

THESIS REPORT

Ph.D.

Analysis and Synthesis of Distributed Systems

by Y. Zhuang

Advisor: J.S. Baras

Ph.D. 94-5



*Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry*

Analysis and Synthesis of Distributed Systems

by

Yan Zhuang

Dissertation submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1994

Advisory Committee:

Professor John S. Baras, Chairman/Advisor
Professor P.S. Krishnaprasad
Professor Carlos A. Berenstein
Professor Eyad H. Abed
Assistant Professor K.J. Ray Liu

Abstract

Title of Dissertation: Analysis and Synthesis of Distributed Systems

Yan Zhuang, Doctor of Philosophy, 1994

Dissertation directed by: Professor John S. Baras

Martin Marietta Chair in Systems Engineering

Department of Electrical Engineering

University of Maryland, College Park

We first model and analyze distributed systems including distributed sensors and actuators. We then consider identification of distributed systems via adaptive wavelet neural networks (AWNNs) by taking advantage of the multiresolution property of wavelet transforms and the parallel computational structure of neural networks. A new systematic approach is developed in this dissertation to construct an optimal discrete orthonormal wavelet basis with compact support for spanning the subspaces employed for system identification and signal representation. We then apply a backpropagation algorithm to train the network to approximate the system. Filter banks for parameterizing wavelet systems are studied. An analog VLSI implementation architecture of the AWNN is also given in this dissertation. This work is applicable to signal representation and compression under optimal orthonormal wavelet bases in addition to progressive system identification and modeling. We anticipate that this work will find future applications in signal processing and intelligent systems.

©Copyright by

Yan Zhuang

1994

Dedication

To my parents

Acknowledgments

I would like to express my sincere gratitude to my thesis advisor Professor John S. Baras for creating more opportunities to explore diverse new areas of research than I have ever imagined. I am grateful for the academic freedom he provided me and for his guidance, trust, support and encouragement throughout these years. His broad knowledge and vision make any adventure a fun.

I would like to thank the other members of the Advisory Committee, professors Krishnaprasad, Abed, Berenstein, and Liu for providing many useful comments and suggestions regarding this dissertation. I also thank the other faculty members and staff members, at the Institute for Systems Research of the University of Maryland, who have influenced and helped me in different aspects of life here.

My fellow graduate students, at the Institute for Systems Research, also deserve thanks for making these post-graduate years more enjoyable especially at times of difficulties. I thank Emmanuil Frantzeskakis, Bernard Frankpitt and M. Kemal Sönmez among them in particular.

Last, but certainly not the least, I am very grateful and indebted to my parents for teaching me many valuable theorems of life and serving as a constant source of love, encouragement, caring and understanding for so many years. To them, with the deepest respect and love, I would like to dedicate this thesis.

Table of Contents

<u>Section</u>	<u>Page</u>
1 Introduction	1
1.1 Background	1
1.2 Contributions of the Thesis	3
1.3 Outline of the Thesis	4
2 Modeling and Synthesis of Distributed Systems	5
2.1 Introduction	5
2.2 System Model	9
2.2.1 Distributed Actuator Made of Smart Materials	9
2.2.2 Model of the Flexible Structure with Actuators	11
2.2.3 Sensor Model	13
2.2.4 The Model of the Composite System	15
2.3 Control Algorithm	16
2.3.1 Background on Stabilization of a Thin Plate	16
2.3.2 Control Algorithm	18
2.4 Modal analysis	21
2.4.1 Galerkin Procedure for Discretization	21
2.5 Multiresolution System Identification and Monitoring	25

2.6	Conclusions	26
3	Optimal Design of Sensors and Actuators	28
3.1	Introduction	28
3.2	Problem formulation	30
3.3	Optimal Control and a Numerical Algorithm	32
3.3.1	Optimal Control	33
3.3.2	A Numerical Algorithm	34
3.3.3	Numerical Examples	35
3.4	Other Considerations	40
3.5	Conclusions	41
4	Compactly Supported Wavelets and QMF banks	49
4.1	Introduction	49
4.2	Background Review	50
4.2.1	Wavelet Transform	50
4.2.2	Multiresolution Approximation	51
4.2.3	Decimators and Expanders in Multirate Systems	53
4.2.4	Decimation Filters and Interpolation Filters	55
4.3	QMF Banks and Wavelets	57
4.3.1	Quadrature Mirror Filter Banks	57
4.3.2	Orthonormal Wavelet Basis and QMF	60
4.4	Conclusion	65
5	Optimal Wavelet Basis for Signal Representation	66
5.1	Introduction	66

5.2	Parameterization of Wavelet Functions and Information Measures	67
5.3	Sensitivity Gradient of Wavelet Components	72
5.4	Analysis of Time and Frequency Allocation	75
5.5	Algorithms	76
5.6	Structure Analysis and System Parameterization	79
5.6.1	Decomposition of Unitary Matrices	79
5.6.2	Givens Rotation and QMF Lattice Structures	82
5.6.3	The Existence of the Optimal Wavelet Basis	84
5.7	Simulated Annealing for Global Optimization	85
5.7.1	Stochastic Relaxation	86
5.7.2	Simulated Annealing	87
5.8	Conclusion	90
6	System Identification via Adaptive Wavelet Neural Networks	92
6.1	Introduction	92
6.1.1	Background in System Identification	93
6.1.2	Computational Structures of Neural Networks and AWNN	94
6.2	Problem Statement	96
6.2.1	Structure of AWNN	96
6.2.2	Optimal Wavelet Basis and AWNN	98
6.2.3	Dynamics of Neurons and AWNN	99
6.3	Network Training	102
6.4	Structured Learning via Vector Quantization and Feature Extrac- tion	108
6.4.1	Optimal Wavelet Basis versus Vector Quantization	108

6.4.2	A Classifier Based Learning Structure	109
6.5	Conclusions	110
7	Analog VLSI Architecture Design for AWNN	112
7.1	Introduction	112
7.2	Design Modules for Analog VLSI Implementation	114
7.2.1	An Analog Neuron Structure	114
7.2.2	Vector Multiplier for Weight Adaptation	115
7.2.3	Design Modules for AWNN Systems	117
7.3	Adaptive Learning Scheme via Analog VLSI Structures	118
7.4	Feasibility of Analog AWNN	120
8	Conclusions and Future Research	122

List of Figures

<u>Number</u>		<u>Page</u>
2.1	The composite beam	9
2.2	The PVDF sensor	13
3.1	Discretizing the sensor and actuator layout	34
3.2	Optimization algorithm	36
3.3	The initial displacement of the beam	38
3.4	Time responses of coordinates $d_k(t)$ and $q_k(t)$ without damping	42
3.5	Time-space responses of displacement $w(x, t)$ (m) and $\Phi(x, t)$ (rad) versus x (m) and t (sec) without damping	43
3.6	System energy $E(t)/mean(E(t))$ versus t (sec) without damping	44
3.7	Time responses of coordinates $d_k(t)$ and $q_k(t)$ with active damp- ing; $a= 0.15$ m, $b=0.0$	44
3.8	Time-space responses of displacement $w(x, t)$ (m) and $\Phi(x, t)$ (rad) versus x (m) and t (sec) with active damping; $a = 0.15$ m, $b = 0$	45
3.9	Control $V(t)$ (V) with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$	45

3.10	System Energy $E(t)/\text{mean}(E(t))$ with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$	46
3.11	Time responses of coordinates $\dot{d}_k(t)$ and $q_k(t)$ with active damping; $a = 0$, $b = 0.15$ m	46
3.12	Time-space responses of displacement $w(x, t)$ (m) and $\Phi(x, t)$ (rad) versus x (m) and t (sec) with active damping; $a = 0$, $b = 0.15$ m	47
3.13	Control $V(t)$ (V) with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$	47
3.14	System Energy $E(t)/\text{mean}(E(t))$ with respect to time t (sec) with active damping; $a = 0.15$, m $b = 0$	48
4.1	The decimator and expander	53
4.2	Decimation for $M = 2$	54
4.3	Demonstration of the expander for $L = 2$	54
4.4	Decimation filter and its response	56
4.5	Interpolation filter and its response	56
4.6	Noble identities for multirate systems	57
4.7	Quadrature mirror filter bank (a) and its response (b)	58
4.8	(a) Wavelet analysis QMF bank and (b) its equivalent four channel system	63
4.9	(a) Wavelet synthesis QMF bank and (b) its equivalent four channel system	63
5.1	Mesh structure of the projection space	70
5.2	Flow chart for computing $\psi(t)$	75

5.3	Block structure of the unit factor	83
5.4	A structure block of Givens rotation	83
6.1	A wavelet neural network identification structure.	96
6.2	An AWNN block.	97
6.3	Separation of a neural network as discrimination and conversion parts. (a) A neural network block. (b) The structure of discrim- ination and conversion.	100
6.4	Feature extraction and classification.	101
6.5	A two layer AWNN emulator.	102
6.6	AWNN training structure.	103
6.7	A tree structured learning scheme.	109
7.1	A neuron implementation structure with input stage, transfer stage and output stage.	115
7.2	A realization of a vector multiplier	116
7.3	Analog VLSI blocks: (a) A vector multiplier. (b) A scalar multi- plier.	117
7.4	An analog VLSI architecture for AWNN	119

Chapter 1

Introduction

1.1 Background

Modeling complex distributed systems, such as flexible structure systems, is a challenge due to their infinite dimensional nature. Extensive efforts have been made to model these systems [7] [6] [5] and their actuators [6] [15] [32]. However, even a reasonable mathematical model may involve a large number of coupled variables, nonlinearities and complicated boundary conditions. When the system is equipped with many sensors and actuators of both distributed and discrete types, monitoring the system and processing sensory data can become difficult if not prohibitive. New techniques need to be developed for system modeling, system identification and sensory data processing.

We have found recent advances in wavelet theory applicable in generating an progressive modeling structure used in both system identification and signal approximation in $L^2(R)$. There has been extensive research interest and activities in wavelet theory and its applications in recent years [12] [19], mainly due to the fact that wavelet transforms decompose a signal into components at different resolutions. Wavelet transforms actually simplify the description of signals and provides analysis at different levels of detail. There are some successful

applications of these properties in the fields of signal processing [43], speech processing and especially in image processing [34]. It was shown [35] that it is possible to derive a mother wavelet function $\psi(x) \in L^2(R)$ such that for $j, l \in Z$, $\{\psi_{j,l}(x)\}_{j,l \in Z}$ with

$$\psi_{j,l}(x) = \sqrt{2^j} \psi(2^j x - l) \quad (1.1)$$

is an orthonormal basis of $L^2(R)$. Any square integrable function $f(x) \in L(R^2)$ can be represented as

$$f(x) = \sum_{j,l} w_{j,l} \psi_{j,l}(x). \quad (1.2)$$

This property is very useful in identification of infinite dimensional systems.

Another useful approach to modeling and system identification is using artificial neural networks which provide distributed computational structures. It has been shown [17] that two layer neural networks can approximate any nonlinear function to an arbitrary degree of accuracy. This approach has been successfully applied to many other fields including pattern classification.

Motivated by the latest developments in wavelet theory and neural networks, we develop new approaches for system identification and progressive modeling and introduce the concept of adaptive wavelet neural networks (AWNNs) for signal representation and system identification. We also develop an optimization scheme for generating the optimal wavelet basis forming the AWNN based on the unknown systems or given signals. AWNNs are applicable to system identification and signal representation.

1.2 Contributions of the Thesis

This thesis presents the results of interdisciplinary research regarding distributed systems using theories and methodology mainly from signal processing and control systems.

- We developed models of distributed systems, flexible structures, proposed an algorithm for designing a layout of the distributed sensors and actuators. This work will facilitate modeling and control of complex flexible systems.
- We worked out a progressive system identification scheme which can be used for model verification and system monitoring.
- We developed a systematic approach to construct a signal based optimal wavelet basis with compact support for signal representation and system approximation.
- We utilized the multiresolution property of wavelet transforms and the distributed computational structure of neural networks and developed the concept and structure of Adaptive Wavelet Neural Networks; AWNN would find application in system identification, signal representation and compression.
- We also demonstrated the feasibility of implementing AWNN via analog VLSI circuits for future real time implementation.

We anticipate that the methodology and concepts developed in the dissertation can find their applications in the areas of signal processing and intelligent systems.

1.3 Outline of the Thesis

This thesis is organized as follows:

Chapter 2 discusses the modeling and synthesis of distributed systems. It deals with a component of flexible structures and analyzes the effects of different sensors and actuators to the system. It also introduces the concept of multiresolution system identification and monitoring of complex systems.

Chapter 3 studies the issue of shaping distributed sensors and actuators to maximize active damping to the flexible system. A design algorithm is also given with the model derived from the previous chapter.

Chapter 4 reviews the connection between compactly supported wavelets and QMF banks. It demonstrates that compactly supported wavelets can be constructed through designing a low pass filter subject to certain constraints.

Chapter 5 provides a systematic approach for constructing an optimal wavelet basis with compact support; such a wavelet basis would be suitable for signal representation and system identification. The existence of the optimal wavelet basis is proved in this chapter.

Chapter 6 introduces Adaptive Wavelet Neural Networks (AWNNs) for identification of infinite dimensional systems and for signal representation. Structured learning is also discussed regarding real time adaptive learning with variable structures.

Chapter 7 demonstrates the feasibility of implementing AWNN systems via analog VLSI circuits. Design modules and system architectures are addressed in term of aggregating the design modules into an AWNN system.

The last chapter suggests future research directions and concludes the thesis.

Chapter 2

Modeling and Synthesis of Distributed Systems

In this chapter, we discuss modeling and synthesis of a distributed system in the form of flexible structures. We also point out the necessity for identification and verification schemes for distributed systems.

2.1 Introduction

Distributed systems cover a wide range of systems in many fields ranging from physics, mathematics to engineering. They can be categorized in a variety of ways depending on the circumstance of the physical world and the fields in which they are used. Most obviously, from their names, one can naturally recognize those spatially distributed or physically continuous plants in the real world as distributed systems. For example, fluid dynamics, thermal dynamics belong to the domain of distributed systems since they deal with continuous media in their subjects. Another example can be image processing, the basic units of a digitized image to be treated are pixels which are spatially distributed; so in some sense, the image system is also a distributed system. In dynamics, when objects of a system can not be treated precisely as rigid bodies and elastic effects

are considered, we say that the system is a distributed system.

Flexible structures are examples of distributed systems and can be modeled in different ways, such as finite element models, discrete models and continuous models governed by partial differential equations. Flexible structure control and stabilization have long been a field of research. Its applications can be found in many areas such as vibration suppression in mechanical engineering and structure control in civil engineering. One of the challenges comes from applications in aerospace systems and robotics. Due to the limited launching load, space structures are usually large in size, light in mass and hence weakly damped. In order to implement attitude control of increasing pointing precision, active damping is required to enhance the system's stability. Problems in controller design for large flexible space structures include quickly damping out the pointing errors resulting from step disturbances, or nonzero initial conditions (e.g., resulting from slewing) and maintaining the desired attitude as close as possible in the presence of disturbances.

Modeling these systems, with structure damping, appropriate interface conditions, different geometric configurations and various composite materials, is a challenging research topic. Even a reasonable mathematical model may involve a large number of coupled variables, nonlinearities and complicated boundary conditions. It is a formidable task to obtain concrete representations of optimal feedback operators or to estimate the decay rate induced [30, p7]. It is natural that current research in dynamic control of flexible structures mainly focuses on basic components such as string, beam and plate of the whole structure. For example, Kim and Renardy [27] proved that the Timoshenko beam can be uniformly stabilized by means of boundary control. Morgül demonstrated that

a dynamical boundary and a dynamical boundary torque can cause the vibration of a Euler-Bernoulli beam to decay exponentially based on a Lyapunov functional based energy functional. Lagnese [30] presented a systematic study of uniform stabilization of the motion of a thin plate through the action of forces and moments applied at the edge of the plate. However, when the system becomes more complicated or is equipped with a large number of sensors and actuators of both distributed and discrete types, monitoring the system and processing sensory data can become difficult. New techniques need to be developed to deal with system modeling, system identification and sensory data processing.

The application of smart materials make it possible to implement distributed sensing and control of distributed systems. The so called “smart material” is one whose constitutive property, i.e., its strain and stress relation, is influenced by the external electric field applied to it. Bonding or embedding segmented elements of such materials in a structure would allow the application of localized strain developed in them to be transferred to the structure to control the deformation. The dynamical behavior of the composite structure can be adjusted by implementing appropriate control algorithms. Distributed sensors are bonded to a beam in a similar way. Typically, the output voltage from a sensor is a functional of the beam curvature[32] [47]. Bailey and Hubbard [4] used distributed actuators to control the vibration of a cantilever Euler-Bernoulli beam. Lyapunov’s second method was used to design the control algorithm. Piezoelectric actuators were also employed as elements of intelligent structures by Crawley and de Luis [15] and the corresponding static and dynamical analytic models were derived for segmented piezoelectric actuators. The actuator used

is spatially distributed, made of piezoelectric ceramic material (PZT) which is glued to the beam. The studies above were all based on Euler-Bernoulli beam model without considering shear effect and rotational inertia of the beam. The study of Lee and Moon [32] showed through modal equations that distributed sensors and actuators could be designed to measure or excite specific modes of one-dimensional plates and beams. Cudney [16] conducted research on control of the Timoshenko beam with four added layers of sensors and actuators with rate feedback for system damping. Tzou [47] provided some detailed explanation of the nature of piezoelectric actuators and sensors. Despite the research efforts above, there is still a need to systematically find the optimal layout of the distributed sensors and actuators for a specific application and to design an efficient sensing processing network in the presence of complicated structures. Attempting to fill the gap, we studied active damping problems using distributed actuators and sensors with the rotational inertia included in the beam model [9]. We also worked on shape optimization [53] for distributed sensors and actuators in order to maximize the damping effect.

In the rest of this chapter, we shall first discuss the modeling of a composite Timoshenko beam with distributed actuators and sensors. We then discuss a stabilizing control algorithm through introducing active damping to the system. We finally investigate the actuator and sensor shapes and their impact on the system's elastic modes. Research presented on flexible structure in this dissertation differs from the previous work in the sense of considering the optimal design and analysis of sensor and actuator shape functions with respect to the Timoshenko beam. Throughout our discussion on distributed systems in this dissertation, we shall use flexible structure systems as examples for treatment

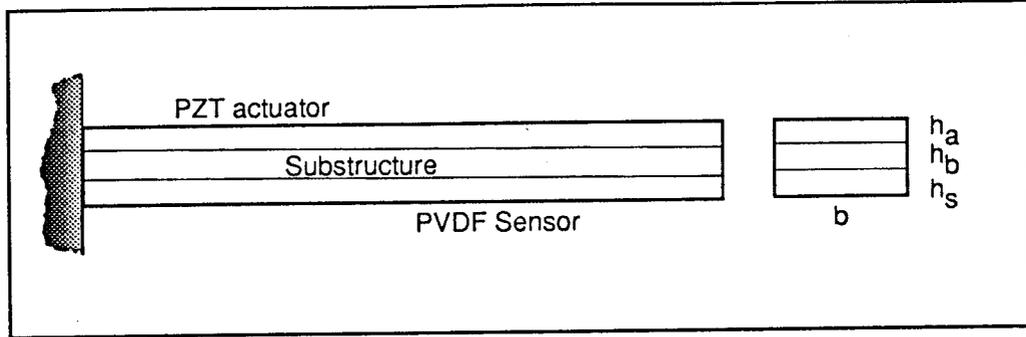


Figure 2.1: The composite beam

with our methodology.

2.2 System Model

The system model consists of a beam substructure with an actuator layer and a sensor layer glued on both sides of the beam. We formulate the model of the composite beam in this section.

2.2.1 Distributed Actuator Made of Smart Materials

One approach to build the desired distributed actuator is to take advantage of the special constitutive properties of certain materials. The actuation produced is due to the property change of the material subject to certain stimulation other than the external actuation force. Since the actuator can be built into the structure the overall structure design can be optimized. One of the advantages of using smart materials as sensors and actuators is that that the after-fact structure change and additional weight can be avoided.

Figure 2.1 shows a simplified structure of a composite cantilever beam with a sensor and an actuator layers glued on both sides of the beam. The actuator layer is made of PZT material while the sensor layer is made of piezoelectric

polymer polyvinylidene fluoride (PVDF) material. In this figure, h_a , h_b and h_s stand for the thickness of the actuator layer, the substructure layer and the sensing layer of the composite beam respectively. The subscripts a , b and s denote quantities and variables associated with the actuator, beam and sensor respectively. We use b to denote the width of the composite beam. For the convenience of our analysis, we assume a perfect bonding condition; we omit the shear effect between the actuator layer and its substructure and that between the sensor layer and the substructure.

The model of the actuator comes from the stress-strain relation of the piezoelectric materials. This relation is similar to that of the thermoelastic materials, with the thermal strain term replaced by the piezoelectric strain Λ . The constitutive equation of the actuator is given by [47]

$$\sigma = E_a(\epsilon - \Lambda) \quad (2.1)$$

where Λ is the actuation strain of the PZT due to an external electric field, and ϵ is the strain of PZT without the external electric field, E_a is the Young's modulus of the PZT and σ represents the PZT stress induced by the strain and the actuation. The actuation strain is given by [4] and [15] as

$$\Lambda(x, t) = \frac{d_{31}}{h_a} V(x, t) \quad (2.2)$$

where d_{31} is the piezoelectric field and strain field constant. $V(x, t)$ is the distributed voltage. The induced strain has the following two effects on the composite beam [4]: One effect is that it introduces a longitudinal strain ϵ_l to insure a force equilibrium along the axial direction. The steady state value of ϵ_l can be derived by solving a force equilibrium equation. The second effect is that the net force in each layer acts through the moment arm with the length from the mid-

plane of the layer to the neutral plane of the beam. The result of the actuation produces the bending moment which is introduced as the control mechanism. Since the composite beam model here has a similar geometric configuration as that in [4], we can express the actuation moment using the linear model used by Bailey [4]

$$M_a = K_a \Lambda(x, t) \quad (2.3)$$

where K_a is a constant depending on the geometry and the materials of the beam. The actuation from the PZT actuator appears as a bending moment which is proportional to the actuation strain of the actuator layer.

2.2.2 Model of the Flexible Structure with Actuators

We use the Timoshenko beam model [46, p331] to describe the dynamical behavior of the beam. Unlike the Euler-Bernoulli beam model, the Timoshenko model contains the rotational inertia and shear effects of the actual beam. The analysis of the latter is more complicated. The beam model is given as

$$\rho A \frac{\partial^2 w}{\partial t^2} = kAG \left(\frac{\partial^2 w}{\partial x^2} - \frac{\partial \Phi}{\partial x} \right), \quad (2.4)$$

$$\rho I \frac{\partial^2 \Phi}{\partial t^2} = EI \frac{\partial^2 \Phi}{\partial x^2} + kAG \left(\frac{\partial w}{\partial x} - \Phi \right). \quad (2.5)$$

Here t is the time variable and x is the space coordinate along the beam in its equilibrium configuration. $w(x, t)$ is the displacement of the centroid from its equilibrium line which is described by $w = 0$. Φ denotes the deflection curve when the shearing force is neglected. The total slope of deflection is

$$\frac{dw}{dx} = \Phi + \beta,$$

where β is the angle of shear. ρ, I and E are mass density, moment of inertia of cross section and Young's modulus respectively. k is a numerical factor depending on the shape of the beam while A and G are area of cross section and modulus of elasticity in shear.

The bending moment of the composite beam without actuation is

$$M_b = EI \frac{\partial \Phi}{\partial x}, \quad (2.6)$$

where

$$EI = E_a I_a + E_b I_b + E_s I_s. \quad (2.7)$$

The bending moment of the beam with actuation moment is

$$M = M_a + M_b \quad (2.8)$$

Using this augmented moment to replace the moment term in the original Timoshenko beam equations, we obtain equations of motion

$$\rho A \frac{\partial^2 w}{\partial t^2} = kAG \left(\frac{\partial^2 w}{\partial x^2} - \frac{\partial \Phi}{\partial x} \right), \quad (2.9)$$

$$\rho I \frac{\partial^2 \Phi}{\partial t^2} = EI \frac{\partial^2 \Phi}{\partial x^2} + kAG \left(\frac{\partial w}{\partial x} - \Phi \right) + c \frac{\partial V}{\partial x} \quad (2.10)$$

with boundary conditions

$$\begin{aligned} w(0, t) &= 0, & \Phi(0, t) &= 0, \\ \frac{\partial w(L, t)}{\partial x} - \Phi(L, t) &= 0, & EI \frac{\partial \Phi(L, t)}{\partial x} &= 0, \end{aligned} \quad (2.11)$$

where

$$c = \frac{d_{31}}{h_a} K_a.$$

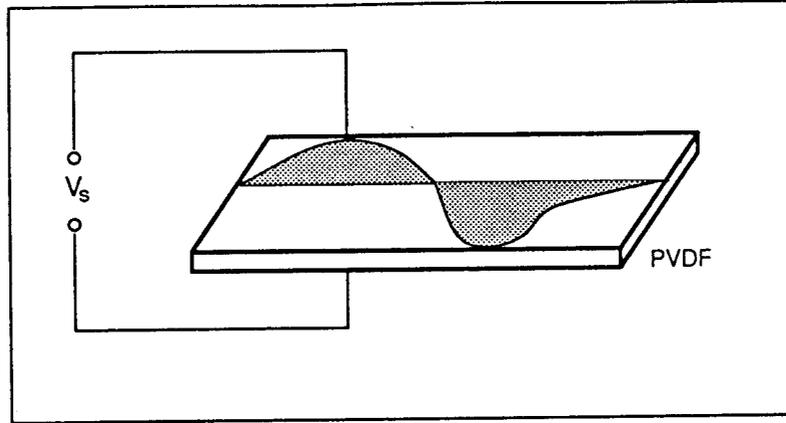


Figure 2.2: The PVDF sensor

The distributed voltage $V(x, t)$ is the control applied to the system. Equations (2.9) (2.10) and boundary condition (2.11) form the control system model. The constant c is determined by the PZT material property and the manufacturing process. The actuation appears in the system in the form of a distributed bending moment in the rotational equation (2.10).

2.2.3 Sensor Model

A spatially distributed sensor is modeled here to provide the sensing signal for the control system in consideration. The distributed sensor is the one whose output is a function of structural responses at different locations. It can be a group of point sensors or a spatially continuous one or the combination of both. These structural responses are obtained either discretely or continuously in space. Using the latter has the advantage of simplifying the complicated computations based on the point measurements since the sensor geometry can be tailored to provide the necessary computation. The spatial aliasing from an array of sensors can be avoided. Typical noncausal sensor dynamics such as gain rolloff without phase shift is possible by using distributed sensors [14].

PVDF is strain sensitive and relies on the applied strain to produce electrical charge. Figure 2.2 shows the structure of the PVDF sensor where the shaded area represents the region covered by electrodes. Only those regions covered by the electrodes play an active role in collection of electric charge due to beam bending. The amount of electrical charge is proportional to the amount of strain induced by the structure. It is also based on the constitutive property of the sensor material. This process is the reverse of piezoelectric actuation [47]. The induced charge per unit length from the strain is

$$q(x, t) = -E_s d_{31} \epsilon_s. \quad (2.12)$$

where E_s is Young's modulus of the sensor material.

The sensor strain is related to the beam curvature by

$$\epsilon_s = \frac{h_b + h_s}{2} \frac{\partial \Phi}{\partial x}. \quad (2.13)$$

This is based on the assumption that the neutral layer of bending is close to the geometric centroid of the composite beam. The electrical charge along the beam is

$$\begin{aligned} Q(x, t) &= \int_0^x q(x, t) F(x) dx \\ &= -E_s d_{31} \frac{h_b + h_s}{2} \int_0^x F(x) \frac{\partial \Phi}{\partial x} dx \end{aligned} \quad (2.14)$$

where $F(x)$ is the shape function of the sensor. The value of $F(x)$ is the local width of the electrodes covering both sides of the sensor material. We may also call $F(x)$ weight function of the sensor since it describes the amount of measurement taken by the distributed sensor at position x along the beam. The output of the sensor is

$$\begin{aligned}
V_s(x, t) &= \frac{Q(x, t)}{C} \\
&= -K_s \int_0^x F(x) \frac{\partial \Phi}{\partial x} dx
\end{aligned} \tag{2.15}$$

where

$$K_s = \frac{E_s d_{31} (h_b + h_s)}{2C} \tag{2.16}$$

is a constant with C being the capacitance between the electrodes of the sensor layer. Suppose the sensor covers the whole beam, then we can drop the spatial variable in the sensor output and have

$$V_s(t) = -K_s \int_0^L F(x) \frac{\partial \Phi}{\partial x} dx. \tag{2.17}$$

Equation (2.17) is the sensor output equation. The output voltage is the weighted integral of the beam curvature. Integrating by parts the right hand side of the above expression once in the spatial variable, we have another form of the sensor output,

$$V_s(t) = -K_s \Phi(L, t) F(L) + K_s \int_0^L \Phi(x, t) \frac{\partial F(x)}{\partial x} dx. \tag{2.18}$$

We shall observe later that the format of Equation (2.18) can be manipulated according to our control needs.

2.2.4 The Model of the Composite System

From the previous analysis, the model of the composite system is expressed by the following equations of motion

$$\rho A \frac{\partial^2 w}{\partial t^2} = kAG \left(\frac{\partial^2 w}{\partial x^2} - \frac{\partial \Phi}{\partial x} \right), \tag{2.9}$$

$$\rho I \frac{\partial^2 \Phi}{\partial t^2} = EI \frac{\partial^2 \Phi}{\partial x^2} + kAG \left(\frac{\partial w}{\partial x} - \Phi \right) + c \frac{\partial V}{\partial x} \tag{2.10}$$

with boundary conditions

$$\begin{aligned} w(0, t) &= 0, & \Phi(0, t) &= 0, \\ \frac{\partial w(L, t)}{\partial x} - \Phi(L, t) &= 0, & EI \frac{\partial \Phi(L, t)}{\partial x} &= 0 \end{aligned} \quad (2.11)$$

and the distributed sensor model

$$V_s(t) = -K_s \int_0^L F(x) \frac{\partial \Phi}{\partial x} dx \quad (2.17)$$

and appropriate initial conditions (IC) to be specified in the next section. The first equation describes displacement of the beam while the second equation shows the rotational movement of the cross section. The sensor function yields a voltage which is a weighted distributed measurement of the curvature of the beam. Up to this point, we have the completed composite beam model for further analysis.

2.3 Control Algorithm

2.3.1 Background on Stabilization of a Thin Plate

Lagnese studied the stabilization of a thin plate modeled by the Mindlin-Timoshenko model [30, p17] which is a two dimensional case of the Timoshenko beam model. The Mindlin-Timoshenko plate model consists of one displacement state $w(x, y, t)$ of the centroid plane, and two rotation states $\phi(x, y, t)$ and $\psi(x, y, t)$ of the cross section with respect the two spatial variable x and y , respectively. Reducing this model to an one dimensional one, we are going to have the standard Timoshenko beam model. In Lagnese's work, an energy functional was used to account for the kinetic energy due to displacement and bending and for the potential energy due to deformation from bending and shear as well. The

energy functional is used to measure the amount of vibration of the plate. The time derivative of the energy functional is given by [30, p27]

$$\frac{dE(t)}{dt} = \int_{\Gamma} [m_1 \frac{d\psi}{dt} + m_2 \frac{d\phi}{dt} + m_3 \frac{dw}{dt}] d\Gamma \quad (2.19)$$

where m_1 , m_2 and m_3 are two bending moments and a force applied to the boundary Γ of the beam. When the control is defined by

$$[m_1, m_2, m_3]' = -F \left[\frac{d\psi}{dt}, \frac{d\phi}{dt}, \frac{dw}{dt} \right]', \quad (2.20)$$

with F being a 3×3 matrix which is positive semidefinite on the boundary Γ , then the resulting closed-loop system is dissipative in the sense that $E(t)$ is non increasing. The following theorem provides conditions for the decay of system energy.

Theorem 2.3.1 (Lagnese [30], p38) *Assume that the boundary Γ is Lipschitz and the plate is clamped and assume that the gain matrix F of (2.20) is positive definite on some nonempty, open subset of Γ . Then every finite energy solution of the Mindlin-Timoshenko plate [30, p26-27] with control given by (2.20) satisfies*

$$\lim_{t \rightarrow \infty} E(t) = 0. \quad (2.21)$$

This theorem states that the vibration of Mindlin-Timoshenko plates can be asymptotically stabilized by boundary control. Similar results can be derived from reducing the model to a one dimensional one for the Timoshenko beam model. In this case, the condition applied to the boundary Γ may be removed. The fact that the Timoshenko beam can be uniformly stabilized by boundary control can be found from the studies of Kim and Renardy [27].

Distributed control using distributed sensors and actuators can follow the same concept of using the energy functional to describe the amount of vibration energy and introducing feedback control to cause energy dissipation. As discussed above, an energy functional is an appropriate measure for the vibration energy.

2.3.2 Control Algorithm

We design the control algorithm by for stabilizing the composite Timoshenko beam described in the previous section subject to certain initial conditions. An energy functional is used to measure the amount of vibration of the system. We seek a control law in such a way that active damping is introduced into the system and the closed loop system is energy dissipative or at least energy non increasing. We shall use the energy functional used in [30] and [27].

The control problem becomes the following: given the system (2.9), (2.10) with boundary conditions (2.11) and an energy functional

$$E(t) = \frac{1}{2} \int_0^L \left\{ \rho A \left[\frac{\partial w}{\partial t} \right]^2 + \rho I \left[\frac{\partial \Phi}{\partial t} \right]^2 + K \left[\frac{\partial w}{\partial x} - \Phi \right]^2 + EI \left[\frac{\partial \Phi}{\partial x} \right]^2 \right\} dx \quad (2.22)$$

where

$$K = kAG,$$

we need to find a control $V(x, t)$ such that

$$\frac{dE(t)}{dt} < 0, \quad \text{for } t > 0. \quad (2.23)$$

The first two terms in the integral are the kinetic energy terms from vertical displacement and cross section rotation of the beam. The third term is the energy due to shear deformation while the last term represents the stored energy from bending.

Taking the derivative of $E(t)$ with respect to time, we have

$$\begin{aligned} \frac{dE(t)}{dt} &= \int_0^L \left\{ \rho A \frac{\partial w}{\partial t} \frac{\partial^2 w}{\partial t^2} + \rho I \frac{\partial \Phi}{\partial t} \frac{\partial^2 \Phi}{\partial t^2} \right. \\ &\quad \left. + K \left(\frac{\partial w}{\partial x} - \Phi \right) \left(\frac{\partial^2 w}{\partial x \partial t} - \frac{\partial \Phi}{\partial t} \right) + EI \frac{\partial \Phi}{\partial x} \frac{\partial^2 \Phi}{\partial x \partial t} \right\} dx. \end{aligned} \quad (2.24)$$

Integrating by parts in the spatial variable and incorporating the system equations (2.9) and (2.10) into it, we obtain the simplified form

$$\frac{dE(t)}{dt} = K \frac{\partial w}{\partial t} \left(\frac{\partial w}{\partial x} - \Phi \right) \Big|_0^L + \frac{\partial \Phi}{\partial t} EI \frac{\partial \Phi}{\partial x} \Big|_0^L + \int_0^L c \frac{\partial \Phi}{\partial t} \frac{\partial V}{\partial x} dx. \quad (2.25)$$

Further, using the boundary conditions (2.11), we arrive at

$$\frac{dE(t)}{dt} = \int_0^L c \frac{\partial \Phi}{\partial t} \frac{\partial V}{\partial x} dx. \quad (2.26)$$

The first two terms in (2.25) vanish.

Using the technique of separation of variables, we let control $V(x, t)$ be decomposed as the product of a function of the spatial variable x and a function of time t , such that

$$V(x, t) = v(x)q(t) \quad (2.27)$$

where $v(x)$ is an actuator shape function and $q(t)$ is the coordinate function of the actuator shape. Substituting $V(x, t)$ into (2.25), we obtain

$$\frac{dE(t)}{dt} = cq(t) \int_0^L \frac{\partial \Phi}{\partial t} \frac{dv(x)}{dx} dx. \quad (2.28)$$

Function $q(t)$ can be determined by using sensor output V_s from (2.18) with $F(x)$ compact on $[0, L]$,

$$\begin{aligned} q(t) &= -\frac{dV_s(t)}{dt} \\ &= -K_s \int_0^L \frac{\partial \Phi}{\partial t} \frac{dF(x)}{dx} dx. \end{aligned} \quad (2.29)$$

The corresponding control $V(x, t)$ is given by

$$V(x, t) = -K_s v(x) \int_0^L \frac{\partial \Phi}{\partial t} \frac{dF(x)}{dx} dx \quad (2.30)$$

Then the rate of energy change with respect to time (2.28) becomes

$$\begin{aligned} \frac{dE(t)}{dt} &= -c q(t) \int_0^L \frac{\partial \Phi}{\partial t} \frac{dv(x)}{dx} dx \\ &= -c K_s \int_0^L \frac{\partial \Phi}{\partial t} \frac{\partial F(x)}{\partial x} dx \int_0^L \frac{\partial \Phi}{\partial t} \frac{dv(x)}{dx} dx. \end{aligned} \quad (2.31)$$

Obviously, the rate of energy decay is a function of both the sensor shape function $F(x)$ and the actuator shape function $v(x)$. The selection of these two function shall have a direct impact of the effect of the control system on the closed-loop system. It is sufficient to select functions $F(x)$ and $v(x)$ to be proportional to each other, i.e.,

$$F(x) = k_{as} v(x) \quad (2.32)$$

where $k_{as} > 0$. In this case,

$$\frac{dE(t)}{dt} \leq 0, \quad (2.33)$$

and the closed loop system is energy dissipative. The feedback control (2.30) is velocity feedback and actually provides damping of Voigt type.

The feedback control (2.30) can be expressed in another form as

$$V(x, t) = -K_s v(x) \frac{\partial \Phi}{\partial t} F(x) \Big|_0^L + K_s v(x) \int_0^L \frac{\partial^2 \Phi}{\partial x \partial t} F(x) dx. \quad (2.34)$$

The second term in (2.34) is a weighted integral of the rate of change of the beam curvature with respect to time. Since the curvature of the beam is proportional to the strain of the beam in this formulation, the controlled beam has an altered constitutive equation. The stress is no longer just proportional to the strain, but the linear combination of both strain and the rate of change of the strain

with respect to time.

We may propose another control algorithm as

$$q(t) = C_1 V_s(t) + C_2 \frac{dV_s(t)}{dt}. \quad (2.35)$$

The corresponding control $V(x, t)$ appears in the form of

$$V(x, t) = -C_1 K_s v(x) \int_0^L \Phi \frac{dF(x)}{dx} dx - C_2 K_s v(x) \int_0^L \frac{\partial \Phi}{\partial t} \frac{dF(x)}{dx} dx. \quad (2.36)$$

This control algorithm would increase the natural frequency of the vibration of the beam in addition to active damping since the integration of the weighted curvature along the beam is introduced in the feedback. It is similar to introducing both position and velocity feedback in the traditional PID control for rigid bodies. The constant C_1 is to be determined for the required performance.

2.4 Modal analysis

We discussed sensor shape and actuator shape and their impact to the control of a cantilever beam in [9]. Different sensor outputs are available by choosing the appropriate sensor shape function $F(x)$. PVDF can be segmented along the beam to collect the signals from local regions. The sensor can provide deflection and strain information of the beam. A sensing network with simple computations will produce different signals in one sample period to meet the need of the control system.

2.4.1 Galerkin Procedure for Discretization

We use a Galerkin procedure to implement modal expansion and to analyze the impact of the induced active damping on the flexible beam. We write the beam

lateral displacement $w(x, t)$ and cross section rotation $\Phi(x, t)$ as

$$w(x, t) = \sum_{k=1}^n P_k(x) d_k(t) \quad (2.37)$$

$$\Phi(x, t) = \sum_{k=1}^n \Psi_k(x) q_k(t) \quad (2.38)$$

where $P_k(x)$ and $\Psi_k(x)$ are the admissible functions satisfying the boundary conditions of the system. These functions can be chosen such that $\{P_k\}$ form a normalized orthogonal basis and $\{\Psi_k\}$ form another normalized orthogonal basis. Functions $d_k(t)$ and $q_k(t)$ are their time coordinates respectively.

We rewrite here the control $V(x, t)$ from (2.27)

$$V(x, t) = v(x)q(t) \quad (2.39)$$

Substituting the modal forms (2.37) (2.38) and (2.27) into the system equations (2.9) and (2.10), multiplying P_l and Ψ_m to both sides of the first and second equation respectively, integrating both equation with respect to the spatial variable, we arrive at

$$\begin{aligned} \rho A \int P_k P_l dx \ddot{d}_k(t) &= K \sum \int P_k^{(2)} P_l dx d_k(t) \\ &\quad - K \sum \int \Psi_k^{(1)} P_l q_k(t) \end{aligned} \quad (2.40)$$

$$\begin{aligned} \rho I \int \Psi_k \Psi_m dx \ddot{q}_k(t) &= K \sum \int P_k^{(1)} \Psi_m dx d_k(t) + \sum (EI \int \Psi_k^{(2)} \Psi_m dx \\ &\quad - K \int \Psi_k \Psi_m dx) q_k(t) + c \int v^{(1)} \Psi_m dx q(t). \end{aligned} \quad (2.41)$$

where \dot{d} and \ddot{d} stands for the first and second time derivative of the function $d(t)$, $P^{(i)}(x)$ stands for the i^{th} spatial derivative of $P(x)$.

The orthogonal property of the base functions $P_k(x)$ and $\Psi_k(x)$ yields

$$\rho A \ddot{d}_l(t) - K \sum \int P_k^{(2)} P_l dx d_k(t) + K \sum \int \Psi_k^{(1)} P_l q_k(t) = 0 \quad (2.42)$$

$$\begin{aligned} \rho I \ddot{q}_m(t) - K \sum \int P_k^{(1)} \Psi_m dx d_k(t) - \sum [EI \int \Psi_k^{(2)} \Psi_m dx \\ - K \int \Psi_k \Psi_m dx] q_k(t) - c \int v^{(1)} \Psi_m dx q(t) = 0 \end{aligned} \quad (2.43)$$

The last term in the second equation is the contribution of the distributed control.

Equations (2.42) and (2.43) can be rearranged into a $2n \times 2n$ matrix form

$$M\ddot{u}(t) + C\dot{u}(t) + Ku = 0, \quad (2.44)$$

where $u(t) = [d_1(t) \cdots d_n(t) q_1(t) \cdots q_n(t)]^t$ is the vector of coordinate functions. M is the mass matrix. It is diagonal in this case with the first n elements being ρA with the rest being ρI . C is the damping matrix whose nonzero elements are given by

$$c_{(n+l)(n+k)} = cK_s \int v^{(1)}(x) \Psi_l(x) dx \int F^{(1)}(x) \Psi_k(x) dx, \quad 1 \leq l, k \leq n \quad (2.45)$$

which are introduced by the feedback control $V(x, t)$. It is known from (2.29) that

$$\begin{aligned} q(t) &= -K_s \int_0^L \frac{\partial \Phi}{\partial t} \frac{dF(x)}{dx} dx \\ &= -K_s \int_0^L \Psi_k F^{(1)} dx \dot{q}_k(t) \end{aligned} \quad (2.46)$$

K is the stiffness matrix with elements

$$\begin{aligned} k_{lk} &= -K \int P_k^{(2)} P_l dx, \\ k_{l(n+k)} &= K \int \Psi_k^{(1)} P_l dx, \end{aligned}$$

$$k_{(n+l)k} = -K \int P_k^{(1)} \Psi_l dx, \quad (2.47)$$

$$k_{(n+l)(n+k)} = K \int \Psi_k \Psi_l dx - EI \int \Psi_k^{(2)} \Psi_l dx$$

where $0 \leq l, k \leq n$.

The induced system damping can be observed from the introduction of damping matrix C whose elements are determined by the feedback control loop and by the sensor shape function $F(x)$ and actuator shape function $v(x)$.

Observing (2.45), we notice that the damping introduced to the system depends on the sensor and actuator shape functions $F(x)$ and $v(x)$. The amount of damping involved is limited by the sensor and actuator coefficients K , and c which are determined by the materials and their manufacturing. The coefficient c_{lk} describes the amount of damping introduced to the l th mode by the control system based on the measurement of the k th mode. The similarity holds for the actuator as well. Hence it is possible to enhance damping to some undesired elastic modes by modal analysis and by choosing the sensor and actuator shape functions $F(x)$ and $v(x)$ accordingly. An approach based on minimizing the average vibration energy over a certain time period can iteratively find the optimal shapes for both the sensors and the actuators [53]. The actual control systems can be implemented by segmented actuators and sensors with a switching network to provide different configurations in each sample period.

It is an interesting question to choose shape functions for the distributed sensors and actuators. The purpose is to introduce damping to different vibration modes efficiently rather than excite some undesired modes. The concentrated points or regions of control moment should be placed away from the nodal points of the vibration modes to assert the maximum control effect. Our case here can

be formulated as optimization of a certain performance measurement functional by choosing the candidate functions $F(x)$ and $v(x)$ subject to some constraints on the control $V(x, t)$. We anticipate that the optimal layout of sensors and actuators provide good performance.

2.5 Multiresolution System Identification and Monitoring

Distributed systems usually need complicated models to describe their dynamical behavior and characteristics as for the flexible structure modeling and control discussed above. The mathematical model governed by partial differential equations can be difficult to establish and to solve due to the variety of boundary conditions and the interfaces with different components and materials. This poses a question of how to systematically model complicated distributed systems in order to yield a workable and accurate model for system representation. One solution is to use system identification and parameter estimation and this method requires certain information about the plant and the key parameters which are not available at all time.

Another aspect is system monitoring and sensory data processing. A distributed system can involve hundreds of sensors and actuators of both distributed and discrete types for sensing, system control and monitoring. The sensory data processing and structural health monitoring can be inhibitive. As we mentioned earlier that the resulting sensor network can yield its own processing due to the application of distributed sensors and actuators. However, special treatments are needed in providing fault tolerance and a structured data processing scheme.

We introduce the concept of multiresolution system identification and monitoring here to facilitate our treatment of distributed systems. This concept is motivated by multiresolution representation in signal and image processing [35] [36] [18] and is useful in generating a framework of multiresolution system identification and monitoring schemes for distributed systems. The key motivation is using the available data from the working sensors to progressively approximate the real system or to yield the patterns for recognition and monitoring. Multiresolution in this setting reflects two meanings: the available system configuration at different levels of complexity and the different levels of approximation or monitoring while keeping a similar configuration.

2.6 Conclusions

We have embedded a static distributed PZT actuator model and a distributed sensor model into the Timoshenko beam model to form a composite beam model. We then designed a closed loop controller to introduce damping to the system. The closed loop system is proved to extract energy from the system. Since the Timoshenko beam model accounts for the rotational inertia and shear effects, it is suitable for situations when high frequency vibration occurs or when the shear effects of the material can not be omitted due to both material properties and the geometry. This composite model is more precise in describing the motion of the beam compared to the Euler-Bernoulli beam model in this sense. We then use modal analysis to further discuss the amount of active damping introduced by the controller and a method to choose the appropriate sensor and actuator shapes. It is possible to monitor and suppress the undesired modes by using suitably distributed sensors and actuators.

Further research is needed regarding the damping of beams with nonlinear deformation and comparing the results from the linear models with those from the nonlinear geometric exact model. An efficient numerical approach is also needed for the latter. The optimal distributed sensor and actuator layout is to be found through optimization with respect to the appropriate criteria. The real impacts of the distributed control system on the beam should be verified and further explored by experiments.

Chapter 3

Optimal Design of Sensors and Actuators

We describe a design technique for optimal control in active structural vibration damping using smart materials. We model the beam by the Timoshenko beam model together with the distributed sensors and actuators. A control law using the weighted integration of vibration velocity is incorporated in the closed loop system. We propose a method to find the optimal layout design of the smart material so as to maximize the damping effect. An objective functional is defined based on the vibration energy of the system. The optimal shapes of the sensor and actuator are determined through minimizing the energy functional of the beam over the admissible shape function space subject to certain geometric constraints. An algorithm has been developed to determine the optimal sensor and actuator layout. This method can be generalized to the plate damping problem and more complicated structures as well.

3.1 Introduction

An important issue in the control system design for flexible systems is the determination of the optimal number and location of the control system components:

sensors and actuators [33] [11]. In general there is a larger number of candidate locations than available sensors and actuators. Based on experience and knowledge on structure dynamics and control objectives, a priori selection is usually available. However, this may not give the optimal effect on the closed loop system. Extensive experimental work is expected to justify the design. For discrete optimal sensor and actuator locations, a method based on the orthogonal projection of structural modes onto the intersection of the controllable and observable subspaces is introduced Lim [33]. The controllability and observability Gramians are used in this research to reflect the degrees of controllability and observability of an actuator/sensor pair. However, this method is based on a second order linear model. In the studies by Choe and Baruh [11] regarding the placement of force and torque actuators for structural control, an objective function is defined based on the elements of the actuator influence matrix, and an optimization study is performed to compare the system performance. This work suggests that a relative even distribution of the actuators can lead to satisfactory results. Again, pointwise sensors and actuators are analyzed here. It is also pointed out [11] that piece-wise continuous distributed actuators have the advantage of reducing the stress level and exciting the higher modes less.

The use of smart materials as sensors and actuators allows the adjustment of geometry and dynamical behavior of flexible space structures. It also provides means for sensory data processing by the sensor geometry. Modal sensors and modal actuators are proposed and developed in [32] to provide sensing signals related to certain elastic modes and to suppress the selected elastic modes. It has been pointed out [15] that PZT actuators should be placed in regions with high average strain and away from the nodes of the specified elastic modes to

be controlled to achieve maximum effect. We discussed the damping effect in Chapter 2 with respect to the shape functions of both distributed sensors and distributed actuators on the system. The choice of the sensor and actuator shapes is also an important factor in the performance of systems. A question arises naturally: what are the optimal shapes of the distributed sensors and actuators made from smart materials? We need a systematic approach to answer this question.

We know that the flexible beam is an infinite dimensional system. In order to faithfully measure and control the system without using a truncated model, there is a need to design control algorithms directly from partial differential equation models to avoid the so called spillover [8]. We can develop certain performance measures to carry out the optimal design.

We consider the design issue associated with the vibration damping control of a cantilever beam. The beam is modeled as the Timoshenko beam. Both sides of the beam are covered by PVDF and PZT materials for sensing and actuation as illustrated in Figure 2.1. Using the control algorithm developed in the previous chapter and in [51], the closed loop system can be stabilized. Based on this result, we want to further determine the optimal layout of the continuous distributed sensors and actuators for the system based on minimizing the vibration energy of the system. It is likely that this approach can lead to a general design methodology or at least provide a design guideline.

3.2 Problem formulation

We model the cantilever beam with the Timoshenko beam model which accounts for shear effects and rotational inertia. The Timoshenko model describes the

physical behavior better than the Euler-Bernoulli model does especially for the high frequency vibration components. The actuator and sensor are the layers made of piezoelectric ceramic (PZT) and piezoelectric polymer polyvinylidene fluoride (PVDF) materials attached to both sides of the beam as illustrated in Figure 2.1.

The equations of motion are given by (2.9) and (2.10) of the previous chapter. For the convenience in reference, we rewrite them as

$$\rho A \frac{\partial^2 w}{\partial t^2} = kAG \left(\frac{\partial^2 w}{\partial x^2} - \frac{\partial \Phi}{\partial x} \right), \quad (3.1)$$

$$\rho I \frac{\partial^2 \Phi}{\partial t^2} = EI \frac{\partial^2 \Phi}{\partial x^2} + kAG \left(\frac{\partial w}{\partial x} - \Phi \right) + c \frac{\partial V}{\partial x} \quad (3.2)$$

and boundary conditions (2.11) as

$$\begin{aligned} w(0, t) &= 0, & \Phi(0, t) &= 0, \\ \frac{\partial w(L, t)}{\partial x} - \Phi(L, t) &= 0, & EI \frac{\partial \Phi(L, t)}{\partial x} &= 0. \end{aligned} \quad (3.3)$$

$$EI = E_a I_a + E_b I_b + E_s I_s, \quad (3.4)$$

and

$$c = \frac{d_{31}}{h_a} K_a.$$

The distributed control $V(x, t)$ appears in (3.2) as the distributed bending moment. Suppose that the beam is initially at rest, we thus have the following initial conditions

$$w(x, 0) = u(x), \quad \Phi(x, 0) = r(x) \quad (3.5)$$

and

$$\frac{\partial w(x, 0)}{\partial t} = 0, \quad \frac{\partial \Phi(x, 0)}{\partial t} = 0. \quad (3.6)$$

We shall refer boundary conditions (3.5) and (3.6) as IC in our discussing for simplicity.

The sensor output is given by (2.17) as

$$V_s(t) = -K_s \int_0^L F(x) \frac{\partial^2 \Phi}{\partial t \partial x} dx \quad (3.7)$$

which is a weighted integral of beam curvature along the longitudinal direction of the beam. K_s is a constant determined by the sensor material. We rewrite the feedback control (2.30) as

$$V(x, t) = K_s v(x) \int_0^L F(x) \frac{\partial^2 \Phi}{\partial t \partial x} dx,$$

where $v(x)$ is the actuator shape function. Similarly to the sensor shape function, $v(x)$ is the width of the electrodes covering the surfaces of the PZT actuator.

The energy functional (2.22) is rewritten as

$$E(t) = \frac{1}{2} \int_0^L \left\{ \rho A \left[\frac{\partial w}{\partial t} \right]^2 + \rho I \left[\frac{\partial \Phi}{\partial t} \right]^2 + K \left[\frac{\partial w}{\partial x} - \Phi \right]^2 + EI \left[\frac{\partial \Phi}{\partial x} \right]^2 \right\} dx. \quad (3.8)$$

The vibration energy defined above is a function of time t and the initial condition IC specified above. Our goal is to maximize system damping through selection of sensor and actuator shape functions with respect to this particular case.

3.3 Optimal Control and a Numerical Algorithm

Unlike point sensors and actuators, the geometry of the spatially distributed sensor and actuator has a function of preprocessing the sensor output signals

and control weight. A judicious choice of the shapes can extract the desired signals and implement the control algorithm. We discussed in [51] a method of choosing the appropriate sensor and actuator shapes for active damping control by means of modal analysis. We would like to develop a systematic approach to deal with this problem here.

3.3.1 Optimal Control

The control (3.2) is a function of sensor and actuator shape functions and the weighted integral of the beam curvature. We have proved that the control (3.2) stabilizes the system [51] through introducing damping into it.

We seek the optimal control in the sense that the energy functional is minimized over all the possible sensor and actuator shapes for a specific vibration state. Our task here is to find the optimal sensor and actuator shape functions $v(x)$ and $F(x)$ so as to minimize the energy functional (2.22) at a certain time. This optimization scheme depends on the initial condition of the beam.

When the beam is initially at rest, the problem is to find functions $v_0(x)$ and F_0 such that

$$J[T, v_0, F_0, IC] = \min_{v \in \mathcal{V}, F \in \mathcal{F}} \frac{1}{2} \int_0^L \left\{ \rho A \left[\frac{\partial w}{\partial t} \right]^2 + \rho I \left[\frac{\partial \Phi}{\partial t} \right]^2 + K \left[\frac{\partial w}{\partial x} - \Phi \right]^2 + EI \left[\frac{\partial \Phi}{\partial x} \right]^2 \right\} dx, \quad (3.9)$$

where \mathcal{V} and \mathcal{F} are the sets of all the admissible actuator and sensor shape functions. The admissible functions here depend on geometry of the structure. For beam and plate like structures, the geometry is usually simple. Since the region of the beam which could be covered with smart materials is assumed to have length L and width b , the sets \mathcal{V} and \mathcal{F} contain the collection of all the piecewise continuous curves within this region. The optimization hence has

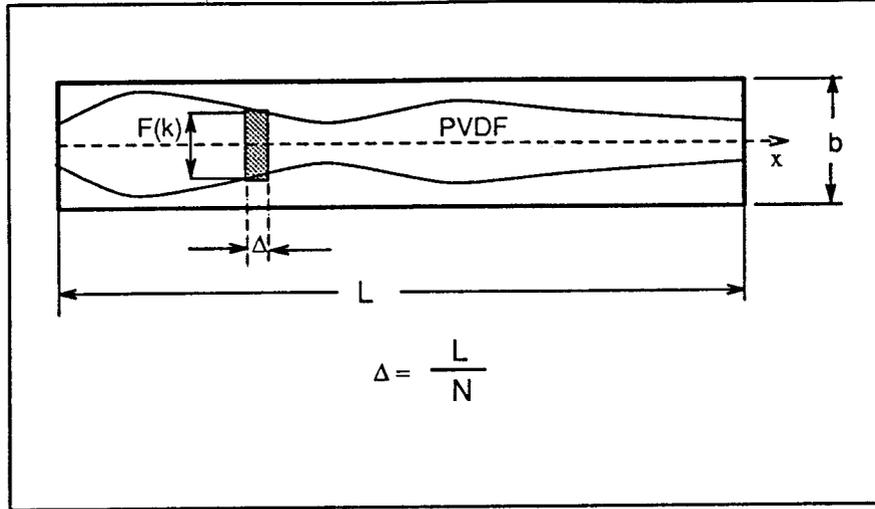


Figure 3.1: Discretizing the sensor and actuator layout

a geometric constraint. The functions $v(x)$ and $F(x)$ denote the width of the electrodes covering the smart materials; we have $0 \leq v(x) \leq b$ and $0 \leq F(x) \leq b$. We are interested in minimizing the amount of vibration energy at time T .

3.3.2 A Numerical Algorithm

For numerical solution of a shape optimization problem, one typically starts by guessing an initial design. One then discretizes the elastic problem using finite elements or using difference method or a Galerkin procedure. After discretization, the optimal design problem becomes a large nonlinear programming problem. Different routines are available for working on the latter.

We start our numerical scheme by discretizing the region along the longitudinal spatial axis as in Figure 3.1. Denote distance along this axis by x . Let N be the total number of segments with equal size. Then the width of each segment is L/N . The discretized shape functions $v(x)$ and $F(x)$ assume constant values $v(k)$ and $F(k)$ inside the k^{th} element. We thus have piecewise constant functions $v(k)$ and $F(k)$ with $k = 1, 2, \dots, N$. The $2k$ members of $v(x)$ and $F(k)$

become the optimization parameters. We then compute the distributed control $V(k, t)$, $k = 1, 2, \dots, N$, based on the initial conditions of the system and the discretized shape functions. The time response to the input can be computed through solving the equations of motion numerically. This procedure yields the value of the cost functional J at time T . An optimal routine shall be followed to search and adjust the piecewise constant sensor and actuator shape functions toward reducing the value of the cost functional (3.9). The new shape functions are then used to generate the system input $V(k, t)$ again. This procedure is repeated until the difference of the energy functional values between two iterations is less than a predetermined threshold. The resulting $v(k)$ and $F(k)$ can be smoothed to give the final optimal shape functions $v_0(x)$ and $F_0(x)$ for the sensors and actuators. The algorithm is given in Figure 3.2.

3.3.3 Numerical Examples

In this section, we provide a numerical example regarding the optimal shape design. We consider an aluminum beam substructure with a PZT actuator layer and a PVDF sensor layer on both side of the beam and investigate the impact of the layout of the distributed sensor and distributed actuator on system energy. Our goal is to show the feasibility of conducting designs based on the energy measure and the method described in this chapter.

We first select the basis functions as discussed in the previous chapter for using a Galerkin procedure to discretize the equations of motions (3.1) and (3.2) as the following

$$P_k(x) = \sin\left(\left(k\pi + \frac{\pi}{2}\right)\frac{x}{L}\right) \quad (3.10)$$

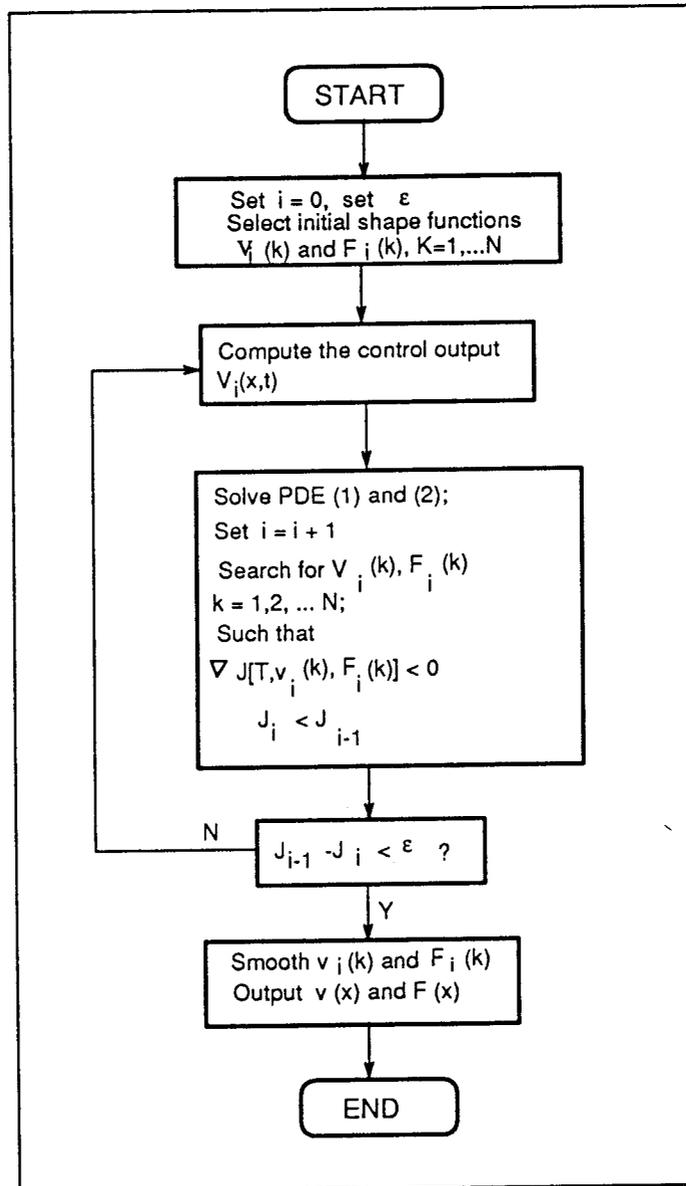


Figure 3.2: Optimization algorithm

for displacement and

$$\Psi_k(x) = \sin\left(\left(k\pi + \frac{\pi}{2}\right)\frac{x}{L}\right). \quad (3.11)$$

It is easy to check that the boundary conditions associated with equations of motions (3.1) and (3.2) are satisfied by the basis functions above. These basis functions span the space of admissible solutions of these two partial differential equations.

A composite aluminum beam model is employed in the simulation. The aluminum beam model used in simulation is $2m$ long, $0.15m$ wide and $7mm$ thick. The Young's modulus of aluminum is $70 GPa$ and the density of aluminum is $2710 kg/m^3$. We assume that the Young's modulus for the materials used in the sensing and actuation layers is $2.0 GPa$ and both the sensor layer and the actuator layer is $3mm$ thick. These parameters are used in the simulation.

The displacement and rotation of the beam are approximated by their first four modes respectively with zero initial velocity and zero shear. The initial conditions in consideration are

$$w(x, 0) = 0.015L \sin\left(\frac{3\pi}{2L}x\right), \quad \Phi(x, 0) = 0 \quad (3.12)$$

and

$$\frac{\partial w(x, 0)}{\partial t} = 0, \quad \frac{\partial \Phi(x, 0)}{\partial t} = 0. \quad (3.13)$$

The nonzero initial displacement is shown in Figure 3.3.

The response of the undamped beam model subject the initial condition above is simulated to analyze its nature behavior. The responses of time coordinates $d_k(t)$ and $q_k(t)$ of the modal equation (2.44) are illustrated in Figure 3.4. It can be seen that the rotation of the cross section of the beam contains high

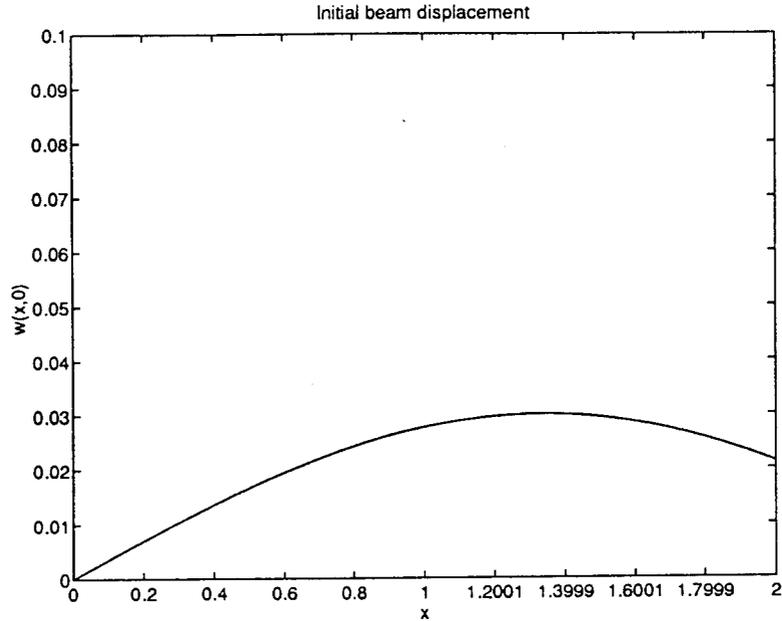


Figure 3.3: The initial displacement of the beam

frequency components compared with the motion of the transversal displacement. The waveforms of the rotational are the modulated wave forms of those of the displacements. The Timoshenko beam model describes the high frequency components of the vibration of the system in this regard. A mesh plot of the time-space response without damping is shown in Figure 3.5 in which the shear of the beam appears to concentrate on the half closest to the clamped end ($x = 0$). The system energy does not decay with respect to time as shown in Figure 3.6.

To concentrate on the concept and methodology of this chapter, we partition the beam along its longitudinal axis x into two segments of equal length $L/2$ where L is the length of the beam. We also let the sensor layer and the actuator layer have equal width at both sides of the same beam location, i.e.,

$$F(x) = v(x).$$

Furthermore, we denote the width of these layers at the first and the second beam segment from the clamped end by a and b respectively. We may then vary a and b , the optimization parameters, and observe the impacts of different configurations on the change of system energy subject to initial conditions (3.12) and (3.13).

Simulation results show that for the given initial conditions the system is mostly damped when the layout of the sensor layer and the actuator layer is chosen to be $a = 0.15 \text{ m}$ and $b = 0$. This matches the phenomenon of the zero damping response shown in Figure 3.4 and Figure 3.5 in which shear deformation concentrates near the clamped end ($x = 0$) of the beam. We can observe the time responses of the first four coordinates $d_k(t)$ of the displacement and the first four coordinates $q_k(t)$ of the rotation from Figure 3.7. The high frequency components in rotations are damped out in less than 10 seconds. The corresponding mesh plots of the time-space responses in Figure 3.8 demonstrate that the vibration of the system dies out in about 10 second. Figure 3.9 provides the control voltage which is used to close the system loop. The vibration energy of the system decays towards zero with an envelop of an exponential type as shown in Figure 3.10.

Different sensor and actuator layouts provide different damping effects. One case in the simulation shows that when the layout is placed near the free end of the beam, i.e., $a = 0$ and $b = 0.15 \text{ m}$, with respect to the same initial conditions, the damping induced is less efficient than that in the previous case. The time history of coordinates of different displacement modes and rotation modes are illustrated in Figure 3.11. Some of these modes display similar patterns to those in the previous case except the fourth rotation coordinate contains high

frequency components. The effects of this layout can be seen from the time-space response in Figure 3.12. These responses are less favorable when compared with those in the previous case shown in Figure 3.8. The control voltage $V(t)$ and the system energy $E(t)$ are shown in Figure 3.13 and Figure 3.12 respectively. The rate of system energy decay is slower than that of the previous case due to the changed layout of the sensor and actuator layers.

System energy is a useful measurement of the vibration. Comparing the cases above, we know that is not sufficient to observe the time history of different coordinates of different vibration modes to design the layout of distributed sensors and actuators. As a consequence, it may not be accurate to judge the design and performance of systems based on modal analysis alone. However, from the system energy profiles of Figure 3.10 and Figure 3.14, we can see that the former one has a faster energy decay rate than the later one does. This phenomenon explains in part the differences in system performances as illustrated by the time-space responses in Figure 3.8 and Figure 3.12.

3.4 Other Considerations

The optimization scheme can be used to deal with more complicated structures. Sensors and actuators may be piecewise continuously distributed on the structures. When structures contain both distributed and pointwise sensors and actuators, our formulation still holds. The corresponding shape functions $v(x)$ and $F(x)$ become both piecewise continuous and pointwise in the relevant regions.

In terms of computation, the bottleneck is the simulation of the systems governed by partial differential equations. Different methods can be implemented

to solve the partial differential equations (2.9) and (2.10).

Modal sensors and actuators can be designed through optimization as well. This may reduce the influence of leak-through, i.e., the crossover effect among different modes [32], to improve the overall performance. Different performance measures and cost functions are required to formulate these optimization problems.

The design algorithm is useful in dealing with complicated systems. The design from the methodology above depends on initial conditions and specific boundary conditions of systems. One may use a segmented sensor layer and a segmented actuator layer or multi-layer structures, and incorporates sensing switch networks to reconfigure the layout dynamically so as to meet the needs in sensing and control of the various initial conditions and boundary conditions. The layout of these segments and the switching rules of the networks can be designed through the optimization scheme above.

3.5 Conclusions

We have developed a method to facilitate optimal designs of active vibration damping using smart materials. Simulation results indicate that the energy based design methodology is feasible. The optimal algorithm described can be expected to yield reasonably good designs for the layout of distributed sensors and actuators. Although the algorithm is developed based on the beam model used, the method can be extended to the plane case. This procedure is expected to work for cases with irregular geometry or nonuniform structural material as well.

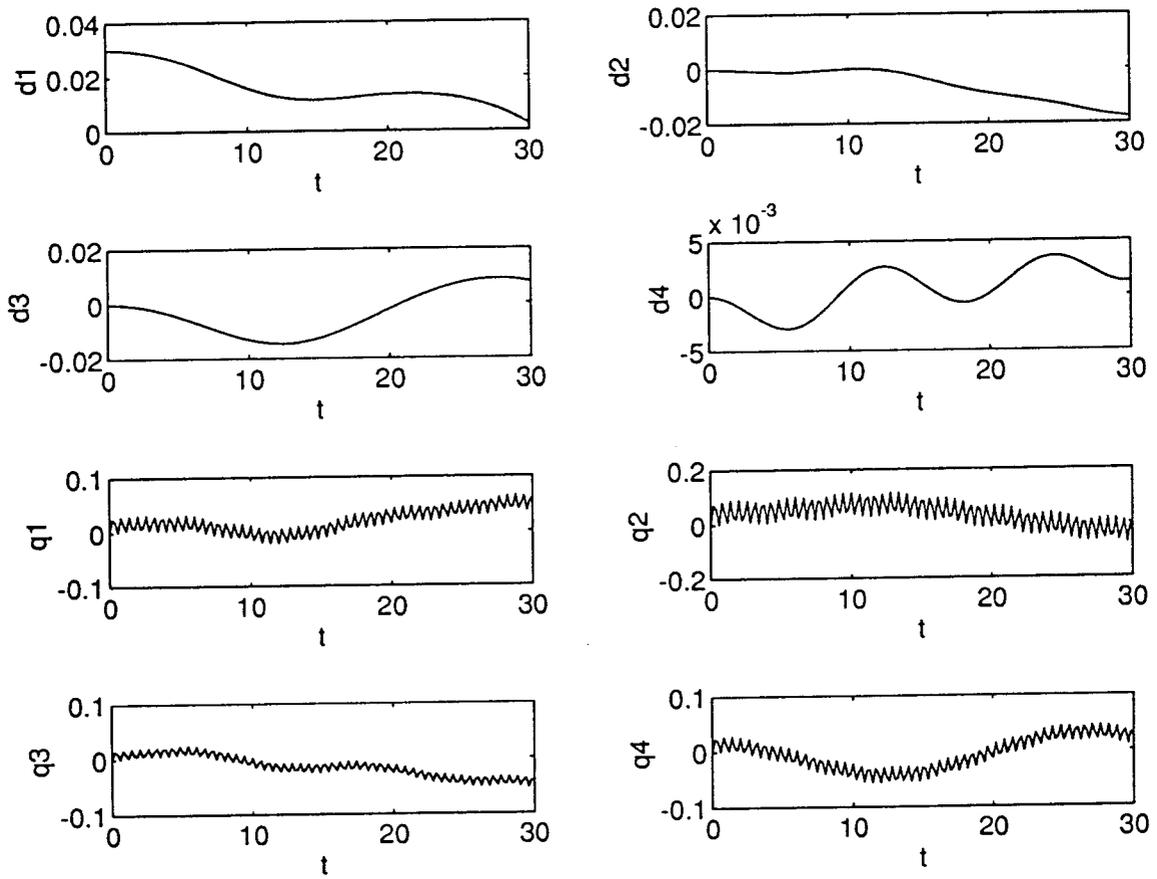


Figure 3.4: Time responses of coordinates $d_k(t)$ and $q_k(t)$ without damping

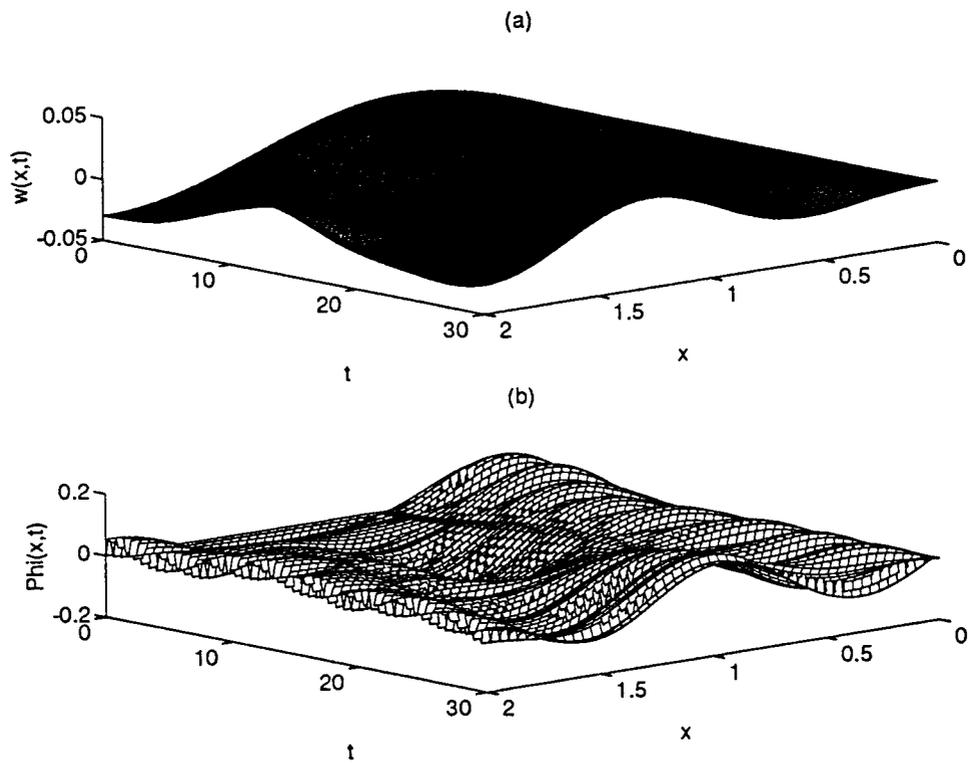


Figure 3.5: Time-space responses of displacement $w(x,t)$ (m) and $\Phi(x,t)$ (rad) versus x (m) and t (sec) without damping

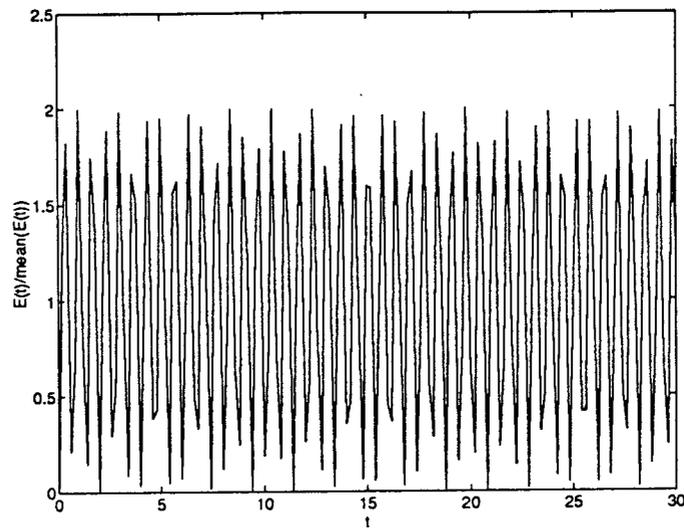


Figure 3.6: System energy $E(t)/\text{mean}(E(t))$ versus t (sec) without damping

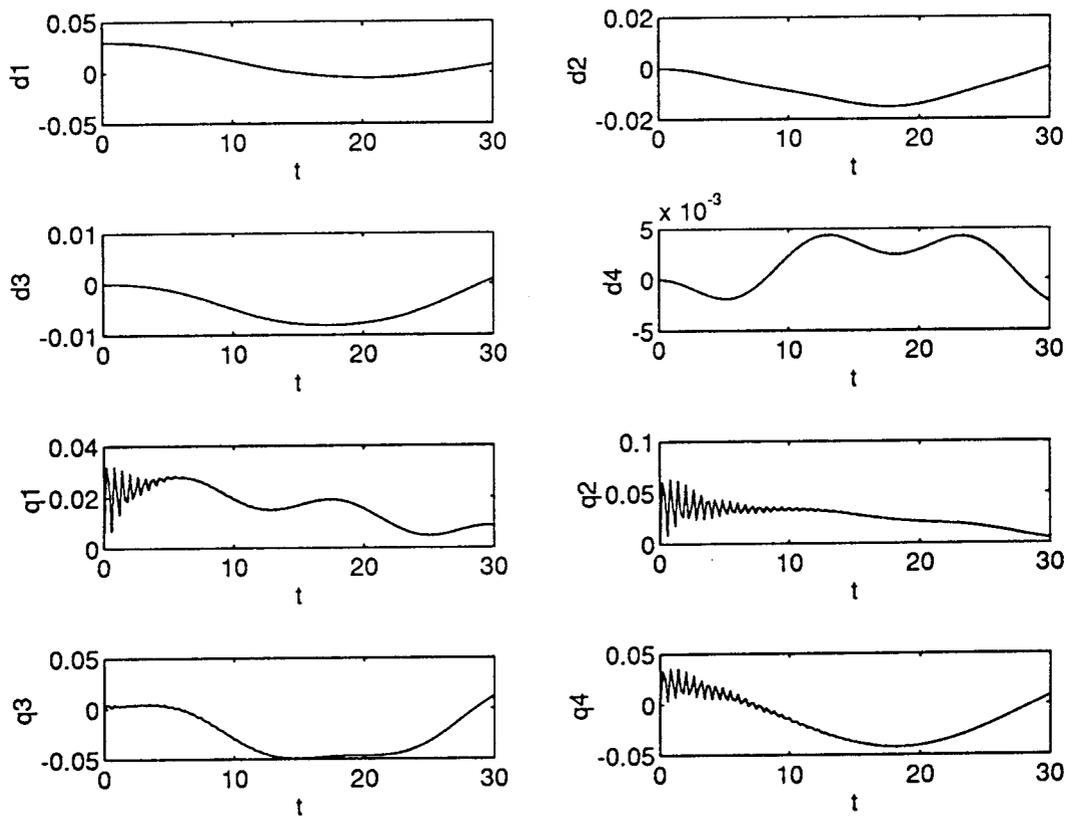


Figure 3.7: Time responses of coordinates $d_k(t)$ and $q_k(t)$ with active damping; $a=0.15$ m, $b=0.0$

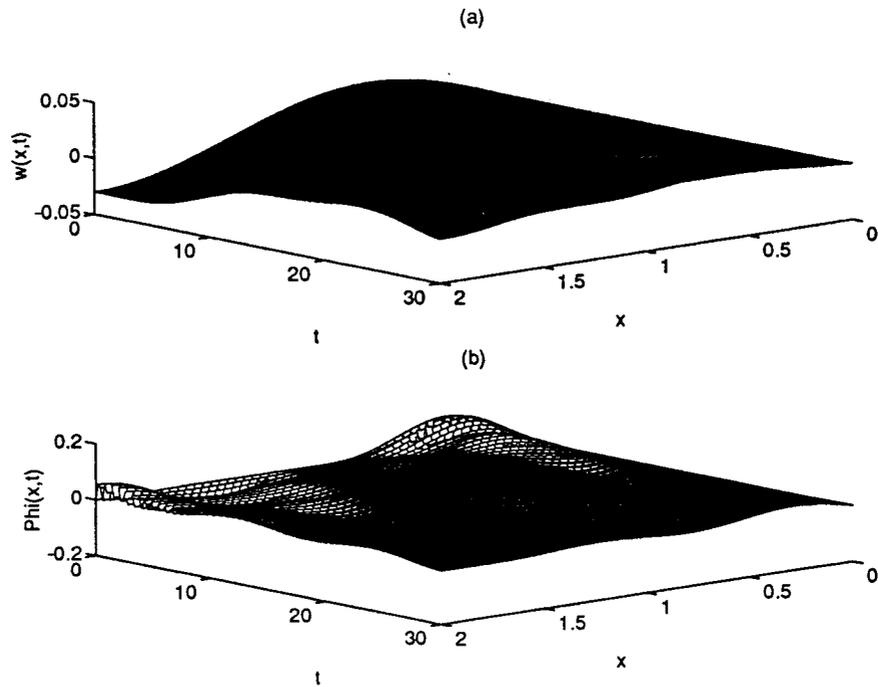


Figure 3.8: Time-space responses of displacement $w(x,t)$ (m) and $\Phi(x,t)$ (rad) versus x (m) and t (sec) with active damping; $a = 0.15$ m, $b = 0$

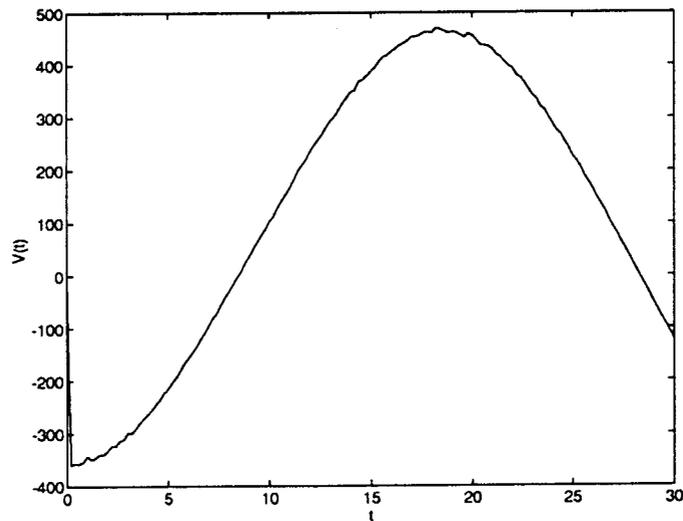


Figure 3.9: Control $V(t)$ (V) with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$

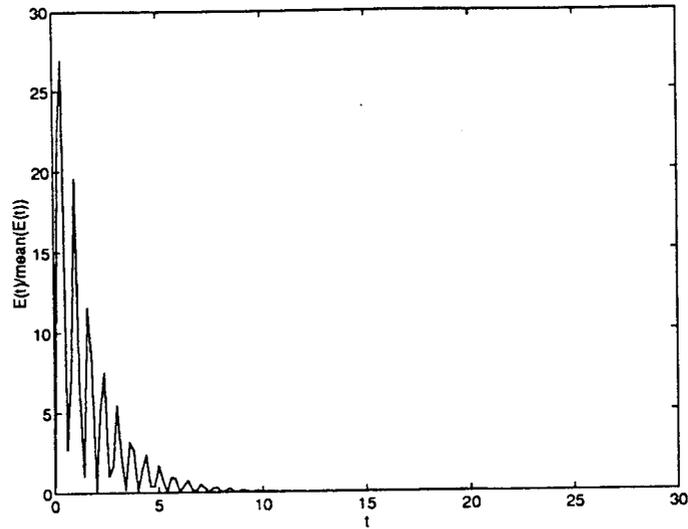


Figure 3.10: System Energy $E(t)/\text{mean}(E(t))$ with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$

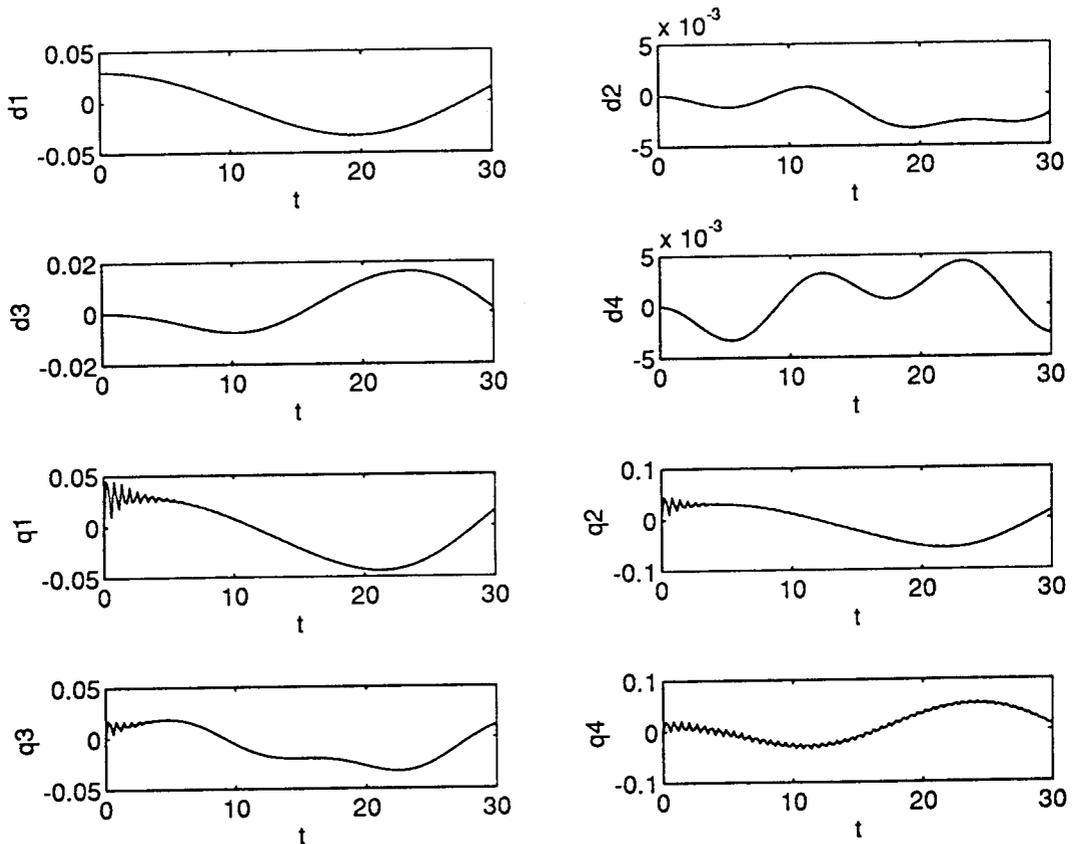


Figure 3.11: Time responses of coordinates $d_k(t)$ and $q_k(t)$ with active damping; $a = 0$, $b = 0.15$ m

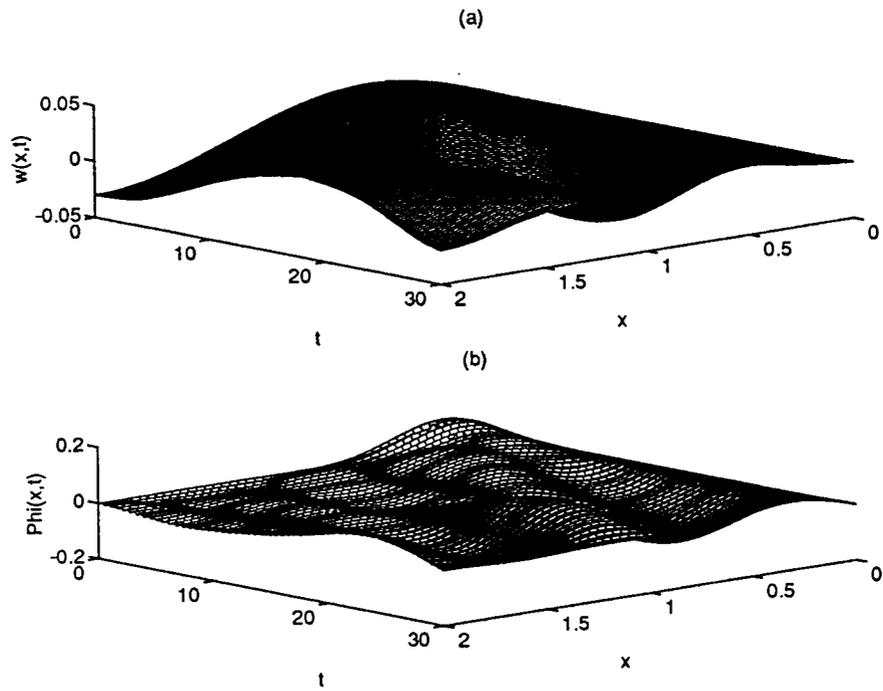


Figure 3.12: Time-space responses of displacement $w(x,t)$ (m) and $\Phi(x,t)$ (rad) versus x (m) and t (sec) with active damping; $a = 0$, $b = 0.15$ m

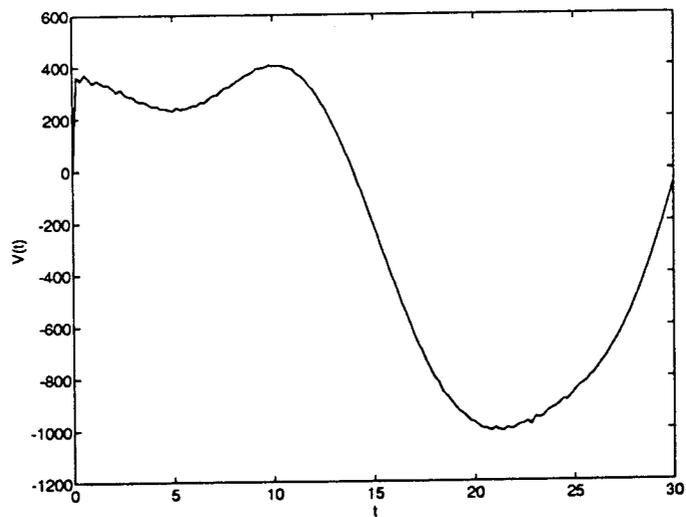


Figure 3.13: Control $V(t)$ (V) with respect to time t (sec) with active damping; $a = 0.15$ m, $b = 0$

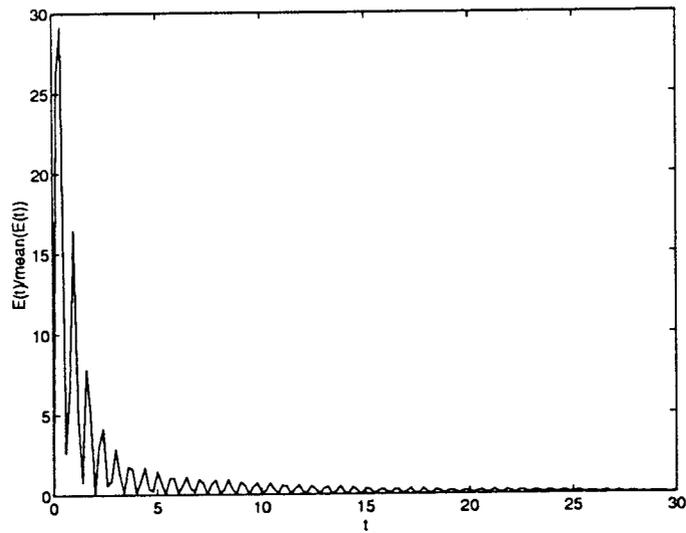


Figure 3.14: System Energy $E(t)/\text{mean}(E(t))$ with respect to time t (sec) with active damping; $a = 0.15$, $m b = 0$

Chapter 4

Compactly Supported Wavelets and QMF banks

4.1 Introduction

We have found recent advancement in wavelet theory encouraging in generating progressive modeling structure for system identification and signal approximation in $L^2(R)$. There have been extensive research interest and activities in wavelet theory and its applications in recent years [12] [19]. The most attractive features of wavelet theory are the multiresolution property and time and frequency localization ability. The wavelet transform decomposes a signal into its components at different resolutions. Its application actually simplifies the description of signals and provides analysis at different levels of detail. There have been successful applications of these properties in the fields of signal processing, speech processing and especially in image processing [43] [36] [34]. The wavelet transform differs from short-time Fourier transforms (STFT) in the sense of producing a varying time-frequency window for signal representation. It admits nonuniform bandwidths, so that the bandwidth may be higher at higher frequencies, allowing implementation of the transform through different levels of decimation in filter banks.

In this chapter, we introduce the background on choosing the optimal wavelet basis with compact support of an appropriate size. We first briefly review the multiresolution property of wavelet functions and the structure of quadrature mirror filter (QMF) banks. We then introduce the conditions for generating a compactly supported discrete wavelet basis in terms of properties of quadrature mirror filter (QMF) banks [48].

This chapter makes use of the multiresolution property of wavelets as studied by Mallat [35], the link between wavelets and filter banks as studied by Daubechies [18], and the structures of multirate systems and quadrature mirror filters (QMF) of Vaidyanathan [48], Akansu [1] and Herley [22]. We follow Vaidyanathan [48] closely in this chapter.

4.2 Background Review

This section reviews the background on the wavelet transform, multiresolution property of wavelets and the components in multirate systems to be used in generating the wavelet transform.

4.2.1 Wavelet Transform

All the basis functions are dilations and shifts of a single function called the base wavelet. A general form is

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), \quad (4.1)$$

where $a \in R^+$, $b \in R$. The parameters a and b provide scaling and shift, respectively, of the original function $\psi(t)$. The wavelet transform is defined as

$$X_w(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi^*\left(\frac{t-b}{a}\right) x(t) dt. \quad (4.2)$$

The discretized version of the wavelet basis functions is

$$\psi_{m,n}(t) = a_0^{-m/2} \psi(a_0^{-m}t - nb_0), \quad m, n \in Z, \quad a_0 > 0, b_0 \neq 0, \quad (4.3)$$

which corresponds to $a = a_0^m$ and $b = na_0^m b_0$, where the size of the shift depends on the scaling factor. We are interested in the dyadic case, that is, the case $a_0 = 2$ and $b_0 = 1$. It was shown [35] that it is possible to derive a base wavelet function $\psi(x) \in L^2(R)$ such that for $j, l \in Z$, $\{\psi_{j,l}(x)\}_{j,l \in Z}$ with

$$\psi_{j,l}(x) = 2^{-j/2} \psi(2^{-j}x - l) \quad (4.4)$$

is an orthonormal basis of $L^2(R)$. Any signal in $L^2(R)$ can be decomposed into its components in different scales in subspaces of $L^2(R)$ of corresponding resolutions and the reverse is true when the regularity condition for the base wavelet $\psi(x)$ is introduced [19] [35]. The base wavelet function $\psi(x)$ plays a central role in this formulation.

4.2.2 Multiresolution Approximation

A multiresolution approximation of $L^2(R)$ due to Mallat [35] is a sequence $\{V_j\}_{j \in Z}$ of closed subspaces of $L^2(R)$ such that the following hold, with Z denoting the set of all integers:

(I)

$$V_j \subset V_{j-1}, \quad \forall j \in Z \quad (4.5)$$

$$\bigcup_{j=-\infty}^{+\infty} V_j \text{ is dense in } L^2(R) \text{ and } \bigcap_{j=-\infty}^{+\infty} V_j = \{0\} \quad (4.6)$$

(II).

$$f(x) \in V_j \iff f(2x) \in V_{j-1}, \quad \forall j \in Z \quad (4.7)$$

(III).

$$f(x) \in V_j \implies f(x - 2^j k) \in V_j, \quad k \in Z \quad (4.8)$$

and there is a scaling function $\phi(x) \in L^2(R)$, such that, for all $j \in Z$,

$$\phi_{j,l} = 2^{-j/2} \phi(2^j x - l)_{l \in Z} \quad (4.9)$$

is an orthonormal basis of V_j with $V_j \subset V_{j-1}$.

With this setting, W_j , the complement of $V_j \subset V_{j-1}$, can be expressed as

$$V_j \oplus W_j = V_{j-1}, \quad (4.10)$$

with

$$V_J = \bigoplus_{j=J+1}^{\infty} W_j. \quad (4.11)$$

For all j , there is a wavelet function $\psi(x)$, such that,

$$\psi_{j,l}(x) = 2^{-j/2} \psi(2^{-j} x - l)_{l \in Z} \quad (4.12)$$

is an orthonormal basis of W_j . The additional information in an approximation at resolution 2^{-j} compared with the resolution 2^{-j+1} is contained in the subspace W_j , the orthogonal complement of $V_j \subset V_{j-1}$. If we define P_{V_j} to be a projection operator in $L^2(R)$ and I to be the identity operator, then

$$P_{V_j} \rightarrow I, \quad \text{as } j \rightarrow -\infty. \quad (4.13)$$

Any square integrable function $f(x) \in L(R^2)$ can be represented as

$$f(x) = \sum_{j,l} w_{j,l} \psi_{j,l}(x), \quad (4.14)$$

the coefficients $w_{j,l}$ carry the information of $f(x)$ near frequency 2^{-j} and near $x = 2^j l$.

When a function in $L^2(R)$ or a transfer function in $H^2(R)$ is unknown, the

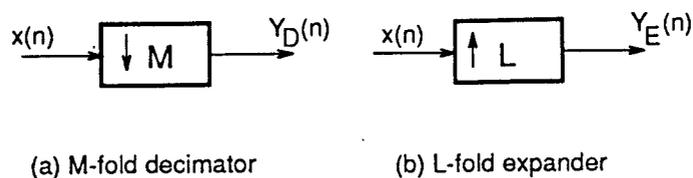


Figure 4.1: The decimator and expander

wavelet system is feasible for its approximation. Affine wavelet decompositions are developed as a means of construction rational approximations to nonrational transfer functions in the studies by Pati [40]. Since a closed expression is usually not available for practical purposes, it is necessary to use a sum of finitely many functions, typically of lower order or of less complexity, to approximate the original transfer function. A wavelet system can be implemented to approximate the unknown system. This process is completed by adjusting the coefficients with respect to the wavelet basis.

4.2.3 Decimators and Expanders in Multirate Systems

Figure 4.1(a) shows the M -fold decimator and L -fold expander. The decimator is also called a downsampler. It takes the input sequence $x(n)$ and produces the output sequence

$$y_D(n) = x(Mn), \quad (4.15)$$

where M is an integer. Only those samples of $x(n)$ which occur at times equal to a multiple of M are retained to form the output sequence. When $M = 2$, the decimation process is shown in Figure 4.2.

Figure 4.1(b) shows the L -fold expander. It takes the input sequence $x(n)$ and produces the output sequence

$$y_E(n) = \begin{cases} x(n/L), & \text{if } n \text{ is integer-multiple of } L \\ 0, & \text{otherwise} \end{cases} \quad (4.16)$$

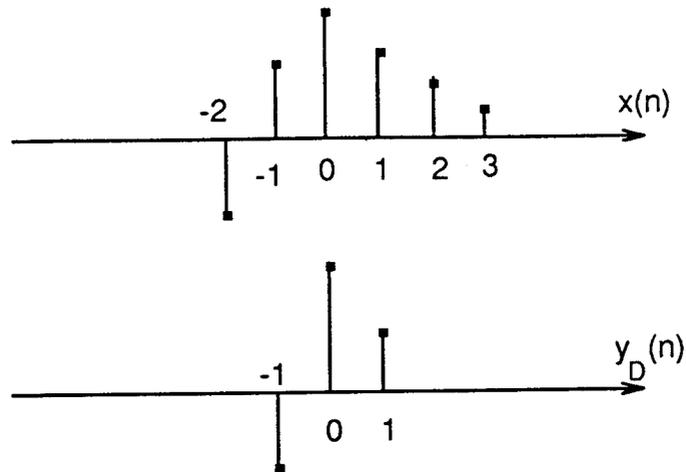


Figure 4.2: Decimation for $M = 2$

where L is an integer. Other names for the expander are sampler rate expander, interpolator and upsampler. Figure 4.3 demonstrates the process of upsampling, in which zeros are inserted between the two adjacent input pulses. The z-domain expression for the decimator is given by

$$y_D(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(z^{1/M} W^k), \quad (4.17)$$

where $X(z)$ is the z transform of the input sequence $x(n)$ and W is defined as

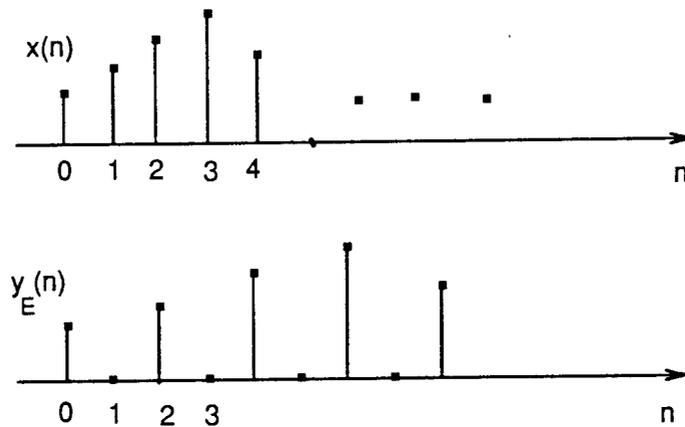


Figure 4.3: Demonstration of the expander for $L = 2$

the M th root of unity

$$W = e^{-j2\pi/M}. \quad (4.18)$$

Consequently, the transform on the unit circle yields

$$y_D(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j(\omega-2\pi k)/M}). \quad (4.19)$$

The transform $y_D(e^{j\omega})$ is obtained by first stretching $X(e^{j\omega})$ by a factor of M and then shifting the stretched version uniformly by the amount $2k\pi$ and taking the summation of all the shifts.

The expander has an easier form of

$$\begin{aligned} y_E(z) &= \sum_{n=-\infty}^{\infty} y_E(n)z^{-n} \\ &= \sum_{k=-\infty}^{\infty} y_E(kL)z^{-kL} = \sum_{k=-\infty}^{\infty} y_E(k)z^{-kL} \\ &= X(z^L). \end{aligned} \quad (4.20)$$

Similar to the decimator, the expander has the expression

$$y_E(z) = X(e^{j\omega L}) \quad (4.21)$$

on the unit circle.

The operation of the expander produces multiple copies of the compressed spectrum in the frequency domain.

4.2.4 Decimation Filters and Interpolation Filters

The process of decimation is similar to the sampling process of a continuous signal regarding to causing possible aliasing. The stretched version of $X(e^{j\omega/M})$ can in general overlap with its shifted copies and prevent recovering the original signal $x(n)$ from the decimated signal $y_D(n)$. A low pass signal with its

bandwidth limited to π/M or a signal passing through a low pass filter with this bandwidth shall avoid signal aliasing, when passing through a M -fold decimator. Figure 4.4 shows the block structure of a decimation and its typical amplitude response. Similarly, we demonstrate the interpolation filter in Fig-

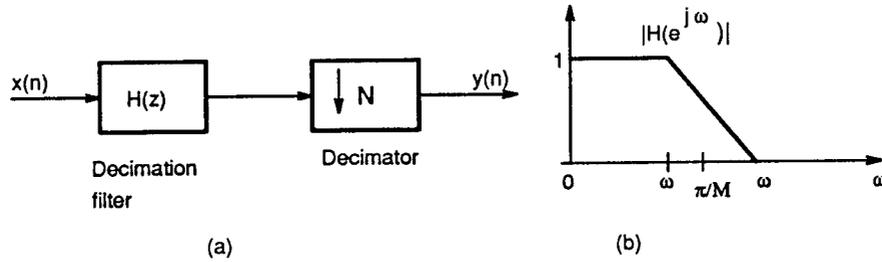


Figure 4.4: Decimation filter and its response

ure 4.5. Fractional sampling rate L/M alteration is possible by first passing the signal through an expander with a factor L and then through a filter followed by a decimator of factor M . The input-output relation in the time domain for

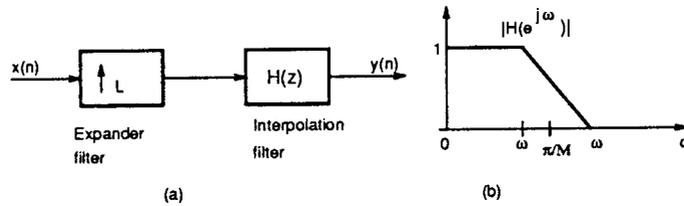


Figure 4.5: Interpolation filter and its response

the decimator, expander and the rational decimator are given by the following:

$$y(n) = \begin{cases} \sum_k x(k)h(nM - k), & M\text{-fold decimator} \\ \sum_k x(k)h(n - kL), & L\text{-fold expander} \\ \sum_k x(k)h(nm - kL), & M/L\text{-fold decimator} \end{cases} \quad (4.22)$$

where M and L are relatively prime in the M/L -fold decimator.

In the application of filter banks, the basic structure blocks above can be reorganized to form an interconnected system. Figure 4.6 demonstrates the Noble identities for changing the cascade orders of the decimator and decimation filter and that of the interpolation filter and expander. These identities are very

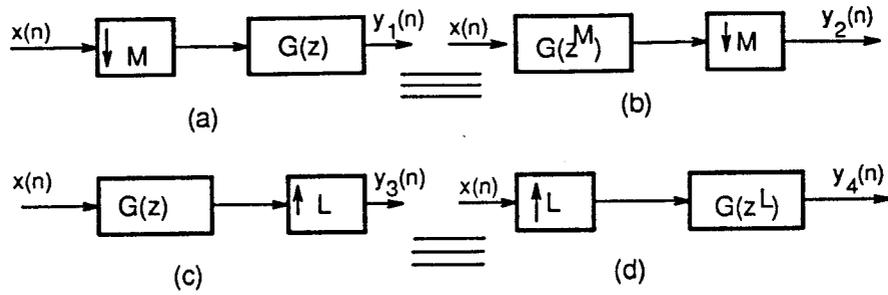


Figure 4.6: Noble identities for multirate systems

useful in the formulation and simplification of multirate systems.

4.3 QMF Banks and Wavelets

4.3.1 Quadrature Mirror Filter Banks

The earliest known quadrature mirror filters are related by the relation

$$H_1(z) = H_0(-z), \quad (4.23)$$

which apparently satisfies $|H_1(e^{j\omega})| = |H_0(e^{j(\pi-\omega)})|$. This means that if $H_1(z)$ is an ideal high pass filter, then $H_0(z)$ is an ideal low pass filter. These two filters form the components of QMF banks. It can be separated into two parts: the analysis bank and the synthesis bank as shown in Figure 4.7. As we shall see in Chapter 5, the QMF bank can be used to generate orthonormal wavelet bases under certain conditions.

Definition 4.3.1 (Vaidyanathan [48] p196) *A filter bank is said to be a perfect reconstruction system if its transfer function is a pure delay, free from aliasing, amplitude distortion and phase distortion, i.e., if $\tilde{x}(n) = cx(n - n_0)$.*

Definition 4.3.2 (Vaidyanathan [48] p288) *A rational transfer matrix $H(z)$ is said to be paraunitary if*

$$\tilde{H}(z)H(z) = dI, \text{ for all } z, \quad (4.24)$$

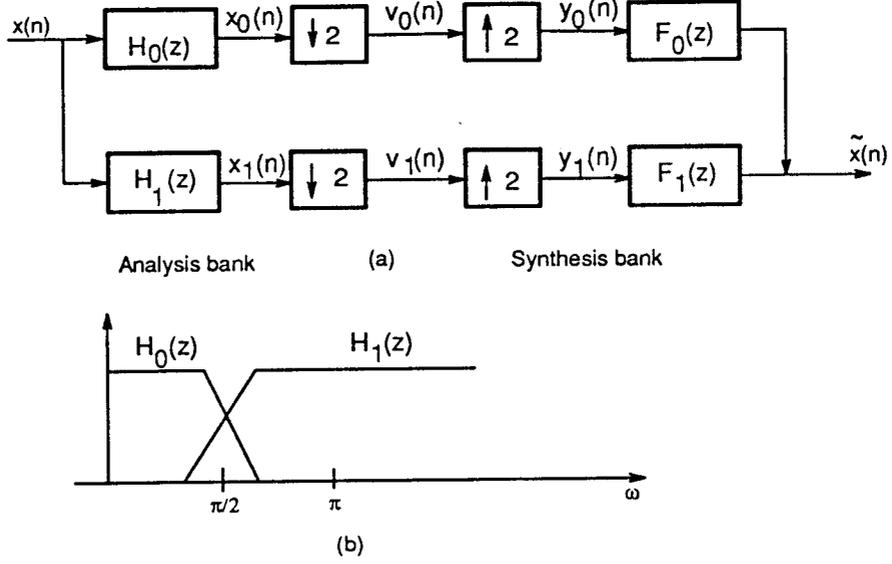


Figure 4.7: Quadrature mirror filter bank (a) and its response (b)

where d is a constant, I is the identity matrix.

The QMF bank in Figure 4.7 is in a general setting. The reconstructed signal is given by

$$\begin{aligned} \tilde{X}(z) &= \frac{1}{2}[H_0(z)F_0(z) + H_1(z)F_1(z)]X(z) \\ &\quad + \frac{1}{2}[H_0(-z)F_0(z) + H_1(-z)F_1(z)]X(-z). \end{aligned} \quad (4.25)$$

The $X(-z)$ term represents the aliasing component of the system due to decimation of the input signal $x(n)$. The matrix notation for the above is

$$\tilde{X}(z) = \frac{1}{2} \begin{bmatrix} X(z) & X(-z) \end{bmatrix} \begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{bmatrix} \begin{bmatrix} F_0(z) \\ F_1(z) \end{bmatrix}, \quad (4.26)$$

where $\mathcal{H}(z)$ is the alias-component (AC) matrix of the system which is given by

$$\mathcal{H}(z) = \begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{bmatrix}. \quad (4.27)$$

The conditions for this filter bank to be free of aliasing are the following [48]

$$F_0(z) = H_1(-z), \quad F_1(z) = -H_0(-z), \quad (4.28)$$

causing the coefficient of the $X(-z)$ to vanish thus cancelling the aliasing component. In fact, only the filter $H_0(z)$ needs to be designed for the system and the other filters are given by the relation above. If we further assume that $H_0(z)$ is power symmetric, i.e.,

$$\tilde{H}_0(z)H_0(z) + \tilde{H}_0(-z)H_0(-z) = 1, \quad (4.29)$$

and assign

$$H_1(z) = -z^{-N}\tilde{H}_0(-z), \quad (4.30)$$

where N is an odd integer, then, the transfer function of the QMF bank reduces to a pure delay of N time units. Thus, $\tilde{x}(n) = 1/2x(n - N)$ and the QMF bank forms a perfect reconstruction system.

For a general filter bank with analysis bank composed of FIR filters, we have

$$H_0(z) = \sum_{n=0}^N h_0(n)z^{-n}, \quad H_1(z) = \sum_{n=0}^N h_1(n)z^{-n}, \quad (4.31)$$

which are obviously causal. By definition, the AC matrix is paraunitary if

$$\begin{aligned} \tilde{H}_0(z)H_0(z) + \tilde{H}_0(-z)H_0(-z) &= d, \\ \tilde{H}_1(z)H_1(z) + \tilde{H}_1(-z)H_1(-z) &= d, \\ \tilde{H}_0(z)H_1(z) + \tilde{H}_0(-z)H_1(-z) &= 0. \end{aligned} \quad (4.32)$$

Lemma 4.3.1 *The AC matrix of (4.27) is paraunitary if $H_0(z)$ is power symmetric and*

$$H_1(z) = cz^{-N}\tilde{H}_0(-z), \quad |c| = 1, \quad (4.33)$$

or equivalently in the time domain,

$$h_1(n) = -c(-1)^n h_0^*(N - n), \quad (4.34)$$

where N is an odd integer.

The order of the filter H_0 is necessary to be odd to satisfy the constraint imposed on a zero-phase half band filter. For the two channel QMF bank, if we let $c = -1$, rearrange the relationship between the analysis filter bank and the synthesis bank, we have as in [48, p301]

$$H_1(z) = -z^N \tilde{H}_0(-z), \quad F_0(z) = z^N \tilde{H}_0(z), \quad F_1(z) = z^N \tilde{H}_1(z), \quad (4.35)$$

with the corresponding impulse responses in the time domain given by

$$h_1(n) = (-1)^n h_0^*(N - n), \quad (4.36)$$

$$f_0(n) = h_0^*(N - n), \quad \text{and} \quad (4.37)$$

$$f_1(n) = h_1^*(N - n), \quad (4.38)$$

respectively. These equations demonstrate that it is only necessary to design the filter $H_0(z)$ and to derive the other filters as above to formulate a perfect reconstruction two channel QMF bank for signal decomposition and synthesis. For filters with real coefficients, if $H_0(z)$ is a low pass filter, then the corresponding $H_1(z)$ is a high pass one. These two filters have a close relationship with orthonormal wavelets.

4.3.2 Orthonormal Wavelet Basis and QMF

A particularly useful set-up for our problem is a discrete orthonormal wavelet basis with compact support. This is useful for real time implementation on digital computers. The compactness of support provides a means of isolation and detection of signals at a certain region which has proven useful in the signal processing community. Our interest is in parameterizing the discrete wavelet

basis functions with a finite number of parameters so as to generate the optimal wavelet basis for system identification and signal representation.

From the multiresolution property of wavelets due to Mallat [35], for $\phi(t) \in V_j$, we have $\phi(2t) \in V_{j-1}$ and $\phi(2t - n)$ is a basis for the space V_{j-1} . Hence, we have the expression for the scaling function $\phi(t)$ with t denoting time as [19]

$$\phi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} c_k \phi(2t - k). \quad (4.39)$$

The corresponding discrete wavelet is given by

$$\psi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} d_k \phi(2t - k), \quad (4.40)$$

where the coefficient $\sqrt{2}$ is for normalization. These are the two fundamental equations for the scaling function $\phi(t)$ and wavelet function $\psi(t)$ which is determined by the scaling function $\psi(t)$ while the latter is to be parameterized by a finite set of parameters as we proceed.

Let us denote $h_0(k) = c_k$ and $h_1(k) = d_k$ and take their Fourier transforms

$$H_0(e^{j\omega}) = \sum_k h_0(k) e^{-j\omega k}, \quad (4.41)$$

and

$$H_1(e^{j\omega}) = \sum_k h_1(k) e^{-j\omega k}. \quad (4.42)$$

The coefficients $\{c_k\}$ and $\{d_k\}$ can be identified as a low pass filter and a high pass filter respectively. The frequency domain version of the fundamental equations are obtained by taking the Fourier transform of Equation (4.39) and Equation (4.40). This yields

$$\Phi(\omega) = \frac{1}{\sqrt{2}} H_0(e^{j\omega/2}) \Phi(\omega/2) \quad (4.43)$$

and

$$\Psi(\omega) = \frac{1}{\sqrt{2}} H_1(e^{j\omega/2}) \Phi(\omega/2) \quad (4.44)$$

where $\Phi(\omega)$ and $\Psi(\omega)$ are the Fourier transforms of $\phi(t)$ and $\psi(t)$, respectively. These two equations can be used recursively to generate the scaling and wavelet functions.

We need to consider the case when $H_0(z)$ is a causal FIR filter, i.e., the case in which only finitely many c_k 's are nonzero for the filter. Without loss of generality, we assume that $c_k \neq 0$ when $k \in [0, K]$ where K is a positive odd integer. The scaling function $\phi(t)$ can be nonzero only on $[0, K]$ due to the finite duration of the sequence $\{c_k\}$. The base wavelet function obtained through $\phi(t)$ is also compactly supported. With the FIR assumption, the two fundamental equations for the scaling function and the wavelet function become

$$\phi(t) = \sqrt{2} \sum_{k=0}^K c_k \phi(2t - k) \quad (4.45)$$

and

$$\psi(t) = \sqrt{2} \sum_{k=0}^K d_k \phi(2t - k), \quad (4.46)$$

respectively. We need to find the conditions for the generated wavelet function to produce an orthonormal basis for a subspace of $L^2(R)$ for function approximation and signal representation. Interestingly, the dyadic orthonormal wavelet functions can be related to binary tree structured QMF banks constructed from the two basis filters which determine the scaling function and the wavelet function.

Figure 4.8 (a) shows a three level dyadic tree structured QMF bank for wavelet transformation. The input sequence $x(n)$ is decomposed into different

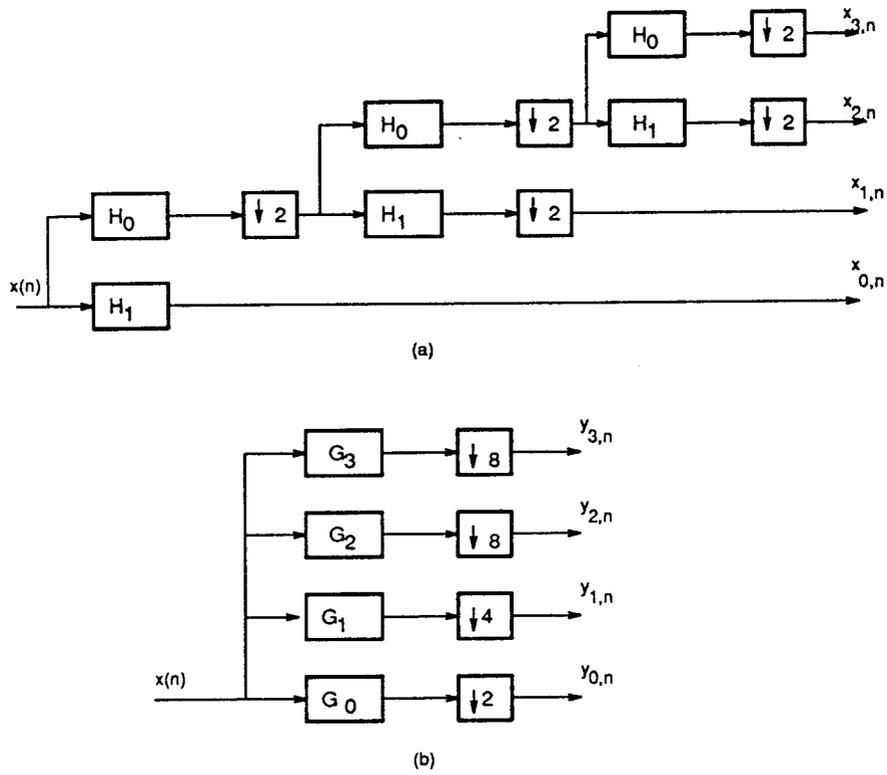


Figure 4.8: (a) Wavelet analysis QMF bank and (b) its equivalent four channel system

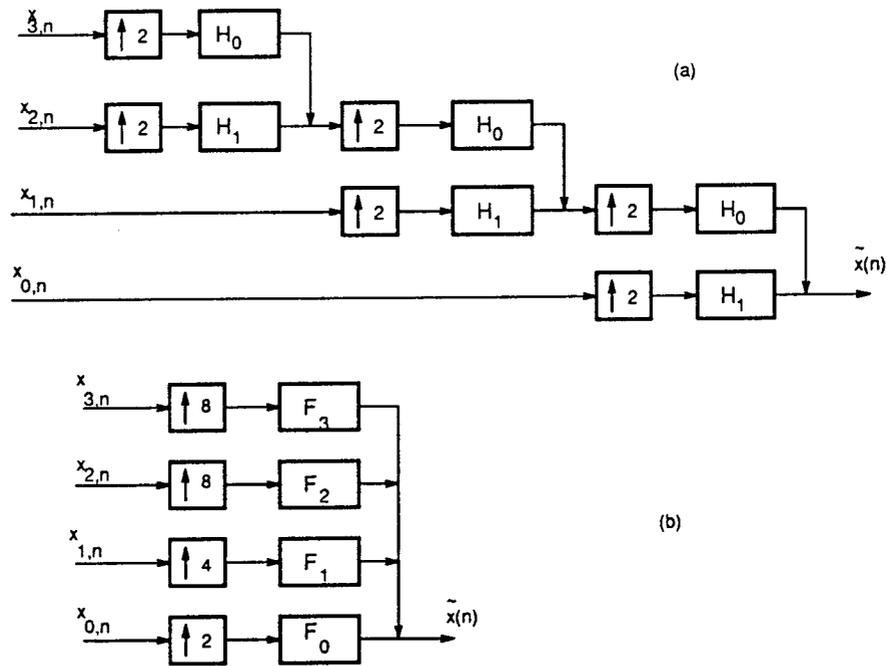


Figure 4.9: (a) Wavelet synthesis QMF bank and (b) its equivalent four channel system

resolutions by passing the signal through the QMF bank. The outputs $x_{i,n}$ are the related wavelet coefficients. Figure 4.8 (b) provides the equivalent four channel filter bank which is derived from (a) through applying the Noble identities of Figure 4.6. Figure 4.9 demonstrates the corresponding synthesis QMF bank whose input is the sequence of the wavelet coefficients while the output is the reconstructed signal $\tilde{x}(n)$ which is the wavelet representation of the original signal. The following theorem is a revised version of a theorem due to Vaidyanathan [48, p530].

Theorem 4.3.1 (Vaidyanathan [48] p530) *Let $H_0(z)$ and $H_1(z)$ be causal FIR filters, then the scaling function $\phi(t)$ and the wavelet function $\psi(t)$ generated by the QMF bank of Figure 4.8 and Figure 4.9 are causal with finite duration Kb_0 . Further, if $H_0(z)$ and $H_1(z)$ satisfy paraunitary condition (4.32), $|H_0(1)| = \sqrt{2}$ and $H_0(e^{j\omega}) \neq 0$ while $|\omega| < \pi/2$, the wavelet functions $\psi_{j,l}(t)$ are orthonormal.*

The condition imposed on orthonormality of wavelets can be relaxed when the number of levels of the QMF tree is finite. In this case, both the scaling function and the base wavelet function are obtained through finite recursion by using Equation (4.43) and Equation (4.44) respectively, i.e., the paraunitary condition alone is enough to guarantee the orthonormality of the wavelet functions. A proof of this fact in the frequency domain is provided in [48]. As a matter of fact, this is the usual situation in practical application and implementation.

We summarize the conditions for compactly supported scaling functions and wavelets to be orthonormal based on the theorem above and the properties of AC matrices of the two channel QMF banks as a lemma as follows.

Lemma 4.3.2 *A compactly supported scaling function and wavelets generated through the finite recursion are orthonormal if the matrix*

$$\mathcal{H}(\omega) = \begin{bmatrix} H_0(e^{j\omega}) & H_1(e^{j\omega}) \\ H_0(e^{j(\omega+\pi)}) & H_1(e^{j(\omega+\pi)}) \end{bmatrix} \quad (4.47)$$

is paraunitary for all ω for the two-channel quadrature mirror filter (QMF) bank.

This is the constraint that the parameters c_k should satisfy in order to generate an orthonormal wavelet basis. In particular, by Lemma 4.3.1, the cross-filter orthonormality implied by the unitary property is satisfied by the choice

$$H_1(z) = -z^{-K} H_0(-z^{-1}), \text{ K odd} \quad (4.48)$$

or, in the time domain,

$$h_1(k) = (-1)^k h_0(K - k). \quad (4.49)$$

As we can see from the above, both the scaling function and the wavelet function depend on the selection of $\{c_k\}$ for $k \in [0, K]$. As a consequence, the dilations and shifts of the base wavelet depend on the selection of this set of parameters subject to the paraunitary condition imposed on the filters of the QMF bank.

4.4 Conclusion

Compactly supported wavelets and QMFs are closely related. Indeed, the design of a good wavelet for signal representation can start from building the appropriate power symmetric low pass filter H_0 and the corresponding QMF bank. In order for the wavelet system to really represent the signal or reconstruct the signal with a simple structure, it is necessary to further introduce optimal design techniques which are the tasks to be introduced in the next chapter.

Chapter 5

Optimal Wavelet Basis for Signal Representation

5.1 Introduction

We know that wavelet functions can be used for function approximation and finite energy signal representation in signal processing and system identification. The wavelet basis is generated by dilating and shifting a single base wavelet function $\psi(t)$. As we addressed in the previous section, that the wavelet function is not unique and its design can be related to that of a power symmetric FIR low pass filter. Obviously, different wavelet $\psi(t)$ shall yield different wavelet bases. However, the choice of the wavelet for signal representation should be appropriate to receive the maximal benefits of this new technique. Different wavelet functions may be suitable for different signals or functions to be represented or to be approximated. It is reasonable to think that if a wavelet contains enough information about a signal it is going to represent, the total complexity of the wavelet system is going to be simplified in terms of the level of resolution required which then reduces the computational complexity of the problem for system implementation.

The key to choosing the optimal wavelet basis for signal representation and

construction of adaptive wavelet neural networks (AWNNs) [52] lies in the appropriate parameterization and the right performance measure in addition to the accurate interpretation of physical phenomena. A method is proposed in [45] [26] for choosing a wavelet for signal representation based on minimizing an upper bound of the L^2 norm of error in approximating the signal up to the desired scale. Coifman et al. derived an entropy based algorithm for selecting the best basis from a library of wavelet packets [13]. However, a direct method to systematically generate a signal based optimal orthonormal discrete wavelet basis with compact support is still to be developed.

In this chapter, we study the problem of choosing the optimal wavelet basis with compact support as the continuation of our previous work on the selection of wavelet bases [55]. We first introduce the concepts of information measure as a distance measure and the optimal discrete orthonormal wavelet basis under the information measure. We then derive the information gradient for construction of the optimal wavelet basis. We also provide a proof of the existence of an optimal wavelet basis. A systematic approach is developed for derivation of the best wavelet basis. This approach may be implemented for real time systems due to our parameterization of the problem.

5.2 Parameterization of Wavelet Functions and Information Measures

We first introduce a distance measure for optimization purposes. Inspired by the work in [13], we define an additive information measure of entropy type and the optimal basis as the following.

Definition 5.2.1 (Coifman et al [13]) *A non negative map \mathcal{M} from a sequence $\{f_i\}$ to R is called an additive information measure if $\mathcal{M}(0) = 0$ and $\mathcal{M}(\sum_i f_i) = \sum_i \mathcal{M}(f_i)$.*

Definition 5.2.2 *Let $x \in R^N$ be a fixed vector and \mathcal{B} denote the collection of all orthonormal bases of dimension N , a basis $B \in \mathcal{B}$ is said to be optimal if $\mathcal{M}(Bx)$ is minimal for all bases in \mathcal{B} with respect to the vector x .*

We shall define a distance measure between a signal and its decompositions to subspaces of $L^2(R)$ motivated by Shannon entropy (Shannon's formula) [24]

$$H(X) = H(P) = - \sum_{x \in X} P(x) \log P(x), \quad (5.1)$$

which is interpreted as a measure of the information content of a random variable X with distribution $P_x = P$ in information theory.

Definition 5.2.3 *Let H be a Hilbert space which is an orthogonal direct sum*

$$H = \oplus \sum H_i, \quad (5.2)$$

a map \mathcal{E} is called decomposition entropy if

$$\mathcal{E}(v, \Psi) = - \sum \frac{\|v_i\|^2}{\|v\|^2} \log \frac{\|v_i\|^2}{\|v\|^2} \quad (5.3)$$

for $v \in H$, $\|v\| \neq 0$, such that

$$v = \oplus \sum v_i, v_i \in H_i, \quad (5.4)$$

and we set

$$p \log p = 0, \text{ when } p = 0. \quad (5.5)$$

Entropy is a good measure for signal concentration in signal processing and information theory. The value of $\exp \mathcal{E}(v)$ is proportional to the number of

coefficients and the length of code words necessary to represent the signal to a fixed mean error and to error less coding respectively. The number $\frac{\|v_i\|^2}{\|v\|^2}$ is the equivalent probability measure in the decomposition entropy which is the stochastic approximation of Shannon entropy since the density function of the signal is unknown. For a source of a finite number of independent signals, such as a digital image considered as a source of independent pixels, its entropy is maximum for uniform distribution [25, p42]. If the entropy value is less than the maximum, then, this implies that a higher concentration of the signal energy over certain bands exists. In our system identification formulation, energy concentration is identified with models of lower orders or networks with less complexity. Using entropy as a performance measure takes advantage of the fact of the nonuniform energy distribution of the signal or systems in consideration. The optimization of the wavelet basis is finding the suitable wavelet for a certain class of signals which have energy concentration at certain frequency bands. In other words, we are seeking a representative of a certain class of signals to generate suitable subspaces in which the decomposition entropy is minimized or equivalently that the energy of the signal is concentrated.

Let $\psi(t)$ be the base wavelet function and let $\Psi(t)$ represent the orthonormal discrete wavelet basis of L^2 generated by dilation and shifting of $\psi(t)$, similarly, we define Ψ_j to be the basis of H_j . We write $\Psi(t) = \{\psi_{j,l}(t)\}$ and $\Psi_j(t) = \{\psi_{j,l}(t)\}_{l \in \mathbb{Z}}$ respectively. We treat both $\Psi(t)$ and Ψ_j as operators and thus define the following.

Definition 5.2.4 *Let Ψ be a given basis as above, a basis operation is defined to be a map from $L^2(\mathbb{R})$ to a set of real numbers, i.e., $\Psi(t)f(t) = \{f_{j,l}\}_{j,l \in \mathbb{Z}}$ where $f_{j,l} = \langle f(t), \psi_{j,l}(t) \rangle$ for all $f(t) \in L^2$.*

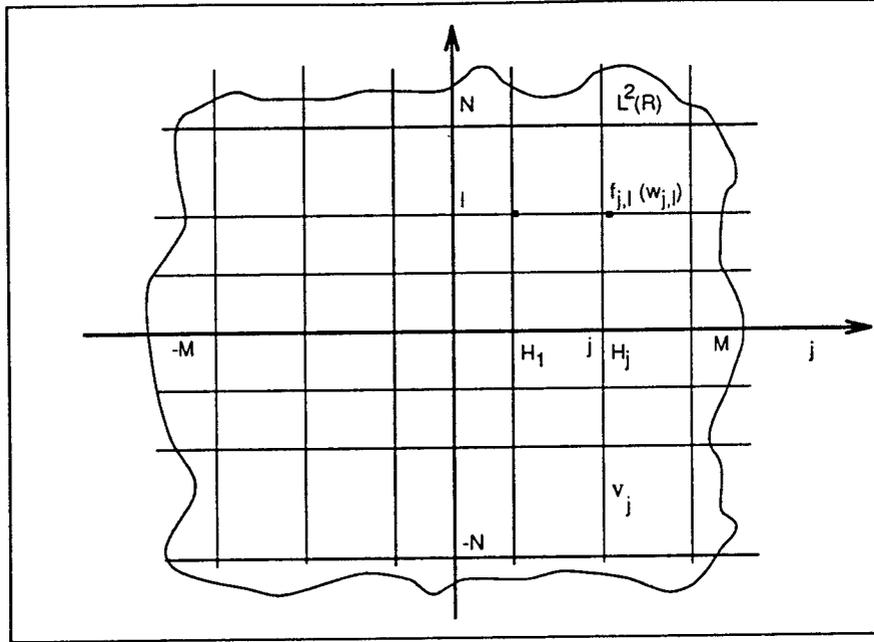


Figure 5.1: Mesh structure of the projection space

Let M and N be appropriate positive integers. We truncate the approximation in Equation (4.14) to a scale up to M

$$f(x) = \sum_{j=-M}^M \sum_{l=-N}^N w_{j,l} \psi_{j,l}(x). \quad (5.6)$$

The subspaces used to approximate the function $f(x)$ has a mesh structure of size $(2M + 1) \times (2N + 1)$ as in Figure 5.1.

Given a function or signal $f(t) \in L^2(R)$ and a base wavelet function $\psi(t)$ with a finite mesh of size $(2M + 1) \times (2N + 1)$, we can decompose the signal into the orthogonal subspaces as

$$f(t) = \sum_{j=-M}^M \sum_{l=-N}^N f_{j,l} \psi_{j,l}(t). \quad (5.7)$$

We are going to find the best wavelet base function $\psi(t)$ for a given signal $f(t)$ such that the additive information measure \mathcal{M} is minimized. The base operation $\Psi f(t)$ yield the weights on the nodes of the mesh. The weights on the vertical line with coordinate j is the number set produced by $\psi_j f(t)$.

Although the decomposition entropy is a good measure for the “distance,” it is not an additive type of map because the norm $\|v\|$ is used to scale the vector. We thus further introduce a cost functional

$$\lambda(\Psi, v) = - \sum_j \|v_j\|^2 \log \|v_j\|^2, \quad (5.8)$$

which relates to the decomposition entropy through

$$\mathcal{E}(v, \Psi) = \|v\|^{-2} \lambda(\Phi, v) + (2M + 1) \log \|v\|^2. \quad (5.9)$$

As shown in the expression above, the cost functional λ takes the wavelet basis Ψ and the signal vector v as its arguments. For any fixed signal, it is a functional of the basis and hence of the base wavelet function $\psi(t)$. The former function in (5.8) is an additive measure. Since the above two functionals share the same set of minimal points, we minimize the functional $\lambda(\Phi, f)$ for seeking the optimal wavelet basis through multiresolution decompositions.

The weight of the decomposition of a signal $f(t)$ on a subspace H_j is measured by the subnorm $\|f_j\|$ defined as

$$\|f_j(t)\| = \|P_{H_j}[f(t)]\| \quad (5.10)$$

where

$$\|f_j\|^2 = \sum_{l=-N}^N f_{j,l}^2. \quad (5.11)$$

Similarly, the norm of the decomposed signal is given by

$$\|f(t)\|^2 = \sum_{j=-M}^M \|f_j\|^2. \quad (5.12)$$

5.3 Sensitivity Gradient of Wavelet Components

We need to further find $\frac{\partial f_{j,l}}{\partial c_k}$ which is a measure of the sensitivity of the component of signal decomposition to a wavelet basis versus the change of the defining parameter set of the base wavelet. One can solve this through numerical methods from the relations and definitions. Based on the definition of information measure and the properties of QMF discussed earlier, we derive an explicit expression for analyzing the sensitivity of a dilated and shifted wavelet function versus the parameter set as follows.

Lemma 5.3.1 *The sensitivity gradient $\frac{\partial \psi_{j,l}}{\partial c_k}$ of the component $\psi_{j,l}$ of the wavelet basis Ψ versus parameter c_k is given by*

$$\begin{aligned} \frac{\partial \psi_{j,l}}{\partial c_k} = & \sqrt{2^{-j+1}} \sum_n \left[(-1)^{K-k} \phi(2^{-j+1}t - 2l - n) \right. \\ & \left. + (-1)^n \sqrt{2} c_{K-n} \phi(2^{-j+2}t - 4l - 2n - k) \right]. \end{aligned} \quad (5.13)$$

Proof:

From the fundamental equation of wavelets (4.46) and the wavelet basis function,

$$\frac{\partial \psi_{j,l}}{\partial c_k} = \sqrt{2^{-j+1}} \frac{\partial}{\partial c_k} \sum_n h_1(n) \phi(2^{-j+1}t - 2l - n). \quad (5.14)$$

This is

$$\frac{\partial \psi_{j,l}}{\partial c_k} = \sqrt{2^{-j+1}} \sum_n \left[\frac{\partial h_1(n)}{\partial c_k} \phi(2^{-j+1}t - 2l - n) + h_1(n) \frac{\partial}{\partial c_k} \phi(2^{-j+1}t - 2l - n) \right]. \quad (5.15)$$

From Equation (4.45), we have

$$\frac{\partial \phi(t)}{\partial c_k} = \sqrt{2} \phi(2t - k). \quad (5.16)$$

hence,

$$\frac{\partial}{\partial c_k} \phi(2^{-j+1}t - 2l - n) = \sqrt{2} \phi(2^{-j+2}t - 4l - 2n - k). \quad (5.17)$$

We need to find $\frac{\partial h_1(n)}{\partial c_k}$. From the time domain relation (4.49) of the QMF, we have ,

$$h_1(n) = (-1)^n h_0(K - n) \quad (5.18)$$

with h_0 being compactly supported on $[0, K]$. Thus,

$$h_1(n) = \frac{\partial}{\partial c_k} (-1)^n c_{K-n}, \quad (5.19)$$

there is only one nonzero term when $K - n = k$. This yields,

$$\frac{\partial h_1(n)}{\partial c_k} = (-1)^{K-k}. \quad (5.20)$$

The lemma is proven through (5.17) and (5.20).

□

This lemma establishes a direct link between the rate of change of the components in the basis Ψ and the variations of the parameters in the fundamental equations of wavelets, which leads to the next theorem. We introduce the following theorem to show the relationship between the information measure and the parameter set c_k . The relation to be described shall provide a clue for developing an algorithm to find the optimal base wavelet function for the AWNN.

Theorem 5.3.1 *Let $\lambda(\cdot, \cdot)$ be the additive information measure as defined in (5.8) and $f(t) \in L^2(\mathbb{R})$ be a fixed signal. Let $[0, K]$ be the compact support for $\{c_k\}$ and let Ψ be the corresponding wavelet basis from dilations and shifts of the mother wavelet $\psi(t)$. Then the gradient of the information measure with respect*

to the parameter set $\{c_k\}$ for the given signal is given by

$$\begin{aligned} \frac{\partial \lambda(\Psi, f(t))}{\partial c_k} &= -\sqrt{2^{-j+2}} \sum_j \sum_l \log 2 \|f_j\|^2 \\ &\quad \cdot f_{j,l} \sum_n \left[(-1)^{K-k} \langle f(t), \phi(2^{-j+1}t - 2l - n) \rangle \right. \\ &\quad \left. + (-1)^n c_{K-n} \langle f(t), \phi(2^{-j+2}t - 4l - 2n - k) \rangle \right]. \end{aligned} \quad (5.21)$$

Proof:

By the chain rule, we have the information gradient

$$\frac{\partial \lambda(\Psi, f(t))}{\partial c_k} = \sum_j \frac{\partial \lambda(\Psi, f(t))}{\partial \|f_j\|^2} \frac{\partial \|f_j\|^2}{\partial c_k}. \quad (5.22)$$

The definition of information measure $\lambda(f(t))$ in (5.8) yields,

$$\begin{aligned} \frac{\partial \lambda(\Psi, f(t))}{\partial \|f_j\|^2} &= -\log \|f_j\|^2 - 1 \\ &= -\log 2 \|f_j\|^2, \end{aligned} \quad (5.23)$$

with 2 being the base of \log function. We use the chain rule again,

$$\begin{aligned} \frac{\partial \|f_j\|^2}{\partial c_k} &= \frac{\partial}{\partial c_k} \sum_l f_{j,l}^2 \\ &= 2 \sum_l f_{j,l} \frac{\partial f_{j,l}}{\partial c_k}. \end{aligned} \quad (5.24)$$

We have so far

$$\frac{\partial \lambda(\Psi, f(t))}{\partial c_k} = -2 \sum_j \sum_l \log 2 \|f_j\|^2 f_{j,l} \frac{\partial f_{j,l}}{\partial c_k}. \quad (5.25)$$

Since

$$\frac{\partial f_{j,l}}{\partial c_k} = \left\langle f(t), \frac{\partial \psi_{j,l}}{\partial c_k} \right\rangle, \quad (5.26)$$

the result from the previous lemma concludes the proof.

□

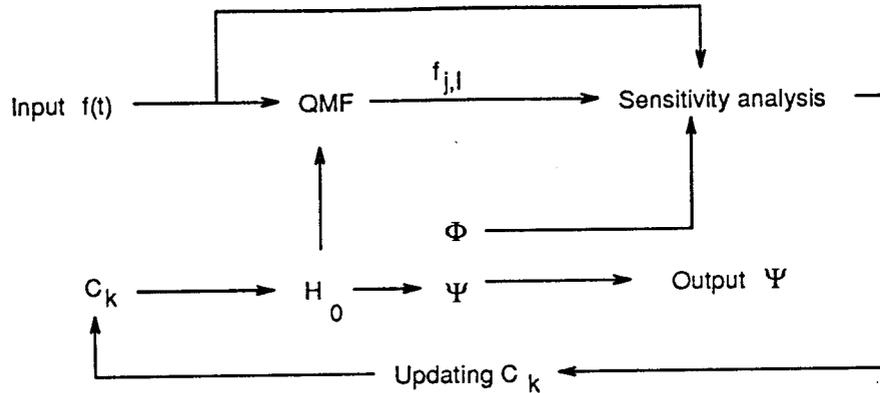


Figure 5.2: Flow chart for computing $\psi(t)$

This theorem demonstrates an explicit relation among the gradient of the additive information measure, parameter set $\{c_k\}$ and the measured signal $f(t)$. It points to the direction for updating the parameter set to reduce the information measure and thus will facilitate the search for the optimal wavelet basis.

We may propose a gradient based scheme of construction of the optimal wavelet basis from the theorem above. This algorithm starts by assigning an initial set of parameters which form the low pass filter of the QMF bank which is followed by the generation of both scaling function $\phi(t)$ and the base wavelet $\psi(t)$ through a recursive process. The wavelet decomposition is implemented by passing the input signal through the QMF bank composed of $H_0(z)$ and $H_1(z)$. The parameter set $\{c_k\}$ is updated after sensitivity analysis. The flow chart in Figure 5.2 describes this process.

5.4 Analysis of Time and Frequency Allocation

The parameter set $\{c_k\}$ is updated based on the result from sensitivity analysis and the final wavelet is generated upon the final set of parameters.

The values of M and N determine the computational complexity of signal

representation and system identification. In our problem, these values can be identified with the size of AWNN. Obviously, when both M and N turn to infinity, the constructed function converges to the original function. However, in real system implementation, one has to use the truncated model instead of the form of infinite summation. As a consequence, the selection of the values of M and N becomes a practical issue. One can also start the computation by an educated guess and uses an adaptive scheme on the level of resolution required and the size of each subspace of the given resolution which is described by the value of N . Knowledge of the system and of the signal may also help choosing the initial size of the mesh in Figure 5.1.

One applicable method is to analyze the time and frequency allocation of the signal from available data. We wish to estimate the bandwidth of the signal and estimate its energy concentration on the time and frequency plane so as to provide a initial guess of M and N . The choice of M and N shall provide a mesh which covers the region where the energy of signal is concentrated. In this way, one can hope to represent the signal with a wavelet network of a reasonable size. This process is not unique and does not have a precise solution. Different methods, prior knowledge about the system and common sense should be used to solve this problem. Some discussion on time-frequency allocation analysis for system approximation can be found in [41].

5.5 Algorithms

We have identified the problem of finding the optimal wavelet basis Ψ with that of finding a parameter set $\{c_k\}$ such that the additive information measure λ is minimized. Once the set $\{c_k\}$ is determined, both the scaling function ϕ and the

base wavelet function ψ can be derived afterwards. Equipped with the above theorem, the information gradient is available, different optimization schemes can be applied to solve this problem. We have developed a basis selection algorithm based on a steepest descent method as follows. To simplify notation, we denote the parameter set $\{c_0, c_1, \dots, c_{K-1}\}$ by a vector C .

Algorithm 5.5.1 *Computation of the optimal wavelet basis*

Step 1: Set $i := 1$,

$$\lambda_0 := 0,$$

mesh parameters M, N ;

Initialize vector C_0 ;

Input $f(t)$.

Step 2: If C_i dose not satisfy the constraint,

then, modify C_i and repeat Step 2.

$$\text{Step 3: } C_i := C_{i-1} + p_{i-1} \frac{\partial \lambda}{\partial C_{i-1}}.$$

Step 4: Compute ϕ and ψ .

Step 5: Compute λ .

Step 6: If $|\lambda_i - \lambda_{i-1}| > \epsilon$,

$i := i + 1$, go to Step 2.

Step 7: Output the optimal basis Ψ and stop.

The mesh size is governed by the choice of parameter M and N . Obviously, when M and N turn to infinity, the supporting subspace spanned by the dilations and shifts of the base wavelet turns to space $L^2(\mathbb{R})$. The size of the mesh is identified with the complexity of the resulted wavelet system. The constraint on the parameter c_k is dominated by the unitary property of the QMF bank which

can be transformed into a set of algebraic equations. The parameters M and N can be predetermined by the time and frequency localization property of the signal in consideration. We can also perform an adaptation scheme to generate a mesh of an appropriate size. This is realized by a modified algorithm as follows.

Algorithm 5.5.2 *Computation of the optimal wavelet basis with variable AWNN size.*

Step 1: Set $i := 1$,

$$\lambda_0 := 0,$$

mesh parameters M, N ;

Initialize vector C_0 ;

Input $f(t)$.

Step 2: If C_i does not satisfy the constraint,

then, modify C_i and repeat Step 2.

$$\text{Step 3: } C_i := C_{i-1} + p_{i-1} \frac{\partial \lambda}{\partial C_{i-1}}.$$

Step 4: Compute ϕ and ψ .

Step 5: Compute λ .

Step 6: If $|\lambda_i - \lambda_{i-1}| > \epsilon$,

$$i := i + 1,$$

$$M := M + 1,$$

$$N := N + 1, \text{ go to Step 2.}$$

Step 7: Output the optimal basis Ψ and stop.

This algorithm starts from an initial mesh size determined by M and N in step 1. While updating the parameter set $\{C_i\}$, the algorithm adjusts the size

of the mesh until the error tolerance is met to finish the iterative process. The sequence of order updating and parameter updating can be organized adequately for reducing computational complexity.

5.6 Structure Analysis and System Parameterization

We have developed in the previous section the formulation of the information measure and the sensitivity analysis of wavelets versus their parameter set which determine the amount of information represented by the base wavelet $\psi(t)$. We also formulated the optimization problem and proposed a gradient based optimization algorithms. However, the algorithms in the previous section are constrained procedures which guarantee the paraunitary property of the filters in the QMF bank. In this section, we are going to parameterize the optimization into an unconstrained optimization through analyzing the structures of unitary matrices over the complex field. The structure analysis shall reduce the number of parameters of the system and should be an economical way to express and generate a family of wavelets.

5.6.1 Decomposition of Unitary Matrices

We shall show that the paraunitary matrix discussed earlier can be decomposed to its minimum forms and thus be parameterized for the optimization purpose. We first introduce some basic definitions and notations to facilitate the description of our problems. Let F be a subfield of C closed under complex conjugation, let U be the multiplication group of two by two unitary matrices over the ring of Laurent polynomials $F(z)$ with $|z| = 1$. Define a natural conjugate operation denoted by \sim which takes the complex conjugate of F and replaces z by its re-

reciprocal. Denote \hat{U} to be the subgroup of matrices in U which have determinant 1 and equal identity matrix when evaluated at $z = 1$. Use UI to represent the two by two multiplication group of unitary matrices over the subfield F . It is obvious that the aliasing component (AC) matrix in (4.27) belongs to U . From the previous definition of a paraunitary matrix, it can easily be converted as a unitary matrix here with a scaling factor d . Without further explanation, we shall identify those two in the rest of this section.

Theorem 5.6.1 (Pollen [42]) *For every element $M \in U$, there exist unique*

$$A \in UI, B \in \begin{bmatrix} 1 & 0 \\ 0 & z^k \end{bmatrix}, C \in \hat{U}$$

with $k \in Z$ such that $M = ABC$.

The uniqueness holds upon the choice of the parameter k of matrix B . To proceed further, we define the order of a matrix in U to be the highest order of the Laurent polynomial in the matrix. This theorem shows [42] that we can factor the matrix in U into the product of a zero order polynomial matrix, a monomial matrix and a matrix which has unit determinant and is equal to the identity matrix upon evaluation at $z = 1$. The process can be repeated when the order of the factor C is greater than one. Hence, we may have

$$\begin{aligned} M &= ABC \\ &= ABA_1B_1C_1 \\ &= ABA_1B_1A_2B_2 \cdots A_mB_mC_m \end{aligned} \tag{5.27}$$

where $m \leq \text{deg}(M)$. The matrices in U have a nice property of being able to be expressed as a unique product of a minimal number of simple factors in \hat{U} .

Definition 5.6.1 A matrix $X \in \hat{U}$ is called a simple factor if its degree is not more than one.

Lemma 5.6.1 A 2×2 matrix X is a simple factor if

$$X = \begin{bmatrix} a + (1 - a)z & b - bz \\ -(b - bz) & a + (1 - a)/z \end{bmatrix}$$

with $|a - \frac{1}{2}|^2 + |b|^2 = (\frac{1}{2})^2$.

The set of simple factors is a subset of the collection of all the degree 0 or 1 elements of UI .

Proof:

Obviously, X is an identity matrix when $z = 1$. The circle condition regarding a and b in the theorem proves that $\det(X) = 1$.

□

The topology of the factor is shown here to be a circle when the field is real, or a 2-sphere for complex field. The set of all simple factors is a subset of the set composed of all the matrices of degree zero and one.

The unique factorization theorem for second order matrices in \hat{U} due to Pollen is introduced here without proof for the parameterization of the unitary matrices.

Theorem 5.6.2 (Pollen [42]) Let $M \in \hat{U}$ with $\deg(M) > 0$, then there exist simple factors A_i and B_i such that

$$M = \tilde{A}_1 B_1 \tilde{A}_2 B_2 \cdots \tilde{A}_m B_m.$$

This expression is the factorization of M as a product of $2m$ simple factor elements and factor inverses. Furthermore, the factorization is unique. A slightly

weaker form of the decomposition is that M is factorized into a product of m elements of degree one, i.e., $M = C_1 C_2 \cdots C_m$. However, these elements may not be simple factors.

This theorem implies that the original parameters of the matrix can be parameterized by the factors in a simple form and the factors can be derived from the coefficient matrices of the original unitary matrix M . This property is especially useful since we consider the paraunitary property of the filter transfer function H_0 and H_1 on the unit circle in the z plane which falls into the realm of this unique factorization theorem.

5.6.2 Givens Rotation and QMF Lattice Structures

This section intends to transform the constrained optimization of the optimal wavelet function into an unconstrained problem through parameterization. The decomposition theorem shows that it is possible to formulate the problem with z on the unit circle which is convenient to be treated by trigonometric functions. We need to find an appropriate mapping between the parameters of the QMF and the corresponding set of unconstrained parameters on the unit circle. We then apply the numerical procedure discussed earlier to find the right QMF bank and thus to find the optimal wavelets. One of the convenient ways is to use Givens rotation as following.

From the decomposition theorem 5.6.1, for any matrix in U with degree greater than or equal to one, we can apply the theorem again until all the factors have degree zero or one as defined above. The resulting factors are of the form of a 2×2 paraunitary transfer matrix

$$\Lambda(z) = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}. \quad (5.28)$$

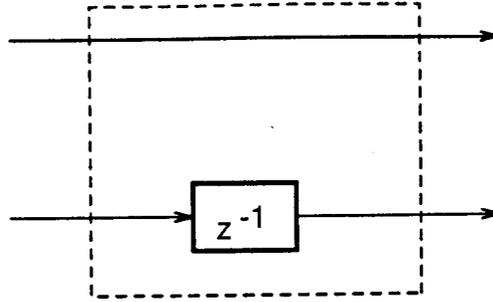


Figure 5.3: Block structure of the unit factor

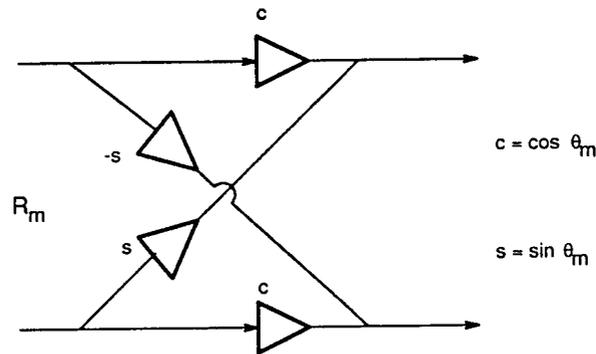


Figure 5.4: A structure block of Givens rotation

Figure 5.3 shows the block structure of this basic factor in U . We notice that the product of two unitary matrices is still unitary. We further introduce the transfer matrix

$$R_m = \begin{bmatrix} \cos\theta_m & \sin\theta_m \\ -\sin\theta_m & \cos\theta_m \end{bmatrix}, \quad (5.29)$$

where θ_m is real. This is a Givens rotation which is used to parameterize the AC matrix on the unit circle. The block structure of the Givens rotation is shown in Figure 5.4. It can be easily shown that the determinant of the AC matrix (4.27) is βz^{-K} where β is a constant coefficient for the low pass filter $H_0(z)$ of degree K . At most, the AC matrix has K such factors in its decomposition form.

Let $H_K(z)$ denote the 2×2 aliasing component matrix of degree K , the

factorization through Givens rotation is given by

$$H_K(z) = \alpha R_0 \Lambda_0 R_1 \Lambda_1 \cdots R_{K-1} \Lambda_{K-1} R_K \Lambda_K \begin{bmatrix} 1 & 0 \\ 0 & \pm 1 \end{bmatrix}, \quad (5.30)$$

where α is a constant. The structure of the factorization is a cascade system in which the basic blocks R_i and $\Lambda(z)$ appear alternatively. The parameters of the QMF bank, i.e., of the low pass filter $H_0(z)$, are functions of those angles θ_m where m runs from zero to K . The number of parameters of the QMF and that of the factorization through applying Givens rotation are the same. Hence, the parameter set $\{c_k\}$ can be expressed as a continuous nonlinear function of those angles which take their values in the closed interval $[0, 2\pi]$. We represent the relation as

$$C = P(\theta), \quad (5.31)$$

where P denotes the nonlinear function of the parameterization, C and θ represent the corresponding $K + 1$ dimensional vectors of parameters. We can minimize (5.9) through the gradient based method with

$$\frac{\partial \lambda(\Psi, f(t))}{\partial \theta_k} = \sum_{m=0}^K \frac{\partial \lambda(\Psi, f(t))}{\partial c_m} \frac{\partial c_m}{\partial \theta_k}. \quad (5.32)$$

The optimization of the cost functional (5.9) turns into an unconstrained optimization on the compact set $[0, 2\pi]$.

5.6.3 The Existence of the Optimal Wavelet Basis

With the parameterization of the low pass filter $H_0(z)$ by the rotation vector θ over the set $[0, 2\pi]^{K+1}$, the existence of the global minimum is guaranteed although the gradient based algorithm may stop at a local minimum. The parameter set C is computed from the final set of θ 's to further improve the optimal base wavelet $\psi(t)$ and the optimal compactly supported orthonormal

wavelet basis. The next theorem is a main result of this chapter.

Theorem 5.6.3 (Existence of an optimal wavelet basis) *Let $\lambda(\cdot, \cdot)$ be the additive information measure as defined in (5.8) and let $f(t) \in L^2(\mathbb{R})$ be an arbitrary finite energy function. Then there exists an optimal base wavelet function $\psi^*(t)$ such that, $\lambda(\Psi^*, f(t)) = \lambda_{\min}$ where Ψ^* is the wavelet basis generated by the dilations and shifts of $\psi^*(t)$.*

Proof:

By the definitions, λ is continuous with respect to its arguments, Ψ , $f(t)$ and the parameter set C , the nonlinear function in (5.31) is a continuous bijection with $\theta \in [0, 2\pi]^{K+1}$, a compact set, as the consequence, the information measure parameterized by the angular vector θ is a continuous functional defined on a compact manifold $[0, 2\pi]^{K+1}$, hence, there exists a vector $\theta^* \in [0, 2\pi]^{K+1}$, at which, λ attains its minimum. The optimal wavelet is thus determined through C , the coefficients of the low pass filter in the QMF bank.

□

The existence theorem guarantees that the global optimal solution does exist. However, as mentioned earlier, the computation may end up at a local minimum depending on the starting point and thus yields a suboptimal solution which in many cases may still provide satisfactory performance.

5.7 Simulated Annealing for Global Optimization

We have proven the existence of the optimal wavelet for signal representation and have developed an algorithm for generating the optimal wavelet in the pre-

vious sections. The resulting gradient method is straightforward to implement. However, it is important to emphasize that the outcome of the optimal base wavelet from the optimization scheme depends on the choice of the initial point. The gradient based method will stop at the local minimum nearest to the initial point. To overcome this limitation, we need to select an appropriate initial point or to resort to other remedies.

5.7.1 Stochastic Relaxation

Stochastic relaxation is a viable option for searching for the global optimal. It makes possible to evade local minima, to reduce or to eliminate the dependency of the outcome on the initial choice of data. At each iteration, the parameter set is perturbed in a random fashion by a prescribed amount. The cost functional is then evaluated at the perturbed state which tends to increase the value of the cost functional randomly. The local change of the parameter set can be modeled by its neighborhood or by the fuzzy set generated by the random disturbance. The magnitude of disturbance generally decreases with time, or, with the increase of the number of iterations. A prescribed threshold decides the acceptance or rejection of the result of each iteration. This threshold is in general a function of time, of the variance of the stochastic disturbance and of the amount of increment of the value of the cost functional. If the cost decreases, the updated state is accepted; if the cost increases, it is accepted with probability. The essence of stochastic relaxation is that the parameter changes which increase cost functional but with lower probability are permitted. Hence, the optimization process is not going to be trapped at local optima. In the other words, the results from the iterative process is irrelevant to the initial point in the space of feasible solutions.

The local change of parameter set depends on a global control parameter called *temperature*. At low temperature, the local conditional distribution concentrates on decreasing the cost functional, on minimizing the energy functional whereas at high temperature, the conditional distribution tends to be uniform. The extreme cases are the gradient based method and random selection method. The gradient of the information measure versus the state provides a guide for reducing the cost functional at each iteration on the perturbed optimization parameters. This technique has been used in vector quantization to generate the optimal codebook by Zeger and Gersho [49]. Another example is an algorithm for computing the maximum of *a posteriori* estimate of the original image from the degraded image based on the Gibbs distribution and stochastic relaxation with convergence in probability to the global optimal by Geman and Geman in [20].

5.7.2 Simulated Annealing

Simulated annealing is a stochastic relaxation technique in which a randomly generated perturbation is added to the set of optimization parameters θ at each iteration. The name annealing came from physics. The key is the analogy between the state of physical systems related to temperature and the random state under Gibbs distribution. Certain materials can be driven to low energy state by reducing their temperature gradually at appropriate rate to produce desired properties. Simulated annealing has an equivalent state most probable under Gibbs distribution with low “temperature.” If a sufficient number of iterations is made at a given temperature, simulated annealing yields a numerical solution close to the *thermal equilibrium* at that temperature at which the probability distribution of that state is stationary. If the temperature is further reduced and

the iteration repeated, the resulting state approaches a new equilibrium at this reduced temperature. The equilibrium reached at very low temperature is the state with minimum energy. We can seek the global optimal wavelet solution via simulated annealing.

Let θ be the parameter vector as appeared in the previous chapter, we may also call it state of the simulated annealing, Θ denote the corresponding state space, $[0, 2\pi]^{K+1}$. Let $\hat{\theta}$ be the perturbed state of θ and the corresponding perturbed space is denoted by $\hat{\Theta}$.

Definition 5.7.1 (Geman and Geman [20]) *A Gibbs distribution relative to $\{\Theta, \hat{\Theta}\}$ is a probability measure π on θ with*

$$\pi(\theta) = \frac{1}{Z} e^{-\lambda(\Psi, t)/T} = \frac{1}{Z} e^{-\lambda(\theta)/T}, \quad (5.33)$$

where T is a constant, which is the temperature, Z is the scaling factor $Z = \sum_{\theta} e^{-\lambda(\theta)/T}$, λ is the additive information measure in (5.9).

The functional $\lambda(\cdot, \cdot)$ is an implicit function of the state θ and plays the role of energy functional of the Gibbs distribution. The “temperature” T controls the peak of the density function. The value of T reduces gradually during the simulated annealing process in order to produce the low energy state, i.e., the optimal parameter vector θ , for generating the optimal mother wavelet and the optimal wavelet basis.

The problem of finding the optimal wavelet basis becomes minimizing λ over Θ through simulated annealing. Let K be the order of the low pass filter $H_0(z)$, i.e., the degree of the paraunitary matrix composed of the low pass and high pass filters. The unconstrained parameter set $\theta = [\theta_0 \theta_1 \cdots \theta_K]$ takes value at time t , with $t = 0, 1, 2, \dots$. At time t , the k th component of θ is denoted by $\theta_k(t)$, we

have $\theta(t) = [\theta_0(t)\theta_1(t)\cdots\theta_K(t)]$ with $\theta_k(t)$ taking value from the interval $[0, 2\pi]$. We want to find the global with an arbitrary starting point. At each step, only one component of θ is updated. Assume n_0, n_1, n_2, \dots be the sequence in which the components of θ is updated, thus, at any instance t , $\theta_s(t) = \theta_s(t-1)$ for all $s \neq n_t$. At each t , with $\theta(t)$ given we generate y randomly from the interval $[0, 2\pi]$. We assign $\theta_{n_t}(t) = y$. This is the local updating scheme for vector θ . We shall demonstrate that as t turns to infinity $\theta(t)$ is irrelevant to $\theta(0)$.

The statistical properties of the random process $\{\theta(t)\}$ can be expressed by the state transition from $\theta(t-1)$ to $\theta(t)$ by the following

$$P(\theta_s(t) = y_s, s \in S) = \pi(\theta_{n_t} = y_{n_t} | \theta_s = y_s, s \neq n_t) \cdot P(\theta_s(t-1) = y_s, s \neq n_t), \quad (5.34)$$

where π is the Gibbs distribution (5.33) which drives the stochastic process for the optimization scheme. The temperature T is a function of time t in the distribution, T decreases as time t increases. If we denote the Gibbs distribution π at time t by $\pi_{T(t)}$, the random process above can be expressed as

$$P(\theta_s(t) = y_s, s \in S) = \pi_{T(t)}(\theta_{n_t} = y_{n_t} | \theta_s = y_s, s \neq n_t) \cdot P(\theta_s(t-1) = y_s, s \neq n_t). \quad (5.35)$$

Let $\Delta = \max_{\theta} \lambda(\theta) - \min_{\theta} \lambda(\theta)$, denote $\Theta_0 = \{\theta \in \Theta, \lambda(\theta) = \min_{\eta} \lambda(\eta)\}$, we then have the following theorem due to [20, Geman and Geman] to prove the convergence of the simulated annealing.

Theorem 5.7.1 *Assume there is an integer $M \geq K + 1$, then for arbitrary $t = 0, 1, \dots$, we have $S \in \{n_{t+1}, n_{t+2}, \dots, n_{t+M}\}$, let $T(t)$ be a decreasing sequence and assume, (a) $\lim_{t \rightarrow \infty} T(t) = 0$, (b) $T(t) \geq (K + 1)\Delta / \log t$, for all $t \geq t_0 \geq 2$,*

then, for every $y_0 \in \Theta$ and for every $y \in \Theta$,

$$\lim_{t \rightarrow \infty} P(\theta(t) = y | \theta(0) = y_0) = \pi_0(\theta). \quad (5.36)$$

This theorem demonstrates two facts. First, the distribution of state $\theta(t)$ converges to π regardless of the selection of initial conditions. Second, the distribution of the state converges to the distribution which minimizes the energy of the system globally as the temperature decreases to zero. Gradually reducing temperature T yields the most probable state under the Gibbs distribution. This state is the low energy state of the system.

The complexity of simulated annealing is determined by the value of K which is the degree of the paraunitary matrix (4.27). The value $K + 1$ is the dimension of the search space for simulated annealing. In finding the optimal wavelet, the state to be randomized is the compact parameter set $[0, 2\pi]^{K+1}$. We observe from the above theorem that the approximate computation time

$$t \geq e^{(K+1)\Delta/T} \quad (5.37)$$

which is the number of total updating of the parameter set. For practical implementation, the product of $(K + 1)\Delta$ may cause a long time iteration process. It is necessary to control the size of the problem and to select the appropriate temperature T . Further research is needed regarding this aspect.

5.8 Conclusion

This chapter has provided a direct approach to construct an optimal orthonormal wavelet basis with compact support for signal representation. First, a cost functional, an additive information measure, is introduced based on the decomposition entropy of a given signal with respect to an initial wavelet basis. This

entropy measures the nonuniform energy concentration of the given finite energy signal. The sensitivity of each dilation and shift of the mother wavelet function $\psi(t)$ with respect to the governing coefficients has been found. This relation establishes the gradient of the information measure versus the parameter set. We have proved the existence of an optimal wavelet basis for signal representation and for system identification.

Simulated annealing is proposed to avoid the influence of the initial condition on the optimization process. Theoretically, simulated annealing will find the global optimum. However, the computation may be very time consuming. As a consequence, one may try to proceed with a combined method of both gradient based and stochastically relaxed approaches.

Our method can find applications in the other fields in addition to signal representation and system identification. In the context of pattern recognition, the methodology described in this chapter is also a way to construct the feature space and to partition the signal space according to its representatives. The parameterization of cost functionals in this chapter is not unique, other forms of measures or cost functions may be introduced according to the contexts of actual physical problems.

Finally, it is worth pointing out that although the optimal wavelet basis described in this chapter is one dimensional, it can be extended to two dimensional nonindependent wavelets due to our general formulation. This demonstrates the potential applications of this method of wavelet basis construction in image processing for obtaining high compression ratios.

Chapter 6

System Identification via Adaptive Wavelet Neural Networks

In this chapter, we discuss identification of infinite dimensional systems via adaptive wavelet neural networks (AWNNs). We first address the background for system identification via neural networks. We then formulate the problems of system identification and self monitoring of distributed systems and illustrate the structure of adaptive wavelet neural networks. Finally, we propose an AWNN training algorithm.

6.1 Introduction

There are two well known types of system identification schemes, parametric and nonparametric. The former depends on the given model structure used in identification and determines the model parameters based on input and output of the unknown systems. The second scheme does not require the information regarding the model structure and gives an estimate of the impulse response of the unknown systems. However, some cases are not suitable to be treated with these conventional approaches due to insufficient analytical knowledge of

the plant, incomplete information on the number of key parameters and the presence of disturbance and uncertainties. Even when enough knowledge about the system is available, the model of the system may be too complicated to be used to design control systems. For this reason, we introduce an adaptive learning scheme to emulate the real system.

6.1.1 Background in System Identification

We are interested in introducing another form of identification scheme which employs a parallel computational structure and uses knowledge from measurement to adapt to different models and structures. This method can be used for both linear and nonlinear system identification. The underlying idea is twofold: first, identify the type or class of the system and pick a simple component or a structure which describes the characteristics of the system; second, start from the simple structure to build a basis to generate or approximate the given system successively in an appropriate functional space.

We consider identification as constructing a suitable subspace of $L^2(R)$ and generating a function to approximate the output of the system with respect to the input since the input responses of a large class flexible structure systems and distributed systems belong to $L^2(R)$. The identification of an input output relation can thus be formulated as the approximation of a function in $L^2(R)$ by its projection on an appropriate subspace of $L^2(R)$. If we can construct a suitable subspace of $L^2(R)$ in an appropriate scale spanned by dilating and shifting a base wavelet function, we should be able to approximate a function in $L^2(R)$ with a function in the subspace of the relevant resolution in the sense of minimizing a norm of the difference between the two functions. Naturally, the best approximation is predetermined by the subspace in consideration and

thus by the base wavelet which determines the dynamical characteristics of the subspace used for approximation. If partial information of the system is available *a priori*, or the class of the function to be approximated is detected, as shown in the previous chapter, an appropriate wavelet basis could be built and the multiresolution property can be used to approximate the function progressively.

6.1.2 Computational Structures of Neural Networks and AWNN

A general representation of a neural network is a computational structure of finite linear combinations of the form

$$g(\mathbf{x}) = \sum_{j=1}^N w_j \sigma(\mathbf{a}_j^T \mathbf{x} + b_j), \quad (6.1)$$

where $\mathbf{x}, \mathbf{a}_j \in R^N$, $b_j \in R$ are fixed. The network is formed from weighted compositions and superpositions of a single, simple nonlinear pattern or response function. The activation function σ is generally nonlinear and depends heavily on the context of the application. The real constant b_j is the bias of each neuron.

Neural networks have found their applications in controls and system identification. A neural network was used as an emulator and controller to control a highly nonlinear truck-trailer docking problem in [39]. Applications of neural networks have been studied and summarized in [23] regarding modeling, identification and control structures. The nonlinear functional mapping properties of neural networks are central to their applications in system identification and controls. It has been proven by Cybenko [17] that a two-layer neural network can approximate a nonlinear function to an arbitrary degree of accuracy. However, the number of neurons required in the neural network may far exceed the limit for practical implementations. This poses a burden for the applications in

on line system identification and in real time system controls. As we see from the structure of artificial neural networks above, no dynamical components are included in the general setting of neural networks and the network is memoryless.

Incorporating proper dynamical components into the network can reduce the number of neurons required for certain performance. An issue in control is the dynamical nature of the system to be considered. It is reasonable to speculate that when proper dynamics are included in the neural networks, the performance of the networks is expected to improve. We anticipate that the information from the wavelet basis will help reducing the number of neurons needed to achieve the same performance provided that the wavelets contain useful information about the system in consideration.

Our thoughts on a unified work on wavelets and neural networks are further encouraged by the work on wavelets and neural networks by Zhang and Benveniste [50]. In their work, a notion of wavelet networks is proposed as an alternative to feedforward neural networks for approximating arbitrary nonlinear functions. We are interested in using both the multiresolution property from wavelet decomposition and the convenience of computational structures of neural networks to approximate the unknown plants; we first introduced the concept and structure of adaptive wavelet neural networks (AWNN) in which a signal based optimal wavelet basis is incorporated and applied AWNN for identification of distributed systems in [52]. The dynamics of the AWNN adapts to the relevant systems and hence yields an AWNN smaller than that of static neural networks.

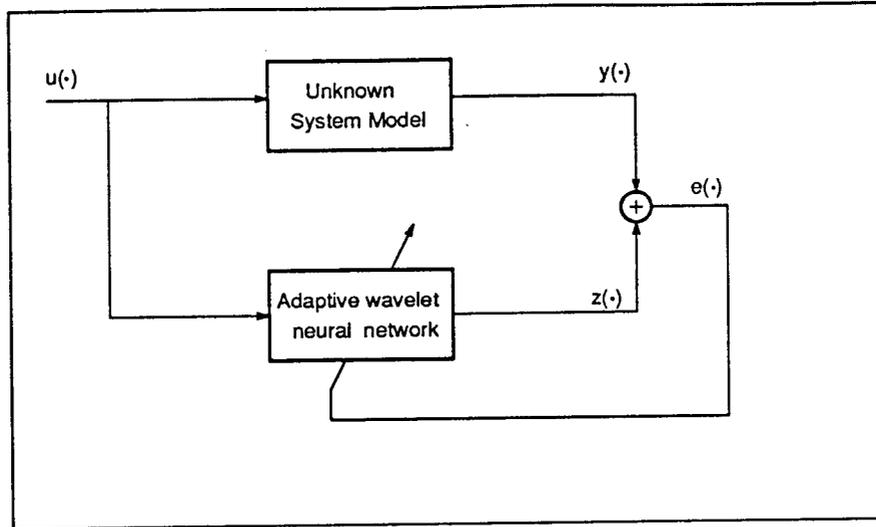


Figure 6.1: A wavelet neural network identification structure.

6.2 Problem Statement

6.2.1 Structure of AWNN

Given an infinite dimensional stable system with unknown input output relationships, we set up an identification structure shown in Figure 6.1, in which $u(\cdot)$ and $y(\cdot)$ are the input and output to and from the unknown system. An adaptive wavelet neural network block (AWNN) is used to approximate the given system with $z(\cdot)$ as its output. The matching error $e(\cdot)$ is defined as the difference between $y(\cdot)$ and $z(\cdot)$. The network is tuned to match the system through minimizing the error $e(\cdot)$.

The structure of an adaptive neural wavelet network is shown in Figure 6.2, in which $u(\cdot)$ is the input to both the system and the network, $z(\cdot)$ is the corresponding output. This network contains a hidden layer of an appropriate wavelet basis $\{\psi_{j,l}\}$ from dilating and shifting a base wavelet $\psi(\cdot)$ which is to be determined via an optimal adaptive scheme of basis selection. The activation function $\sigma(\cdot)$ is a nonlinear function. One of the possible forms is a sigmoidal

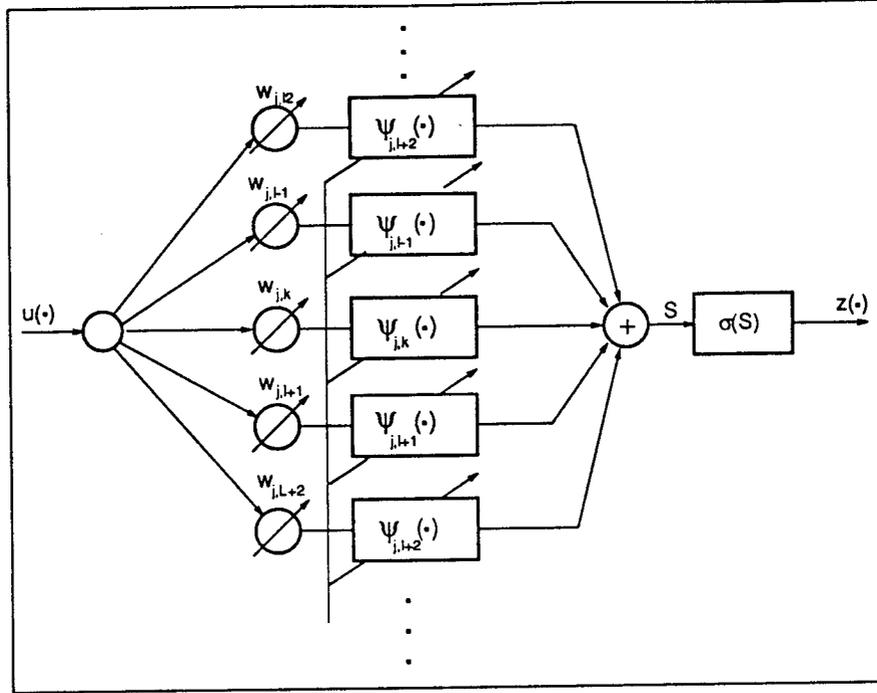


Figure 6.2: An ANN block.

function

$$\sigma(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}. \quad (6.2)$$

Although this is a general setting, the dynamics of the activation function can be selected either as a linear or a nonlinear function according to the dynamics of the wavelet blocks. The output of the network is given by

$$z(\cdot) = \sigma\left(\sum_{j,l} w_{j,l} \psi_{j,l}(\cdot)\right) u(\cdot), \quad (6.3)$$

where

$$\hat{G}(\cdot) = \sigma\left(\sum_{j,l} w_{j,l} \psi_{j,l}(\cdot)\right) \quad (6.4)$$

is the estimated input-output relation of the unknown system. The function $\hat{G}(\cdot)$ approximates the input-output relation to a certain level of resolution which depends on the resolution of the subspaces spanned by the wavelet functions.

6.2.2 Optimal Wavelet Basis and AWNN

The mother wavelet function $\psi(x)$ determines the dynamical nature of the adaptive wavelet neural networks. We use x to denote the variable in the domain of consideration. We are interested in using a mother wavelet function which describes the dynamical behavior of the systems in consideration most closely. When incorporated into the network, the wavelet network should have the best performance for a certain complexity or to provide a certain performance level with a minimum complexity. This is true when $\psi(x)$ is chosen to contain information regarding the class of the given system. We shall use this optimal wavelet function $\psi(x)$ in our wavelet neural networks for system approximation. In chapter 5, we discussed in detail the formulation and solution of the optimal wavelet basis selection. We shall use the optimal wavelet basis described as the basic component in the AWNN.

We define the random error at instant k by the random sample (u_k, y_k) as the difference between y_k and z_k , with the system output y_k as the desired output for the neural network. The error at the k^{th} instant is defined by

$$e_k = y_k - z_k. \quad (6.5)$$

The square of error at step k is

$$E_k = \frac{1}{2}[y_k - z_k]^2. \quad (6.6)$$

The accumulated error E ,

$$E = \sum_k E_k \quad (6.7)$$

sums the errors of the first k iterative steps. The network with a minimal matching error E is required to approximate the unknown system. The identifi-

cation problem transforms into trajectory learning in the corresponding discrete domain.

Our problem becomes two-fold: selecting the best wavelet basis for a wavelet neural network; training the AWNN afterwards to match the unknown plant. First, we need to find the optimal base wavelet function $\psi^*(x)$ such that the positive cost measure \mathcal{M} is minimized for the detected dynamical behavior of a given system, i.e.,

$$\psi^*(x) = \arg \min \mathcal{M}_\psi(\psi, f(x)). \quad (6.8)$$

Secondly, we need to train the network to emulate the given system in the sense of finding the optimal weights $\{w_{j,l}\}$ to minimize the cost index J which is

$$J_{opt} = \min_w E[w]. \quad (6.9)$$

The input-output relation of the trained neural wavelet network is used to represent the transfer function of the given system to facilitate the design of control systems. This forms a self-tuning system identification scheme via an AWNN.

6.2.3 Dynamics of Neurons and AWNN

An artificial neural network is an interconnection of computational components, called neurons or nodes. It provides a distributed computational structure and can be processed in parallel. The function of each neuron may not be the same unlike the elegant mathematical expressions in general. We can imagine artificial neural networks as maps from stimulus to responses subject to either reward or penalty depending on the setting of real physical systems.

The neural network is shown as layers of recognition and adaptive response

as described in Figure 6.3 [2, p30]. The first layer takes the complex stimulus

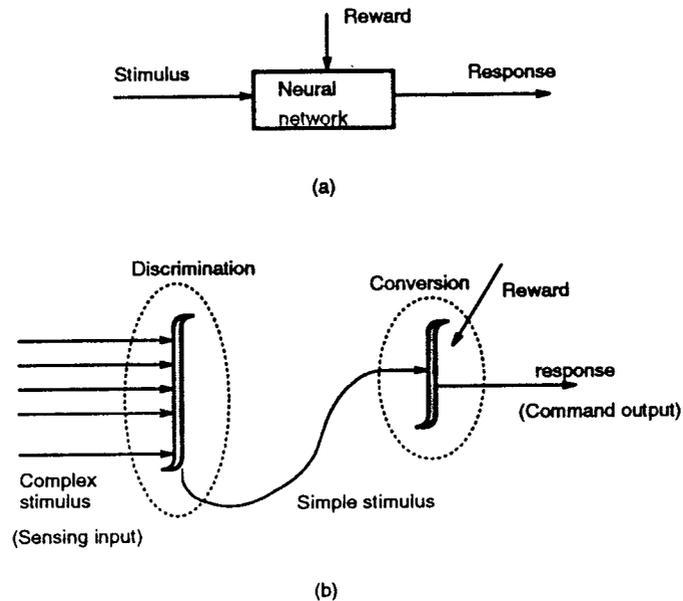


Figure 6.3: Separation of a neural network as discrimination and conversion parts. (a) A neural network block. (b) The structure of discrimination and conversion.

from the physical systems, i.e., sensory data for the artificial neural network, it then extract the feature of the complex input and produces a representative, a simple stimulus, to excite the next layer. The second layer converts the stimulus into either a command or a response according to adaptive learning schemes. This is one of the local computational models of neural networks.

A low level description of the process above is illustrated in Figure 6.4 [2, p31] as three separated layers. The first two layers act as a feature extractor and a classifier respectively. The first layer extracts features form the sensory data. This process is highly parallel and can be implemented with analog parallel circuits. The second layer takes the outputs from the preprocessed signals and represents them in a simplified way at a higher level. Those patterns in the second layer can be classified based on theory form vector quantization.

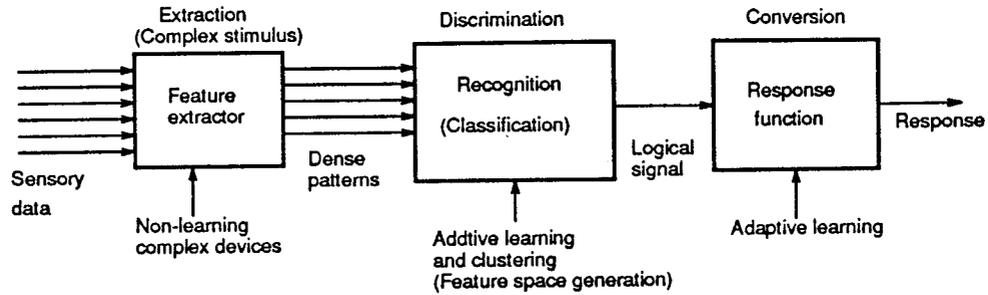


Figure 6.4: Feature extraction and classification.

The system identification scheme can be realized through using a two-layer AWNN. We may call it an emulator. A two layer AWNN emulator is shown in Figure 6.5. The first layer consists the AWNN blocks illustrated in Figure 6.2; the second layer takes the weighted sun of the outputs from the first layer and sends it to the linear neuron in the layer. This structure can approximate the input-output relation of an unknown system and is similar to those represented by Equation (6.1).

The AWNN system has its special adaptability to outside changes while maintaining the basic functions of general artificial neural networks. The first layer of the AWNN emulator consists of wavelet blocks generated from the optimal mother wavelet described in Chapter 5. The process of the basis selection plays the role of feature extraction in the signal space. As we discussed earlier, the resulting optimal mother wavelet serves as the representative of a class of signals from the physical system. The second layer of the AWNN emulator produces responses to the stimuli preprocessed by the wavelet blocks of the first layer. The adaptive nature of the AWNN is realized though the signal dependent basis and network training.

Definition 6.2.1 (Cybenko, [17]) *We say σ is discriminatory if for a mea-*

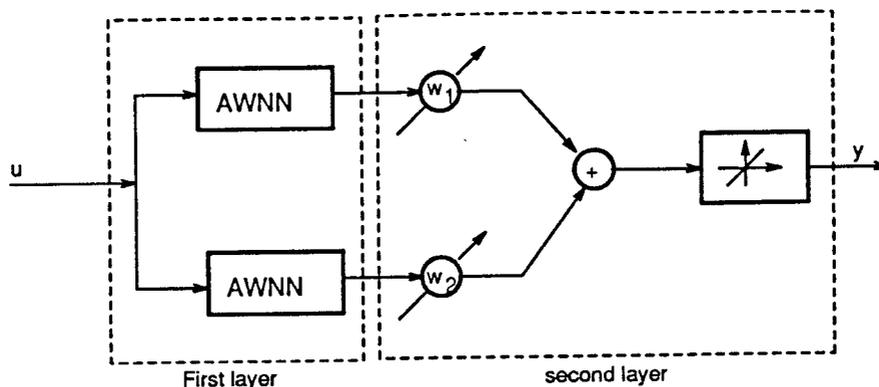


Figure 6.5: A two layer AWNN emulator.

sure $\mu \in M(I_n)$

$$\int_{I_n} \sigma(y^T x + \theta) d\mu(x) = 0$$

for all $y \in R^n$ and $\theta \in R$ implies $\mu = 0$.

Theorem 6.2.1 (Cybenko, [17]) *Let σ be any discriminatory function, then the finite sums of the form*

$$G(x) = \sum_{j=1}^N \alpha_j \sigma(y_j^T x + \theta_j)$$

are dense in $C(I_n)$. In other words, given any $f \in C(I_n)$, for arbitrary $\epsilon > 0$, there is a partial sum, $G(x)$, of the above form, for which,

$$|G(x) - f(x)| < \epsilon, \quad \forall x \in I_n.$$

This important theorem tells us that a two layer neural network of (6.1) can approximate any nonlinear function. This theorem foresees the basis for system identification and function approximation via neural networks.

6.3 Network Training

This section describes a supervised learning process of the AWNN as stochastic approximation of an unknown function. The training of an AWNN consists of

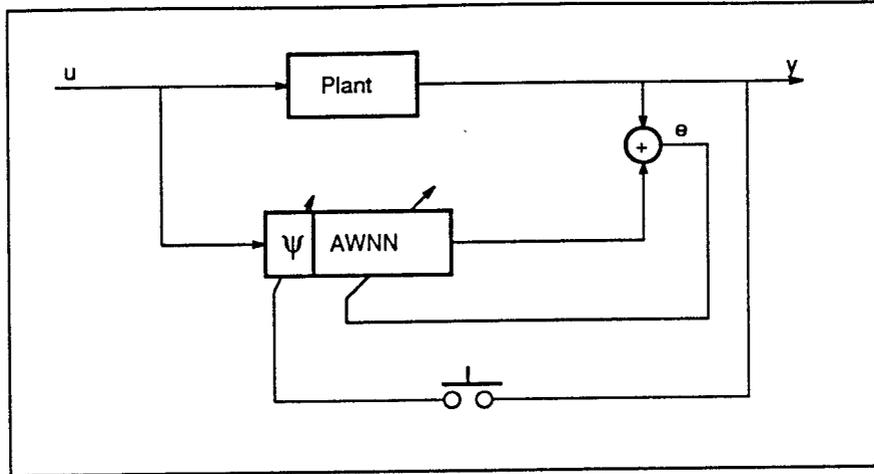


Figure 6.6: AWNN training structure.

two stages: a pre-training procedure and an actual training scheme via weight updating. The pre-training is a preparation process of configuration or adjusting the basis of the network based upon the output measurements from the unknown system excited by a test signal. The purpose is to equip the network with the appropriate dynamics and generate an AWNN of a manageable size. The network is trained afterwards with a supervised learning process. The training structure is shown in Figure 6.6.

During the first stage, the network takes the output of the unknown system excited by a test signal and looks for the best wavelet basis with the switch at the closed position. The algorithm given in Chapter 5 is used to generate the best wavelet basis Ψ for the AWNN. The dynamical behavior of the AWNN is thus determined by this process. This stage also provides appropriate initial weights for the network training to start with. Since the basis contains the measured information of the unknown system, the required size of the network is reduced compared with a neural network without the dynamical components. This will speed up the network training process.

The next stage is the network training which is a goal-directed learning aimed at minimizing the relevant cost functional. It is supervised learning since certain pattern of $\psi(x)$ related to the unknown system is used during training. Different training algorithms were discussed in [29], [44] and [10]. Due to the convenience of our problem formulation, we use the backpropagation algorithm in [29] to train the AWNN. The backpropagation algorithm, an extension of LMS algorithm, modifies the weights at each step with nonlocal error information. This is an implied feedback which closes the loop for adapting weights of the AWNN. The backpropagation provides a suboptimal solution in the sense of using finitely many wavelet blocks to approximate the infinite dimensional system. For convenience, we rewrite Equation (6.9) as

$$J_{opt} = \min_w E[w]. \quad (6.10)$$

The task here is to minimize the cost functional J of Equation (6.10).

From the structure of the AWNN, we have

$$S = \sum_{j,l} w_{j,l} \psi_{j,l}(s) u(s) \quad (6.11)$$

as the input to the sigmoidal function $\sigma(\cdot)$. We update the weight $w_{j,l}(k)$ at k^{th} iteration by a stochastic difference equation

$$w_{j,l}(k+1) = w_{j,l}(k) + q_k \Delta w_{j,l}(k) \quad (6.12)$$

where

$$\Delta w_{j,l}(k) = -\frac{\partial E_k}{\partial w_{j,l}}. \quad (6.13)$$

with the learning coefficients q_k 's satisfying [29, p186],

$$\sum_k q_k = \infty \quad (6.14)$$

$$\sum_k q_k^2 < \infty. \quad (6.15)$$

The condition (6.14) constrains the sequence $\{q_k\}$ to decrease slowly, while (6.15) constrains to decrease q_k quickly. The combined effect is to guarantee the appropriate learning rate.

The gradient of the cost functional with respect to the weight $w_{j,l}$ is expressed as

$$\frac{\partial J}{\partial w_{j,l}} = \sum_k \frac{\partial E_k}{\partial w_{j,l}}. \quad (6.16)$$

We refer to the definition of the square of error at step k in Equation (6.6) and use the subscript k of a variable to denote the value of the variable at the instant k . By the chain rule, we have

$$\begin{aligned} \frac{\partial E_k}{\partial w_{j,l}} &= -(y_k - z_k) \frac{\partial z_k}{\partial w_{j,l}} \\ &= -(y_k - z_k) \frac{\partial z_k}{\partial S_k} \frac{\partial S_k}{\partial w_{j,l}} \\ &= -(y_k - z_k) \sigma'(S_k) \psi_{j,l} u_k. \end{aligned} \quad (6.17)$$

Hence

$$\begin{aligned} \Delta w_{j,l}(k) &= (y_k - \sigma(S_k)) \sigma'(S_k) \psi_{j,l} u_k \\ &= (y_k - \sigma(\sum_{j,l} w_{j,l} \psi_{j,l} u_k)) \sigma'(\sum_{j,l} w_{j,l} \psi_{j,l} u_k) \psi_{j,l} u_k \end{aligned} \quad (6.18)$$

as the weights updating scheme. The general backpropagation algorithms can be found in [29]. This process starts by assigning $y_{j,l}$, the coefficients from the basis operation Ψy of the measured output $y(s)$ to the wavelet basis of the AWNN, to $w_{j,l}(0)$. The trained neural wavelet network shall be used to implement control system design. The reconstruction from the given wavelet basis is the approximation of the plant up to a certain resolution. Summarizing

the above yields the following algorithm.

Algorithm 6.3.1 *AWNN training scheme*

Step 1: Set $i := 1$,

$$J_0 := 0,$$

Basis selection, input Ψ ,

Set $w_{j,l}(0) := y_{j,l}$.

Step 2: $w_{j,l}(i) := w_{j,l}(i - 1) + q_{i-1} \Delta w_{j,l}(i - 1)$.

Step 3: Compute J_i .

Step 4: If $|J_i - J_{i-1}| > \epsilon$,

$i := i + 1$, go to Step 2.

Step 7: Stop.

The first step of the AWNN training scheme begins with the optimal base wavelet selection which has been discussed in detail in Chapter 5. This procedure equips the AWNN with the appropriate dynamical component for system identification owing to the fact that the decomposition entropy of the output of the system is minimized. The optimal feature space in which the unknown system is to be approximated is optimized in such a way that the energy distribution of the system or the signal concentrates in it. The complexity is expected to be reduced due to the dynamical wavelet blocks embedded. This is one of the key ideas of AWNNs. The second step updates the weights using the back-propagation algorithm. The next step check the ending condition for stopping the iteration process.

The gradient based training algorithm converges to a local minimum. As discussed in Chapter 5, stochastic relaxation may be used to reduce the chance

of being trapped by a local minimum, however, this will increase the cost for computation and put burden for real time implementation. One usually starts the procedure from a randomized initial point and searches for the minimum and hopes that the local minimum is good enough for practical purposes. Due to the combined structure of the wavelet system and the neural network, one can start from the wavelet coefficients resulting from the basis search algorithm. This paradigm is reasonable since the wavelet decomposition should be close to the optimal solution before truncating the full representation into finitely many terms. The QMF bank in Chapter 4 can implement this system and produce the discrete wavelet coefficients. We shall accept the solution starting from the wavelet decomposition. At least, it is one of the suboptimal solution having a better chance of being closed to the optimal solution.

Implementation of AWNNs is another way of curve fitting to available data. They have both advantages and disadvantages. They are conceptually simple and easy to use and are adaptable to complicated problems or suitable to deal with problems which do not have a modeled structure or are too complicated to model. Another advantage is that neural networks offer a distributed, parallel processing ability thus provide integrity and possible fault tolerance. The function of each neuron is usually a simple function which is easy to implement. The most obvious disadvantage is that neural networks do not recognize and preserve the structures of the systems they deal with and there is no systematic way to determine the structures of the networks either. Embedding system dependent dynamical components into the networks will be useful in overcoming these disadvantages. Our attempt in designing an AWNN will be of research potential in this regard.

6.4 Structured Learning via Vector Quantization and Feature Extraction

We have discussed the training of AWNN in the previous section in which both the training structure and algorithm were given. We shall propose a general learning structure to conclude this chapter.

6.4.1 Optimal Wavelet Basis versus Vector Quantization

Adaptive wavelet neural networks (AWNNs) are dynamical systems whose components are adjusted according to the real system. Consequently, the AWNNs contain the information of the systems to be approximated and the signal to be represented. The training procedure is based on the adaptation of the optimal wavelet basis. In the first step of the training, given a test signal, the optimal algorithm in Chapter 5 produces the corresponding optimal wavelet in the sense of minimizing the decomposition entropy of the system. The basis selection algorithm can be performed off line due to the complexity of the computation. We may use techniques in vector quantization to systematically complete both the off line optimization and the on line search for the optimal basis.

Given an input sequence, we can have a corresponding output sequence and produce the optimal basis using the given algorithm. These two sequences can be formed into a vector, called signal vector, then the process becomes establishing mapping from a vector to a functional. If we collect all the wavelet bases corresponding to the representatives of the elements of the vector space called signal space which is composed of all the input-output vectors, we can use vector quantization schemes [21] to pair a signal vector with the corresponding wavelet

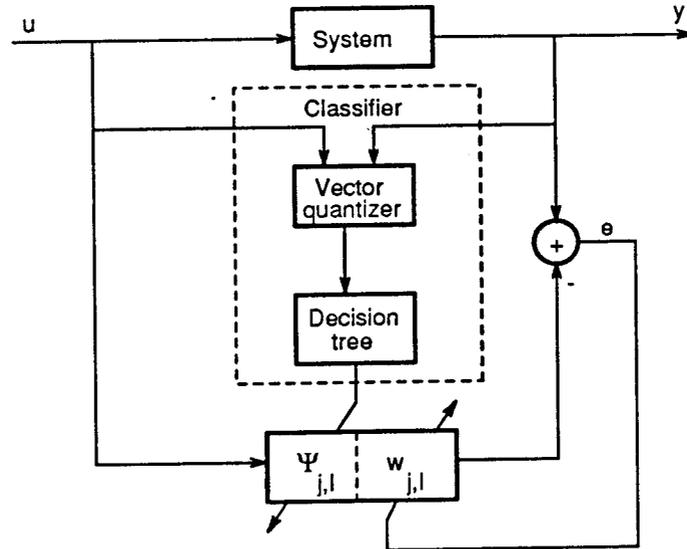


Figure 6.7: A tree structured learning scheme.

basis and partition the signal space into disjoint subspaces whose indices correspond to those of the wavelet bases. The generalized Lloyd algorithm employing the nearest neighbor condition and the centroid condition [21, p195] is suitable for generating the optimal partition and the corresponding set of the optimal wavelet bases, i.e., the codebook here. A vector quantizer and a decision tree are employed to choose the feature space partition corresponding to classes of signals and search for the index of the optimal wavelet bases to be used. In the context of pattern recognition, the collection of the optimal wavelet bases with compact support form a feature space. Detailed explanation and analysis can be found in [54] on multiresolution feature extraction.

6.4.2 A Classifier Based Learning Structure

A classifier can be used to generate a pointer to specify the wavelet basis in use for a given input and output pair of the system. Figure 6.7 demonstrates the principle of the structured learning scheme via classification of the signal space through employing a vector quantizer and a decision tree. Given a input and the

corresponding output signals in the form of a signal vector with the appropriate length, we pass the vector through the vector quantizer and the decision tree, i.e., the classifier of the system. The classifier maps the signal vector to the pointer of the corresponding optimal wavelet basis, and the adaptive wavelet neural network thus input this optimal wavelet with respect to this input-output pair into the system. The system identification and signal representation is performed through the weight updating algorithm discussed earlier.

The vector quantizer and the decision tree based AWNN provides flexibility in progressive learning, modeling and signal representation. The VQ based AWNN described above offers both structural and parameter adaptations to the encountered situations. It is of research potential in this aspect.

6.5 Conclusions

We have developed algorithms for identification of infinite dimensional systems via an adaptive wavelet neural network. We first work on the problem of selecting the compactly supported optimal wavelet basis for spanning the subspaces in which the unknown system is approximated up to a predetermined resolution. An algorithm is given for constructing the optimal basis Ψ for the network emulator based on the measurements of the output from the unknown system. We then apply a backpropagation algorithm to train the resulting AWNN for system approximation. This is an efficient way of approximating an infinite dimensional system up to a certain resolution in a subspace of $L^2(R)$ spanned by the dilations and shifts of the optimal base wavelet and the size of the AWNN can be reduced compared with the neural networks without the dynamical components.

We proposed an adaptive scheme for the AWNN which adjusts its structure

through the optimal basis selection and thus construct the feature space for approximating the transfer function of the system. The approximation algorithm is based on the selection of an optimal basis. Our method combines the advantage of multiresolution property of wavelet decompositions and the convenience of the computational structures of neural networks. The marriage of the best from both fields should provide a powerful tool kit for solving problems of a much wider range. Our approach can be generalized to the N dimensional case with signals from $L^2(\mathbb{R}^N)$. We also introduced the principle of a structured learning scheme via vector quantization and illustrated the classifier based adaptation structure for system identification and signal representation. This structured learning scheme offers additional flexibility for adaptive systems subject to the environmental changes.

The methodologies developed in this chapter are expected to be useful not only for system identification and progressive modeling but also for signal classification, compression and reconstruction. Future research is needed on these aspects.

Chapter 7

Analog VLSI Architecture Design for AWNN

In this chapter, we address the implementation of AWNN via analog VLSI circuits. We first analyze the design modules of the major components of the artificial neural networks. We then provide the architectural design for real time implementation.

7.1 Introduction

Neural networks and the AWNN provide parallel computational structures for real time system identification and signal representation. More important, they offer a parallel processing power and means of possible real time implementation for system identification, systems monitoring and equally important, sensory data processing of distributed systems and complex systems. In a more general sense, neural networks and AWNN perform data exchange, signal processing and data association with their nodes being processors for operation at different levels of a distributed system, and the functions of different nodes can be custom designed to meet specific requirements based upon the context of the physical problems. The significance and advantage of using neural networks

lie in the potential power in dealing with nonlinear systems and in real time implementation.

Many neural networks are implemented through digital systems and software and the computation is performed off line. In such systems, weights and parameters are usually stored and recalled in local memories of the network nodes while the activation function is computed through a look up table of each node using limited amount of hardware. However, the multiplication of between inputs and weights can be the bottleneck of the problem. In addition, fetching variable values takes extra time during computation. Computation speed is a critical issue in digital neural network implementation. Effort has been made to speed up the operation through employing a digital neural network without multipliers [37] with parameters being restricted to power-of-two.

Analog circuits have several features in the implementation of neural networks. Principles, background and basis circuit components in analog VLSI and neural systems can be found in the book by Mead [38]. For realizing certain nonlinear functions such as those of sigmoidal type, analog circuits can take advantage of the physics property such as the Boltzmann distribution [38, p25] to yield a direct and simple system realization which could be time and area consuming if implemented digitally. Although the precision of analog circuits may not exceed the accuracy of 7-8 bits in the next few years, they can still satisfy some important tasks such as navigation and tracking which have incoming data precision of usually not more than one percent [28].

We are interested in designing modularly structured analog VLSI implementations of ANNs in which relatively simple structures aggregate to generate the whole system. We shall first illustrate and analyze some basic modules of

nonlinear functions, vector multipliers and weight adaptation blocks. We then produce a frame work for the analog AWNNs.

7.2 Design Modules for Analog VLSI Implementation

In this section, we illustrate several existing design modules for the implementation of analog AWNN. We then address the refreshing scheme and circuit structures for the weight adaptation.

7.2.1 An Analog Neuron Structure

We use the analog CMOS circuitry shown in Figure 7.1 [31] as a basic component of neurons for the AWNN. The circuit consists of three major parts: an input layer, a nonlinear function part and an output stage.

The input of the neuron block is a current denoted by $i_{j,l}$. The output of the circuit is a voltage denoted by v_o . This structure allows the input to be the sum of several current signals. The input current is converted into a voltage V by an operational amplifier with controlled feedback by differential resistance R_g . This differential resistance is a gain factor determined by

$$v = R_g i_{j,l}, \quad (7.1)$$

$$R_g = \frac{1}{K_N \frac{W_g}{L_g} (V_{g1} - V_{g2})} \quad (7.2)$$

where K_N , W_g and L_g denote the transconductance parameter, the channel width and the channel length of the relevant four transistors in the feedback loop, respectively.

The sigmoidal function is implemented by the hyperbolic tangent function.

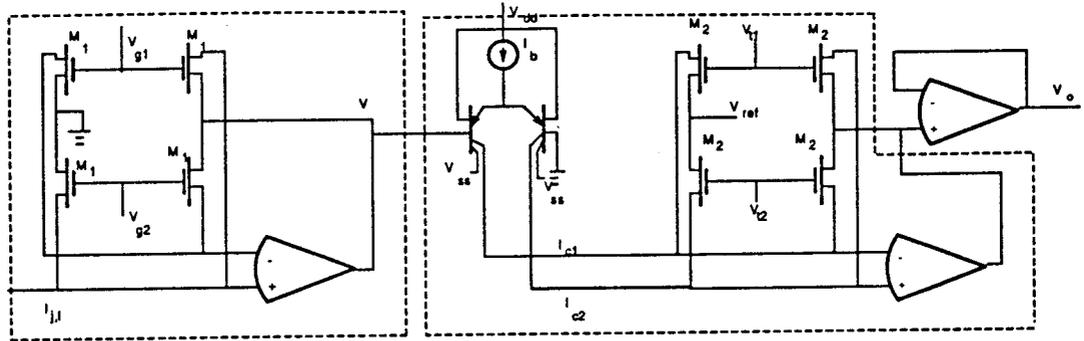


Figure 7.1: A neuron implementation structure with input stage, transfer stage and output stage.

This function is realized through the differential pair of transistors with the input and output relation

$$v_o = V_{ref} + R_t I_b \alpha \tanh(V/(2V_t)) \quad (7.3)$$

where V_t is the thermal voltage and $\alpha = -i_c/I_E$, I_b is the bias of the differential pair of transistors. The output of the differential pair goes through the second operational amplifier with the same feedback loop as that in the input stage. The transfer function of the neuron is given by [31]

$$V_o = V_{ref} + R_t I_b \alpha \tanh(R_g I_{j,l}/(2V_t)) \quad (7.4)$$

where R_t is the controlled differential resistance which is used to control the gain of the system. Voltage V_{ref} serves as the bias of the output stage.

7.2.2 Vector Multiplier for Weight Adaptation

The next major component is the one responsible for synapse updating or weight updating. We use the circuit illustrated in Figure 7.2 [31] for this purpose.

The basic computational structure in artificial neural networks is the sum of products. This operation is equivalent to the inner product of two vectors.

Figure 7.2 [31] illustrates the structure of a vector multiplier. The input and

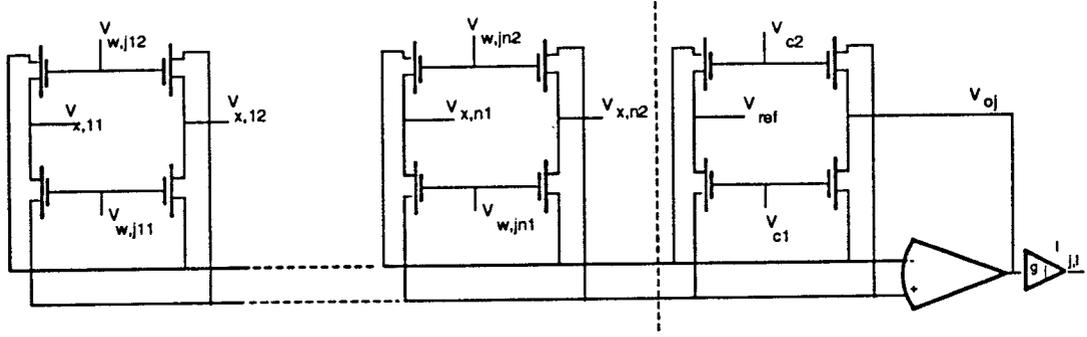


Figure 7.2: A realization of a vector multiplier

output relation of the vector product is given by

$$i_{j,l} = g_j \cdot (v_o - V_{ref}) \quad (7.5)$$

$$= \frac{g_j}{(W/L)_0(v_{c1} - v_{c2})} \cdot \sum_{i=1}^n (W/L)_i (v_{w,ji1} - v_{w,ji2})(v_{x,i1} - v_{x,i2}), \quad (7.6)$$

where g_j is the transconductance of the output stage of the j th neuron. The $(v_{w,ji1} - v_{w,ji2})$ and $(v_{x,i1} - v_{x,i2})$ represent the voltage differences of both the weigh parameters $w_{j,l}$ and the input vector $x(k)$. The value of W/L is the width and length ratio of the channel and provides the weights for the terms in the sum of product together with the control voltage $(v_{c1} - v_{c2})$. Each weight coefficient of a neuron is supported by the above structure. The vector multiplier is a basic component for weight updating, additional auxiliary circuits are needed for addressing and for weight refreshing. One option is using an A/D and D/A conversion scheme; each weight voltage is read and transformed through an A/D converter followed by a D/A conversion to write to the storage capacitor [3] for weight updating. This method can be used for the implementation of AWNN. We can see from these modules the significance of reducing the size of neural networks and the advantage of implementing AWNNs.

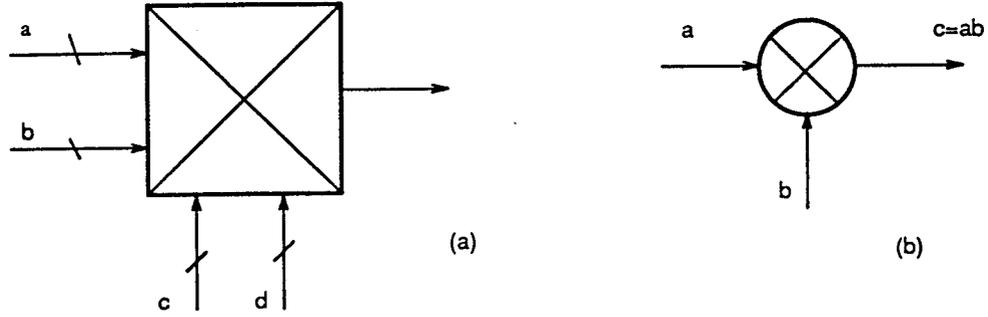


Figure 7.3: Analog VLSI blocks: (a) A vector multiplier. (b) A scalar multiplier.

7.2.3 Design Modules for AWNN Systems

We now have the design modules for both neurons and synapses. For clarity, we introduce the structure blocks illustrated in Figure 7.3 for implementing the AWNN system. The function of the vector multiplier is given by

$$\begin{aligned}
 V &= (a - b)^T Q (c - d), \\
 &= \sum_i q_i (a - b)_i (c - d)_i
 \end{aligned} \tag{7.7}$$

where a , b , c and d are vectors of the same dimension, Q is the diagonal real matrix whose elements are determined by the transconductance, control voltage and the channel geometry of the transistor as appeared in (7.5). The output of the scalar multiplier is given by $c = ab$.

In addition to the above basic design modules, several auxiliary components are needed such as weight refreshing circuits, addressing circuits and control circuits for connecting the basic components for analog VLSI system implementation.

7.3 Adaptive Learning Scheme via Analog VLSI Structures

We shall illustrate that it is feasible to implement the AWNN learning algorithm in Chapter 6 through using the analog VLSI modules for real time implementation and thus increase computation speed for on line system identification and signal representation. The training scheme in the above chapter is backpropagation, a supervised learning algorithm based on the gradient of the error square with respect to the weight vector w .

We know that a two layer neural network can approximate any nonlinear function arbitrarily close. However, the number of neurons required may far exceed the feasible limit for any practical real time implementation. With the AWNN architecture and the application of the multiresolution property of wavelets, one layer of AWNN can approximate any function with finite energy to any precision and the resulting AWNN could be smaller in size compared with its static counterpart since the AWNN contains information extracted from the system. For this reason, we consider AWNN architectures with one dynamical layer only. More complicated AWNNs will follow suit.

For convenience, we rewrite the training algorithm for weight updating (6.12), (6.18) and (6.11) respectively as

$$w_{j,l}(k+1) = w_{j,l}(k) + q_k \Delta w_{j,l}(k), \quad (7.8)$$

$$\Delta w_{j,l}(k) = (y_k - \sigma(S_k)) \sigma'(S_k) \psi_{j,l} u_k \quad (7.9)$$

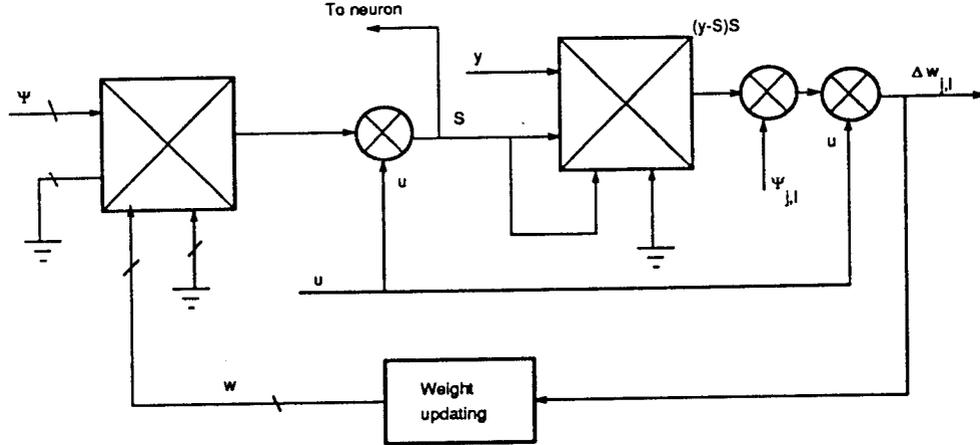


Figure 7.4: An analog VLSI architecture for AWNN

where

$$S = \sum_{j,l} w_{j,l} \psi_{j,l}(\cdot) u(\cdot). \quad (7.10)$$

We use a linear function to replace the sigmoidal function in the case of one layer AWNN to reduce the complexity of implementation. Let $\sigma(x) = x$, and consider the weight increment updating scheme

$$\Delta w_{j,l}(k) = (y_k - S_k) S_k \psi_{j,l} u_k, \quad (7.11)$$

with

$$S_k = \sum_{j,l} w_{j,l} \psi_{j,l}(\cdot) u_k, \quad (7.12)$$

while keep the updating scheme (7.8).

We use the design blocks in the previous section for AWNN system implementation. The architecture of analog implementation of the adaptive wavelet neural network is shown in Figure 7.4. The system is realized by using two vector multipliers and three scalar multipliers. The first multiplier takes the vector inputs from both the optimal wavelet basis Ψ generated from the algorithm for constructing the optimal wavelet basis and the weight vector w ; for clarity, the

subscripts are omitted here. The output of the first stage is multiplied by the input u to the system to generate S_k which is being sent to both the neuron and to the second multiplier to produce the increment of $w_{j,l}$ for updating the weight vector w . The second vector multiplier takes actually scalar in this case and its structure can be simplified accordingly. The signal $\Delta w_{j,l}$ is forwarded to the weight updating unit for producing the weight vector. This shows the principle of the implementation of the adaptive learning scheme of the AWNN structure in Chapter 6.

The design above is for each individual weight component of a single node which is connected to the neuron function block to approximate the system output. As we can see, the system is modular and can be aggregated to form a complete system with full functionality. Auxiliary control circuits and addressing schemes are needed to complete the whole AWNN system.

7.4 Feasibility of Analog AWNN

AWNN can hope to be used for real time system identification and for signal representation. It contains the information extracted from the unknown systems and the signals to be represented. The application of AWNN shall reduce the size of neural networks.

In this chapter, we demonstrate that it is feasible to implement the AWNN using analog VLSI design moduluses which can be arranged in a distributed fashion to realize the AWNN algorithm. The significance of analog AWNNs lie in its parallel structures and high operation speed. Although this chapter deals with only an framework of the architectural design, it shows the feasibility and the hardware structure for future real time system implementations.

Further research needs to be carried out in producing an implementation which saves area and provides fast operation speed. Also, issues on nonlinearity of the multipliers need to be studied to yield the best possible circuit precision.

Chapter 8

Conclusions and Future Research

We have studied the analysis and synthesis of distributed systems and discussed modeling and control of flexible structural systems and related problems such as design of distributed sensors and actuators, system identification and signal representation. We developed a systematic approach for generating the optimal orthonormal wavelet basis with compact support and proposed the AWNN structure for signal representation. The work presented in this thesis bridges theories and methodologies from signal processing and control systems.

We considered identification of distributed systems via adaptive wavelet neural networks (AWNNs). We took advantage of the multiresolution property of wavelet systems and the computational structure of neural networks to approximate the unknown plant successively. A systematic approach was developed in this thesis to find the optimal discrete orthonormal wavelet basis with compact support for spanning the subspaces employed for system identification. We then applied a backpropagation algorithm to train the network to emulate the unknown system. This work is applicable to signal representation and compression under the optimal orthonormal wavelet basis in addition to progressive system

identification and modeling. We anticipate that this work will be useful for practical applications in the areas of signal processing and intelligent systems.

In the last chapter of the thesis, we demonstrated that it is feasible to implement our AWNN concept via analog VLSI for real time implementation.

Further research needs to be carried out on possible real time optimal wavelet selection and for implementing the AWNN for both system identification and for signal representation in real time. Also, the concept of structured learning via vector quantization and feature extraction can find applications in different areas of research including system identification, signal representation, data compression and in different intelligent systems.

Bibliography

- [1] A.N. Akansu and R.A. Haddad. *Multiresolution Signal Decomposition*. Academic Press, Inc., Boston, 1992.
- [2] A.M. de Callataÿ. *Neural and Artificial Intelligence: Misconceptions about Brains and Neural Networks*. North-Holland Elsevier Science Publishers B.V., Sara Burgerhartstraat 25, P.O. Box 211, 1000 AE Amsterdam, The Netherlands, 1992.
- [3] B. Linares-Barranco and E. Sánchez-Sinencio, and A. Rodríguez-Vázquez and J. Huertas. A CMOS analog adaptive BAM with on chip learning and weight refreshing. *IEEE Trans. on Neural Networks*, 4(3):445–455, May 1993.
- [4] T. Bailey and J.E. Hubbard Jr. Distributed piezoelectric-polymer active vibration control of a cantilever beam. *Journal of Guidance, Control and Dynamics*, 8(5):605–611, Sept.-Oct. 1985.
- [5] P.M. Bainum and C.M. Diarra. The dynamics and control of the orbiting spacecraft control laboratory experiment (SCOLE) during station keeping. *The Journal of the Astronautical Sciences*, 37(4):513–527, Oct.-Dec. 1989.
- [6] A.V. Balakrishnan. A mathematical formulation of the SCOLE control problem, Part 1. Technical Report CR 172581, NASA, May 1985.

- [7] A.V. Balakrishnan. *On a Large Space Structure Control Problem*, volume Vol 97, pages 3–15. Springer-Verlag, New York, 1987.
- [8] M. Balas. Active control of flexible structures. *Journal of Optimization Theory and Applications*, 25:415–436, July 1978.
- [9] J.S. Baras and Y. Zhuang. Active vibration damping using smart material. In *Proceedings of the Workshop on Distributed Parameter Modeling and Control of Flexible Aerospace Systems*, Williamsburg, Virginia, June 8-10 1992. NASA Langley Flight Research Center.
- [10] S. Billings and S. Chen. Neural networks and system identification. In K. Warwick, G.W. Irwin, and K.J. Hunt, editors, *Neural Networks for Control and Systems*, chapter 9, pages 181–205. Peter Peregrinus Ltd., London, United Kindom, 1992.
- [11] K. Choe and H. Baruh. Actuator placement in structural control. *Journal of Guidance, Control and Dynamics*, 15(1):40–48, Jan.-Feb. 1992.
- [12] C.K. Chui. *Wavelets: A tutorial in Theory and Applications*. Acamemic Press, Inc., 250 Sixty Ave, San Diego, CA 92101, 1992.
- [13] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. on Information Theory*, 38(2):713–718, Mar. 1992.
- [14] S.A. Collina, D.W. Miller, and A.H. von Flotow. Piezopolymer spatial filters for active structural control. In C.A. Rogers and C.R. Fuller, editors, *Recent Advances in Active Control of Sound and Vibration*, pages 219–234,

VPI&SU, Blacksburg, Virginia, April 15-17 1991. NASA Langley Research Center and others, Technomic Publishing Co. Inc.

- [15] E.F. Crawley and J. de Luis. Use of piezoelectric actuators as elements of intelligent structures. *AIAA Journal*, 25(10):1373–1385, 1987.
- [16] H.H. Cudney. *Distributed Structural Control Using Multilayered Piezoelectric Actuators*. PhD thesis, SUNY Buffalo, NY, 1989.
- [17] G. Cybenko. Approximations by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4):303–314, 1989.
- [18] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41:909–996, 1988.
- [19] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, 3600 University City Science Center, Philadelphia PA, 1992.
- [20] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, Nov. 1984.
- [21] A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, 1992.
- [22] C. Herley. *Wavelets and Filter Banks*. PhD thesis, Columbia University, 1993.
- [23] K.J. Hunt and D. Sbarbaro. Studies in neural network based control. In K. Warwick, G.W. Irwin, and K.J. Hunt, editors, *Neural Networks for Control and Systems*, chapter 6, pages 95–122. Peter Peregrinus Ltd., London, United Kindom, 1992.

- [24] I. Csiszár and J. Körner. *Information Theory*. Akadémiai Kiadó, Budapest, Hungary, 1981.
- [25] A.K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1989.
- [26] P. Jorgensen. Choosing discrete orthogonal wavelets for signal analysis and approximation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages III-308-311, Minnesota, Minneapolis, April 27-30 1993.
- [27] J.U. Kim and Y. Renardy. Boundary control of the Timoshenko beam. *SIAM Journal of Control and Optimization*, 25(6):1417-1429, Nov. 1987.
- [28] C. Koch. Implementation early vision algorithms in analog networks: An overview. In H.G. Schuster, editor, *Applications of Neural Networks*, pages 3-23. VCH Verlagsgesellschaft mbH, Weinheim, Germany, 1992.
- [29] B. Kosko. *Neural Networks and Fuzzy Systems*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632, 1992.
- [30] J.E. Lagnese. *Boundary Stabilization of Thin Plates*. SIAM, Philadelphia, 1989.
- [31] J.A. Lansner and T. Lehmann. An analog CMOS chip set for neural networks with arbitrary topologies. *IEEE Trans. on Neural Networks*, 4(3):441-444, May. 1993.
- [32] C. K. Lee and F. C. Moon. Modal sensors/actuators. *Journal of Applied Mechanics*, 57:434-441, June 1990. Trans. of the ASME.

- [33] K.B. Lim. Method for optimal actuator and sensor placement for large flexible structures. *Journal of Guidance, Control and Dynamics*, 15(1):49–57, Jan.-Feb. 1992.
- [34] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Trans. on Pattern Analysis and Machine intelligence*, 14(7):710–732, July 1992.
- [35] S.G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$. *Transactions of The American Mathematical Society*, 315(1):69–87, Sept. 1989.
- [36] S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.
- [37] M. Marchesi, G. Orlandi, F. Piazza, and A. Uncini. Fast neural networks without multipliers. *IEEE Trans. on Neural Networks*, 4(1):53–62, Jan. 1993.
- [38] C. Mead. *Analog VLSI and Neural Systems*. Addison-Wesley VLSI System Series. Addison-Wesley Publishing Company, Inc., 1989.
- [39] D.H. Nguyen and B. Widrow. Neural networks for self-learning control systems. *IEEE Control Systems Magazine*, 10(3):18–23, April 1990.
- [40] Y.C. Pati. *Wavelets and Time-Frequency Methods in Linear Systems and Neural Networks*. PhD thesis, University of Maryland, College Park, MD 20742, 1992.

- [41] Y.C. Pati and P.S. Krishnaprasad. Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations. *IEEE Trans. on Neural Networks*, 4(1):73–85, Jan. 1993.
- [42] D. Pollen. $SU_I(2, F[z, 1/z])$ for F a subfield of C . *Journal of the American Math. Society*, 3(3):611–624, July 1990.
- [43] O. Rioul and M. Vetterli. Wavelets and signal processing. *IEEE Signal Processing Magazine*, pages 14–38, Oct. 1991.
- [44] N. Tepedelenlioglu, A. Rezgui, R. Scalerò, and R. Rosario. Fast algorithms for training multilayer perceptrons. In B. Souček, editor, *Neural and Intelligent Systems Integration*, chapter 4, pages 107–133. John Wiley & Sons. Inc., 1991. Sixth-Generation Computer Technology Series.
- [45] A.H. Tewfik, D. Sinha, and P. Jorgensen. On the optimal choice of a wavelet for signal representation. *IEEE Trans. on Information Theory*, 38(2):747–765, Mar. 1992.
- [46] S. Timoshenko and D.H. Young. *Vibration Problems In Engineering*. D.Van Nostrand Co.,Inc., Canada, third edition, 1955.
- [47] H.S. Tzou. Integrated distributed sensing and active vibration suppression of flexible manipulators using distributed piezoelectrics. *Journal of Robotic Systems*, 6(6):745–767, 1989.
- [48] P.P. Vaidyanathan. *Multirate Systems and Filterbanks*. Prentice Hall Signal Processing Series. P T R Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1993.

- [49] K. Zeger and A. Gersho. A stochastic relaxation algorithm for improved vector quantiser design. *Electronics Letters*, 25(14):896–898, July 1989.
- [50] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. on Neural Networks*, 3(6):889–898, Nov. 1992.
- [51] Y. Zhuang and J.S. Baras. Distributed control of a Timoshenko beam. In *Proceedings of the Third International Conference on Adaptive Structures*, pages 216–227, San Diego, California, Nov. 9-11 1992. Technomic Publishing Co., Inc.
- [52] Y. Zhuang and J.S. Baras. Identification of infinite dimensional systems via adaptive wavelet neural networks. Tech. Report 93-64, Institute for Systems Research, The University of Maryland, College Park, MD 20742, Aug. 1993.
- [53] Y. Zhuang and J.S. Baras. Optimal design of distributed sensors and actuators made of smart materials. In *Proceedings of The First IEEE Regional Conference on Aerospace Control Systems*, pages 65–69, Westlake Village, California, May 25-27 1993. IEEE Control Systems Society.
- [54] Y. Zhuang and J.S. Baras. Feature extraction and classification of signals via adaptive wavelet neural networks. In preparation, 1994.
- [55] Y. Zhuang and J.S. Baras. Optimal wavelet basis selectin for signal representation. In Harold H. Szu, editor, *Wavelet Applications*, pages 200–211, Vol. 2242, 1994. SPIE–The International Society for Optical Engineering, The Society of Photo-Optical Instrumentation Engineers, P.O. Box 10, Bellingham, 98227-0010, USA.