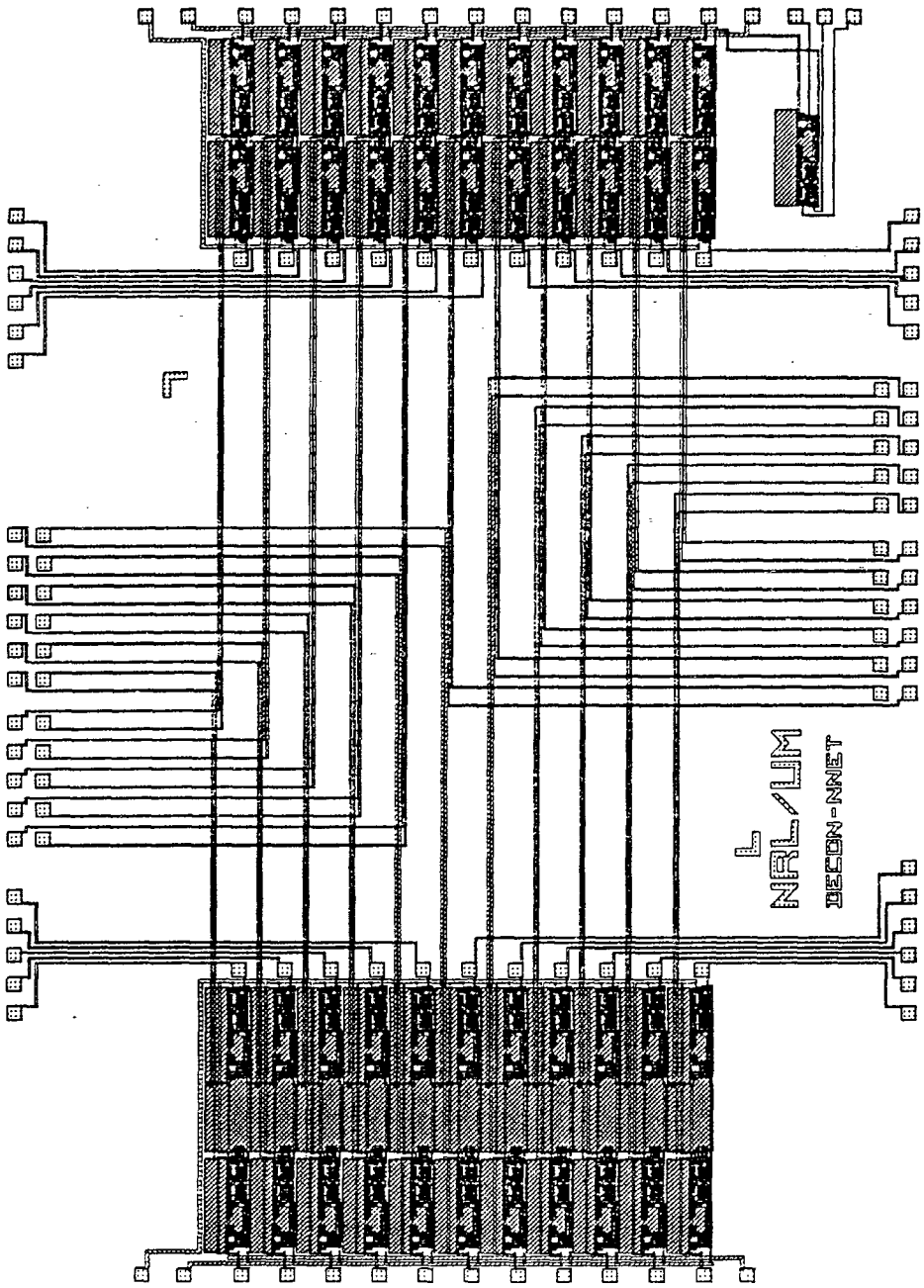


Figure 5.7: *Layout of integrated circuit chip containing all amplifiers for an eleven channel deconvolution network*

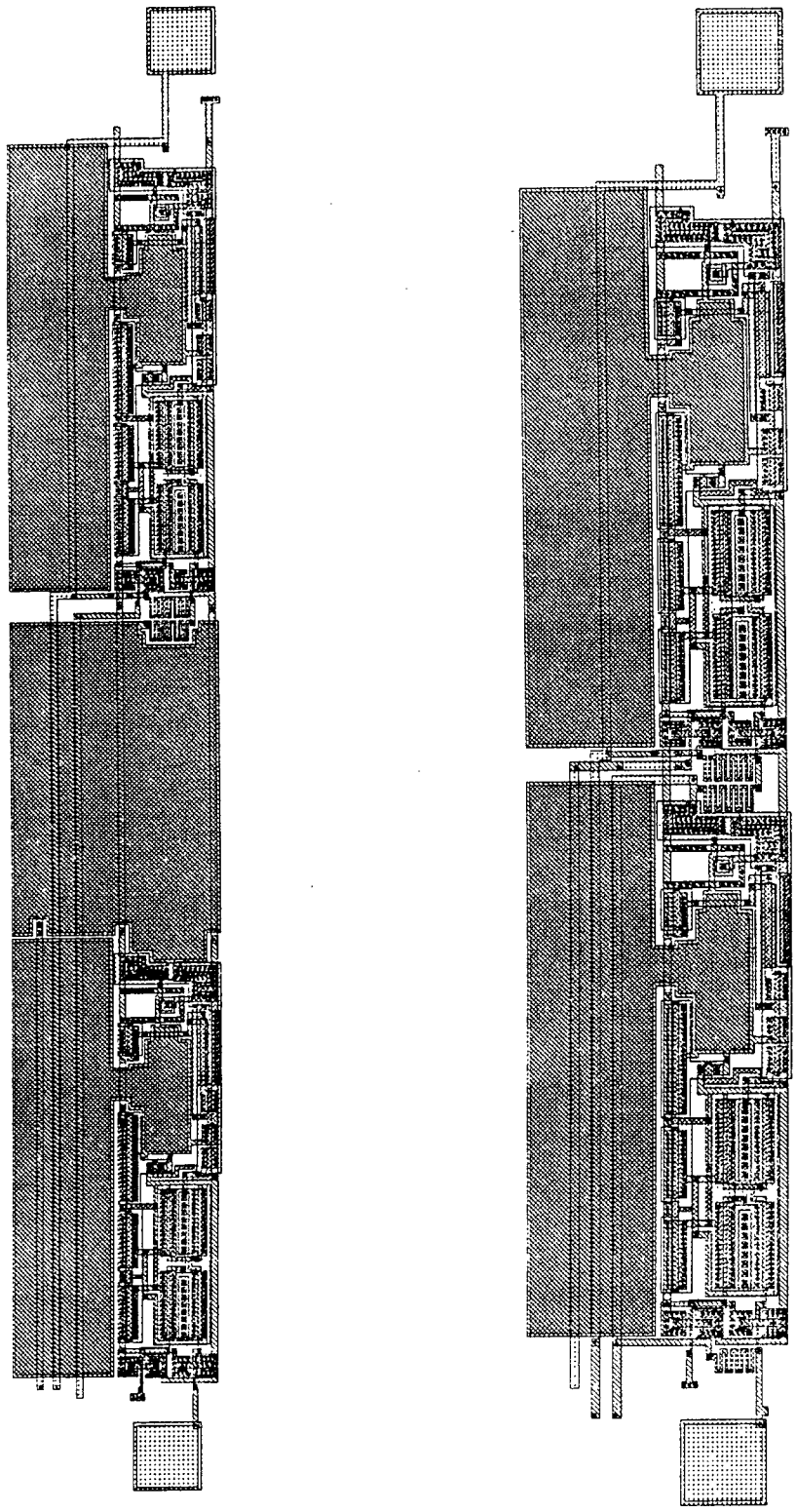


Figure 5.8: *Left: Layout of signal plane amplifier; Right: Layout of constraint plane amplifier.*

The basic building block of the amplifier chip is the operational amplifier. Figure 5.9 shows a circuit schematic of the operational amplifier designed for this purpose. Design of the amplifier takes into consideration the current driving requirements for use in the deconvolution network. Inputs on each amplifier are diode protected against spikes. A source follower output stage is used on every amplifier so as to meet the current driving requirements while maintaining relatively low circuit complexity (compared to, for example a push-pull output stage).

As can be seen from Figure 5.9, an internal compensation capacitor of 5pf is required for every operational amplifier. In addition to this, external feedback capacitors of about 10pF are required at the first stage of every signal plane node. Since the technology available did not include an additional electrode layer (i.e. a second layer of polysilicon to be used in forming capacitors), all capacitors in the network are formed as parallel plate capacitors with polysilicon as one electrode and the first metal layer as the second electrode. Capacitance between the first metal layer and polysilicon is approximately a factor of eight less than the capacitance between polysilicon and an electrode layer. Hence about eight times as much area is used to form the capacitors, compared to capacitors formed using an electrode layer. Since about half of the area used by constraint plane nodes and two thirds of the area used by the signal plane nodes is used to form the capacitors, about seven twelveths of the total area used by the amplifiers is taken up by capacitors. Hence the use of a separate electrode layer would result in about a 50% reduction in space utilization. Therefore a network with twenty two inputs and twenty two outputs is a trivial extension of the current structure (which includes the space

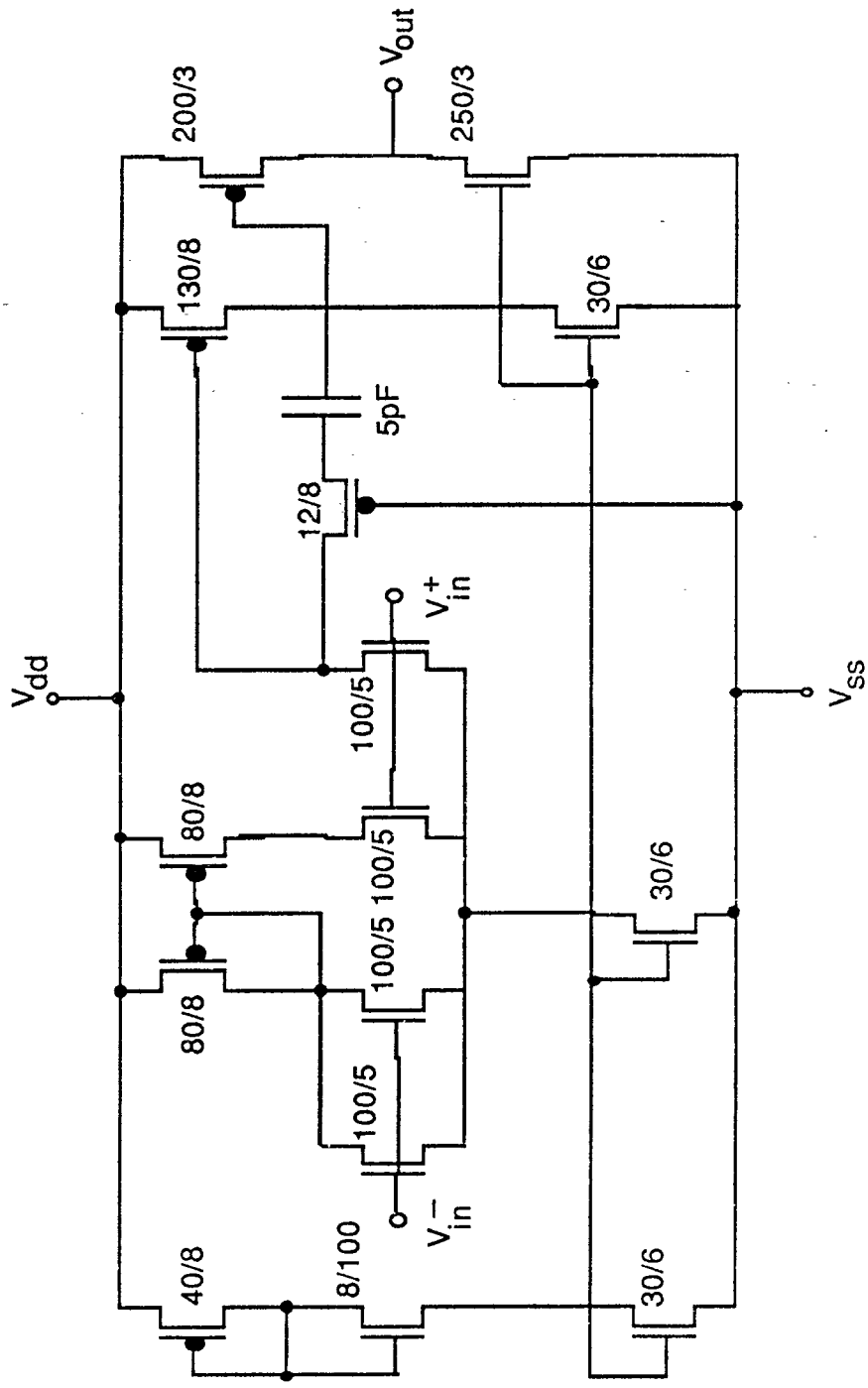


Figure 5.9: *Circuit diagram of operational amplifiers designed for integrated circuit implementation of deconvolution network*

provided for the resistive matrix) using a comparable $3\mu m$ technology.

5.2.2 The Resistive Interconnection Matrix Chip

In any implementation of a matrix of resistive elements, it is necessary to consider the effects of process induced variations on the performance of the network with which it is to be used. Current integrated circuit process technology may introduce variations in the value of any given resistor as large as 20%. It must either be established that degradation in performance of the network due to such variations is irrelevant, or an approach to implementation must be taken which preserves the essential characteristics of the interconnection matrix in the face of process induced variations. The latter approach was taken in this implementation.

Since in the case of the deconvolution network, the resistive interconnection matrix is a discretization of the elastic convolution kernel, it is reasonable to try to preserve the shape of the connectivity profile. That is, it is the ratiometric relationship of the connective weights that is truely important since any other variations correspond to a simple scaling of the inputs or outputs. In an attempt to preserve the shape of the connectivity profile in the face of processing induced variations, the connective weights are quantized. Figure 5.10 shows the layout of the resistive matrix. A discrete resistive element is formed by etching a $5\mu m \times 5\mu m$ opening in the oxide layer between two metal wires and then evaporating silicon into the opening (see Figure 5.11). Amorphous silicon, therefore forms the resistive material. Each such discrete resistor has a resistance value that depends on the thickness of the oxide layer and the area of the opening. For the oxide thickness

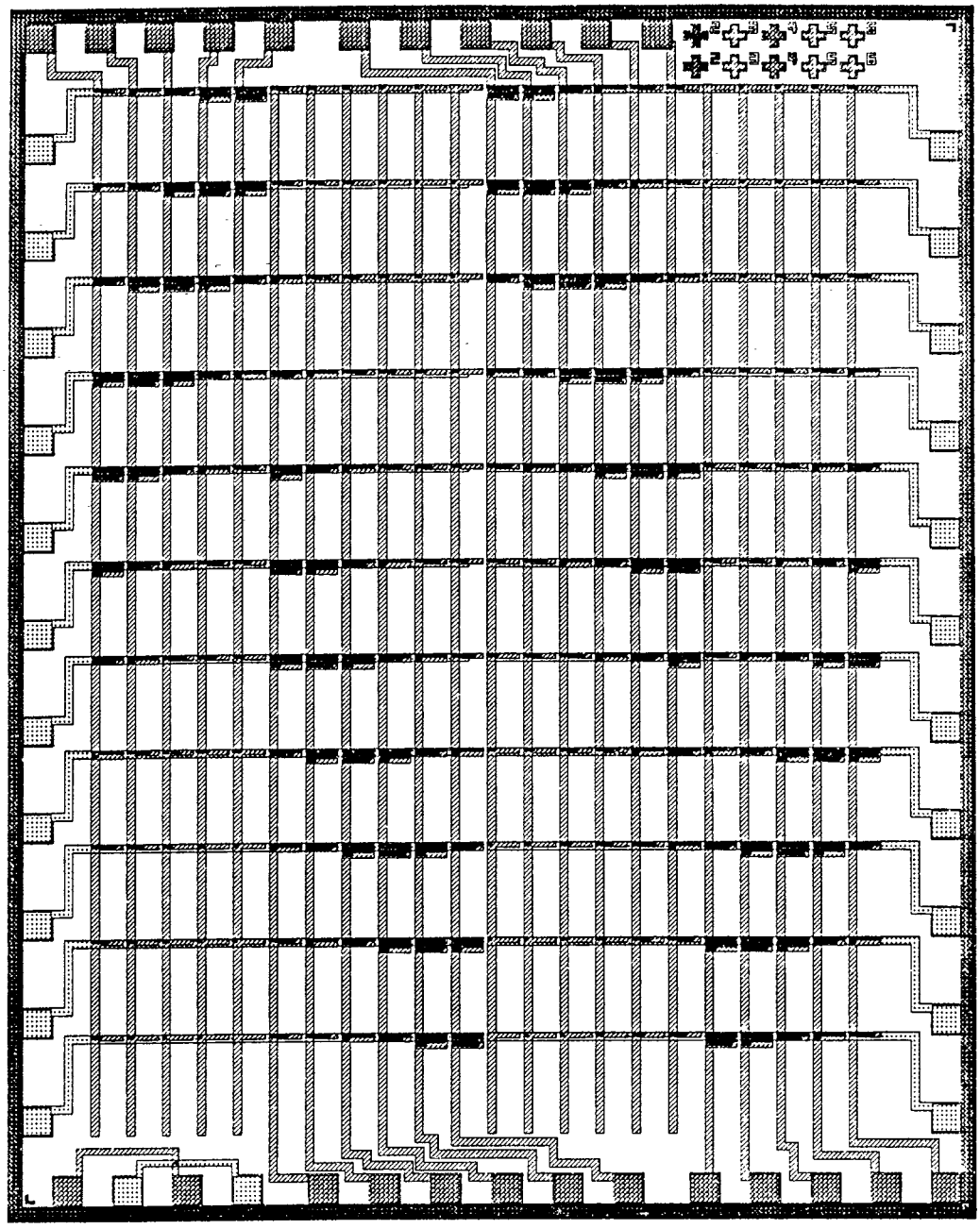


Figure 5.10: *Layout of resistive interconnection matrix chip.*

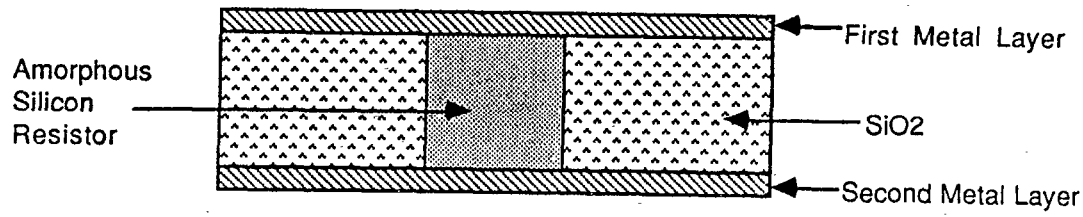


Figure 5.11: *A discrete resistive element as used in the resistive matrix chip.*

used (and $5\mu m \times 5\mu m$ openings) the value of a single discrete resistor was measured to be approximately 35KΩ. A wire grid is formed using the first metal layer to construct parallel wires in one direction and the second metal layer to construct wires in a orthogonal direction (to the wires formed in the first metal layer). At each point on the grid, a connective weight is defined by placing a number of discrete resistors of the type described at that point. The strength of connection at any point on the grid is determined by the number of discrete resistors placed there. Ratiometric relationships between elements of the connection matrix should for the most part be preserved in this approach since the ratio of two connective weights is primarily determined by the ratio of the number of discrete resistors forming each connection Any variations in the ratios are the effects of phenomena such as nonuniform oxide thickness over the area of the die, nonuniform deposition of amorphous silicon which are relatively small effects.

CHAPTER
SIX

CONCLUSIONS AND FUTURE DIRECTIONS

In this thesis, we considered the inverse problem of recovering stress distributions over regions of contact from samples of strain provided by an array of tactile sensors. In the case where stress is applied to the surface of a compliant material and strain is measured at a fixed depth beneath the surface, the inverse problem was shown to be a problem of deconvolution. It was shown in Chapter 2 that this inverse problem of tactile sensing is ill-posed in the sense defined by Hadamard. Also in Chapter 2, it was shown that the technique of regularization could be used to introduce *a priori* knowledge into the problem in order to obtain solutions. The constraints of non-negativity and smoothness were imposed by choosing an entropy regularizer for the recovery of normal surface stress. Solutions to the inverse problem could then be obtained by minimization of a cost functional.

In Chapter 3, it was shown that under certain hypotheses, energy functions may be explicitly determined for nonlinear RC electrical networks. Stable equilibria of the network were shown to correspond to local minima of the energy function and conditions for stability of the network were determined.

An analog neural network for regularized solution of the inverse problem was presented in Chapter 4. Using the results of Chapter 3, it was shown that the energy

function of the proposed network corresponds to the variational principle described in Chapter 2 for solution of the inverse problem of tactile sensing. Stability of the network, in terms of electrical circuit analysis, was shown to be guaranteed by monotonicity of the characteristics of the signal plane amplifiers. It was also determined that any regularizer which could be written in the form of equation (4.2.14) and satisfied the monotonicity requirements for the signal plane amplifier characteristic g, could be incorporated into the energy function for the network. Computer simulations demonstrated the ability of the deconvolution network to accurately recover normal surface stress even when the sensor outputs were severely corrupted by noise.

A breadboard prototype of the deconvolution network was used to demonstrate the computational speed achievable by such a hardware implementation. Convergence time for the breadboard prototype was measured to be approximately 10μsec. An integrated circuit implementation of the proposed deconvolution network was undertaken. An hierarchical approach was taken in the integrated circuit implementation to provide flexibility in the design of an interconnection matrix.

Several aspects of tactile sensing are of a very similar nature as problems encountered in computational vision. Essential differences lie in the fact that unlike vision sensors, which are remotely located with respect to their target, tactile sensors are required to be in physical contact with their targets. The 'deblurring' problem considered in this thesis is an example of a problem which also arises in computational vision. Edge detection is used in vision to determine boundaries within the visual field and is required in tactile sensing to identify physical edges

of objects, and locate holes to determine shape. A similar analogy can be drawn between motion detection in vision and identifying slippage in tactile sensing.

The set of processes that recover physical attributes of visible three dimensional objects from two-dimensional visual (intensity) images, is collectively termed as *early vision.* In a similar manner, the elements of the second hierarchical level of tactile sensing (described in Chapter 1) may be collectively termed *early touch.* Early touch then can be defined as the set of processes that recover the three-dimensional attributes of an object and properties of the established contact, from two-dimensional arrays of sensor measurements. As yet, the set of problems of which early touch is comprised have not been clearly defined, but recovery of stress over the contact region, edge detection, and identification of slippage are likely among them.

Most early vision problems are ill-posed and regularization has successfully been applied in solving many of them (see e.g. [35] and [3]). However, regularization as a technique for solving ill-posed problems has also demonstrated limitations. Among the limitations of standard regularization theory, is its inability to effectively cope with discontinuities [36]. If the operators K and P in equation (2.6.24) are linear (as in standard regularization theory), the solution space is essentially restricted to generalized splines. Hence in some cases the resultant solutions will be overly smooth and cannot be trusted at discontinuities such as edges. It has been suggested [37] that after standard regularization, locations where the resultant solution originates a 'large' error in the second (regularizing) term of equation (2.6.24) can be identified as locations of discontinuities. A second second stage

of regularization can then be performed, using the locations of discontinuities as boundary conditions. This approach requires the choice of a threshold for error in the regularizing term in order to identify discontinuities. Furthermore the task of locating discontinuities is hindered by the smoothing due to the first stage of regularization. Marroquin in [38] proposes a nonquadratic stabilizer which preserves discontinuities in reconstructions of surfaces from depth data by embedding prior knowledge about the geometry of discontinuities.

Another approach to overcome some of the limitations of standard regularization theory, is based on Bayesian estimation and Markov random field models of the image (tactile or visual). This approach (see [36] uses prior knowledge represented in terms of appropriate probability distributions instead of directly restricting the solution space. It can be shown [39] (see also [36] and [35]), that maximizing the *a posteriori* probability, is equivalent to minimizing an expression of the form (2.6.24). However, in this case the functional to be minimized is not in general quadratic as a whole.

It is clear that the class of problems for which quadratic variational principles are sufficient, is limited. For every quadratic variational principle, it can be shown that there exists a corresponding linear analog electrical network consisting of resistors, voltage sources, and current sources, which has the same solutions. This fact is used in [3] to synthesize analog resistive networks to solve problems in early vision. In [40], the approach taken to solving a nonquadratic variational principle employs a hybrid analog-digital network which at each iteration (on the digital time scale) solves a quadratic variational principle (in analog). In general,

nonquadratic variational principles may posses numerous local minima in addition to a global minimum (which may or may not exist). The deterministic gradient descent approach taken in [40] demonstrated an ability to perform well (qualitatively) in comparison to statistical annealing which converges to the global solution with probability one if appropriately applied. However, convergence to the global solution cannot be guaranteed and the hybrid analog-digital nature of the network introduces additional hardware complexity.

In this thesis, it was shown that a strictly analog network can be structered so as to solve a nonquadratic variational principle. Convergence to the global solution, in this case, is guaranteed since the variational principle of (2.7.27) is strictly convex. It was also shown that there exists a class of nonquadratic variational principles (see equation (4.2.14)) which can be solved by similar networks by choosing appropriate characteristics $g(\cdot)$ for the signal plane amplifiers. In the case of this larger class of nonquadratic variational principles, convergence to global solutions cannot in general be guaranteed since multiple minima may exist. Techniques such as adding noise to the network can be used to aid in escaping from local minima in an attempt to find the global solution. Two interesting questions arise in this context: (i)For any nonquadratic variational principle, under what conditions does there exist a (possibly nonlinear and active) analog network with the same solution? and (ii)How can optimality of the solutions be guaranteed?

So far the discussion in this thesis has been confined primarily to the problems of early touch. The highest level of the tactile sensing hierarchy, which has been ignored so far, is crucial to the usefulness of any tactile sensing system. A higher

level description of the tactile environ environment is the next step beyond the low-level description provided by the processes of early touch. For instance, although a low-level description of a grasped object may be sufficient to secure the object in a stable grasp, it is not adequate to directly identify the object. In this aspect of tactile sensing as well, neural networks may provide a solution. Adaptive neural networks have demonstrated a remarkable ability to 'learn' complex representations and successfully classify patterns based on these representations. Among other applications of such neural network classifiers are handwritten character recognition [32], identification of faces [31], and classification of superposed radar return signals [?]. Successful VLSI implementation of an adaptive neural network classifier for recognition of grasped objects may further reduce the computational load of the central processor.

It was shown by construction of a breadboard circuit (in Chapter 5) that analog hardware implementation of the network proposed in Chapter 4 leads to convergence times in the order of 10μsec. In order to compare this with digital computation, we note that simple inversion of an $n \times n$ symmetric Toplitz matrix, is of computational complexity $O(n^2)$. For the purpose of a biased comparison (biased in favor of digital computation), we can ignore the regularizer and assume that it takes exactly n operations to invert the matrix A. Then for a modest array of 25 tactile sensors, it would be necessary to perform,

$$25^2 \frac{operations}{solution} \times 10^5 \frac{solutions}{second} = 62.5 \frac{Million \ \ operations}{second}$$

in order to keep up with the processing speed of the analog network. This is clearly

not possible for local digital computation. Also as the size of the sensor array increases, the processing time required for digital computation increases quadraticaly. It was noted in Chapter 4 that in the case of the analog network, convergence time actually decreases as the size of the problem increases. In [2] it was established through survey that processing times of approximately 1-2ms are sufficiently fast for most tactile sensing applications. Additional available time resulting from the use of analog network processors could be utilized to perform other tasks such as predicting slippage based upon prior and current processing results. An initial approach to analog integrated circuit implementation of the deconvolution network was also discussed in Chapter 5. A hierarchical design strategy was employed to permit some flexibility in the pattern of interconnections in the network. Mating of the amplifier chip and resistive matrix chip in the manner discussed in Chapter 5 is not however optimal and this approach was only used for experimental purposes. Technologies such as 'Indium bump' bonding are more suitable for such a design and would permit greater density of circuitry on both the amplifier and resistive matrix chips. In this technology, the two chips would also be stacked one on top of the other, but connections between the two chips would be made by placing small 'bumps' of Indium in corresponding locations on both chips and then heating the whole structure which results in the Indium melting and alignment of corresponding bumps on both chips due to surface tension.

It is clear that there remain a great many unaddressed and unsolved problems in the area of tactile sensing. Tactile sensing has not received the attention of researchers to the same extent as vision has. As in vision, there is a need in tactile

sensing to identify and formalize the problems involved and then devise sensible solutions to these problems. We have suggested in this thesis that analog neural networks may provide a solution to some of the low-level processing aspects of tactile sensing and possibly even some of the higher level tasks such as object recognition. However, we have only addressed a small portion of a large and complex problem.

BIBLIOGRAPHY

[1] G. Gordon Ed., *Active Touch.* Oxford: Pergamon Press, 1978.

[2] L. Harmon, "Automated tactile sensing," *Intl. J. Robotics Research*, vol. 1, no. 2, pp. 3–32, 1982.

[3] T. Poggio and C. Koch, "Ill-posed problems in early vision: from computational theory to analogue networks," *Proc. R. Soc. London*, vol. B 226, pp. 303–323, 1985.

[4] J. Hadamard, *Lectures of the Cauchy Problem in Linear Partial Differential Equations.* New Haven, Conn.: Yale University Press, 1923.

[5] P. Dario, D. DeRossi, C. Domenici, and R. Francesconi, "Ferroelectric polymer tactile sensors with anthropomorphic features," in *Proceedings Intl. Conf. on Robotics, Atlanta, Ga.*, pp. 332–339, March 13-15 1984.

[6] A. Grahn and L. Astle, "Robotic ultrasonic force sensor arrays," Preprint. Bonneville Scientific, Salt Lake City, Utah.

[7] P. Dario and G. Butazzo, "An anthropomorphic robot finger for investigating artificial tactile perception," *Intl. J. of Robotics Research*, vol. 6, Fall 1987.

[8] M. Raibert, "An all digital vlsi tactile array sensor," in *Proceedings Intl. Conf. on Robotics, Atlanta Ga.*, pp. 314–319, March 13-15 1984.

[9] D. Mott, M. Lee, and H. Nicholls, "An experimental very high resolution tactile sensor array," in *Proceedings 4th International Conference on Robot Vision and Sensory Control, London*, 1984.

[10] C. T. Yao, M. C. Peckerar, J. Wasilik, C. Amazeen, and S. Bishop, "A novel three dimensional microstructure fabrication technique for a triaxial sensor," Preprint. Naval Research Laboratory, Code 6804, Washington D.C., 1987.

[11] J. J. Clark, "A magnetic field based compliance matching sensor for high resolution, high compliance tactile sensing," in *Proceedings IEEE Inter. Conf. On Robotics and Automation, Philadelphia, Pa.*, pp. 772–777, April 1988.

[12] S. Timoshenko and J. Goodier, *Theory of Elasticity.* New York: McGraw Hill, 1951.

[13] R. Yang, *Tactile perception for Multifingered Hands.* Master's thesis, Univ. Of Md., College Park, MD., 1987. Systems Research Center Tech. Report No. SRC-TR-87-126.

[14] I. Gohberg and S. Goldberg, *Basic Operator Theory.* Boston, MA.: Birkhauser, 1980.

[15] W. Rudin, *Real and Complex Analysis.* McGraw Hill, 1986.

[16] A. Tikhonov,*Sov. Math. Dokl*, vol. 4, pp. 1035–1038, 1963.

[17] A. Tikhonov and A. V.Y., *Solutions of Ill Posed Problems.* Washington, D.C.: Winston Press, 1977.

[18] J. C. Maxwell, "A treatise on electricity and magnetism," vol. I 3rd Ed., pp.407.

[19] B. Tellegen, "A general network theorem with applications," *Phillips Research Reports*, vol. 7, 1952.

[20] C. Cherry, "Some general theorems for nonlinear systems possesing reactance," *Phil. Mag.*, vol. 42, p. 1161, 1951.

[21] W. Millar, "Some general theorems for nonlinear systems possessing resistance," *Phil. Mag.*, vol. 42, p. 1150, 1951.

[22] R. Brayton and J. Moser, "A theory of nonlinear networks i and ii," *Quarterly Of Applied Mathematics*, vol. 22, pp. 1–33,81–184, 1964.

[23] T. Matsumoto, "On several geometic aspects of nonlinear networks," *Journal of the Franklin Institute*, vol. 301, January-February 1976.

[24] C. Desoer and F. Wu, "Trajectories of nonlinear rlc networks: a geometric approach," *IEEE Transactions on Circuit Theory*, vol. CT-19, November 1972.

[25] T. Matsumoto, "On dynamics of electrical networks," *J. of Differential Equations*, vol. 21, May 1976.

[26] T. Matsumoto, "Eventually passive nonlinear networks," *IEEE Transactions on Circuits and Systems*, vol. CAS-24, May 1977.

[27] S. Smale, "On mathematical foundations of electrical circuit theory," *Journal of Differential Geometry*, vol. 7, pp. 193–210, 1972.

[28] D. Tank and J. Hopfield, "Simple 'neural' optimization networks: an a/d converter, signal decision circuit and a linear programming circuit," *IEEE Transactions on Circuits and Systems*, vol. CAS-33, pp. 533–541, May 1986.

[29] C. Marrian and M. Peckerar, "Electronic neural net algorithm for maximum entropy deconvolution," in *Proceedings IEEE First Annual Inter. Conf. On Neural Networks, San Diego, Ca.*, June 1987.

[30] H. Conway and e. al., "Normal and shearing contact stress in indented strips and slabs," *Intl. J. Eng. Sci.*, vol. 4, pp. 343–359, 1966.

[31] M. Fan, J. Koninckx, L. Wang, and A. Tits, "Console:a cad tandem for optimization-based design interacting with arbitrary simulators," Technical Report, University of Maryland, Systems Research Center, 1987.

[32] Y. S. Abu-Mostafa and D. Psaltis, "Optical neural computers," *Scientific American*, pp. 88–95, March 1987.

[33] H. P. Graf, L. D. Jackel, and W. E. Hubbard, "Vlsi implementation of a neural network model," *IEEE Computer Magazine*, pp. 41–49, March 1988.

[34] F. Kub, I. Mack, K. Moon, C. Yao, and J. Modolo, "Programmable analog synapses for microelectronic neural networks using a hybrid digital-analog approach," in *Proceedings IEEE International Conference On Neural Networks, San Diego, Ca.*, July 1988.

[27] S. Smale, "On mathematical foundations of electrical circuit theory," *Journal of Differential Geometry*, vol. 7, pp. 193–210, 1972.

[28] D. Tank and J. Hopfield, "Simple 'neural' optimization networks: an a/d converter, signal decision circuit and a linear programming circuit," *IEEE Transactions on Circuits and Systems*, vol. CAS-33, pp. 533–541, May 1986.

[29] C. Marrian and M. Peckerar, "Electronic neural net algorithm for maximum entropy deconvolution," in *Proceedings IEEE First Annual Inter. Conf. On Neural Networks, San Diego, Ca.*, June 1987.

[30] H. Conway and e. al., "Normal and shearing contact stress in indented strips and slabs," *Intl. J. Eng. Sci.*, vol. 4, pp. 343–359, 1966.

[31] M. Fan, J. Koninckx, L. Wang, and A. Tits, "Console:a cad tandem for optimization-based design interacting with arbitrary simulators," Technical Report, University of Maryland, Systems Research Center, 1987.

[32] Y. S. Abu-Mostafa and D. Psaltis, "Optical neural computers," *Scientific American*, pp. 88–95, March 1987.

[33] H. P. Graf, L. D. Jackel, and W. E. Hubbard, "Vlsi implementation of a neural network model," *IEEE Computer Magazine*, pp. 41–49, March 1988.

[34] F. Kub, I. Mack, K. Moon, C. Yao, and J. Modolo, "Programmable analog synapses for microelectronic neural networks using a hybrid digital-analog approach," in *Proceedings IEEE International Conference On Neural Networks, San Diego, Ca.*, July 1988.

[35] C. Marcus and R. Westervelt, "Stability of analog neural networks with delay," In Press. Physical Review A.

[36] C. R. K. Marrian, M. C. Peckerar, I. Mack, and Y. C. Pati, "Electronic neural net for solving ill-posed problems with anentropy regularizer," in *Proceedings 8th Inter. Maximum Entropy Workshop*, (Cambridge, U.K.), Kluwer, 1988. J. Skilling Editor.

[37] T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature*, vol. 317, no. 6035, pp. 314–319, 0000.

[38] J. Marroquin, S. Mitter, and T. Poggio, "Probabilistic solution of ill-posed problems in computational vision," *Journal of the American Statistical Association*, vol. 82, p. , March 1987.

[39] D. Terzopoulus, *Multiresolution Computation of Visible-Surface Representation.* PhD thesis, Masachusettes Institute of Technology, Cambridge, MA., 1984. Ph.D. Thesis, Department of Electrical Engineering and Computational Science.

[40] J. Marroquin Memo 792, MIT, Artificial Intelligence Lab, 1984.

[41] J. Marroquin, "Probabilistic solution of inverse problems," Technical Report 860, MIT, Artificial Intelligence Lab, 1985.

[42] J. Hutchinson, C. Koch, J. Luo, and C. A. Mead, "Computing motion using analog and binary resistive networks," *IEEE Computer Magazine*, pp. 52–63, March 1988.

[43] A. Teolis, Y. C. Pati, M. C. Peckerar, and S. Shamma, "Cascaded neural networks for classification of superposed radar return signals in the presence of noise," To be published. Naval Research Laboratory, Code 6804, Washington D.C. and Systems Research Center, Univ. of Md., College Park, MD., 1988.

[44] Y. C. Pati, D. Friedman, P. S. Krishnaprasad, C. T. Yao, M. Peckerar, R. Yang, and C. R. K. Marrian, "Neural networks for tactile perception," in *Proceedings IEEE Inter. Conf. On Robotics and Automation, Philadelphia, Pa.*, pp. 134–139, April 1988.

[45] J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci., U.S.A.*, vol. 79, pp. 2554–2558, April 1982.

[46] D. Tank and J. Hopfield, "Neural computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, pp. 141–152, 1985.

[47] M. W. Hirsch, "Convergence in neural nets," in *Proceedings IEEE International Conference On Neural Networks, San Diego, Ca.*, June 1987.

[48] J. Hopfield, "Neurons with graded responses have collective computational properties like those of two-state neurons," *Proc. Natl. Acad. Sci., U.S.A.*, vol. 81, pp. 3088–3092, May 1984.

[49] L. Jackel, R. Howard, H. Graf, B. Straughn, and J. Denker, "Artificial neural networks for computing," *J. Vac. Sci. Technol. B*, vol. 4, pp. 61–63, January/February 1986.

[50] R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, pp. 4–22, April 1987.

[51] D. Schwartz, R. Howard, J. Denker, R. Epworth, H. Graf, W. Hubbard, L. Jackel, B. Straughn, and D. Tennant, "Dynamics of microfabricated electronic neural networks," *Appl. Phys. Lett*, vol. 50(16,20), April 1987.

[52] L. Harmon, "Touch sensing technology: a review," Tech. Rep., Society of Manufacturing Engineers, One SME Drive, P.O. Box 930, Dearborn, Mi. 48128, 1980. Technical Report MSR80-03.

[53] R. Fearing and J. Hollerback, "Basic solid mechanics for tactile sensing," *Intl. J. of Robotics Research*, vol. 4, Fall 1985.

[54] E. Guillemin, *Theory of Linear Physical Systems.* New York: John Wiley and Sons, Inc., 1963.