Optical Signal Processors: Novel Computer Models and Experiments

by

David Sun

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED

Dr. Richard O. Claus
Co-Chairman

Dr. Kent A. Murphy
Co-Chairman

Dr. Anbo Wang

April 7, 1994

Blacksburg, Virginia

c.2

LD
5655
V855
1994
S86
c.2

Optical Signal Processors: Novel Computer Models and Experiments

by

David Sun

Electrical Engineering

(ABSTRACT)

The concept of incorporating optical fiber delay lines as signal processors is investigated. These signal processors are discrete time filters implemented using optical fiber delay lines in the form of optical fiber couplers Advantages of such a filter include the ability to eliminate the need for electro-optical conversion or at least postpone the conversion to a point where the electronics may be simplified. In addition, these optical signal processors exhibit all the advantages afforded traditional optical fiber systems. Furthermore, these devices allow high speed signal processing rates greater than 1GHZ which rival the performance of the most advanced electronic counterparts available today.

In this thesis, optical signal processors are investigated in detail. First, the fundamental signal processing concepts and optical component background is discussed for the analysis of the devices are developed. Subsequently, various configuration of optical signal processors are discussed and novel computer models for these configurations using Z-transform theory are presented. Lastly, experimental verification is presented of the rudimentary models and a unique approach is presented that will allow the creation of optical signal processors with a flexibility never achievable before.

ACKNOWLEDGMENTS

I wish to thank the members of my committee for their help in fostering my education. I am grateful to Dr. Claus for providing the support and ability for me to pursue a graduate degree at Virginia Tech and with FEORC. I have also greatly appreciated the pearls of wisdom imparted to me from Dr. Murphy as well as the times he has allowed me to kick out his door stop. I would also like to thank Dr. Wang for his help and showing me as an undergraduate how all the pieces of my education actually do fit together.

I would also like to thank all of the people of FEORC and F&S, past and present, that have helped me learn, challenged me to excel and made me laugh. Without them, I probably would have been dumber, unhappier, and older by the time I finished this work.

In addition, I would like to acknowledge the Scuba Club at Virginia Tech and its members. My affiliation with them has provided me with the escape that I needed to keep my sanity. Whenever burnout and frustration was setting in, I was able to focus on something else. For a weekend or only a night, that was what I needed to revitalize me and keep me going.

Finally, I am grateful to my parents, Jack and Teresa and my sisters, Mina and Susan. They have provided me with the inspiration and motivation for my continued schooling.

TABLE OF CONTENTS

Chapter 1: Introduction 1

Chapter 2: Elementary Signal Processing 5

2.1 Z-Transforms 5

2.2 Transfer Function 6

2.2.1 Stability 7

2.2.2 Frequency Response 8

2.3 Discrete Time System Structures 11

2.3.1 Direct Structures 11

2.3.2 Lattice Structures 14

2.4 Group Delay 14

Chapter 3: Optical Components 17

3.1 Delays 17

3.2 Multipliers 19

3.2.1 Polished 20

3.2.2 Fused 22

3.2.3 Asymmetric 25

3.3 Junctions 26

Chapter 4: Filter Configurations 27

4.1 FIR Filter 27

4.2 Lattice Filter 29

4.3 Single Loop Filter 32

4.4 Gain Amplified Filter 34

4.5 Asymmetric Coupler Filter 38

4.6 Double Loop Filter 40
Chapter 5: Computer Modeling 44
5.1 FIR Filter 44
5.2 Lattice Structure Filter 46
5.3 Single Loop Filter 48
5.4 Gain Amplified Filter 53
5.5 Asymmetric Coupler Filter 57
5.6 Double Loop Filter 59
Chapter 6: Experiments 65
6.1 Single loop filter 65
6.2 Tunable Single Loop Filter 71
6.3 Future Experiments 76
6.3.1 Modeling of Higher Order Systems 77
6.3.2 Fabrication of Higher Order Systems 77
6.3.3 Twist Tunable Coupler Manufacturing Issues 78
6.3.4 Photofluidic Systems 78
6.3.5 Adaptive Optical Signal Processors 79
Chapter 7: Conclusions 80
Chapter 8: References 82
Chapter A: Appendix 86

TABLE OF FIGURES

Figure 1.1. Optical fiber sensor based smart structure. 2

Figure 1.2. Revised diagram of a smart structure with an optical signal processor. 3

Figure 2.1. Sample pole zero plot. 7

Figure 2.2. Magnitude response of example filter. 10

Figure 2.3. Sample signal flow graph [2.1]. 12

Figure 2.4. Direct form II of example filter [2.1]. 13

Figure 2.5. Lattice Structures. a) Basic section b) IIR lattice system. 14

Figure 2.6. Phase distortion of a signal after passing through an IIR filter. 16

Figure 3.1. Simplified schematic of filter implemented by Chang *et al.* 18

Figure 3.2. Schematic of an optical fiber coupler. 19

Figure 3.3. a) Geometry of a polishing block. b) Crossection of an assembled coupler [3.9]. 21

Figure 3.4. Fabrication of a fused biconical tapered coupler using an oxy-butane torch [3.10]. 23

Figure 4.1. Optical FIR filter. a) Fiber configuration. b) Data structure. 28

Figure 4.2. Lattice Filter. a) Fiber configuration. b) Data structure. 30

Figure 4.3. Single loop signal processor. a) Fiber configuration. b) Data structure. 32

Figure 4.4. Magnitude response of an ideal filter with K=1/3. 33

Figure 4.5. Gain Amplified Filter. a) Fiber configuration. b) Data structure. 34

Figure 4.6. Frequency response of single loop filter with G=2.0. 36

Figure 4.7. Pole-zero plot of filter in Figure 4.5. 37

Figure 4.8. Amplified loop with noise injected. a) Fiber configuration. b) Data structure. 38

Figure 4.9. Asymmetric Coupler Filter. a) Fiber configuration. b) Data structure. 39

Figure 4.10. Double Loop Filter. a) Fiber configuration. b) Data structure. 41

Figure 5.1. Possible zero locations in the Z-plane for optical FIR filters of order 1-6 respectively. 45

Figure 5.2. Possible pole locations in the Z-Plane for optical lattice filters of order 1-6 respectively. 47

Figure 5.3. Magnitude vs. frequency vs. coupling coefficient for a single loop filter. 48

Figure 5.4. Single loop filter responses for loss factors of a) 0.15, b) 0.45 and c) 0.80 respectively. 50

Figure 5.5. Amplitude at 0.50 Fs vs. coupling ratio vs. loop loss. 51

Figure 5.6. Interpolated coupling ratio-loop loss combinations. 52

Figure 5.7. Gain Amplified Filter responses with G=a) 1.00, b) 1.30, c) 1.80 and d) 2.20 respectively. 54

Figure 5.7. Gain Amplified Filter responses with G= e) 3.30, f) 3.90 and g) 5.00 respectively. 55

Figure 5.7. Asymmetric filter response for B= a) 0.09 and b)0.39 respectively. 57

Figure 5.7. Asymmetric filter response for B= c) 0.69, d) 0.89, e) 0.95, and f) 1.00 respectively. 58

Figure 5.9. Random analysis of the possible pole locations for the ideal double loop filter. 60

Figure 5.10. Double loop configuration with a) B≅0, b) B=0.10, c) 0.25 and d) 0.30 respectively. 61

Figure 5.10. Double loop configuration with e) B=0.35, f) 0.40, g) 0.45 and h) 0.59.... 62

Figure 5.10. Double loop configuration with i) B=0.85. 63

Figure 6.1. Equipment setup for experiments. 66

Figure 6.2. Frequency response of the laser system. 67

Figure 6.3. FFT of the input waveform (Top) and output waveform (Bottom). 68

Figure 6.4. Theoretical (solid) and measured (dotted) frequency response of the filter..... 69

Figure 6.5. Comparison of filter response at 3.1 MHz. 70

Figure 6.6. Comparison of filter response at 6.0 MHz. 70

Figure 6.7. Coupling coefficient as a function of rotation. 72

Figure 6.8. Coupler loss as a function of rotation. 72

Figure 6.9. FFT of the input waveform (Top) and output waveform (Bottom). 73

Figure 6.10. Plot of frequency response. 74

Figure 6.11. FFT of the input waveform (Top) and output waveform (Bottom)..... 75

..... 76

Figure 6.12. Theoretical (solid) and measured (dotted) frequency response of the re-tuned filter..... 76

Figure 6.13. Photofluidic system with an optical signal processor. 79

Chapter 1: Introduction

Optical fibers have become widely accepted as the communications platform of today's community. Their applications are typically in networks ranging from existing long-haul communication systems and their repeaterless versions to the highly touted information superhighway. In addition to communications, fiber optics has recently gained acceptance in their applications as sensors. These sensors are small, light weight and flexible, they can operate in a wide range of environmental conditions and can be configured to respond to many physical mesurands [1.1].

Signal processing plays a particularly important role in fiber optics for both sensing and communication applications. The outputs of optical fiber sensors often requires signal processing due to their high sensitivity to multiple measurands. Similarly, for communication networks, the long distance between transmitters and receivers poses a potential problem. As the distance increases, the optical signal propagating in the fiber is

prone to degradation. As a result, a balance is required between the allowable signal degradation and the cost and performance of signal reconditioners. Much of the performance of these systems is dictated by the performance of the receivers in the system, specifically the rate at which the optical signals can be processed.

The physical marriage of these two disciplines presents a breakthrough in both fields with the ability to benefit a broad scope of other areas of study. In particular, these signal processors present a significant breakthrough in the ongoing research of optical computing and optical integrated systems. These advances may also add many new and exciting dimensions to fiber sensor-based smart structures and systems. In an envisioned smart structure environmental sensing, mechanical actuation and structural control are integrated internally [1.2]. Figure 1.1 shows a diagram of an optical fiber sensor-based smart structure.

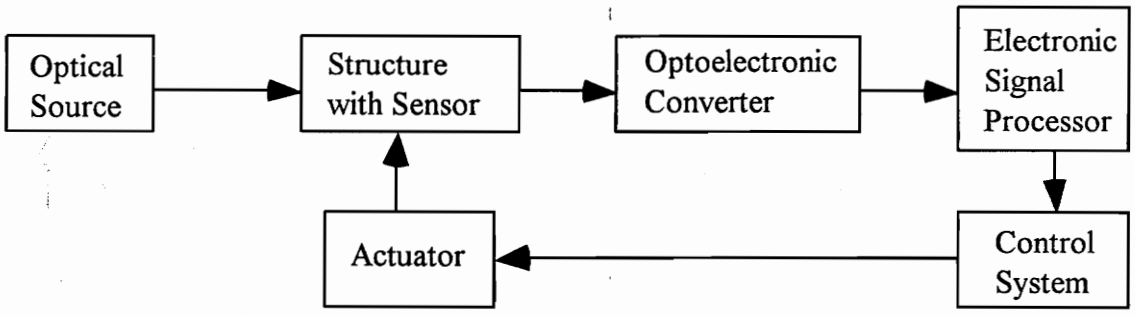


Figure 1.1. Optical fiber sensor based smart structure.

Traditional signal processing in a such an optical-based control system involves the detection of the signal, electronic filtering and remodulating an optical source. This approach has not only proved costly but the process often degrades the signal to noise ratio by more than 50dB [1.3]. By creating these filters using optical fiber delay lines the signal processing may be done either without performing the electro-optical conversion or

at least postponing the conversion to a point where the electronics may be simplified. These optical signal processors exhibit all the advantages afforded optical fiber systems such as small packaging, electromagnetic interference immunity and ultra-high bandwidth. In addition, the devices have low, frequency-independent loss, negligible dispersion and polarization independence. Furthermore, these devices allow high speed signal processing rates greater than 1GHz which rival the performance of their most advanced electronic counterparts available today. Figure 1.2 shows a revised diagram of a smart structure with an optical signal processor.

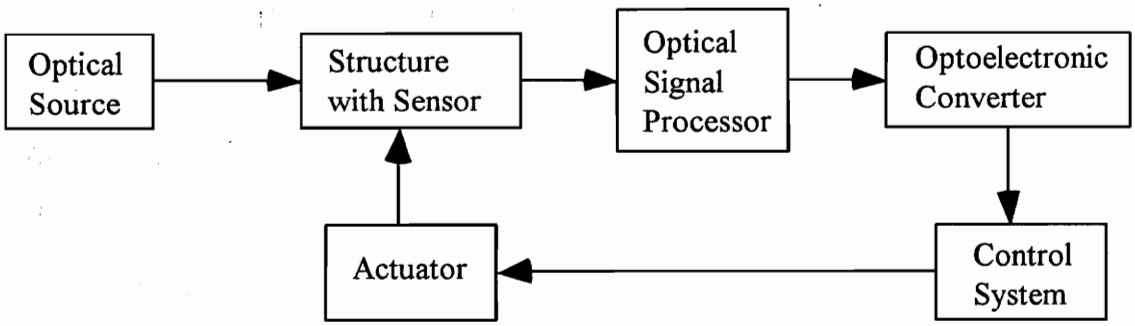


Figure 1.2. Revised diagram of a smart structure with an optical signal processor.

This filtering phenomena is achieved by an interference of the modulation instead of the optical carrier. As a result, it is required that the coherence time of the modulated source be less than the unit delay of the filter. Although the signal is continuous, the unit delay lines in the filter generate discrete time samples. In addition, since these systems are linear and time-invariant, their analysis can be facilitated using Z-transform theory.

In this thesis, optical signal processors are investigated in detail. Chapter 2 establishes the fundamental signal processing concepts necessary for the analysis of the devices. Chapter 3 provides a background investigation of the optical components necessary to create the

signal processing devices. Chapter 4 describes various configurations of optical signal processors. Novel computer models for these configurations using Z-transforms are presented in Chapter 5. These models are coded in Matlab 4.0 and the subroutines responsible for each model are referred to in the text. These subroutines and their support functions are included in the Appendix. Although selected frames from these models are included in this text, the most knowledge will be gained from these models by allowing them to perform in real time. Lastly, Chapter 6 will present experimental verification of a rudimentary model as well as demonstrate a unique approach which applies these models. This approach will allow the creation of optical signal processors with a flexibility never achievable before.

Chapter 2: Elementary Signal Processing

In order to properly understand the operation of an optical signal processor, it is important to understand the foundation of knowledge upon which they are based. As they operate in a fashion similar to electronic filters, a good starting point would be to introduce elementary signal processing concepts. This chapter will briefly discuss the fundamental concepts of discrete time signal processing. The purpose of this discussion is to familiarize these concepts to the reader and by no means encompasses the complete field of signal processing. For a more detailed discussion, the reader is refereed to the texts referenced in this chapter.

2.1 Z-Transforms

Optical signal processors are devices whose output consists of scaled, time delayed versions of its input. This output is generated using weighted fiber taps separated by

lengths of optical fiber acting as delays. These delays are constructed using fiber whose length is an integer multiple of a chosen unit length, or delay. Although a more detailed discussion of these devices will follow, it is evident that since the taps are separated by integer multiples of a unit delay, the evaluation of complex systems can be facilitated by the use of the Z-transform. By definition, the Z-transform of a discrete signal is described as

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}, \tag{2.1}$$

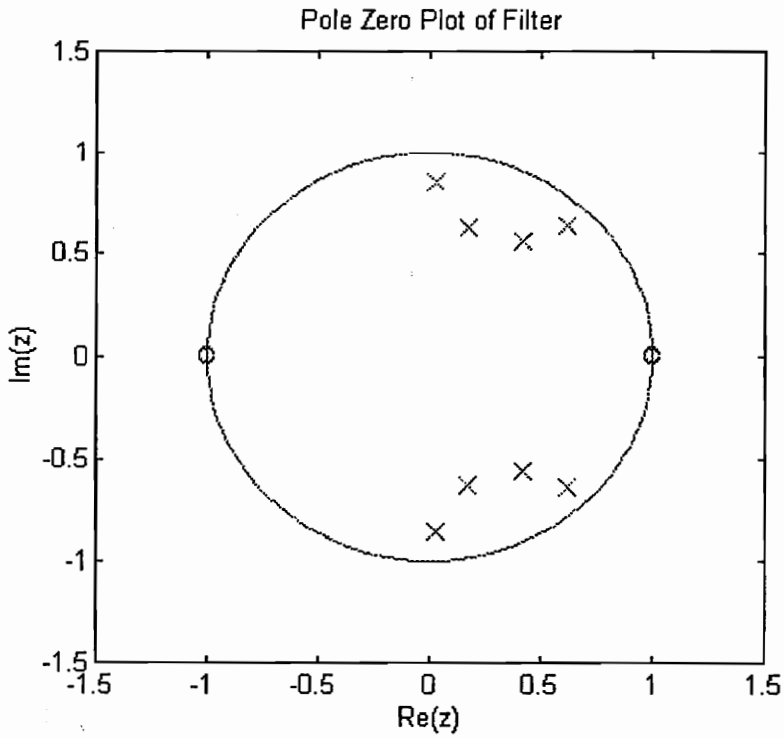
where z is the complex transformation variable [2.1]. A close examination will show that a Z-transform is also a Laurent series and as such, is subject to all the properties of complex variables. In particular, Cauchy's Integral and Residue theorems are powerful tools that have many applications [2.2].

By evaluating the system functions in the Z domain, many powerful signal processing techniques may be realized with results that may be understood intuitively. In using Z-transform techniques however, we must assume that these systems are linear and time invariant (LTI).

2.2 Transfer Function

A system can be readily described by examining its transfer function. The transfer function may be obtained by taking the ratio of the Z-transform of the output to the Z-transform of the input. Taking the roots of the denominator and numerator polynomials and plotting these poles and zeros, much insight can be read into the response of the system. By

plotting the real and imaginary components of these poles and zeros on the Z-plane and comparing them with a circle centered at the origin of unity radius, the systems stability and frequency response can be easily examined. Figure 2.1 shows a sample Pole-Zero plot for a given discrete system.



$$H(z) = \frac{0.10 - 0.04z^{-2} + .06z^{-4} - .04z^{-6} + 0.10z^{-8}}{1 - 2.47z^{-1} + 4.31z^{-2} - 4.89z^{-3} + 4.48z^{-4} - 2.91z^{-5} + 1.52z^{-6} - 0.50z^{-7} + 0.12z^{-8}}$$

Figure 2.1. Sample pole zero plot.

2.2.1 Stability

The fundamental criteria for stability in digital systems is that all repeated poles of the system must lie inside the unit circle. A system with poles outside of the unit circle is inherently unstable and systems with poles on the unit circle are considered marginally

stable. This analysis of stability is similar to that of continuous time systems and an in depth analysis would show that there is a mapping between the stability region in the continuous time case and the unit circle [2.3].

2.2.2 Frequency Response

In addition to stability, the pole-zero plot is also able to describe the frequency response of the system. This effect is a result of the relationship between the Z-transform and the Fourier Transform. Equation (2.1) gives the definition of the Fourier Transform,

$$X(e^{j\omega t})=\sum_{n=-\infty}^{\infty}x(n)e^{-j\omega nt}. \tag{2.1}$$

It can be seen upon comparison with (2.1) that the Fourier Transform is simply the Z-transform where z is defined as $z=e^{j\omega t}$. By doing so we have restricted the magnitude of z to unity and as a result, the Z-transform evaluated at $|z|=1$ is the Fourier Transform where it exists [2.4]. This unity magnitude results in the common phrase, "evaluating the Z-transform along the unit circle" as each point on the unit circle now represents a certain frequency.

A useful geometric approach is also available for the calculation of the frequency response of a system. By writing the transfer function of the system in factor form we obtain

$$H(z)=\frac{b_0\prod_{m=1}^{M}(1-z_m z^{-1})}{a_0\prod_{k=1}^{K}(1-p_k z^{-1})}, \tag{2.3}$$

where z_m are the zeros of the system, p_k are poles of the system and b_0 and a_0 are gain coefficients. Equation (2.3) can be rewritten as

$$H(z) = \frac{b_0 z^{N-M} \prod_{m=1}^{M} (z - z_m)}{a_0 \prod_{k=1}^{K} (z - p_k)}. \tag{2.4}$$

Bearing in mind that z is a complex variable, (2.4) may be rewritten as

$$H(z) = z^{N-M} \frac{b_0 \prod_{m=1}^{M} B_m e^{j\theta_m}}{a_0 \prod_{k=1}^{K} A_k e^{j\phi_k}}, \tag{2.5}$$

where the roots are considered to be vectors of magnitude B_m or A_m and angle θ or ϕ respectively. By evaluating (2.5) on the unit circle one can show that

$$|H(z)| = \frac{b_0 \prod_{m=1}^{M} B_m}{a_0 \prod_{k=1}^{K} A_k}, \tag{2.6}$$

and

$$\angle H(z) = (N-M)\omega T + \sum_{m=1}^{M} \theta_m - \sum_{k=1}^{K} \phi_k. \tag{2.7}$$

Equation (2.6) implies that the magnitude response of a system at a certain frequency may be obtained by finding the product of the lengths of vectors between each zero and the frequency of interest on the unit circle, divided by the product of the lengths of vectors from each pole. The phase response at a certain frequency is described by (2.7) as the

difference between the sum of the zero vector angles and the sum of the pole vector angles.

It can easily be shown from (2.6) that the closer a zero is to the unit circle, the smaller the magnitude response will be for that particular frequency. In addition, the closer a pole is to the unit circle, the larger the magnitude response. Another implication of this effect is that the closer the roots are to the unit circle the sharper the transition. Figure 2.2 shows the magnitude response of the system previously given. As can be seen, the magnitude falls to zero at frequencies where zeros lie on the unit circle. In addition, it can be seen how a sharp transition may be implemented by placing roots on or near the unit circle.

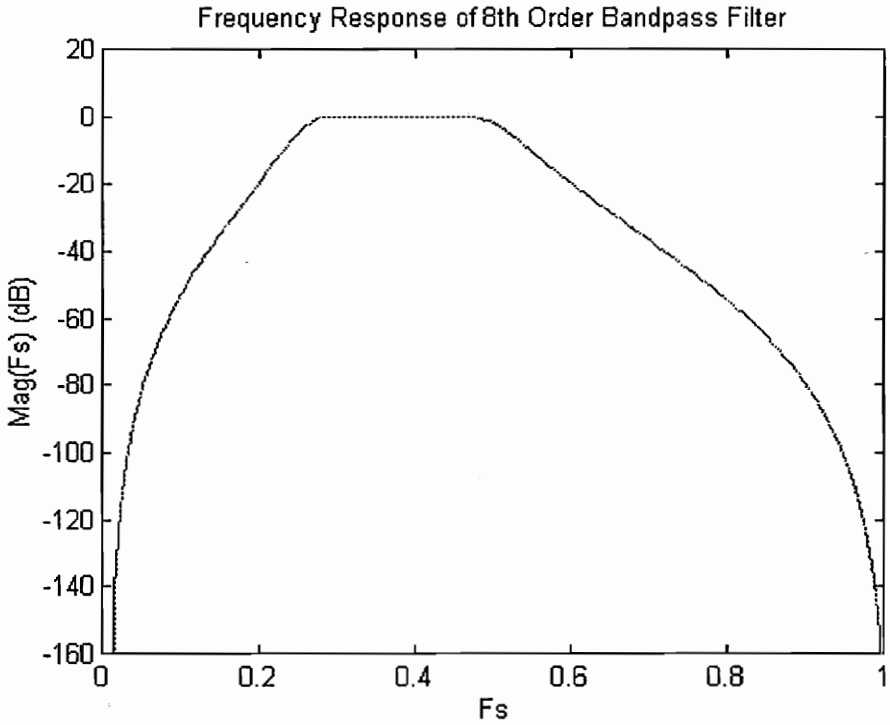


Figure 2.2. Magnitude response of example filter.

To conclude our discussion of frequency response, it is relevant to note its symmetry. For x(n) real we have that $x(n)=x(n)^*$. Recalling (2.2), we then have

$$X(e^{j\omega t}) = \sum_{n=-\infty}^{\infty} x(n)^* e^{-j\omega n t}$$

$$= \left[\sum_{n=-\infty}^{\infty} x(n) e^{-jn(-\omega t)} \right]^*$$

$$= \left[X(e^{-j\omega t}) \right]^*. \tag{2.8}$$

From (2.8) we can see that for x(n) real, the magnitude response of a system is an even function in ω while the phase response is an odd function. In addition, both of these responses are periodic in ω with a the same period as $e^{j\omega t}$; that is $2\pi/T$.

2.3 Discrete Time System Structures

Graphical representations are sometimes helpful in understanding the functionality of complex systems. The most common graphical representations are the block diagram and signal flow graph. Two specific forms of block diagrams are presented, as they pertain to optical signal processors.

2.3.1 Direct Structures

While the pole-zero plot allows one to visualize the frequency response of a filter, signal flow graphs allow one to visualize the implementation of the filter. These signal flow graphs, sometimes referred to as block diagrams, are commonly used in control theory as well as signal processing. Figure 2.3 depicts a typical signal flow graph for a recursive system. A recursive system is a filter implementation that incorporates feedback as well as

feed-forward. These systems are also termed Infinite Impluse Response (IIR) systems as their impulse response would become infinitely long due to the feedback. The converse of IIR systems is the Finite Impulse Response (FIR) system which incorporates only feed-forward.

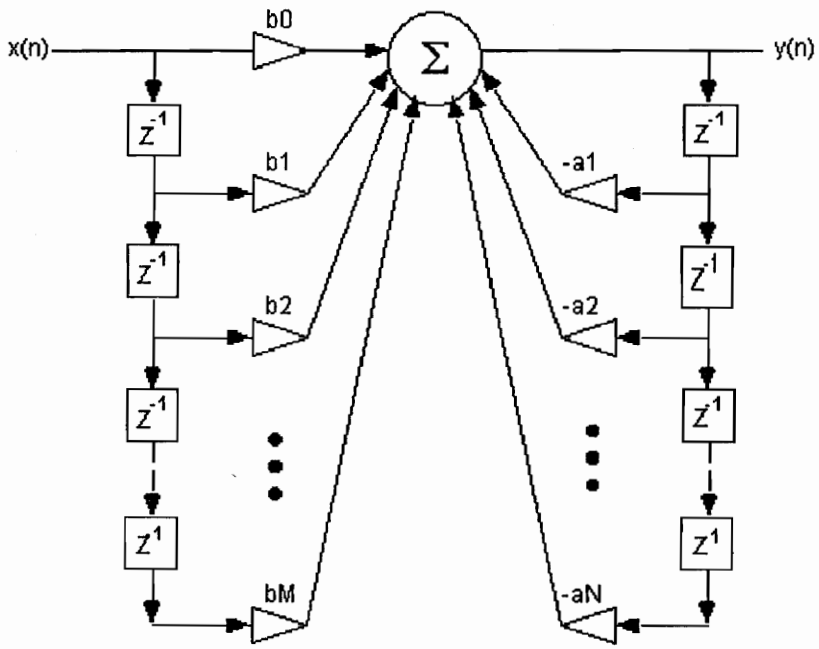


Figure 2.3. Sample signal flow graph [2.1].

As can be seen in this Figure 2.3, an input is branched and sent to multipliers b_m and a_k. These multipliers represent the coefficients of the transfer function and the Z-transform operator, z-1, represents a unit delay. The output signal is generated by sending the various signals to a summer junction where the signals are combined. In typical filters, the delays are realized using a storage element such as a shift register. In general, all digital filters consist of certain basic components: summers, branchers, multipliers, and delays.

The filter depicted in Figure 2.3 can be reconstructed to utilize the least number of delays. One such canonic structure is called the Direct Form II structure. Figure 2.4 shows the example filter in Direct Form II.

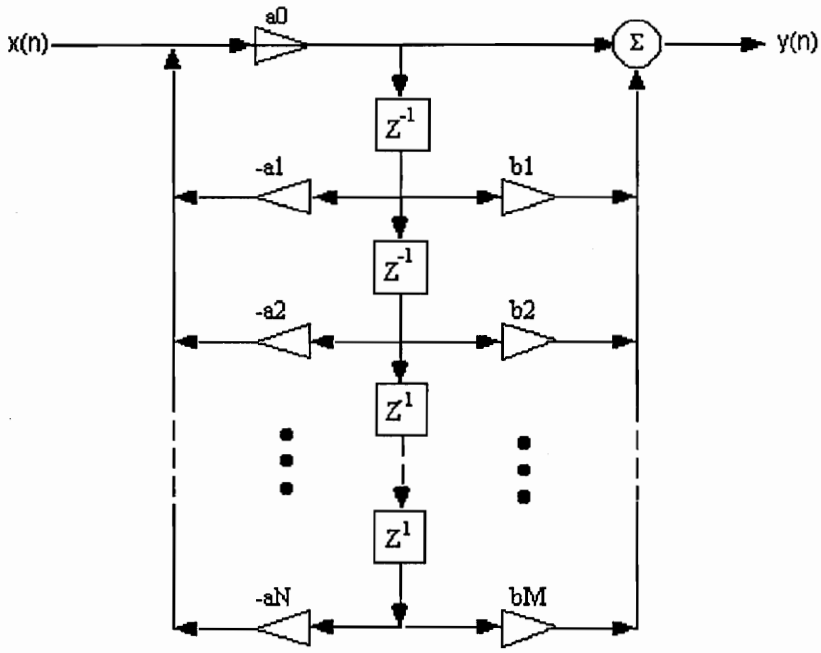


Figure 2.4. Direct form II of example filter [2.1].

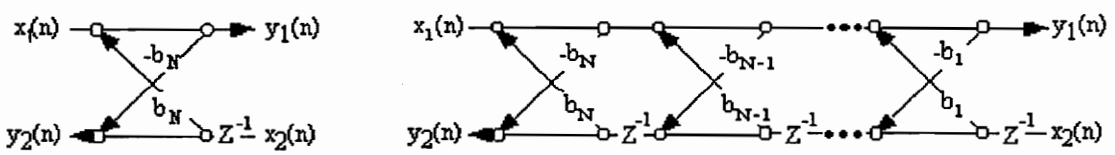
As can be seen in Figure 2.4, the numerator of the transfer function is on the feed-forward portion of the graph while the denominator is implemented on the feedback portion. In addition, the magnitude of the multipliers are identical to the polynomial coefficients as well as in the same order.

Unfortunately, most optical signal processors are not able to incorporate the elegance of such a representation. However a thorough understanding of these principles facilitates the understanding of optical systems.

2.3.2 Lattice Structures

Another class of structures for discrete time systems are called lattice structures. Lattice structures are motivated by autoregressive signal modeling theory [2.5]. While a detailed discussion of lattice structures is beyond the scope of this thesis, a brief introduction will be included as they pertain to optical signal processors.

A basic recursive lattice section is shown in Figure 2.5a, while Figure 2.5b shows a general all-pole IIR lattice system. As can be seen, a lattice system can consist of a cascade of basic two-port devices. This lattice structure allows high order filters to be easily constructed by cascading multiple sections of lower order. By selectively choosing the ports, different filter responses can be obtained.



a) b)

Figure 2.5. Lattice Structures. a) Basic section b) IIR lattice system.

2.4 Group Delay

In our previous discussion of system structures we introduced the concepts of recursive and non-recursive systems. While there are many differences between IIR and FIR systems, group delay can be the deciding factor in choosing a particular structure or configuration. The group delay of a system is defined as

$$G(\omega)=\frac{d}{d\omega}\angle H(z). \tag{2.9}$$

Physically, this can be described as the delay each frequency component experiences in the system. If the group delay is not constant, frequency components which enter the system simultaneously will experience a different delay and as a result, will reach the output at different times. The end result is a distorted waveform at the output. This distortion is termed phase distortion since group delay is the derivative of the phase response, and linear phase is required for a constant group delay. It is of interest to note that while the output is distorted in time, the frequency content remain consistent with expected results.

Figure 2.6 shows an example of phase distortion. A signal composed of multiple frequencies all in the passband is sent through a filter. The solid line shows that the frequency content of the input waveform is maintained while the dashed line shows the distorted waveform after it is passed through the filter.

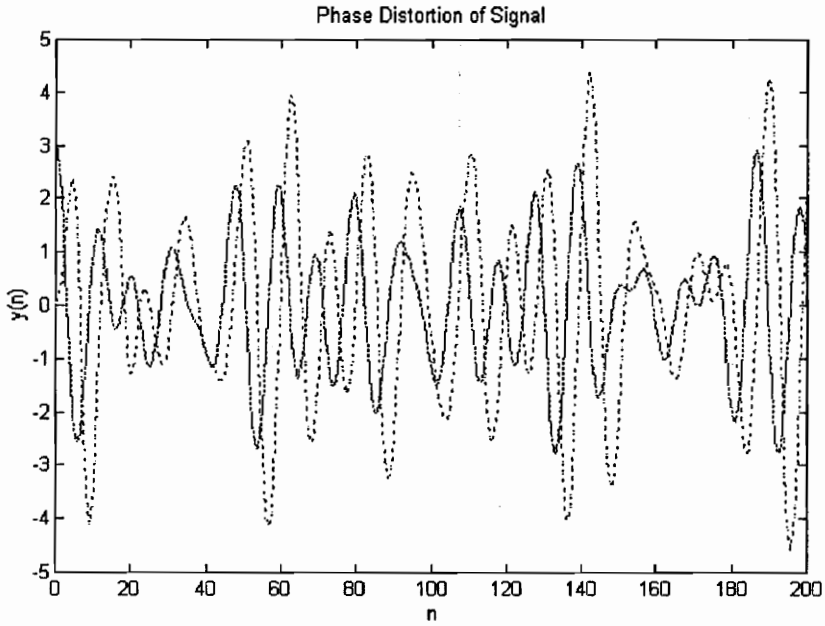


Figure 2.6. Phase distortion of a signal after passing through an IIR filter.

FIR filters are popular due to the ease in which they achieve a linear phase response. FIR filters are commonly used in voice communications and other areas in which phase distortion is unacceptable. Unfortunately, FIR filters generally require a filter order ten times greater than IIR filters to achieve the same frequency response.

Chapter 3: Optical Components

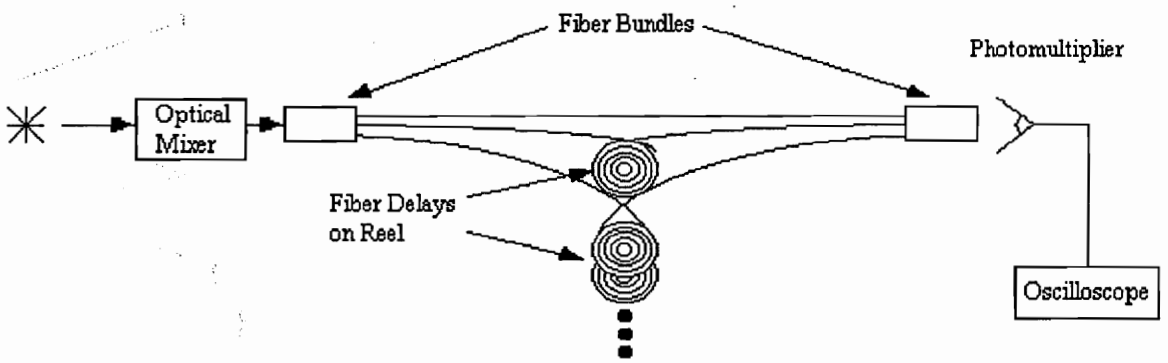
As mentioned previously, all digital filters consist of a few basic components. In order to implement an all optical digital signal processor, it is necessary to find optical equivalents to these components. This chapter will discuss these equivalent components and their functions for an optical signal processor.

3.1 Delays

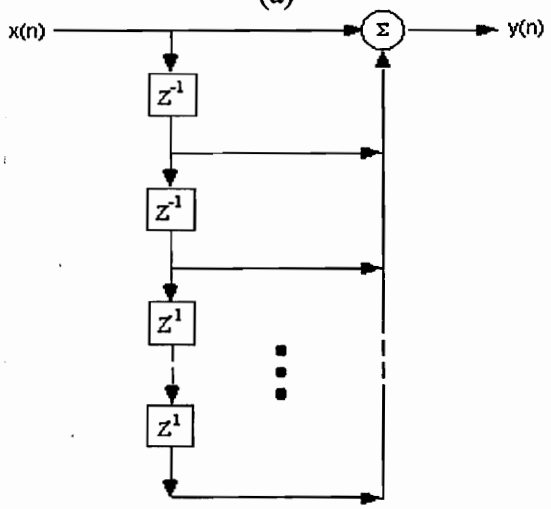
One of the integral components of digital filter is the unit delay represented in signal flow graphs by z^{-1}. These components can be implemented optically by using optical fibers cut to a precise length. The time delay is then defined as the time required for light to propagate from one end of the fiber to the other. Assuming a refractive index of n=1.5, common to all fiber sections in the network, the propagation delay is approximately 5 ns per meter. By choosing a given length of fiber, a unit time delay of the system can be

defined and the sampling frequency would be the inverse of the propagation delay. Because of the dispersive nature of multimode fibers, their use in optical signal processors is limited. In addition, the low-loss nature of singlemode fiber makes the choice easier.

A system requiring only summers, branches, and delays was successfully implemented by Chang *et al.* [3.1]. A simplified schematic of this filter is depicted in Figure 3.1a with its equivalent signal flow graph in Figure 3.1b. The requirement that all coefficients of the transfer function equal unity in this system, results in a limited functionality.



(a)



(b)

Figure 3.1. Simplified schematic of filter implemented by Chang *et al.*

3.2 Multipliers

By including multipliers in the system, one can achieve a greater range of filter responses. These multipliers are implemented in an optical processor through the use of fiber optic couplers. In such a system, the multiplier is equal to the coupling coefficient of the coupler. This correlation is possible due to the nature of the device.

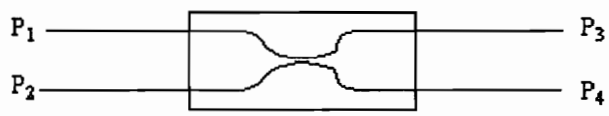


Figure 3.2. Schematic of an optical fiber coupler.

The input-output relationship of a 2x2 directional coupler shown in Figure 3.2 can be described by the following transfer matrix

$$\begin{pmatrix} P_3 \\ P_4 \end{pmatrix} = \gamma \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix}; \tag{3.1}$$

where P1, P2 and P3, P4 are the respective input and output powers of the fields propagating in the corresponding ports and γ is the insertion loss factor which is usually determined by taking the ratio of the output to input optical power of the coupler. Assuming that the coupler is symmetric, (3.2) is reduced to

$$\begin{pmatrix} P_3 \\ P_4 \end{pmatrix} = \gamma \begin{pmatrix} 1-K & K \\ K & 1-K \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix}, \tag{3.2}$$

where K is referred to as the coupling coefficient. These parameters and relationships hold true for singlemode and multimode couplers regardless of their manufacturing process.

The basic theory of singlemode fiber optic coupler operation will be investigated here to provide the foundation for understanding of optical signal processor implementations. It is interesting to note however that as the coupling coefficient determines the magnitude of the multiplier, by the nature of the device, couplers can only function as multipliers between zero and one. This restriction generates a paradox in optical signal processing. While filters which incorporate only couplers as multipliers are guaranteed for stability, they also restrain these filters to the realm of all positive systems. These implications will be discussed in further detail in Chapter 4.

3.2.1 Polished

The polished coupler was first created using mulitmode fiber [3.6]. A singlemode version was developed after refinements were made in the manufacturing techniques [3.6,3.7]. These refinements in manufacturing techniques were necesscary due to the mechanism of coupling in a polished coupler.

Coupling between two fibers in a polished coupler is achieved by the interaction of the evanescent waves in each fiber. In order to achieve this interaction, an area of each fiber is polished so that their cores are partially exposed. This polishing is usually done by placing each fiber in a block, or substrate, with curved slots for the fiber. By adjusting the radius of curvature for the slots in each polishing block, the interaction length for the

coupler is determined. These blocks are then polished with a mechanical lapping wheel until the appropriate depth is achieved. Figure 3.3 shows the geometry of the polishing blocks as well as a crossection of an assembled coupler.

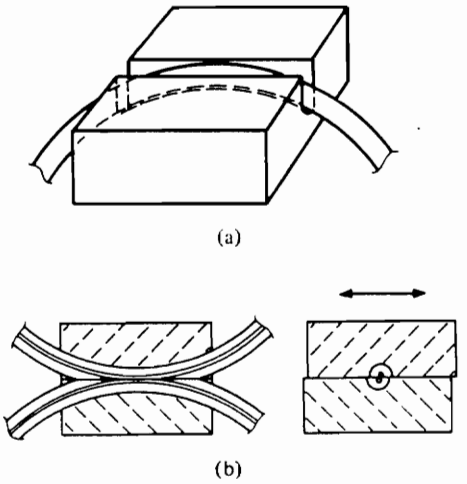


Figure 3.3 a) Geometry of a polishing block. b) Crossection of an assembled coupler.[3.9]

Once the desired polishing depth is achieved, the two polishing blocks are mated together to align the cores of each fiber. By constructing the polishing blocks of a transparent material, one can ensure a proper and clean mating by generating an interference pattern at the interface of the substrates. When the interference pattern consists of less than one fringe, an index matching liquid may be introduced between the substrate through capillary action [3.7,3.9].

Once the substrates are mated and the index matching fluid applied, the coupler is tuned to the desired coupling ratio. This tuning is a result of varying the overlapping of the evanescent fields in each fiber and is achieved by translating the fibers parallel to themselves. In a typical coupler, maximum coupling is achieved when the cores of the

fibers are superimposed. As the cores are offset, the coupling decreases. However this correlation does not hold if the interaction length is greater than the coupling length defined as $\pi/2c$. In this situation, overcoupling is said to occur and the peak coupling is achieve with an offset of the cores [3.9].

This tunability offers an advantage for polished couplers in developing optical signal processors. This advantage is a result of the tunability allowing the same coupler to be incorporated for multipliers of different magnitudes.

3.2.2 Fused

Fused Biconical Tapered (FBT) couplers have been called the most successful of the singlemode couplers. Their success is attributed to their stability and low-loss operation as well as their ease in manufacture. In addition these components allow an arbitrary branching ratio, reduced polarization dependence and a broad range of wavelength operation [3.8]. Like polished couplers, FBT couplers have their origins with multimode fibers [3.5].

Although attempts have been made to describe the operation of FBT couplers using evanescent field coupling, they have been unsuccessful. This problem results because in the waist of the coupler, the propagating fields becomes detached from the cores of the fiber and are guided in the fiber by the air-cladding interface. This detachment is a necessary result of the manufacturing technique.

FBT couplers are made by placing the two fibers in parallel that have had their coatings removed. The parallel placement of the cores is achieved by twisting the two fibers together. The twisted area is often called the biconical-taper structure or the neck of the coupler. This structure is then attached between two translation stages as heat source is applied and the two stages separated. The result is a symmetric tapering of the two fibers and a fusing of their claddings in the neck of the coupler. Figure 3.4 depicts to fabrication of a FBT coupler.

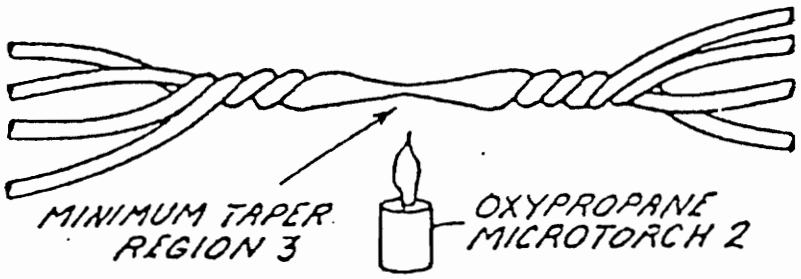


Figure 3.4. Fabrication of a fused biconical tapered coupler using an oxy-butane torch.

By tapering the fiber, the normalized frequency or V-number of the fiber is decreased. V-number is defined by

$$V=\frac{2\pi a}{\lambda}(n_1^2-n_2^2)^{1/2}, \tag{3.3}$$

where 'a' is the core diameter, λ is the wavelength of the light propagating in the fiber and n_1 and n_2 are the indices of refraction for the core and cladding respectively. It can be seen that this tapering decreases the V-number by reducing the radius of the cores.

In the neck area of the coupler, as the light propagates into the tapered area, the mode-field diameter of the guided mode increases as it is loosely bound by the core. This effect is due to the reduction of the V-number. As the two fields become more loosely bound to their respective fiber, they are able in interact and exchange power. The segment of the coupler where power exchange occurs is called the interaction length.

As the light exits the tapered region, the core diameters begin to increase causing an increase in the V-number. As the V-number increases, the light becomes more tightly bound and the mode-field diameter decreases. Thus the interaction and power exchange decreases as the light exits the interaction region.

By varying the interaction length of the coupler, different coupling ratios may be achieved. Unfortunately, the FBT coupler is not as easily tuned as the polished coupler. Tuning usually occurs during manufacture and as a result, a packaged coupler has an invariable coupling ratio and insertion loss. A few techniques have been developed that allow the changing of the coupling ratio without the actual modification of the taper. These techniques usually involve longitudinally stretching the coupling region or inducing a bend or twist in the coupling region [3.11, 3.12, 3.13].

Twist tuning involves the twisting of the coupling region after the coupler is manufactured and allowed to cool. In theory, the introduction of a twist causes a change in the effective index of refraction of the fiber in the coupling region. This concept of effective index is similar to that used in describing bends in optical waveguides [3.14]. This effective index can be shown to be

$$n^2_{eff} = \left[1 + \frac{(x^2 + y^2)}{b^2}\right] n_o^2, \tag{3.4}$$

where $2\pi b$ is the twist pitch, n_0 the effective index distribution and x and y are Cartesian coordinates. It can be seen that the effective index is quadratically dependent of the distance from the axis of the waveguide. This effect causes a redistribution of the mode fields inside the waveguide and changes the power coupled between them [3.15].

While initial limitations in tunability pose as a disadvantage for FBT couplers in developing optical signal processors, the possibility of twist induced tuning overcomes this problem. Additionally, once the proper magnitudes for the multipliers are decided, the ease in manufacture for traditional FBT couplers should make production of optical signal processors more economical.

3.2.3 Asymmetric

Operationally, an asymmetric coupler differs from a symmetric coupler in its transfer matrix. The transfer matrix of an asymmetric coupler is shown in Equation 3.5 where A and B are the coupling coefficients in the different directions and γ_1 and γ_2 are the insertion losses in those directions respectively

$$\begin{pmatrix} P_3 \\ P_4 \end{pmatrix} = \begin{pmatrix} 1 - A\gamma_1 & B\gamma_2 \\ A\gamma_2 & 1 - B\gamma_1 \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix}. \tag{3.5}$$

It is interesting to note that, unlike a symmetric coupler, A and B are independent of each other and can be varied during the manufacturing process of the coupler. This asymmetry

has traditionally found use in local area network applications where it allows for an optical bus with tap-on and tap-off fiber configuration [3.16, 3.17].

Asymmetric couplers can be made using several different techniques. In a polished coupler, the asymmetry may be caused by an unequal polishing of the two fibers. As a result, the two coupling coefficients are a function of the polishing depth of each fiber. Many different methods have been demonstrated for asymmetric FBT couplers. Generally, the coupling coefficients of an asymmetric FBT are controlled by inducing a physical asymmetry of the fiber claddings. Techniques for inducing this asymmetry include pre-tapering or pre-etching one of the fibers before fusing or using two dissimilar fibers. It has also been shown that the coupling ratios are strongly dependent and therefore can be controlled by the degree of fusion of the coupler [3.18, 3.19].

3.3 Junctions

The remaining basic components of a digital filter to be discussed are summers and branchers. The optical equivalents of both branchers or summers are 1xN and Nx1 symmetric couplers. By using couplers which have equal coupling ratios for all branches, a signal may be split or summed evenly with all components. Another alternative is to use only one input or output of an NxN coupler for branchers or summers respectively. However, in using couplers for such devices, it is critical to ensure that the injection losses are equal for all ports as any variance would cause a change in optical power transmission for different branches. The result of any variance would equate to a change in the multipliers used for each branch.

Chapter 4: Filter Configurations

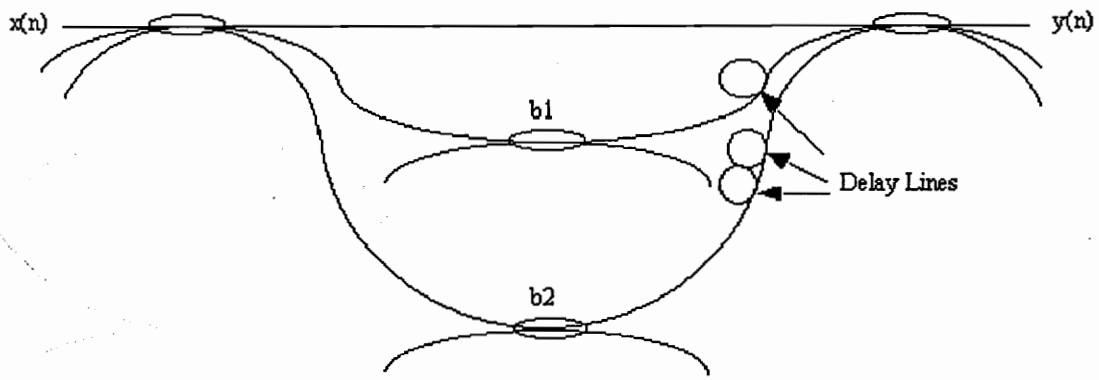
Having established the basic components of every digital filter and their optical equivalents, it is now possible to investigate different configurations of optical signal processors and their performance characteristics. Five different filter configurations will be described and analyzed. In addition, their significance to optical signal processing will be discussed.

4.1 FIR Filter

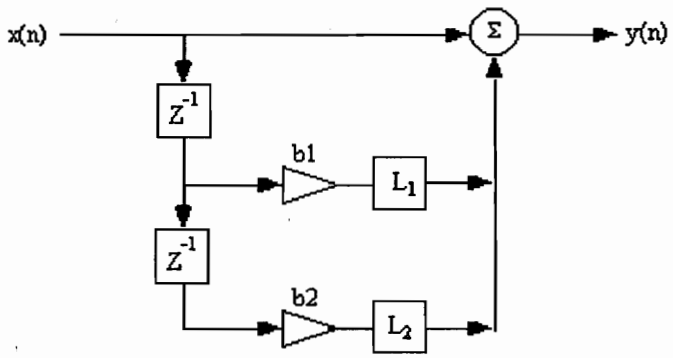
The optical signal processor with the most simple data structure to understand is the FIR filter. This filter is identical to an electronic FIR filter in form and function. A schematic and signal flow graph of an FIR optical filter is shown in Figure 4.1 with the transfer function

$$H(z) = 1 + b_1 L_1 z^{-1} + b_2 L_2 z^{-1}, \tag{4.1}$$

where b_n and L_n are the coupling coefficient and loop losses of the nth coupler. The loop loss is the product of the injection losses of the coupler and the fiber attenuation of the delay.



a)



b)

Figure 4.1. Optical FIR filter. a) Fiber configuration. b) Data structure.

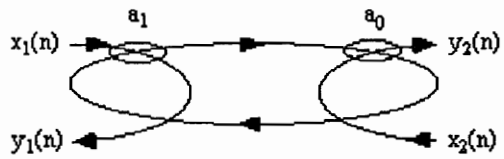
As can be seen from Figure 4.1b, this data structure is identical to those for electronic filters. Because of this structure, the FIR optical signal processor has all the advantages as well as disadvantages of its electronic version. This optical filter has a linear phase

response and an ensured stability due to the absence of any poles. Unfortunately, the drawbacks of FIR filters are also present, specifically, the need for filters of higher order to achieve the equivalent response of lower order IIR filters.

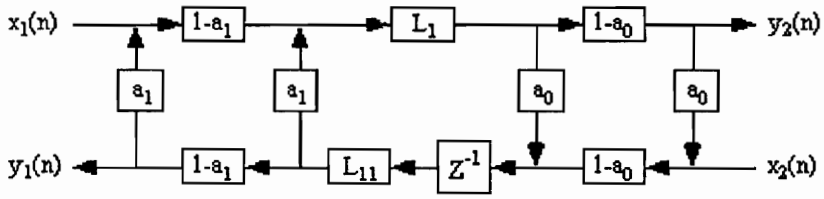
The implementation of optical FIR filters of high order tends to be difficult in practice although some simple versions have been made reported [4.1]. This arises from the number and type of components necessary for this filter. Optical FIR filters require branchers and summers of 1xN or NxN where N is order of the filter. In high order implementations, the optical power in each branch may become too low to retain useful information. In addition, this type of filter is extremely component intensive which leads to high costs.

4.2 Lattice Filter

Another type of optical signal processors that allows for an increased filter order incorporates the use of lattice structures. In contrast to the FIR filter which is an all-zero filter, the lattice filter is an all-pole filter. As mentioned previously, a lattice filter structure involves the cascading of multiple recursive sections. This connection of multiple sections is facilitated by configuring the couplers, as shown in Figure 4.10, to form each lattice section.



(a)



(b)

Figure 4.2. Lattice Filter. a) Fiber configuration. b) Data structure.

As can be seen in Figure 4.2, this configuration is very similar to the lattice structure for electronic filters. As a result, multiple filter sections can be easily cascaded to form a filter with a true feedback structure. The transfer function of the filter using input port 1 and output port 2 is shown in

$$H_{2\ 1}(z) = \frac{(1-a_1)(1-a_0)L_{11}}{1-a_1 a_0 L_1 z^{-1}} \tag{4.2}$$

where L_1 is the fiber attenuation factor from coupler a_1 to coupler a_0 and L_{11} is the attenuation factor of the feedback fiber loop as shown in Figure 4.2. However, due to the lattice structure nature of the filter stage, it is best to describe the transfer function in general terms. By cascading N stages, the transfer function for the output at port 2 becomes [4.2]

$$H_{2,N}(z) = \frac{(1-a_N)H_{2,(N-1)}l_{1N}}{1-a_N H_{1,(N-1)} l_N z^{-1}}, \tag{4.3}$$

where

$$H_{1,N}(z) = \frac{a_N + (1-2a_N)H_{1,(N-1)}l_N z^{-1}}{1 - a_N H_{1,(N-1)} l_N z^{-1}}, \tag{4.4}$$

and the initial parameters are

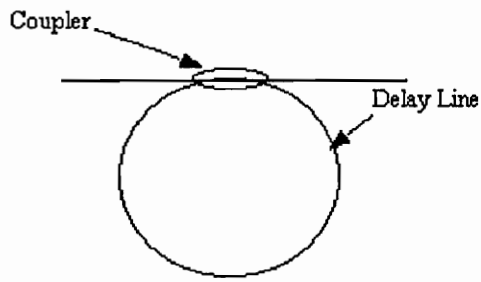
$$H_{1,0} = a_0, \tag{4.5a}$$

$$H_{2,0} = 1 - a_0. \tag{4.5b}$$

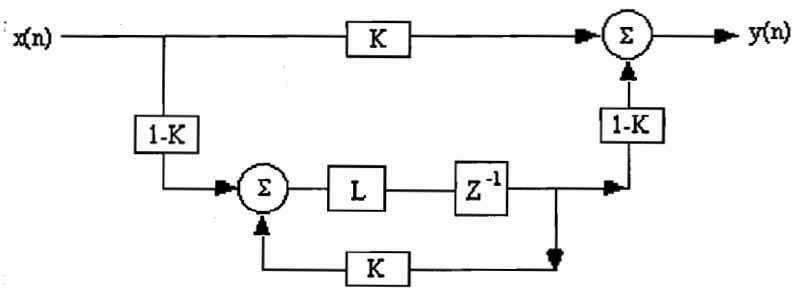
A detailed analysis will show that these filters are autoregressive or all-pole filters and as such can be described using autoregressive modeling.

This autoregressive nature is a result of the configuration of the couplers in each stage. While two couplers are required for each stage, this configuration has versatility not available with other structures. This versatility will be discussed in further detail in Chapter 5.

4.3 Single Loop Filter



(a)



(b)

Figure 4.3. Single loop signal processor. a) Fiber configuration. b) Data structure.

A filter structure that is able to greatly reduce component requirement is the single loop structure. This structure has been traditionally referred to as the recirculating delay line [4.3]. Figure 4.3 shows a schematic and the signal flow graph for this single loop filter.

By simplifying Figure 4.3b, the transfer function can be determined to be

$$H(z) = \frac{K - L(2K-1)z^{-1}}{1 - LKz^{-1}}, \tag{4.6}$$

where K is the coupling coefficient of the coupler and L is filter losses expressed as a loss factor. A loss factor of 1 would indicate no losses in the filter; an ideal case.

This configuration is an IIR filter with poles and zeros defined by

$$p = LK, \tag{4.7}$$

and

$$z = \frac{L(2K-1)}{K} \tag{4.8}$$

respectively. For this filter, the maximum attenuation is achieved when a zero falls on the unit circle. From the previous discussion on single-order filters, the root of this filter falls on the negative real axis and therefore can only have a magnitude of unity at -1. This requires that the coupling ratio be 1/3 for an ideal coupler. Figure 4.4 shows the magnitude response for such a filter which was generated by plotting the output of the Matlab command [h,w]=singfab(1/3,1).

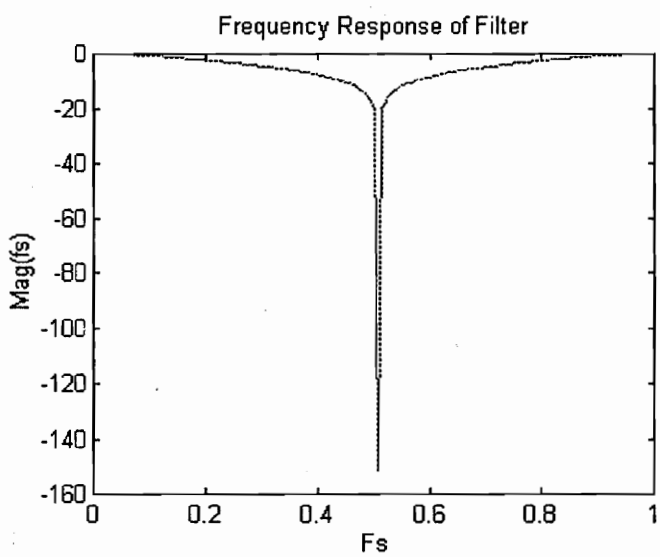
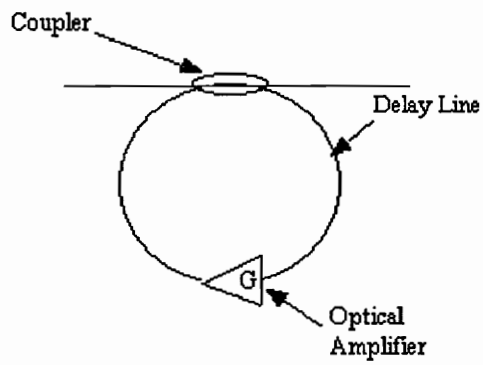


Figure 4.4. Magnitude response of an ideal filter with K=1/3.

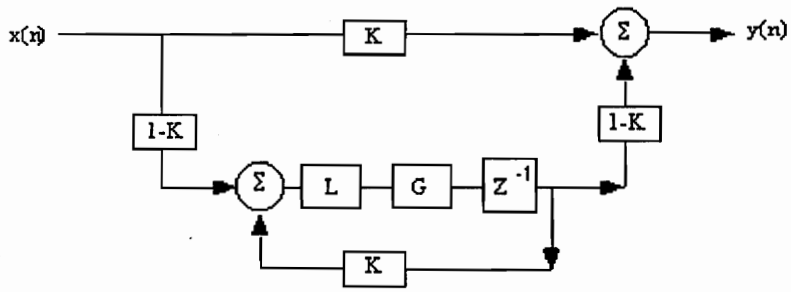
4.4 Gain Amplified Filter

With single loop filters, it was shown that a sharp transition is achievable by placing a zero directly on the unit circle. A steeper transition could be obtained if in addition to the zero, a pole could be placed near the unit circle at +1. However, this is not possible in a single loop system as a single coupling coefficient is not able to satisfy both (4.7) and (4.8). By the addition of gain in the loop, such a response is possible [4.3].

Figure 4.5 shows a single loop filter with an amplifier in the loop and its corresponding signal flow graph.



(a)



(b)

Figure 4.5. Gain Amplified Filter. a) Fiber configuration. b) Data structure.

While the two structures appear to be very similar, the addition of an optical amplifier adds a very significant degree of freedom without increasing the order of the filter. This can be seen by examining the transfer function of the filter given as

$$H(z) = \frac{K - LG(2K-1)z^{-1}}{1 - LGKz^{-1}}. \tag{4.9}$$

Solving for the roots of the equation we obtain

$$z = \frac{LG(2K-1)}{K}, \tag{4.10}$$

and

$$p = LGK. \tag{4.11}$$

By adding the amplifier it can be easily shown that pole-zero placements are feasible that would not be otherwise possible.

Intuitively, it can be seen that the addition of gain affects the filter by allowing for more recirculations of the optical signal in the loop before it is attenuated by losses to the noise floor. This effect can be seen as the gain presents itself in the transfer function with the loop losses and can therefore cancel the losses to present an ideal single loop filter with a non-ideal coupler. By extending the concept further than loss compensation, the amplifier is able to increase the number of recirculations in the loop past the limits imposed by the

tap off power budget of the loop. This effect is then able to cause frequency responses normally unrealizable due to power constraints.

In addition to a change in frequency response, stability effects can be seen in certain regions as the gain is increased. As mentioned previously, the affect of loop losses is to contract the pole-zero locations. The inclusion of gain is able to cause the opposite effect and cause the root locations to expand radially from the origin. This expansion however can lead to problems not previously encountered. Since the stability criteria of filters is to have the poles inside the unit circle, excessive amplification can lead to poles expanding past the unit circle driving the filter to become unstable. This undesirable effect that can render the filter useless and should be avoided. Figure 4.6 shows the frequency response with the coupling coefficient set to 0.40 and the gain equal to 2.5.

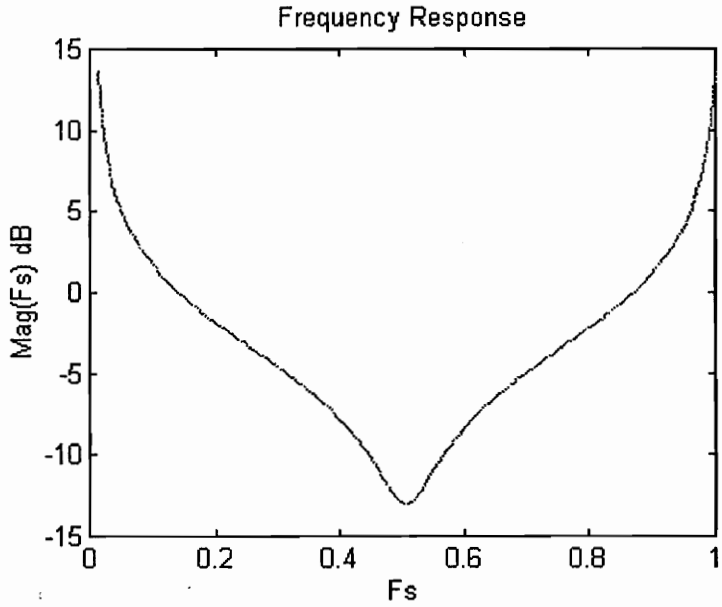


Figure 4.6. Frequency response of single loop filter with G=2.0.

In the region around Fs, the filter has a +14 dB amplification of the signal, which is the result of the pole on, or near, the unit circle. The pole-zero plot for this filter in Figure 4.7 confirms this conclusion.

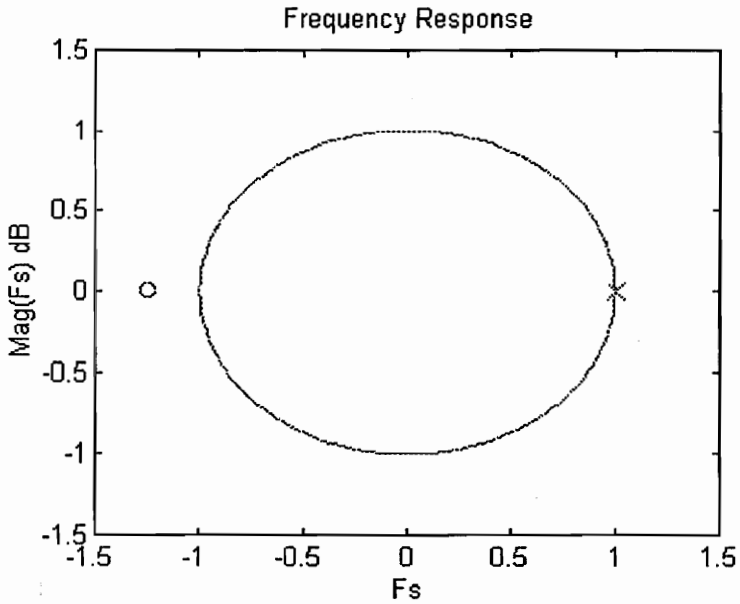


Figure 4.7. Pole-zero plot of filter in Figure 4.5.

One consideration unique to an amplified loop configuration is the effect of amplified spontaneous emission noise. This noise is the result of the optical amplifier when excited ions spontaneously relax to their ground state. These spontaneous emissions contribute random increases in the electromagnetic energy in the loop and are amplified with the signal [4.4].

This noise is analogous to the gaussian noise in electronic filters. As such, they can be modeled similarly by adding injection noise in the system. The logical point in the system to add this noise is in the loop as depicted in Figure 4.8.

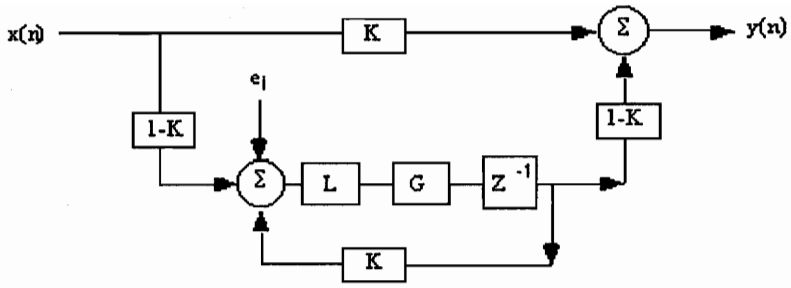
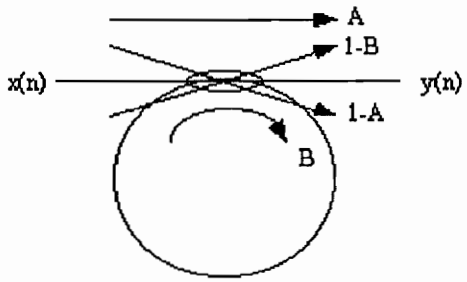


Figure 4.8. Amplified loop with noise injected. a) Fiber configuration. b) Data structure.

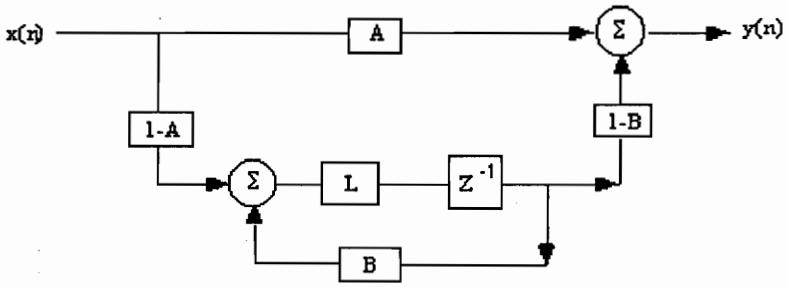
4.5 Asymmetric Coupler Filter

It has been shown that the use of an optical amplifier in the loop of an optical signal processor enables frequency responses not otherwise achievable. However, the current costs associated with these amplifiers prohibits their widespread use. Current Erbium doped fiber amplifiers (EDFAs) add approximately $20,000 to system costs. It can be shown that similar results may be attained by using an asymmetric coupler in the filter. While these devices are not as common place as their symmetric counterparts, they are available and can be manufactured at considerable lower costs that EDFAs. In addition a passive device will have no power consumption and high dependability.

In signal processors, asymmetric couplers can replace the symmetric coupler in a single loop configuration. This asymmetry results in a filter loop which has different coupling ratios for signals coming in to the loop that for signals exiting the loop. This filter schematic is shown in Figure 4.9 with a signal flow graph where A is the coupling coefficient for the tap fiber and B is the coupling coefficient for the bus fiber.



(a)



(b)

Figure 4.9. Asymmetric Coupler Filter. a) Fiber configuration. b) Data structure.

By enclosing the bus fibers in the coupler, to form the loop, the tap off ratio can be lower than the tap-on ratio. The net result is a filter that allows an increased number of recirculations in the loop before the signal is attenuated to the noise floor. This effect is similar to that with single loop filter using an amplifier in the loop and as a result, could allow for different filter responses.

This similarity can be seen by examining the transfer function of the filter

$$H(z) = \frac{A - L(B + A - 1)z^{-1}}{1 - LBz^{-1}}. \tag{4.12}$$

Solving for the poles and zeros we have

$$z = \frac{(B + A - 1)}{A}, \tag{4.13}$$

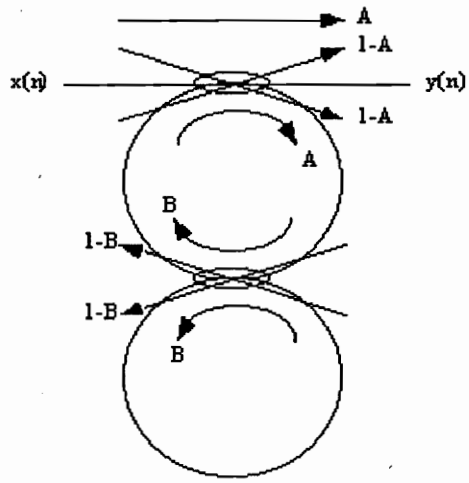
and

$$p = LB. \tag{4.14}$$

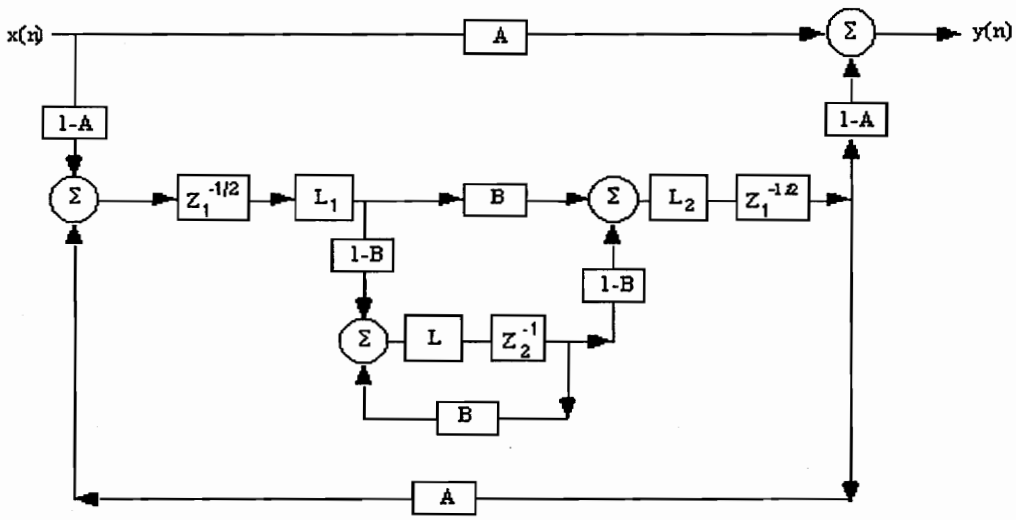
As the equations indicate, the location of the poles are placed solely as a function of B, while the zeros are determined by A and B. In addition, since rules of all positive systems applies, the poles may only be located on the positive real axis within the unit circle while the zeros may fall anywhere on the real axis.

4.6 Double Loop Filter

All of the recursive filters discussed so far have consisted of the single-loop filter structure with some modifications. Unfortunately, none of these modifications have increased the order of the filter. As a result, root placement has been limited by mathematical principles that restrict the roots of first order equations. Although it is possible to cascade multiple loop stages to form higher order filters with various frequency responses, their performance is restricted. This restriction is due to the inability of the feedback of one filter stage to interact with another. This restriction can be overcome to some degree with a two loop system. Figure 4.10 shows the double loop filter configuration.



(a)



(b)

Figure 4.10. Double Loop Filter. a) Fiber configuration. b) Data structure.

The transfer function of the filter structure is

$$H(z) = \frac{A + L_1L_2(1-2A)Bz_1^{-1} - ALz_2^{-1} + L_1L_2L[(1-2B)(1-2A)]z_1^{-1}z_2^{-1}}{1 + ABL_1L_2z_1^{-1} - LBz_2^{-1} - AL_1L_2L(1-2B)z_1^{-1}z_2^{-1}} \quad (4.15)$$

where L1 and L2 are the fiber attenuation factors in the first and second half of the top loop respectively, and L is the fiber attenuation factor of the bottom loop. In (4.15), z_1^{-1} and z_2^{-1} are the delays from the top loop and bottom loop respectively. It is important to differentiate between the delays created by the individual loops. This need is due to an inconsistency in using the Z-transform with this particular configuration. Without making this distinction, extreme coupling coefficients of 0 and 1 cause an incorrect transfer function. For example, ignoring the loss terms and combining the delays, the numerator is

$$B(z) = A + [(1-2A)B - A]z^{-1} + [(1-2B)(1-2A)]z^{-2}. \tag{4.16}$$

Taking B to equal 1, the equation becomes

$$B(z) = A + (1-3A)z^{-1} + [-(1-2A)]z^{-2}. \tag{4.17}$$

However, a coupling coefficient of B=1 physically means that all the optical power is coupled to the loop and no light enters the lower loop. As a result, the lower loop essentially does not exist and $z_2^{-1}=0$. The correct expression for the numerator in (4.15) is then

$$B(z) = A + [(1-2A)]z^{-1}, \tag{4.18}$$

which corresponds to the solution found in (4.10). The same discrepancy exists at the other extremes where B=0 and A=1 or 0.

The ability of the double loop structure to implement second order IIR filters leads one to believe that third order systems could be created by adding another delay line. This third order configuration is entirely possible but an investigation of these higher-order systems is beyond the scope of this discussion. Systems have also been implemented that incorporate previously discussed concepts to create a two loop filter configuration with optical amplifiers in each fiber segment to construct filters with great flexibility [4.5].

Chapter 5: Computer Modeling

The implementation of optical signal processors compter model may help to facilitate their design. Multi-dimensional models will be presented which simulate the performance of the various filter configurations previously described in Chapter 4. The use of these models can help the designer to understand the capabilities and limitations of these optical signal processor.

5.1 FIR Filter

Since an FIR filter is easily implemented in higher orders, a random analysis can quickly be performed to describe possible root placements for the transfer function of the system. Since an FIR filter consists of only zeros, their placement is responsible for the magnitude response of the filter. Figures 5.1(a)-(f) maps some possible zero locations in the Z-Plane for optical FIR filters of order 1 through 6.

From these maps it is seen that the zeros for these types of filters occur in certain discrete areas and that these areas are near the unit circle. These plots clearly show the limitations imposed on the root placement of optical signal processors as a result of their all positive nature. This limitation is analogous to the quantization effect observed in traditional electronic digital signal processors. By knowing these quantization effects and being able to model them, greater insight is gained in understanding them.

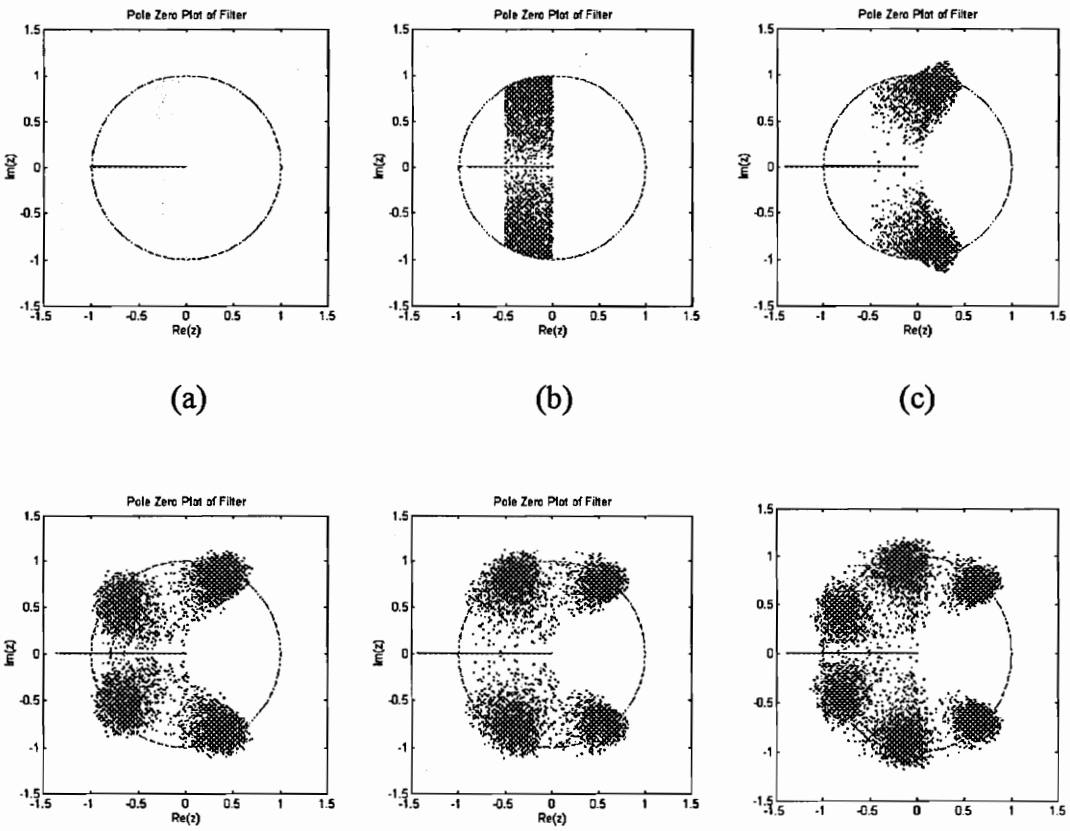


Figure 5.1. Possible zero locations in the Z-plane for optical FIR filters of order 1-6 respectively.

The plots were generated in Matlab using the command [z]=randfir (2000). This command calls upon the RANDFIR.M function and randomly generates 2000 sets of possible coupling coefficients for each multiplier. The roots of the resulting transfer function are then calculated and plotted.

As can be seen from Figure 5.1, the roots of the transfer function tend to cluster in certain areas. For the first order filter, the roots are found on the negative real axis. The zeros of second order filters have a magnitude less with a real component between -0.5 and 0. In addition, since the roots of the filter are complex, the zeros occur in complex conjugate pairs. The remaining plots show the maps for higher order filters. This model assumes no loss in the system. Inclusion of coupler insertion losses and fiber attenuation would alter the maps by contracting the patterns of the maps.

5.2 Lattice Structure Filter

Due to their autoregressive nature, the pole placements are solely responsible for the frequency response of the filter. As with the FIR filter, a random analysis enables one to see the possible filter responses. Figure 5.2 shows the quantization effects of this filter.

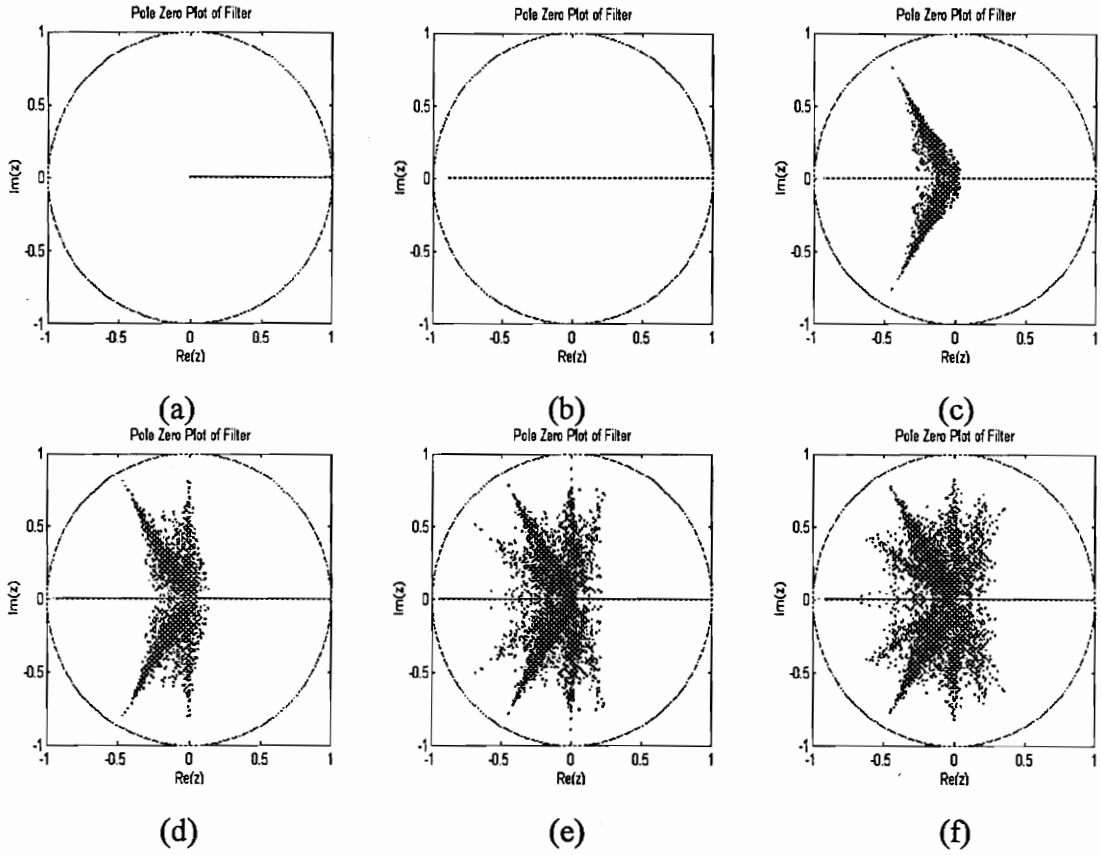


Figure 5.2. Possible pole locations in the Z-Plane for optical lattice filters of order 1-6 respectively.

These plots were generated using the Matlab subroutines RECIRH21, RECIRH22, RECIRH23, RECIRH24, RECIRH25 and RECIRH26 respectively with 5000 iterations. These routines are similar to the RANDFIR.M function in that it randomly generates 5000 sets of possible coupling coefficients for each multiplier and plots the poles of the resulting transfer function.

As with the FIR case there is a distinct pattern to the possible placements for the poles. This pattern corresponds to branches centered about $\pm 2\pi/N$ and integer multiples where N is the order of the filter. This quantization leads to the conclusion that for such a filter

structure it is not possible to generate highpass filters. However by cascading sections, to form higher order filters, the attenuation and cutoff frequencies are easily dictated.

5.3 Single Loop Filter

This result can also be verified by plotting the magnitude response vs. frequency for various coupling coefficients. Figure 5.3 shows a three dimensional plot with an additional contour plot which varies these parameters. The plot was generated using the SING.M subroutine. As can be seen, the maximum attenuation is obtained at 0.50Fs with a coupling ratio of approximately 1/3. This result correlates with the results in Chapter 4.

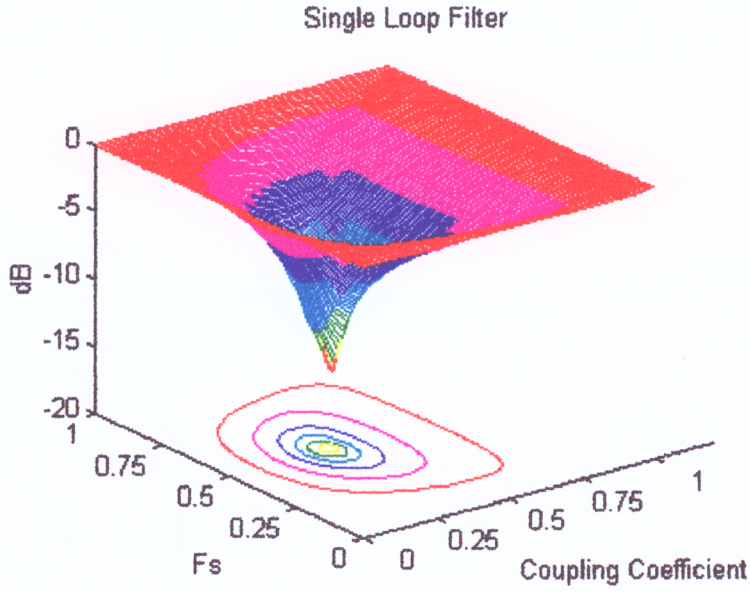
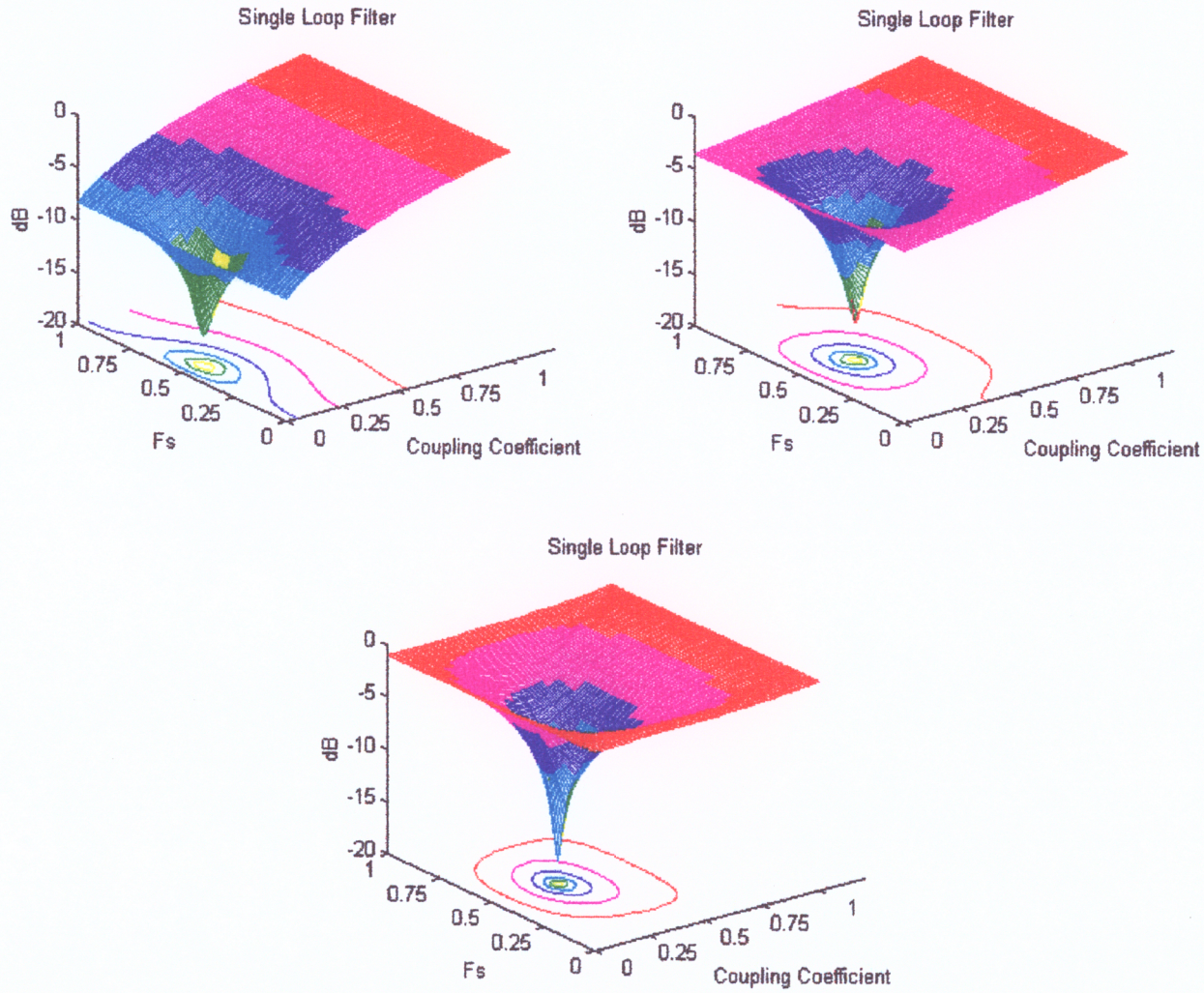


Figure 5.3. Magnitude vs. frequency vs. coupling coefficient for a single loop filter.

However for non-ideal filters, similar frequency responses are still possible as the varying loop losses and coupling coefficients compensate each other. In order to investigate these instances, models must be constructed that are able to simulate the variation of all the

parameters. These models should show the magnitude response as a function of frequency, coupling coefficient and loop losses. Such a four-dimensional model would be difficult to portray effectively in a single plot, but may be visualized using a three dimensional plot which varies in time. This technique would allow three of the variables to be shown as the fourth varies over time.

The Matlab subroutine MOVSING.M generates this four dimensional model as a plot of magnitude vs. frequency vs. coupling ratio and loop losses varied over time. Three of those frames are shown in Figure 5.4 for loss factors of 0.15, 0.45 and 0.80 respectively.



(c)

Figure 5.4. Single loop filter responses for loss factors of a) 0.15, b) 0.45 and c) 0.80 respectively.

By comparing these frames much information can be obtained. Examination of the zero coupling region shows that the frequency response is flat regardless of the loop losses. This result corresponds well with theory as no coupling means that the signal only propagates through the fiber and there is no interference of the modulation. In addition, as loop losses decrease, the attenuation decreases. This effect is also predicted by theory since injection losses of a filter with no coupling results in a pure loss for light propagating through the fiber.

Figure 5.4 also shows the compensation for loss that the non-ideal filter is able to achieve by shifting the target coupling ratio from 1/3. Specifically, Figure 5.4b shows that even with greater than 3dB loss in the loop, a -17 dB response is achievable by using a coupling ratio of approximately 0.25. The relationship between coupling ratio and injection losses for optimum filter performance is best described by Figure 5.5. By assuming that the maximum attenuation is achieved at 0.50 Fs, a plot of amplitude at 0.50 Fs vs. coupling ratio vs. loop loss is generated.

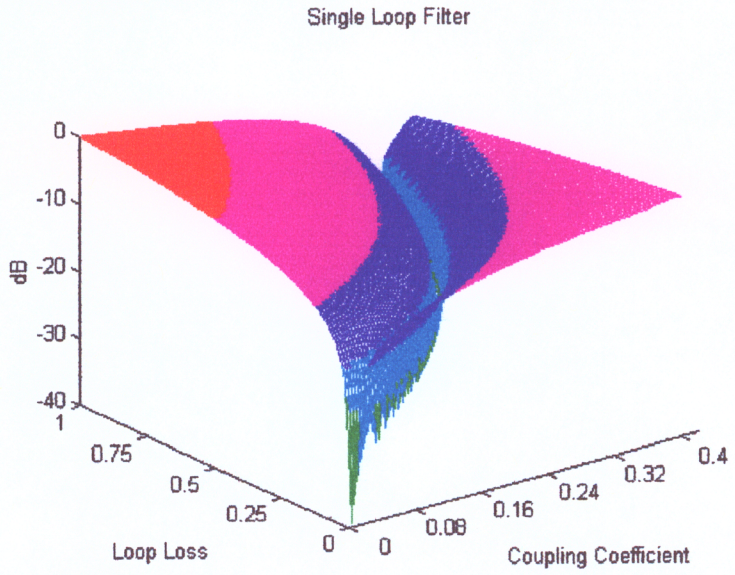


Figure 5.5. Amplitude at 0.50 Fs vs. coupling ratio vs. loop loss.

The minimum response at 0.50Fs for each coupling ratio-loop loss combination can be better described using the contour lines shown in Figure 5.6. This plot of the K-L plane shows the pattern of stopband attenuation depth.

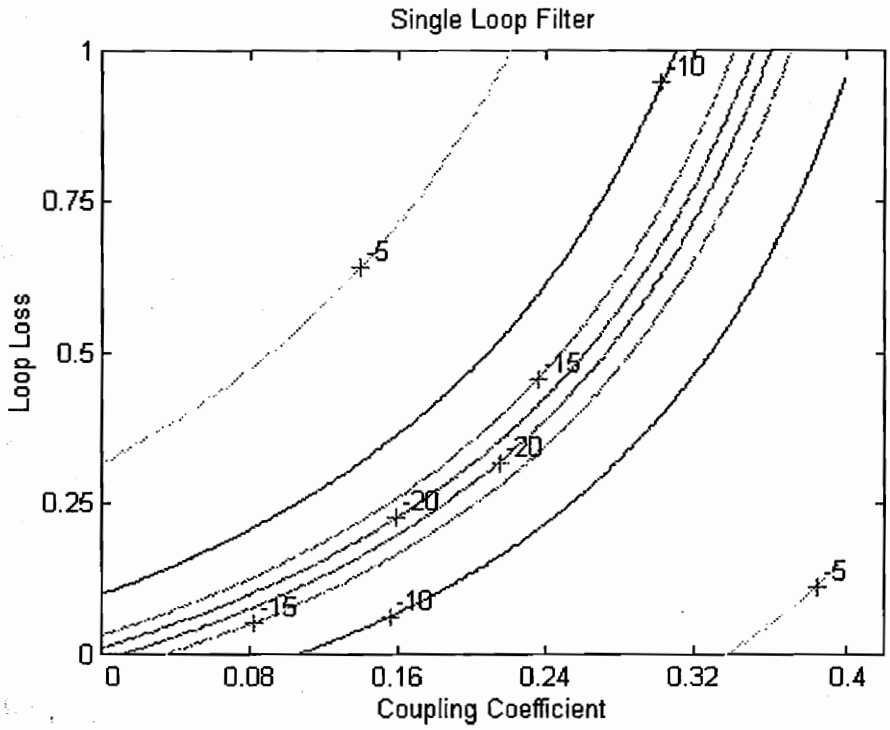


Figure 5.6. Interpolated coupling ratio-loop loss combinations.

As can be seen in Figure 5.6, the maximum attenuation is achieved when the loop loss/coupling coefficient combination falls between the -20dB contour lines. This ability to offset coupler losses with coupling coefficient gives great latitude in the manufacturing process and also allows a versatility in the filters that will be exploited later.

5.4 Gain Amplified Filter

The flexibility previously mentioned that an optical amplifier provides on the frequency response of the filter can be demonstrated with a model that shows the magnitude response vs. frequency vs. coupling ratio. While in the single loop configuration loss was varied in time, gain is now the variable of temporal change. Figure 5.7 shows frames of a sequence generated by MOVGAIN.M. Since any loop losses in this filter can be absorbed in the gain, they are assumed to be zero for this model.

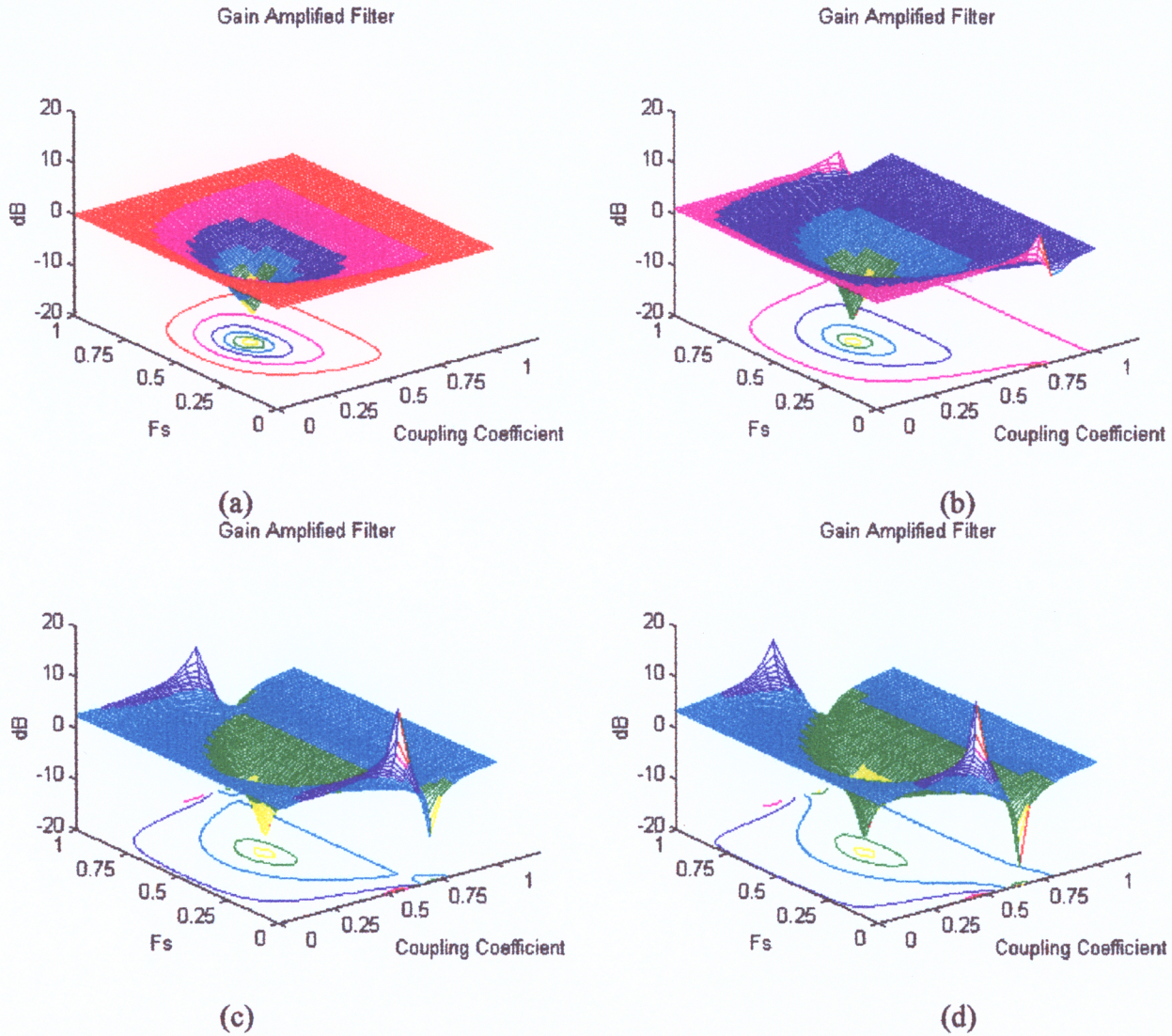
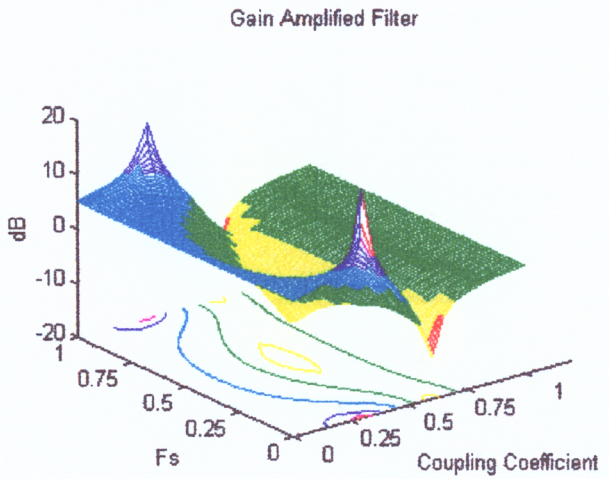
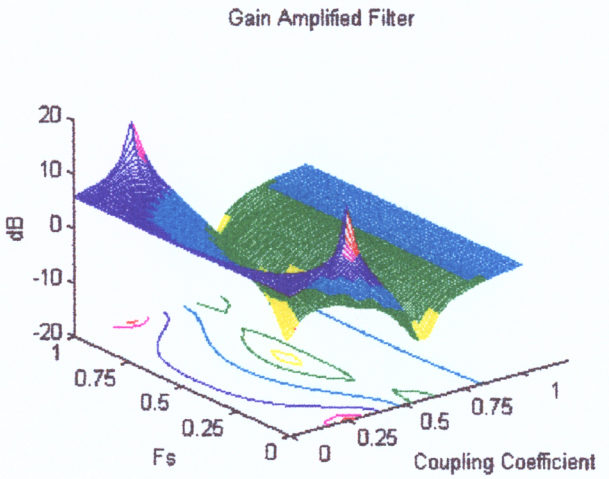


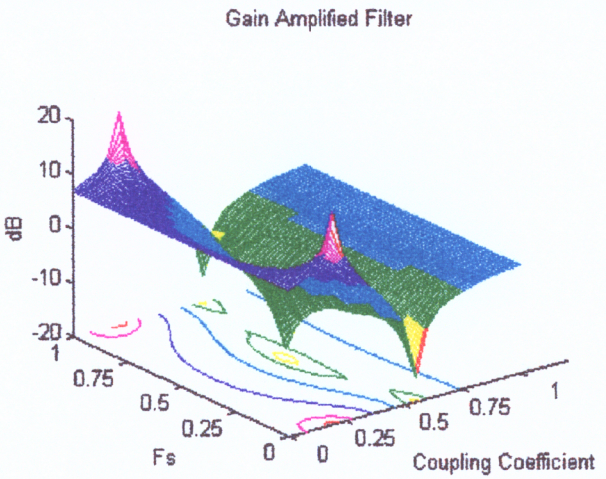
Figure 5.7. Gain Amplified Filter responses with G=a) 1.00, b) 1.30, c) 1.80 and d) 2.20 respectively.



(e)



(f)



(g)

Figure 5.7. Gain Amplified Filter responses with G= e) 3.30, f) 3.90 and g) 5.00 respectively.

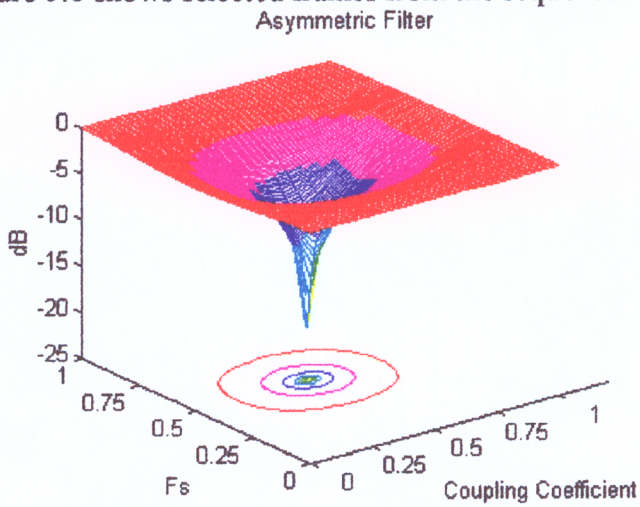
Frame (a) shows the response with G=1.00. As expected, this response is identical to the frequency response of the ideal single loop filter. Frames (b)-(g) show the response for gain equal to 1.30, 1.80, 2.20, 3.30, 3.90 and 5.00 respectively. As can be seen, the filter response becomes quite different from the single loop filter as the gain is increased.

In particular, it is interesting to note that with this filter configuration, multiple types of filter responses are possible. As depicted in Figure 5.7d for example, a lowpass filter results from a coupling coefficient of 0.40. However, with the same insertion losses, at K=0.75 a highpass can be constructed. Not only is this interesting because two different type of filter responses are possible with the same device, but also that one of the possibilities is a highpass filter. This response again shows the flexibility that this filter configuration allows.

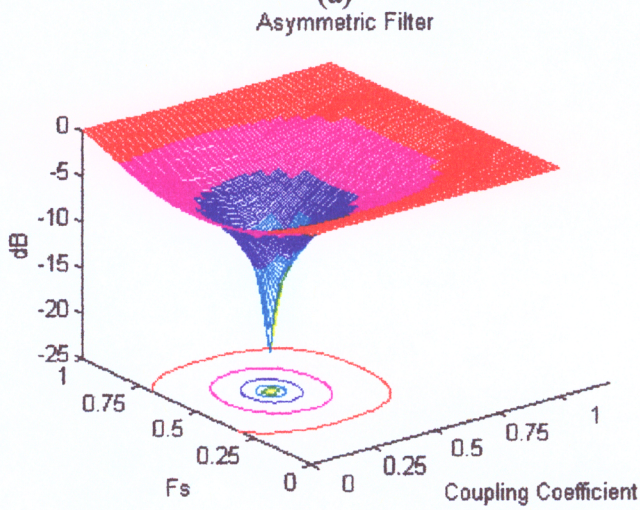
As mentioned in Chapter 4, instabilities in the filter arise from the incorporation of gain. It can be seen in the sequence of frames that the instabilities begin to manifest themselves with gain as small as G=1.30 in the higher coupling coefficients, and as the gain increases, the instabilities appear with lower coupling coefficients. Intuitively, it can be seen that this result is due to the high coupling coefficient causing a large amount of optical power to be coupled into the loop where it is amplified resulting in the instability. As the amplification is increased, lower coupling coefficients produce the same result. In reality however, this effect is not as detremental to the system as may be thought as the optical amplifier would quickly reach saturation halting any great catastrophes.

5.5 Asymmetric Coupler Filter

The effect of the asymmetric coupler on frequency response is best demonstrated by the Matlab subroutine MOVASYM.M which shows a three dimensional plot of magnitude vs. frequency vs. tap on ratio A for an ideal asymmetric coupler. The tap off ratio, B, is varied in time. Figure 5.8 shows selected frames from the sequence.



(a)



(b)

Figure 5.7. Asymmetric filter response for B= a) 0.09 and b)0.39 respectively.

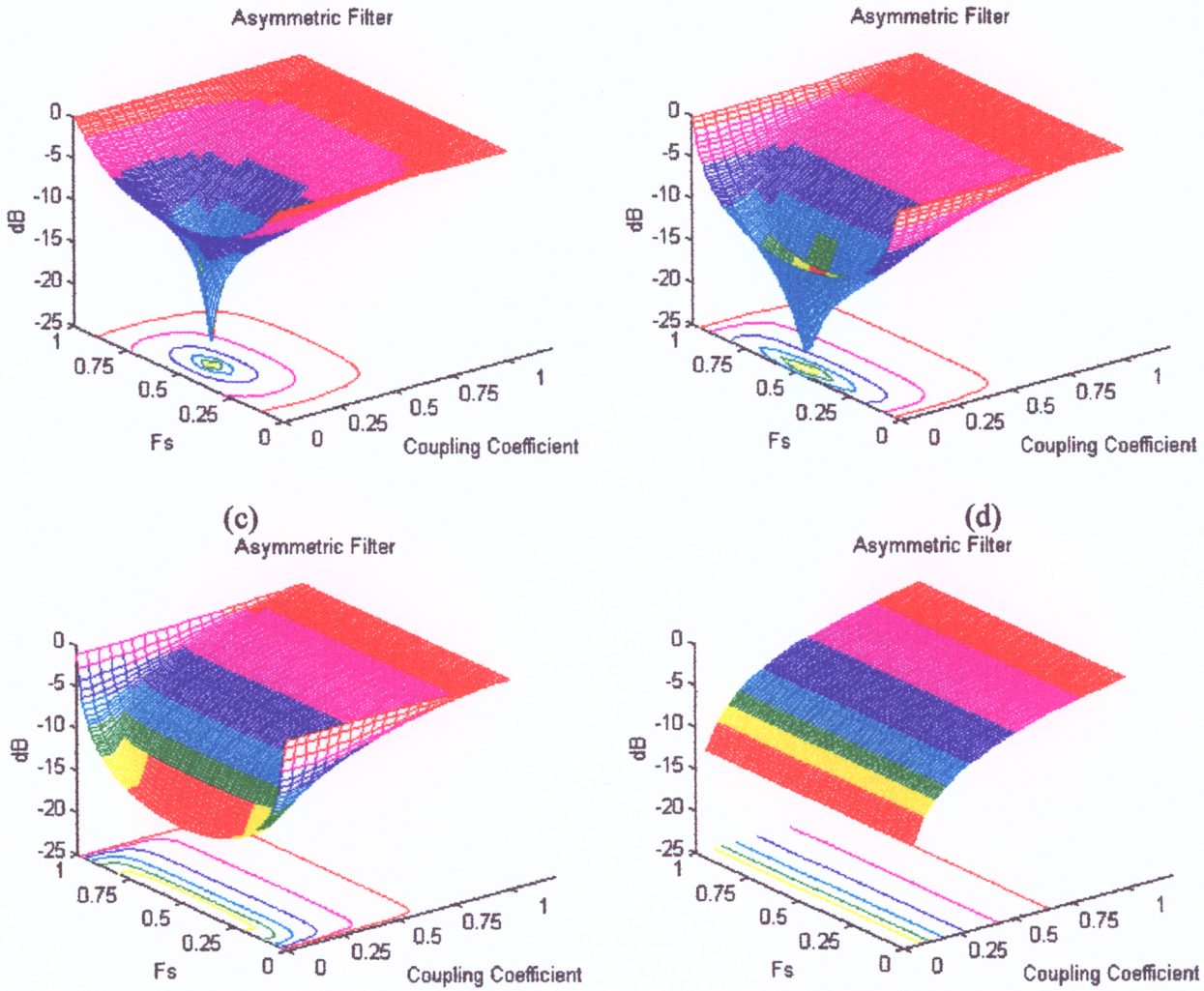


Figure 5.7. Asymmetric filter response for B= c) 0.69, d) 0.89, e) 0.95, and f) 1.00 respectively.

As can be seen from these frames, the substitution of an asymmetric coupler allows for a different combination of possible frequency responses for the filter. In Figure 5.7, frame (a) shows that for low B, the responses are similar to a single loop configuration. However frames (c) and (d) show that as B becomes larger, the area of maximum attenuation migrates toward the lower tap on ratios. This migration allows lowpass filters with relatively narrow passbands at low tap on ratios while at higher tap on ratios the lowpass filters similar to that achieved in the single loop configuration are achieved. In frame (e), as B increases even larger, this narrow band lowpass effect becomes even more significant for low tap on ratios until this type of response dominates. At the extreme, as approaches B=1.00, frame (f) shows that the filter basically functions as an signal attenuator for all frequency components.

From these frames it can be concluded that as the difference between B and A increases, the similarities between the asymmetric configuration and the single loop configuration decrease. This trend is caused because the degree of asymmetry in the coupler increases. Conversely, in regions where the deference in coupling ratios is small, the magnitude response of the filter is similar to the single loop configuration.

5.6 Double Loop Filter

As mentioned previously, the double loop configuration allows for more possible placements of the roots due to their second order nature. Figure 5.9 shows a random analysis of the possible pole placements. Notice that while the gain and asymmetric filter

configurations only allowed for the poles to be located on the positive real axis, this configuration allows for the poles to be placed on the complete real axis. This random analysis was conducted on an ideal filter over 5000 iterations by the Matlab subroutine RANDTLOO.M.

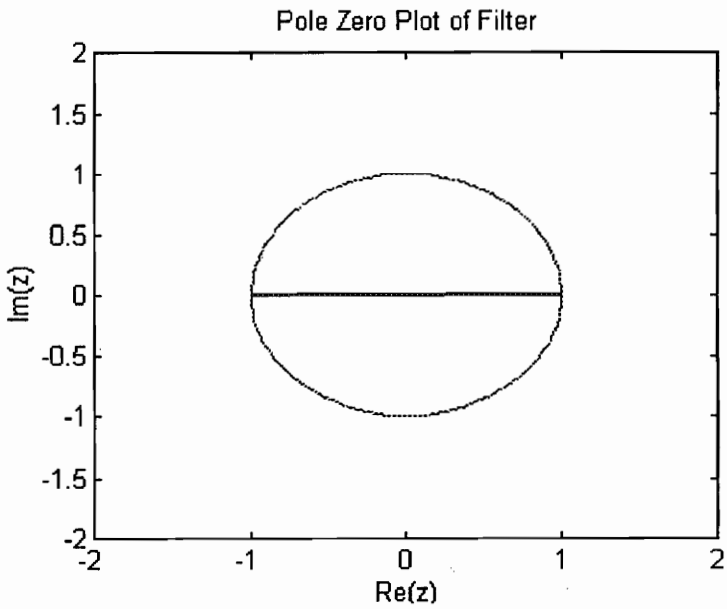


Figure 5.9. Random analysis of the possible pole locations for the ideal double loop filter.

As a result of the increased possible number of pole placements, the variety of possible frequency responses is increased. As with the previous configurations, these possible frequency responses are best demonstrated with a sequence of three dimensional figures. Figure 5.10 shows such a sequence, where B is varied in time, of Magnitude vs. Frequency vs. A. This sequence was generated using the MOVTLOOP.M subroutine.

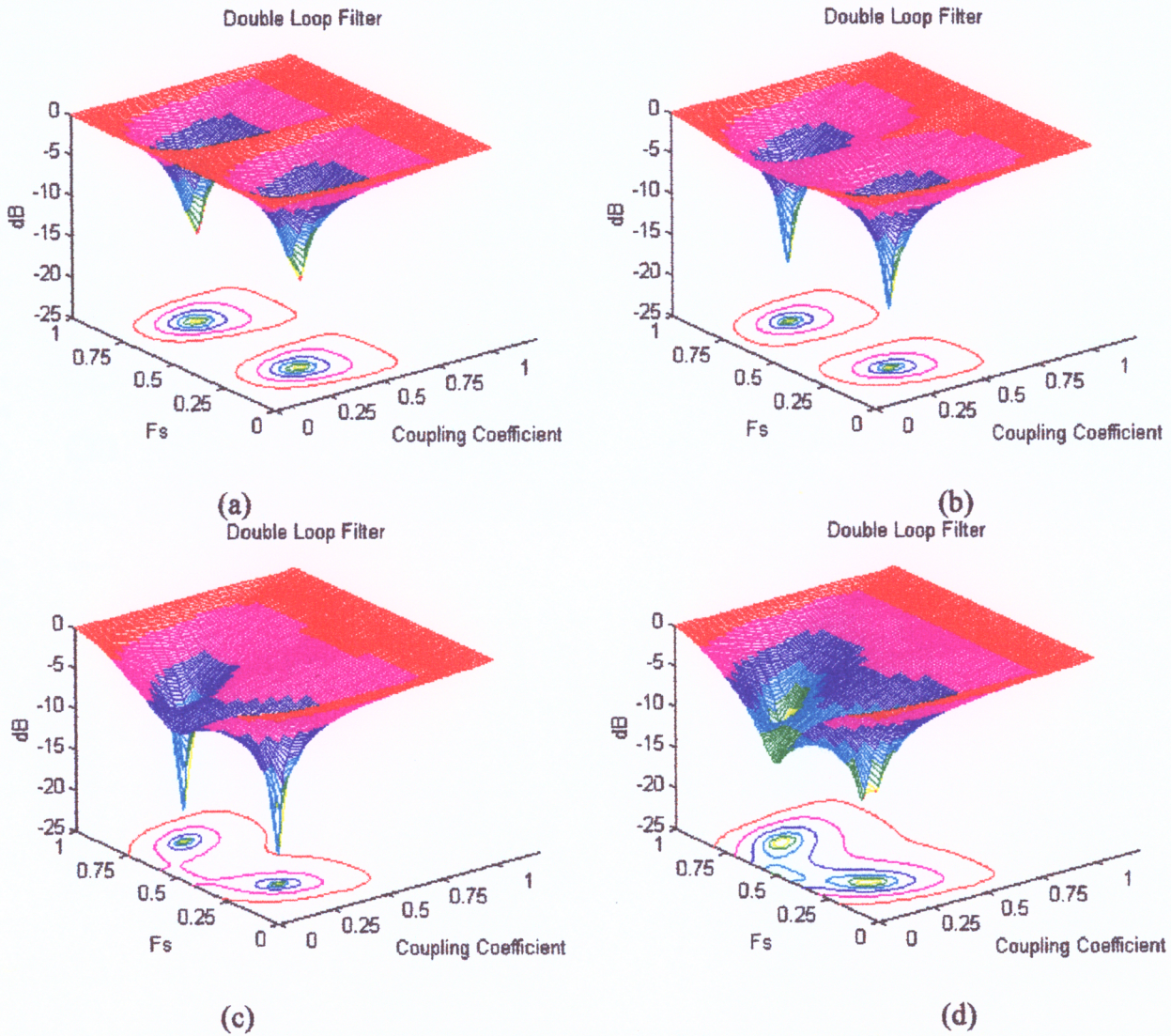


Figure 5.10. Double loop configuration with a) B≅0, b) B=0.10, c) 0.25 and d) 0.30 respectively.

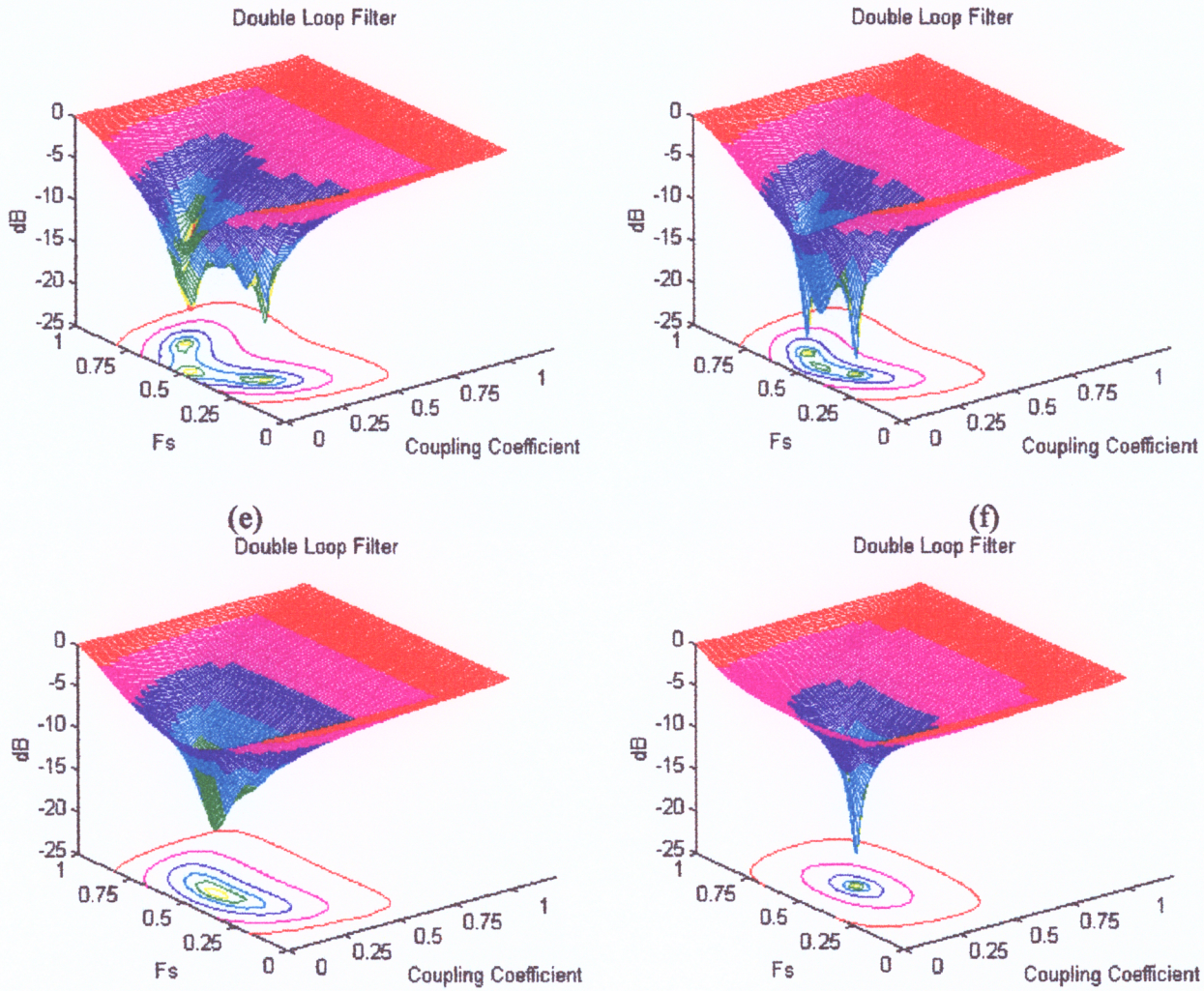
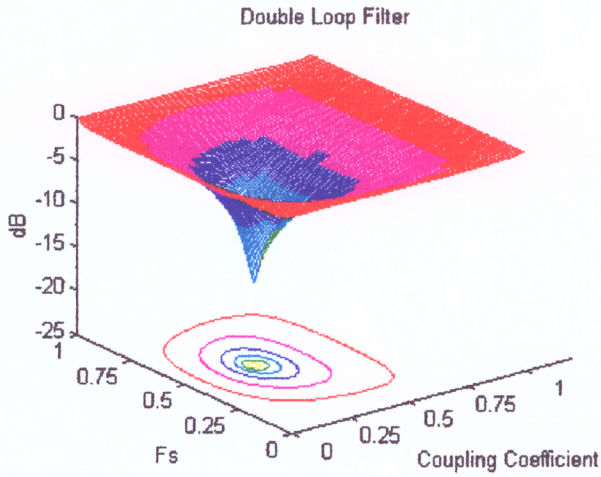


Figure 5.10. Double loop configuration with e) B=0.35, f) 0.40, g) 0.45 and h) 0.59



(i)

Figure 5.10. Double loop configuration with i) B=0.85.

As can be seen in Figure 5.10 frame (a), when B is near zero, the filter exhibits a notch filter response near 0.25Fs. This notch tends to migrate toward 0.50Fs as B is increased until it approaches 0.50 where it then exhibits the familiar lowpass response of the single loop filter as shown in frames (h) and (i). However as with the gain amplified filter, different filtering characteristics prevail at different coupling coefficients. These characteristics are especially significant in the range of B=0.30-0.40 as shown in frames (d)-(f). For example, in frame (d), for A near zero, a lowpass filter is exhibited while for A near 0.25, a notch filter is possible. Frame (e) shows that for a B=0.35 it is actually possible to obtain lowpass filters with a cutoff near either 0.30Fs or 0.50Fs. Both of these filters are capable of approximately -15dB attenuation. Frame (f) shows how these cutoff frequencies continue to move toward 0.50Fs as B is increases. Past B=0.50 however, the filter begins to exhibit the familiar lowpass response that has been seen many times.

These effects that increasing B causes can be reasoned by qualitatively examining the significance of B. In the double loop configuration, B dictates the optical power circulating in the second loop. As B is increased, less of the optical power is coupled into the second loop. As a result, the second order nature of the filter becomes less

pronounced. As a result, as B is increased, the filter response becomes more like the first order single loop filter. It is therefore evident that to fully exploit the advantages of the double loop configuration, couplers with a low coupling coefficient should be chosen for the second loop.

Chapter 6: Experiments

Having described optical signal processors as well as presented computer models of various filter configurations, it is necessary to provide experimental verification of the underlying theory. In the following chapter, experimental data will be provided to verify the basic theory. In addition, experimental verification of the ability for twist-induced tuning in fused-biconical tapered couplers to provide flexibility for the filters not previously possible will be presented. Lastly possible future experiments and directions will be presented that will allow further exploration and application of the optical signal processors.

6.1 Single loop filter

As most of the models presented are an adaptation of the single loop filter, it is important to verify the underlying concepts upon which the derivations are based. To verify these

concepts, a single loop filter was fabricated using the fused biconical taper method and its magnitude response compared with theory. The experimental setup is depicted in Figure 6.1.

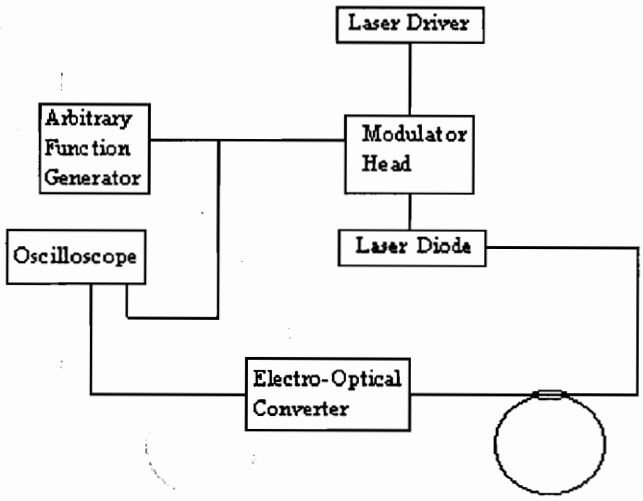


Figure 6.1. Equipment setup for experiments.

The electro-optical converter was a Textronix OR502 E/O Converter plug-in that is specified by the manufacturer for operation to 1.5GHz.

In this setup, the modulator was the AVX-SRB modulator head from Avtech Electronics. This modulator head accepts an RF waveform and superimposes this on the diode driver current. To eliminate a potential source of error, the frequency response of the laser system was measured. This measurement was performed by coupling the output of the laser diode directly into the E-O converter and normalizing the measured output power with the input power. The normalized frequency response of the laser system is plotted in Figure 6.2

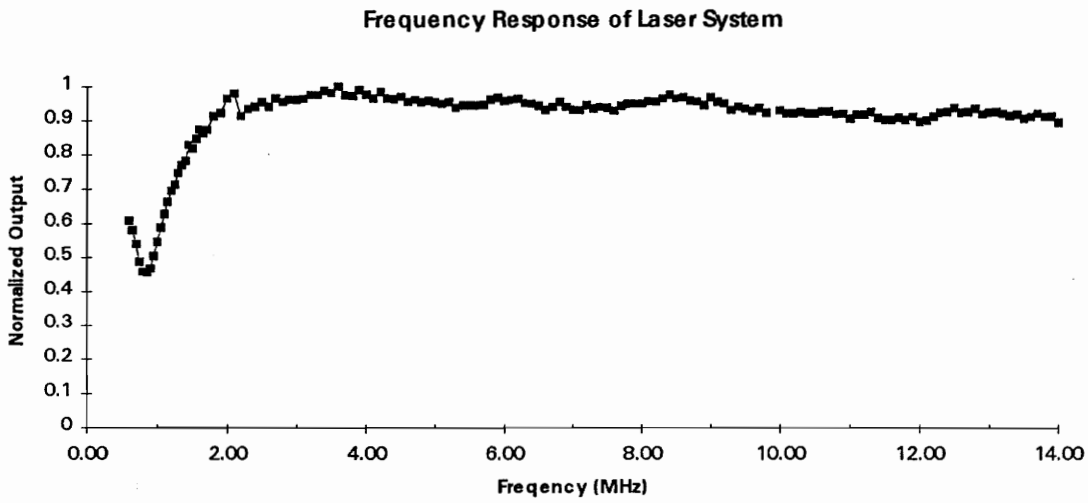


Figure 6.2. Frequency response of the laser system.

While the modulator head is specified by the manufacturer for frequencies of 10MHz to 1.5 GHz, it was found to have a relatively flat frequency response until 2 MHz. Below 2 MHz however, the frequency response of the system appeared to fluctuate. Therefore, the lower frequency cutoff was set to 2 MHz.

The single loop filter was manufactured by enclosing a continuous length of Corning SMF-28™ single mode at 1300nm fiber upon itself. This loop was then placed on a FBT coupler manufacturing stage. The frequency response of the filter was monitored during manufacture by modulating the laser with a square wave of 1.1 MHz. The first three harmonics of the square wave occurred at approximately 1.1, 3.3,and 5.5 MHz respectively. By monitoring the ratio of each harmonic relative to the others the filter response can be estimated. Figure 6.3 shows the FFT of the input and output waveforms.

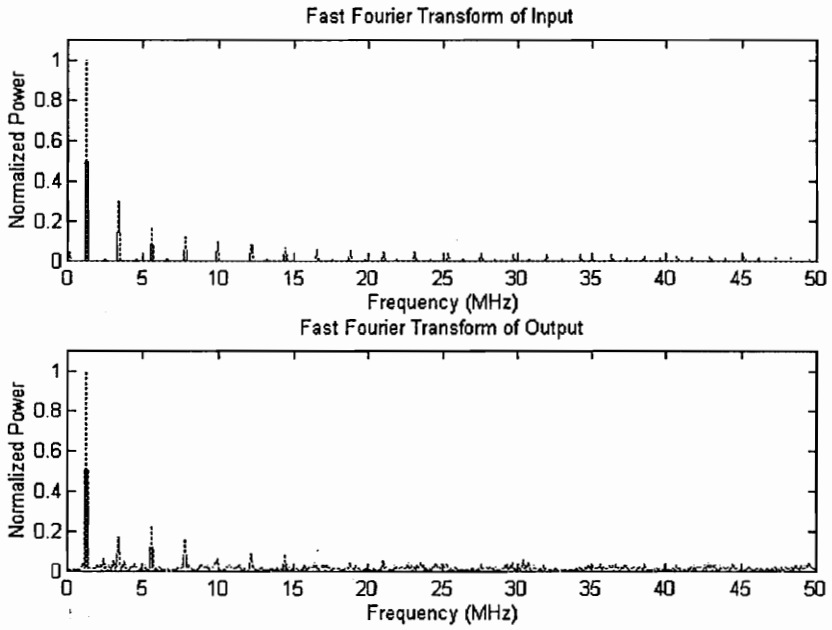


Figure 6.3. FFT of the input waveform (Top) and output waveform (Bottom).

It can be seen that there is a significant reduction of power in the input signal around 3.3 MHz as well as 10 MHz. This is a good indication of the attenuation in the first and second harmonic of the filter's stopband. However, the most reliable method of determining the frequency response is to send a pure sinusoid into the filter and measure the spectral power at the output. This was done by manually changing the frequency of the input signal. It was found that filter exhibited some instabilities at low frequencies. These instabilities were manifested in the form of a fluctuating output spectral power. These fluctuations are most likely the result of the small degree of fusion in the coupler and the relatively poor performance of the modulator head at the low frequencies. Once past the first harmonic of the filter, these fluctuations became minimal.

After the frequency response of the filter was measured, the loop was severed so that the parameters of the coupler could be measured. The coupling ratio, K, was measured to be

12.1 and the insertion loss factor of the coupler was 0.9416. In addition, the loop length was measured to be 30.03m. The core index of refraction and attenuation were specified by Corning to be 1.4675 and 0.40 dB/km respectively. Given these parameters the filter specifications are determined to be:

Sampling Frequency (Fs)	6,191,075 Hz,
Coupling Ratio (K)	12.1,
Losses (L)	0.8990.

Figure 6.4 shows the theoretical frequency response as a solid line with the measured response. As can be seen, there is a good correlation between the two plots

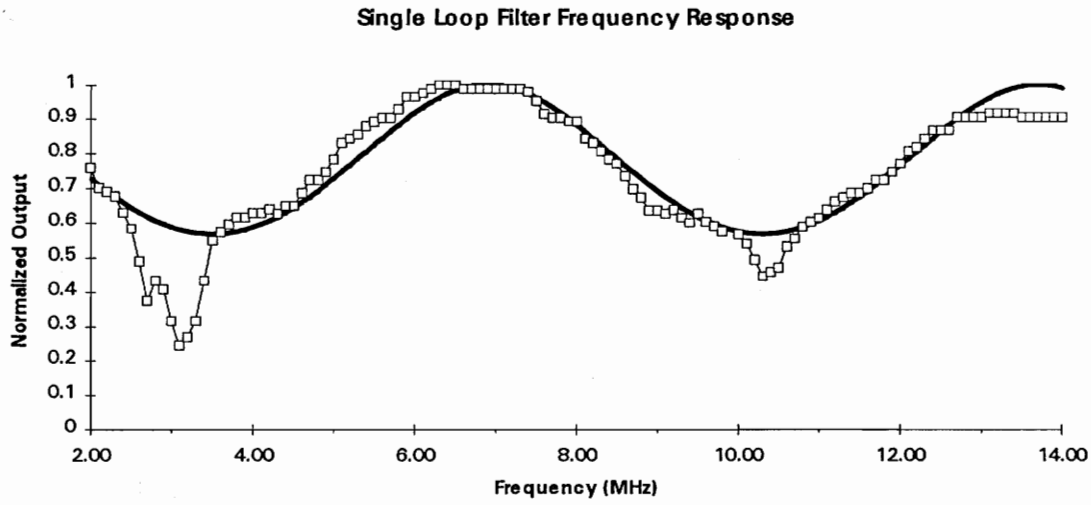


Figure 6.4. Theoretical (solid) and measured (dotted) frequency response of the filter.

Figure 6.5 compares the input waveform with the output waveform of the filter at 3.0MHz. In the middle of the stopband, the filter is able to successfully attenuate the signal to the noise floor.

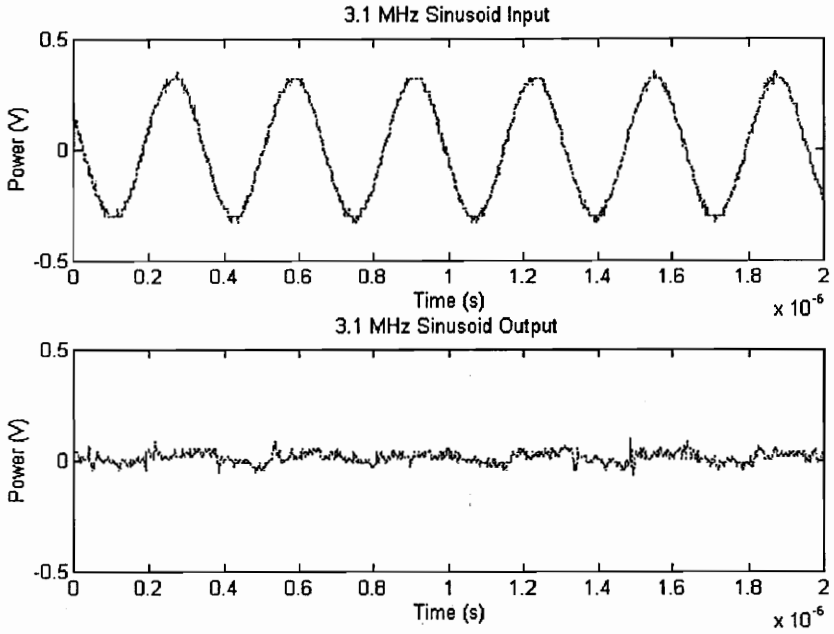


Figure 6.5. Comparison of filter response at 3.1 MHz.

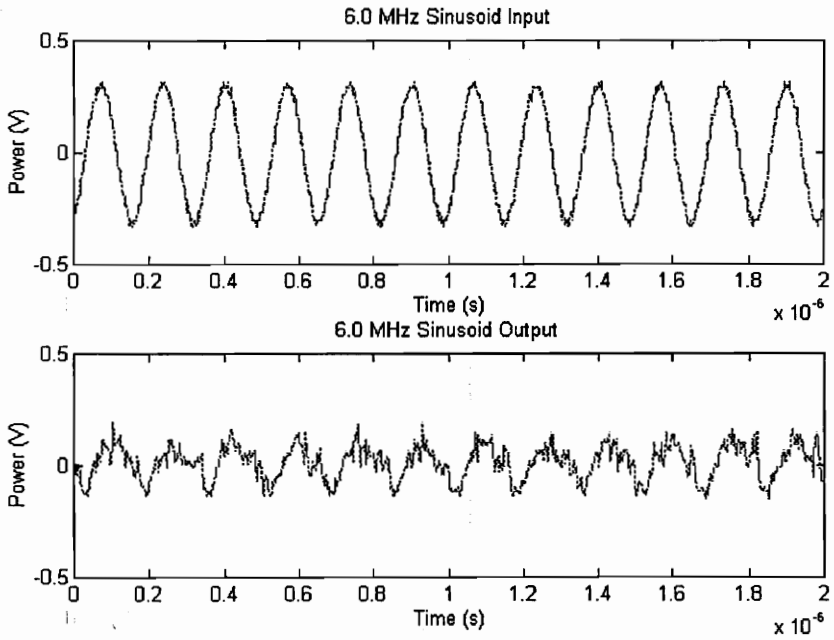


Figure 6.6. Comparison of filter response at 6.0 MHz.

Figure 6.6 compares the filter input and output for a 6.0MHz sinusoid. As this signal is near the filter passband, the attenuation of the signal is slight. However, the output waveform exhibits the phase distortion expected from the recursive nature of the filter.

6.2 Tunable Single Loop Filter

As mentioned previously, one of the disadvantages of optical signal processors, especially those made by the FBT technique, is the inability to tune the filter after manufacture. In Chapter 5, Figure 5.5, the dependency of stopband attenuation on both coupling coefficient and loop losses was established. It will be shown that by incorporating twist-induced tuning with the conventional optical signal processors, not only can the notch depth of the filter be changed, but also its sampling frequency.

The experimental setup is identical to that in Figure 6.1. However, the coupler station was substituted with an 8x8 star coupler station as it allowed for the axial twisting necessary with minimal bending of the coupler. Two separate lengths of fiber, 1 meter and approximately 30 meters in length were used to form a traditional 2x2 coupler. The twisted fibers were heated and pulled through one coupling cycle (i.e. all the optical power returned to the injection leg).

The coupler was then twisted through two complete revolutions. Figures 6.7 and 6.8 shows the coupling coefficient and insertion losses of the coupler as a function of rotation respectively.

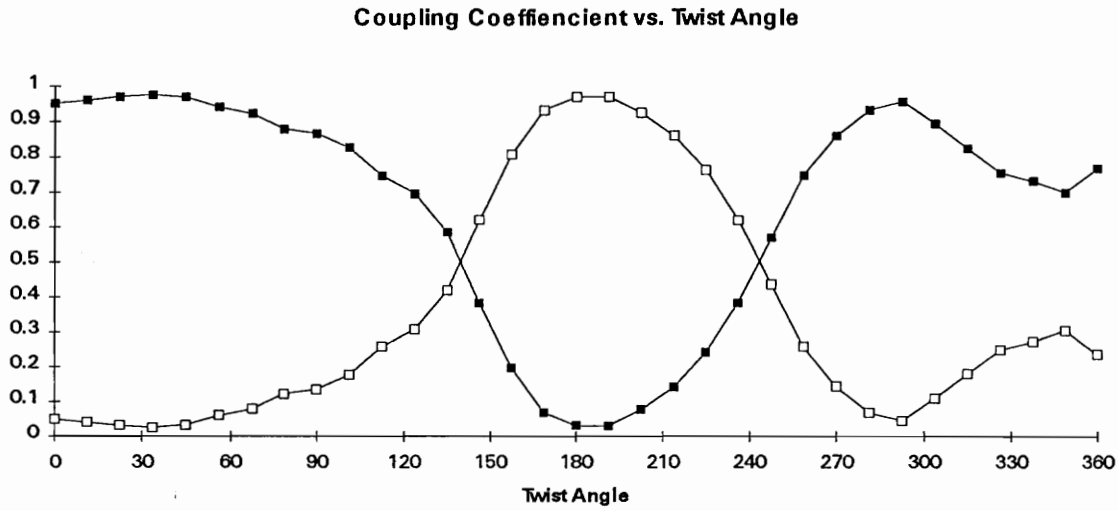


Figure 6.7. Coupling coefficient as a function of rotation.

As can be seen, both parameters vary as a function of rotation. This change verifies the ability to move about the K-L plane describe in Figure 5.5. Given this ability, it is now possible to compensate changes in coupler losses by varying the coupling coefficient.

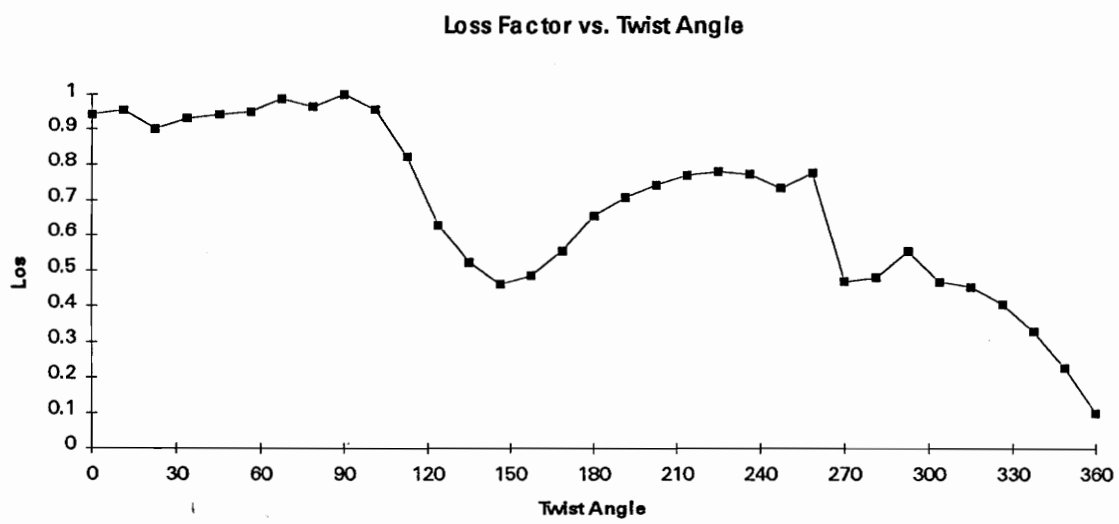


Figure 6.8. Coupler loss as a function of rotation.

To prove this ability, the non-active legs of the coupler were connected using a GTE thermoelastic mechanical splice tube to form the loop. The coupler was then twisted until the desired frequency response was achieved, 326 degrees. This response occurred when the twisting caused the correct combination of coupling coefficient and loop losses to be achieved and is shown in Figure 6.9.

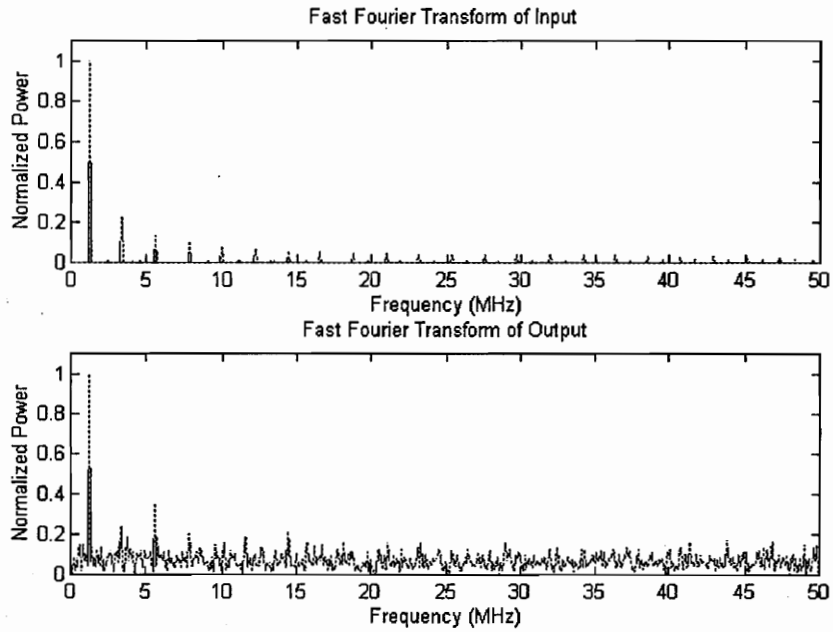


Figure 6.9. FFT of the input waveform (Top) and output waveform (Bottom).

The frequency response of this filter was measured in the same manner as the first filter and is shown in Figure 6.10. While a theoretical response is not available due to the absence of the coupling coefficient value, the values of Fs and its multiples are available. The sampling frequency of 6,450,909 Hz was calculated using a loop length of 31.69 meters which was measured after the filter was disassembled. The expected maxima and minima of the frequency response given the sampling frequency are denoted by markers.

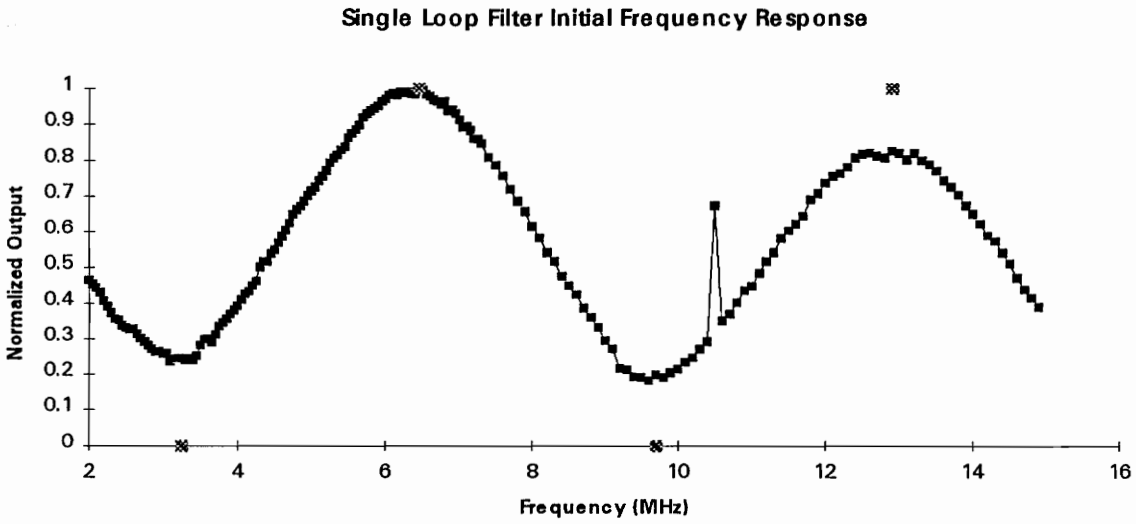


Figure 6.10. Plot of frequency response.

As can be seen in Figure 6.10, the tunability of the coupler successfully allows for a lowpass filter to be created. In addition, the notch depth of the filter may be varied. This variation is possible because there is a much greater flexibility in changing the combination of coupling coefficient and loop losses.

In addition to the notch depth, the sampling frequency of the filter can be easily changed by increasing or decreasing the loop length. Traditionally, changing loop length would require the insertion of a splice in the loop. This splice as well as a change in loop attenuation would modify the loss in the coupler, thus creating an unbalance in the coupling coefficient/loss combination. Since changes in the loop losses may now be compensated by changing the coupling ratio, splices in the loop can be tolerated after the manufacture of the filter. To decrease the sampling frequency of the filter, for example, a length of fiber may be spliced into the loop. Where before, the increase in loop losses

would have completely offset the balance of the filter with coupling ratio, the required coupling ratio may now be achieved by twist tuning.

A loop length of 20.44 meters was mechanically spliced into the loop of the filter characterized in Figure 6.20. The additional splice and length of fiber increased the loop losses by 0.41 dB. By inserting this length of fiber, the sampling frequency was decreased to 3,921,529 Hz. The coupler was then twisted to 304 degrees causing the desired frequency response to return as shown in Figure 6.11.

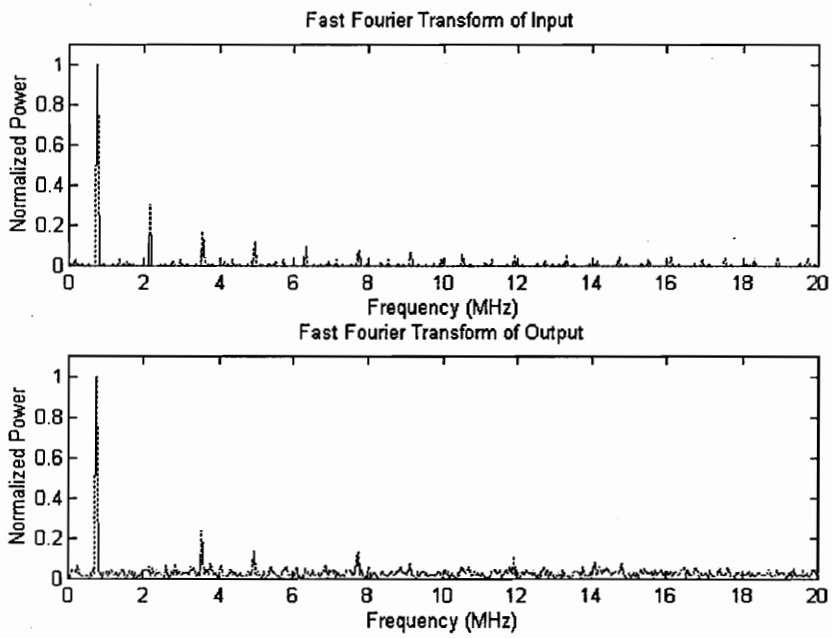


Figure 6.11. FFT of the input waveform (Top) and output waveform (Bottom).

Figure 6.12 shows the frequency response of the filter as sinusoids of varying frequency were passed though. The theoretical response is superimposed for reference. This curve was derived using measured parameters of loop length equal to 52.13 meters, K=0.37 and L=0.327.

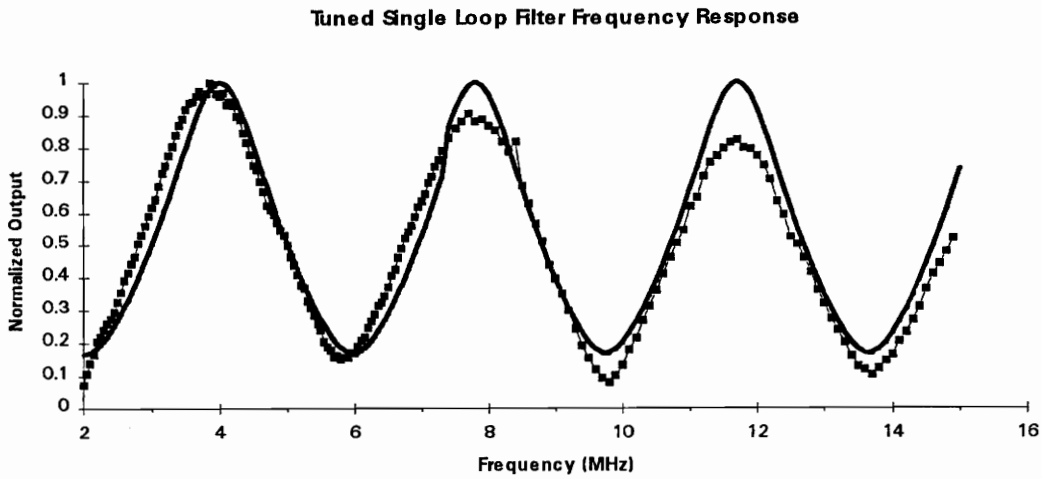


Figure 6.12. Theoretical (solid) and measured (dotted) frequency response of the re-tuned filter.

As can be seen in Figure 6.12, the notch depth of the filter is slightly deeper than the original filter. The most important feature however is that the stopbands and passbands have obviously been changed for this filter while using the same coupler. By extension of this technique, it is obvious that this phenomena can be utilized to create flexible and inexpensive optical signal processors.

6.3 Future Experiments

With the incorporation of inexpensive tunable couplers, a multitude of practical applications exist for optical signal processors. As mentioned previously, the main shortcoming of incorporating fiber optic signal processors is the inflexibility of their performance after manufacture. By integrating twist tuning technology in the manufacturing process, these filters become much more flexible. Below is a brief description of experiments which would help improve optical signal processor technology and demonstrate its applicability in signal processing.

6.3.1 Modeling of Higher Order Systems

By extending the concepts developed in this thesis, computer modeling of triple, quadruple or larger loops could improve the theoretical knowledge of these optical signal processors. Such an analysis will help to demonstrate other types of filtering responses available by using a higher order system. In addition, such an analysis may lead to recursive relationships like those available for the lattice structure. These relationships would greatly improve the analysis of these higher order systems.

In addition to higher order systems, the modeling of combined systems may prove insightful. Such modeling could involve the cascading of FIR and lattice structures. Since these sections are constructed of independent couplers, this would allow complex systems with poles and zeros which may be adjusted independently of one another. One possible type of filter which may result from this configuration is an optical allpass filter. Allpass filters have a constant frequency response but a varying phase response. Such filters are frequently implemented in systems with recursive sections to reduce the phase distortion.

6.3.2 Fabrication of Higher Order Systems

The fabrication of higher order systems, such as those modeled in this thesis, would help to foster a growth of knowledge in the analysis of these filters. The practical knowledge gained through recurring fabrication would enhance the existing theoretical foundation.

Models of the double loop structure have shown that multiple types of frequency responses are possible. Because of the large number of possibilities, this would make an ideal first structure to fabricate and test. The incorporation of tunable couplers would facilitate the fabrication of this filter as well as the generation of the multiple responses possible without a plethora of couplers.

6.3.3 Twist Tunable Coupler Manufacturing Issues

Twist induced tuning has demonstrated its ability to facilitate the fabrication of optical signal processors. With such a coupler, filters may be assembled without the need for coupler manufacturing stages or oxy-propane torches. Various filters may be constructed by a technician with a twist of the wrist. Such ease will help to make the use of these filters, despite their restrictions, more acceptable to the scientific community.

This acceptance however is based on the availibility of reliable tunable couplers. To imporve the reliability of these devices, much research into their manufactureing technique is necesscary. Issuses such as ease in manufacturing, device loss, reliability and packaging are the main concerns. The resolution of these issuses however would provide more than just practical optical signal processors. The application of these devices as tunable fiber couplers would also posess a high commercial value.

6.3.4 Photofluidic Systems

Fluidic devices have found applications is machine control, medical equipment, aerospace and military systems. These devices offer advantages similar to optical fiber components

in terms of ability to operate in harsh environments. Currently, photofluidic interfaces are being developed to integrate fiber sensor technology with these systems [6.1]. These systems could benefit greatly from the incorporation of optical signal processors as signal conditioning is not feasible otherwise. Figure 6.13 shows a diagram of such a proposed incorporation.

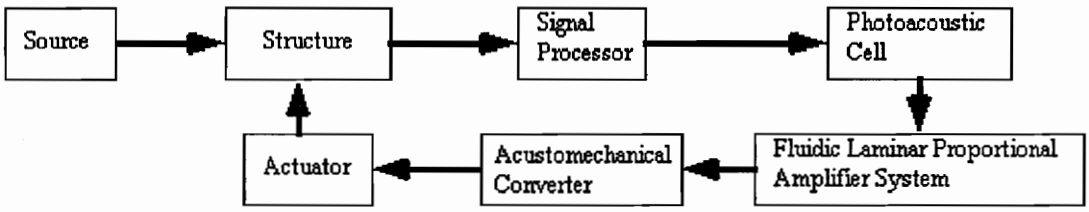


Figure 6.13. Photofluidic system with an optical signal processor.

In an application of a beam structure, the filter could be used to filter out certain harmonics of the beams vibrational frequency. This filtering would be of great use in vibration control algorithms.

6.3.5 Adaptive Optical Signal Processors

Because of the ability of twist tuned couplers to compensate for varying coupler losses, the potential exists for adaptive optical signal processors. This flexibility could be achieved using optical fiber bus switches and rotational actuators. The filtering function may be changed by switching between different delay lines and possibly couplers. The rotational actuators would twist the couplers to the appropriate rotation to achieve the desired filtering function. Use of this technology in such an application would require the repeatability in performance of the twisted couplers to be addressed.

Chapter 7: Conclusions

Optical signal processors in the form of optical fiber couplers have been investigated in great detail. Various configurations of these devices have been described. Computer analysis has been performed that demonstrates the functional characteristics of these configurations. It has been shown that while the single loop filter is the simplest in design, it is also the least flexible. By inserting an EDFA in the loop, an extra degree of freedom may be achieved. However due to the expensive nature of these devices, asymmetric couplers may be substituted with little loss in flexibility while guaranteeing stability. In addition, double loop filters were explored to show an example of increasing the filter order and allowing different types of filtering responses. In addition to filters based on the single loop filter, other filters based on electronic FIR filters and lattice structures have been discussed. Each has shown their benefits as well as their drawbacks.

In addition to filter configurations, the ability to change the filtering response of the loop after manufacture was explored. By mapping out the K-L plane of the single loop filter, it was shown that the same filtering responses were possible to different coupler parameters. This open the door to using twist induced tuning of the coupler to allow this change in filtering response. This breakthrough in filter manufacture will allow optical signal processors to become easily manufactured at less expense.

With the availability of inexpensive tunable filters, a multitude of practical applications exist for optical signal processors. Some of these applications have been mentioned while numerous others exist. Much of these applications will require the further investigation of the twist tuning affect in couplers.

Chapter 8: References

[1.1] R.O. Claus, "Preface," Proceedings of the 5th Conference on Fiber Optic Sensor-Based Smart Materials and Structures, IOP Publishing, Philadelphia, PA, 1992.

[1.2] R.O. Claus, "Overview of fiber optic sensor-based smart materials and structures," Proceedings of the 4th Conference on Fiber Optic Sensor-Based Smart Materials and Structures, Technomic Publishing Co, Lancaster, PA, 1991.

[1.3] J.E. Bowers, S.A. Newton, W.V. Sorin and H.J. Shaw, "Filter response of single-mode fibre recirculating delay lines," Elec. Lett. vol. 18, p. 110-111, 1982.

[2.1] L.B. Jackson, *Digital Filters and Signal Processing*, Kluwer Academic Publishers, Boston, MA, 1989.

[2.2] L. Pennisi, *Elements of Complex Variables*, Holt, Rinehard and Winston, New York, NY, 1962.

[2.3] E.W. Kamen, *Introduction to Signals and Systems*, Macmillan Publishing Company, New York, NY, 1990.

[2.4] A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[2.5] J.D. Markel and A.H. Gray, *Linear Prediction of Speech*, Springer-Verlag, New York, NY, 1976.

[3.1] C.T. Chang, J.A. Cassaboom and H.F. Taylor, "Fibre-optic delay-line devices for R.F. signal processing," Elect. Lett., vol. 13, p. 678-680, 1977.

[3.3] S.K. Sheem and T.G. Giallorenzi, Optics Lett., vol. 4, p. 29, 1979.

[3.4] E. Weidel and D. Gruchmann, Elect. Lett., vol. 15, p. 738, 1979.

[3.5] B.S. Kawasaki and K.O. Hill "Low-loss access coupler for multimode optical fiber distribution networks", App. Opt., vol. 16, p. 1794-1795, 1977.

[3.6] Y. Tsujimoto, H. Serizawa, K. Hattori, and M. Fukai, "Fabrication of low-loss 3dB couplers with multimode optical fibres," Elect. Lett., vol. 14, p. 157-158, 1978.

[3.7] R. A. Bergh, K. Kotler and H.J. Shaw, "Single mode fibre optic directional coupler," Elect. Lett. vol. 16, p. 260-261, 1980.

[3.8] B. S. Kawasaki, K.O. Hill and R.G. Lamont,"Biconical-taper single-mode fiber coupler," Opt. Lett. vol. 6, p. 327-328,1981.

[3.9] M.J.F. Digonnet and H.J. Shaw, "Analysis of a tunable single mode optical fibre coupler," J. Quantum Electronics, vol. QE-18, p. 746-754, 1982.

[3.10] K.A. Murphy, "Method of fabricating a low loss fused biconical taper fiber optic coupler," U.S. Patent No. 4,426,215, 1984.

[3.11] S. Celaschi, J.T. De Jesus and F. M. Smolka, "All-fiber Tunable Beam Splitter," Technical Digest, conference on Lasers and Electro-Optics, OSA, 1988.

[3.12] B.S. Kawasaki, M. Kawachi, K.O. Hill and D.C. Johnson, "A single-mode-fiber coupler with a variable coupling ratio," J. Light. Tech., vol. LT-1, p. 176-178, 1983.

[3.13] T.A. Birks, "Practical tuning mechanism for fused-tapered couplers," Opt. Lett., vol. 13, p. 1126-1128, 1988.

[3.14] D. Marcuse, "Field deformation and loss caused by curature of optical fibers," J. Optical Society of America, vol. 66, p. 311-320, 1976.

[3.15] T.A. Birks, "Twist-induced tuning in tapered fiber couplers," App. Opt., vol. 28, p. 4226-4233, 1989.

[3.16] R. Griffin, J.D. Love, P.R.A. Lyons, D.A. Thorncraft and S.C. Rashleigh, "Asymmetric multimode couplers," J. Light. Tech. vol. 9, p. 1508-1517, 1991.

[3.17] R.J. Boughton, M. Farhadiroushan, D. Kreit, R.C. Youngquist and J.P. Giles, "Optical-fiber systems with nonreciprocal couplers," Opt. Lett. vol. 12, p. 205-207.

[3.18] D.B. Mortifore, "Wavelength-flattened fused couplers," Elect. Lett., vol. 21, p.742-743, 1985.

[3.19] R.G. Lamont, K.O. Hill and D.C. Johnson, Digest of Topical Meeting on Optical Fiber Communication, p. 78, 1985

[4.1] D.M. Gookin and M.H. Berry, "Finite impulse response filter with large dynamic range and high sampling rate," App. Opt., vol. 29, p. 1061-1062, 1990.

[4.2] B. Moslehi, J.W. Goodman, M. Tur and H.J. Shaw, "Fiber-optic lattice signal pocessing," Proceedings of the IEEE, vol. 72, p. 909-930, 1984.

[4.3] B. Moslehi and J.W. Goodman, "Novel amplified fiber-optic recirculation delay line processor," J. Light. Tech., vol. 10, p. 1142-1147, 1992.

[4.4] E. Desurvire, "The golden age of optical fiber amplifiers," Physics Today, p. 20-27, January, 1994.

[4.5] M.C. Vazquez, B. Vizoso, M. Lopez-Amo and M.A. Muriel, "Single and doule amplified recirculating delay lines as fibre-optic filters," Elect. Lett., vol. 28, 1992.

[6.1] R.O. Claus, K.A. Murphy, B.R. Fogg, D. Sun and A.M. Vengsarkar," Photofluidics for integrating fiber sensors with high-authority mechanical actuators," Proceedings of SPIE/OE Fibers, Boston, MA, 1991.

Chapter A: Appendix

The following is a listing of the Matlab 4.0 Code that was developed for the computer models. All the codes are considered copyrighted with all rights reserved.

A.1 FIR Filter

A.1.1 randfir.m

```
function [z]=randfir(iter)
%function [z,p]=randfir(iter)
%       Random analysis if an filter with a 5th FIR filter over ITER
%       interations.
for i=1:iter
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        a3=rand(1);
        a4=rand(1);
        a5=rand(1);
        z(:,i)=roots([1 a0 a1 a2 a3 a4 a5]);
end
%pole zero plot
f=[0:.01:1];
hold off
plot(exp(j*2*pi*f))
```

```
hold on
plot(real(z),imag(z),'.r')
axis (1.5*[-1 1 -1 1]);
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
```

A.2 Single Loop Filter

A.2.1 randsing.m

```
function [z,p]=singloop(iter)
%function [z,p]=singloop(iter)
%       Random analysis of a single loop filter over ITER interations.

g=1;
for i=1:iter
        a=rand(1);
        z1=a;
        z2=-g*(2*a-1);
        p1=1;
        p2=-g*a;
        z=[z1 z2];
        p=[p1 p2];
        z(:,i)=roots([z1 z2]);
        p(:,i)=roots([p1 p2]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
hold off
circle=exp(j*2*pi*f);
%axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
%plot(real(z),imag(z),'.r')
plot(real(p),imag(p),'xr')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.2.2 sing.m

```
%plots a 3D graph of coupling coiefficient v. Fs v. amplitude
%call singfab.m

coup=[0:.05:1]; %sets range of K

for i=1:length(coup)
        [h,w]=singfab(coup(i),1);
        m(:,i)=h;
end

z=-20;
meshcc(m,z)
axis([1 25 1 161 z 20])

%set axes
x=0:.25:1;
sx=alabel(x);
set (gca,'XTickMode','manual')
set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
set (gca,'xticklabels',sx);

y=[0 0.25 0.50 0.75 1];
sy=alabel(y);
set (gca,'YTickMode','manual')
set (gca,'Ytick',[1 41 81 121 161])        %0 index not allowed in Matlab
set (gca,'yticklabels',sy);

%label axes
xlabel('Coupling Coefficient')
ylabel('Fs')
zlabel('dB')
title('Single Loop Filter')
```

A.2.3 movsing.m

```
%Creates a movie of Mag v. Frequency v. Coupling coiefficient with
%loss being time varying

z=-25;  %sets altitude for contour plot
lim=([1 25 1 161 z 0]);
loss=(.05:.05:1);          %sets range for loss
coup=[0:.05:1];            %sets range for coupling coiefficient

%compensate for frames that hit "hot" spots that skew plots
coup(11)=.49;
coup(16)=.74;

M=moviein(length(coup));

for j=19:length(loss)                %loop for movie
        for i=1:length(coup)         %loop for each frame
                [h,w]=singfab(coup(i),loss(j));
                m(:,i)=h;
        end

        meshcc(m,z)
        axis(lim)
        M(:,j)=getframe;

        %set axes
        x=0:.25:1;
        sx=alabel(x);
        set (gca,'XTickMode','manual')
        set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
        set (gca,'xticklabels',sx);

        y=[0 0.25 0.50 0.75 1];
        sy=alabel(y);
        set (gca,'YTickMode','manual')
        set (gca,'Ytick',[1 41 81 121 161])          %0 index not allowed in Matlab
        set (gca,'yticklabels',sy);

        %label axes
        xlabel('Coupling Coefficient')
        ylabel('Fs')
        zlabel('dB')
        title('Single Loop Filter')

end

movie(M,20)
```

A.2.4 singfab.m

```
function [h,w,z,p]=singfab(a,g)
%function [h,w,z,p]=singfab(a,g)
%       Gives the frequency response of a single loop given the coupling
%       coiefficient [a] and loop losses/gain [g]

% g is sum of the losses/gains due to the amplifier and loop losses
% i.e. if no amplifier (gain=1) and loop loss of .8 then g=1*.8=1
% if gain=.5 and loop loss is .8 then g=.5*.8=.4

%calculate coiefficients
z1=a;
z2=-g*(2*a-1);
p1=1;
p2=-g*a;

%calculate roots
b=[z1 z2];
a=[p1 p2];

z=roots(b);
p=roots(a);
[h,w]=freqz(b,a,161,'whole');     %161 points so that the 3D plot looks
                                  %better and allows a easy division of
                                  %Fs
h=10*log10(abs(h));
```

A.2.5 singloss.m

```
%Plots a 3D graph of Amp v. coupling coiefficient v. loop loss/gain
%Amplitude is point 80 of 161 points in the magnitude of the
%frequency response
%since h is 161 long, 0.50 Fs occurs at approximately the 80th point

coup=[.1:.01:.4];         %varies K
gain=[0:.01:1];           %varies loss

for i=1:length(coup)
        for j=1:length (gain)
                [h,w]=singfab(coup(i),gain(j));
                m(j,i)=h(80);                   %takes just the lowest point
        end
end

meshc(m)
axis([0 100 0 100 -15 0])

%sets axes
x=0:.25:1;
```

```
sx=alabel(x);
set (gca,'xticklabels',sx);

y=[0 0.333 0.667 1];
sy=alabel(y);
set (gca,'yticklabels',sy);

%labels axes
xlabel('Coupling Coefficient')
ylabel('Loop Loss')
zlabel('dB')
title('Single Loop Filter')
```

A.3 Gain Amplified Filter

A.3.1 gain.m

```
%plots a 3D graph of coupling coiefficient v. Fs v. amplitude
%calls singfab.m

gain=[1:.5:5]; %sets range for gain

for i=1:length(gain)                    %loop for gain
        [h,w]=singfab(.5,gain(i));
        m(:,i)=h;
end

z=-20;
meshcc(m,z)
axis([1 25 1 161 z 10])

%set axes
x=0:.25:1;
sx=alabel(x);
set (gca,'XTickMode','manual')
set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
set (gca,'xticklabels',sx);

y=[0 0.25 0.50 0.75 1];
sy=alabel(y);
set (gca,'YTickMode','manual')
set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
set (gca,'yticklabels',sy);

%label axes
xlabel('Coupling Coefficient')
ylabel('Fs')
zlabel('dB')
title('Gain Amplified Filter')
```

A.3.2 movgain.m

```
%Creates a movie of Mag v. Frequency v. Coupling coiefficient with
%gain being time varying

z=-30;  %sets altitude for contour plot

lim=([1 25 1 161 z 10]);

%compensate for frames that hit "hot" spots that skew plots
loss=(1:.1:5);
loss(11)=1.99;
loss(21)=2.99;
loss(36)=4.49;

M=moviein(length(loss));
coup=[0:.05:1];  %sets range for gain

for j=1:length(loss)                    %loop for movie
        for i=1:length(coup)        %loop for each frame
                [h,w]=singfab(coup(i),loss(j));
                m(:,i)=h;
        end
        meshcc(m,z)
        axis(lim)
        M(:,j)=getframe;
        %set axes
        x=0:.25:1;
        sx=alabel(x);
        set (gca,'XTickMode','manual')
        set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
        set (gca,'xticklabels',sx);

        y=[0 0.25 0.50 0.75 1];
        sy=alabel(y);
        set (gca,'YTickMode','manual')
        set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
        set (gca,'yticklabels',sy);

        %label axes
        xlabel('Coupling Coefficient')
        ylabel('Fs')
        zlabel('dB')
        title('Gain Amplified Filter')
end
movie(M,20)
```

A.4 Asymmetric Coupler Filter

A.4.1 randasym.m

```
function [z,p]=asymloop(iter)
%function [z,p]=asymloop(iter)
%       Random analysis of an asymetric coupler filter over ITER interations.

for i=1:iter
        a=rand(1);
        b=rand(1);
        z1=a;
        z2=-(b+a-1);
        p1=1;
        p2=-b;
        z(:,i)=roots([z1 z2]);
        p(:,i)=roots([p1 p2]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
hold off
circle=exp(j*2*pi*f);
axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
plot(real(z),imag(z),'.r')
plot(real(p),imag(p),'xr')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.4.2 asym.m

```
%Plots the 3D mesh and contour graph of an asymmetric coupler filter
%Calls asymfab

coup=[0:.05:1];%varies A
B=.9;
for i=1:length(coup)
        [h,w]=asymfab(coup(i),B,1,1);
        m(:,i)=h;
end

z=-25;
meshcc(m,z)%plots mesh and puts contour at -25dB
axis([1 25 1 161 z 0])

%set the axes
x=0:.25:1;
sx=alabel(x);
set (gca,'XTickMode','manual')
set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
set (gca,'xticklabels',sx);

y=[0 0.25 0.50 0.75 1];
sy=alabel(y);
set (gca,'YTickMode','manual')
set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
set (gca,'yticklabels',sy);

z=[-25 -20 -15 -10 -5 0];
sz=alabel(z);
set (gca,'ZTickMode','manual')
set (gca,'Ztick',[-25 -20 -15 -10 -5 0])
set (gca,'Zticklabels',sz);

%label the axes
xlabel('Coupling Coefficient')
ylabel('Fs')
zlabel('dB')
title('Asymmetric Filter')
```

A.4.3 movasym.m

```
%Creates a movie of Mag v. Frequency v. Coupling coiefficient with
%coupling coiefficient B time varying

z=-25;  %sets altitude for contour plot
lim=([1 25 1 161 z 0]);

coupa=[0:.05:1]; %sets range for A
coupb=[0:.05:1]; %sets range for B

%compensate for frames that hit "hot" spots that skew plots
coupb(3)=.09;
coupb(5)=.19;
coupb(7)=.29;
coupb(9)=.39;
coupb(13)=.59;
coupb(15)=.69;
coupb(17)=.79;
coupb(19)=.89;

M=moviein(coupb);

for j=1:length(coupb)                %loop for movie
        for i=1:length(coupa)        %loop for each frame
                [h,w]=asymfab(coupa(i),coupb(j),1,1);
                m(:,i)=h;
        end

        meshcc(m,z)
        axis(lim)
        M(:,j)=getframe;

        %set the axes
        x=0:.25:1;
        sx=alabel(x);
        set (gca,'XTickMode','manual')
        set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
        set (gca,'xticklabels',sx);

        y=[0 0.25 0.50 0.75 1];
        sy=alabel(y);
        set (gca,'YTickMode','manual')
        set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
        set (gca,'yticklabels',sy);

        zl=[-25 -20 -15 -10 -5 0];
        sz=alabel(zl);
        set (gca,'ZTickMode','manual')
        set (gca,'Ztick',[-25 -20 -15 -10 -5 0])
        set (gca,'Zticklabels',sz);
```

```
        %label the axes
        xlabel('Coupling Coefficient')
        ylabel('Fs')
        zlabel('dB')
        title('Asymmetric Filter')
end

movie(M,20)
```

A.4.4 asymfab.m

```
function [h,w,z,p]=asymfab (a,b,ga,gb)
%[h,w,z,p]=asymfab (a,b,ga,gb)
%       Gives the magnitude response and roots of an asymmetric coupler
%       filter given the coupling coiefficient [a] and [b] and the loop
%       losses/gains of path [a] and [b] by [ga] and [gb] respectively.

%calculate coiefficients
z1=a;
z2=-(b+a-1);
p1=1;
p2=-b;
b=[z1 z2];
a=[p1 p2];

%calculate roots
z=roots(b);
p=roots(a);

[h,w]=freqz(b,a,161,'whole');     %161 points so that the 3D plot looks
                                  %better and allows a easy division of
                                  %Fs
h=10*log10(abs(h));
```

A.5 Double Loop Filter

A.5.1 randtloo.m

```
function [z,p]=twoloop (iter)
%function [z,p]=twoloop (iter)
%       Random analysis of a double loop filter over [iter] iterations.

for i=1:iter
        a=rand(1);
        b=rand(1);
        z1=a;
        z2=-2*b;
        z3=(1 - 2*b)*(1 - 2*a);
        p1=1;
```

```
        p2=-(a + 1)*b;
        p3=a*(2*b - 1);
        z(:,i)=roots([z1 z2 z3]);
        p(:,i)=roots([p1 p2 p3]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
circle=exp(j*2*pi*f);
axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
plot(real(z),imag(z),'.r')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.5.2 tloop.m

```
%Plots the 3D meshc graph of a double loop filter
%Calls tloopfab

coup=[0:.05:1]; %varies A

for i=1:length(coup)
        [h,w]=tloopfab(coup(i),.85,1,1);
        m(:,i)=h;
end

z=-25;
meshcc(m,z)
axis([1 25 1 161 z 0])

%set axes
x=0:.25:1;
sx=alabel(x);
set (gca,'XTickMode','manual')
set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
set (gca,'xticklabels',sx);

y=[0 0.25 0.50 0.75 1];
sy=alabel(y);
set (gca,'YTickMode','manual')
set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
set (gca,'yticklabels',sy);

z=[-25 -20 -15 -10 -5 0];
sz=alabel(z);
```

```
set (gca,'ZTickMode','manual')
set (gca,'Ztick',[-25 -20 -15 -10 -5 0])
set (gca,'Zticklabels',sz);

%label axes
xlabel('Coupling Coefficient')
ylabel('Fs')
zlabel('dB')
title('Double Loop Filter')
```

A.5.3 movtloop.m

```
%Creates a movie of Mag v. Frequency v. Coupling coiefficient with
%coupling coiefficient B time varying

z=-25;  %sets altitude for contour plot
lim=([1 25 1 161 z 0]);

coupa=[0:.05:1];%sets range for A
coupb=[0:.05:1];%sets range for B

%compensate for frames that hit "hot" spots that skew plots
coupb(1)=.000001;       %keeps model away from B=0
coupb(11)=.49;
coupb(13)=.59;
coupb(21)=.999999;

M=moviein(length(coupb));

for j=1:length(coupb)               %loop for movie
        for i=1:length(coupa)       %loop for each frame
                [h,w]=tloopfab(coupa(i),coupb(j),1,1);
                m(:,i)=h;
        end
        meshcc(m,z)
        axis(lim)
        M(:,j)=getframe;

        %set axes
        x=0:.25:1;
        sx=alabel(x);
        set (gca,'XTickMode','manual')
        set (gca,'Xtick',[1 6 11 16 21])%compenates for 0 index disallowed in Matlab
        set (gca,'xticklabels',sx);

        y=[0 0.25 0.50 0.75 1];
        sy=alabel(y);
        set (gca,'YTickMode','manual')
        set (gca,'Ytick',[1 41 81 121 161])%compenates for 0 index
        set (gca,'yticklabels',sy);

        zl=[-25 -20 -15 -10 -5 0];
```

```
        sz=alabel(zl);
        set (gca,'ZTickMode','manual')
        set (gca,'Ztick',[-25 -20 -15 -10 -5 0])
        set (gca,'Zticklabels',sz);

        %label axes
        xlabel('Coupling Coefficient')
        ylabel('Fs')
        zlabel('dB')
        title('Double Loop Filter')
clear m
end
movie(M,20)
```

A.5.4 tloopfab.m

```
function [h,w,z,p]=tloopfab (a,b,ga,gb)
%[h,w,z,p]=tloopfab (a,b,ga,gb)
%       Gives the magnitude respons of a double loop filter given the
%       coupling coiefficients of the 1st and 2nd couplers [a] and [b]
%       respectively and the loop losses/gains of the 1st and 2nd
%       couplers/loops [ga] and [gb] respectively.  Special problems
%       arise if [a] or [b] = 1 or 0.
%       Can be converted to give poles/zeros for a specific filter

%calculates coiefficients
z1=a;
z2=ga*(b-2*a*b)-gb*a*b;
z3=ga*gb*(1 - 2*b)*(1 - 2*a);
p1=1;
p2=-(gb*b + a*b*ga);
p3=a*(2*b - 1)*ga*gb;

%calculate roots
b=([z1 z2 z3]);
a=([p1 p2 p3]);
z=roots(b);
p=roots(a);

[h,w]=freqz(b,a,161,'whole');
h=10*log10(abs(h));
```

A.6 Lattice Filter

A.6.1 recirh21.m

```
function [z,p]=recirc21(iter)
%function [z,p]=recirc21(iter)
%       Random analysis of a 1st order lattice structure filter over
```

```
%       ITER interations.

for i=1:iter
        a0=rand(1);
        a1=rand(1);
        z1=(1-a1)*(1-a0);
        z2=0;
        p1=1;
        p2=-a1*a0;
        z(:,i)=roots([z1 z2]);
        p(:,i)=roots([p1 p2]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
hold off
circle=exp(j*2*pi*f);
axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
%plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.6.2 recirh22.m

```
function [z,p]=rech22(iter)
%function [z,p]=recirc22(iter)
%       Random analysis of a 2nd order lattice structure filter over
%       ITER interations.

for i=1:iter
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        z1=(1 - a2)*(1 - a1)*(1 - a0);
        p1=1;
        p2=-(a0*a1 + a1*a2);
        p3=-a0*a2*(1 - 2*a1);
        z(:,i)=roots([z1 0 0]);
        p(:,i)=roots([p1 p2 p3]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
clg
```

```
circle=exp(j*2*pi*f);
axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
%plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.6.3 recirh23.m

```
function [z,p]=rech23(iter)
%function [z,p]=recirc23(iter)
%       Random analysis of a 3rf order lattice structure filter over
%       ITER interations.

for i=1:iter
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        a3=rand(1);
        z1=(1 - a3)*(1 - a2)*(1 - a1)*(1 - a0);
        p1=1;
        p2=-(a2*a3 + a0*a1 + a1*a2);
        p3=-a0*a2*(1 - 2*a1) - a1*a3*(1 - 2*a2) + a0*a1*a2*a3;
        p4=-a0*a3*(1 - 2*a2)*(1 - 2*a1);
        z(:,i)=roots([z1 0 0 0]);
        p(:,i)=roots([p1 p2 p3 p4]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
clg
circle=exp(j*2*pi*f);
%axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.6.4 recirh24.m

```
function [z,p]=rech24(iter)
%function [z,p]=recirc24(iter)
%       Random analysis of a 4th order lattice structure filter over
%       ITER interations.

for i=1:iter
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        a3=rand(1);
        a4=rand(1);
        z1=(1 - a3)*(1 - a2)*(1 - a1)*(1 - a0);
        p1=1;
        p2=-(a0*a1 + a1*a2 + a2*a3 + a3*a4);
        p3=-(a0*a2 - 2*a0*a1*a2 + a1*a3 - 2*a1*a2*a3 - a0*a1*a2*a3 + a2*a4 - a0*a1*a3*a4 -
2*a2*a3*a4 - a1*a2*a3*a4);
        p4=-(a0*a3 - 2*a0*a1*a3 - 2*a0*a2*a3 + 4*a0*a1*a2*a3 + a1*a4 - 2*a1*a2*a4 -
a0*a1*a2*a4 - 2*a1*a3*a4 - a0*a2*a3*a4 + 4*a1*a2*a3*a4 + 4*a0*a1*a2*a3*a4);
        p5=-(a0*a4 - 2*a0*a1*a4 - 2*a0*a2*a4 + 4*a0*a1*a2*a4 - 2*a0*a3*a4 + 4*a0*a1*a3*a4 +
4*a0*a2*a3*a4 - 8*a0*a1*a2*a3*a4);
        z(:,i)=roots([z1 0 0 0]);
        p(:,i)=roots([p1 p2 p3 p4 p5]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
clg
circle=exp(j*2*pi*f);
%axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
%plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.6.5 recirh25.m

```
function [z,p]=rech25(iter)
%function [z,p]=recirc25(iter)
%       Random analysis of a 5th order lattice structure filter over
%       ITER interations.

for i=1:iter
```

```
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        a3=rand(1);
        a4=rand(1);
        a5=rand(1);

        p1=-1;
        p2=(a0*a1 + a1*a2 + a2*a3 + a3*a4 + a4*a5);
        p3=(a0*a2 - 2*a0*a1*a2 - a0*a1*a2*a3 + a1*a3 - 2*a1*a2*a3 - a0*a1*a3*a4 -
a1*a2*a3*a4 + a2*a4 - 2*a2*a3*a4 - a0*a1*a4*a5 - a1*a2*a4*a5 - a2*a3*a4*a5 + a3*a5 -
2*a3*a4*a5);
        p4=(a0*a3 - 2*a0*a1*a3 - 2*a0*a2*a3 + 4*a0*a1*a2*a3 - a0*a2*a3*a4 + ...
        2*a0*a1*a2*a3*a4 - a0*a1*a2*a4 + 2*a0*a1*a2*a3*a4 + a1*a4 - ...
        2*a1*a2*a4 - 2*a1*a3*a4 + 4*a1*a2*a3*a4 - a0*a2*a4*a5 + ...
        2*a0*a1*a2*a4*a5 + a0*a1*a2*a3*a4*a5 - a1*a3*a4*a5 + ...
        2*a1*a2*a3*a4*a5 - a0*a1*a3*a5 + 2*a0*a1*a3*a4*a5 - a1*a2*a3*a5 + ...
        2*a1*a2*a3*a4*a5 + a2*a5 - 2*a2*a3*a5 - 2*a2*a4*a5 + 4*a2*a3*a4*a5);

        p5=(a0*a4 - 2*a0*a1*a4 - 2*a0*a2*a4 + 4*a0*a1*a2*a4 - 2*a0*a3*a4 + ...
        4*a0*a1*a3*a4 + 4*a0*a2*a3*a4 - 8*a0*a1*a2*a3*a4 - a0*a3*a4*a5 + ...
        2*a0*a1*a3*a4*a5 + 2*a0*a2*a3*a4*a5 - 4*a0*a1*a2*a3*a4*a5 - ...
        a0*a2*a3*a5 + 2*a0*a1*a2*a3*a5 + 2*a0*a2*a3*a4*a5 - 4*a0*a1*a2*a3*a4*a5 - ...
        a0*a1*a2*a5 + 2*a0*a1*a2*a3*a5 + 2*a0*a1*a2*a4*a5 - 4*a0*a1*a2*a3*a4*a5 + ...
        a1*a5 - 2*a1*a2*a5 - 2*a1*a3*a5 + 4*a1*a2*a3*a5 - 2*a1*a4*a5 + ...
        4*a1*a2*a4*a5 + 4*a1*a3*a4*a5 - 8*a1*a2*a3*a4*a5);

        p6=(a0*a5 - 2*a0*a1*a5 - 2*a0*a2*a5 + 4*a0*a1*a2*a5 - 2*a0*a3*a5 + ...
        4*a0*a1*a3*a5 + 4*a0*a2*a3*a5 - 8*a0*a1*a2*a3*a5 - 2*a0*a4*a5 + ...
        4*a0*a1*a4*a5 + 4*a0*a2*a4*a5 - 8*a0*a1*a2*a4*a5 + 4*a0*a3*a4*a5 - ...
        8*a0*a1*a3*a4*a5 - 8*a0*a2*a3*a4*a5 + 16*a0*a1*a2*a3*a4*a5);

        p(:,i)=roots([p1 p2 p3 p4 p5 p6]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
clg
circle=exp(j*2*pi*f);
%axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.6.6 recirh26.m

```
function [z,p]=rech26(iter)
%function [z,p]=recirc26(iter)
%       Random analysis of a 6th order lattice structure filter over
%       ITER interations.

for i=1:iter
        a0=rand(1);
        a1=rand(1);
        a2=rand(1);
        a3=rand(1);
        a4=rand(1);
        a5=rand(1);
        a6=rand(1);

p7 = -(a0*a6) + 2*a0*a1*a6 + 2*a0*a2*a6 - 4*a0*a1*a2*a6 + 2*a0*a3*a6 - ...
  4*a0*a1*a3*a6 - 4*a0*a2*a3*a6 + 8*a0*a1*a2*a3*a6 + 2*a0*a4*a6 - ...
  4*a0*a1*a4*a6 - 4*a0*a2*a4*a6 + 8*a0*a1*a2*a4*a6 - 4*a0*a3*a4*a6 + ...
  8*a0*a1*a3*a4*a6 + 8*a0*a2*a3*a4*a6 - 16*a0*a1*a2*a3*a4*a6 + 2*a0*a5*a6 -...
  4*a0*a1*a5*a6 - 4*a0*a2*a5*a6 + 8*a0*a1*a2*a5*a6 - 4*a0*a3*a5*a6 + ...
  8*a0*a1*a3*a5*a6 + 8*a0*a2*a3*a5*a6 - 16*a0*a1*a2*a3*a5*a6 - ...
  4*a0*a4*a5*a6 + 8*a0*a1*a4*a5*a6 + 8*a0*a2*a4*a5*a6 - ...
  16*a0*a1*a2*a4*a5*a6 + 8*a0*a3*a4*a5*a6 - 16*a0*a1*a3*a4*a5*a6 -...
  16*a0*a2*a3*a4*a5*a6 + 32*a0*a1*a2*a3*a4*a5*a6;


p6=(-(a0*a5) + 2*a0*a1*a5 + 2*a0*a2*a5 - 4*a0*a1*a2*a5 + 2*a0*a3*a5 - ...
    4*a0*a1*a3*a5 - 4*a0*a2*a3*a5 + 8*a0*a1*a2*a3*a5 + 2*a0*a4*a5 - ...
    4*a0*a1*a4*a5 - 4*a0*a2*a4*a5 + 8*a0*a1*a2*a4*a5 - 4*a0*a3*a4*a5 + ...
    8*a0*a1*a3*a4*a5 + 8*a0*a2*a3*a4*a5 - 16*a0*a1*a2*a3*a4*a5 - a1*a6 + ...
    2*a1*a2*a6 + a0*a1*a2*a6 + 2*a1*a3*a6 + a0*a2*a3*a6 - 4*a1*a2*a3*a6 - ...
    4*a0*a1*a2*a3*a6 + 2*a1*a4*a6 - 4*a1*a2*a4*a6 - 2*a0*a1*a2*a4*a6 + ...
    a0*a3*a4*a6 - 4*a1*a3*a4*a6 - 2*a0*a1*a3*a4*a6 - 4*a0*a2*a3*a4*a6 + ...
    8*a1*a2*a3*a4*a6 + 12*a0*a1*a2*a3*a4*a6 + 2*a1*a5*a6 - ...
    4*a1*a2*a5*a6 - 2*a0*a1*a2*a5*a6 - 4*a1*a3*a5*a6 - 2*a0*a2*a3*a5*a6 + ...
    8*a1*a2*a3*a5*a6 + 8*a0*a1*a2*a3*a5*a6 + a0*a4*a5*a6 - ...
    4*a1*a4*a5*a6 - 2*a0*a1*a4*a5*a6 - 2*a0*a2*a4*a5*a6 + ...
    8*a1*a2*a4*a5*a6 + 8*a0*a1*a2*a4*a5*a6 - 4*a0*a3*a4*a5*a6 + ...
    8*a1*a3*a4*a5*a6 + 8*a0*a1*a3*a4*a5*a6 + 12*a0*a2*a3*a4*a5*a6 -...
    16*a1*a2*a3*a4*a5*a6 - 32*a0*a1*a2*a3*a4*a5*a6);

p5=(-(a0*a4) + 2*a0*a1*a4 + 2*a0*a2*a4 - 4*a0*a1*a2*a4 + 2*a0*a3*a4 - ...
    4*a0*a1*a3*a4 - 4*a0*a2*a3*a4 + 8*a0*a1*a2*a3*a4 - a1*a5 + ...
    2*a1*a2*a5 + a0*a1*a2*a5 + 2*a1*a3*a5 + a0*a2*a3*a5 - 4*a1*a2*a3*a5 - ...
    4*a0*a1*a2*a3*a5 + 2*a1*a4*a5 - 4*a1*a2*a4*a5 - 2*a0*a1*a2*a4*a5 + ...
    a0*a3*a4*a5 - 4*a1*a3*a4*a5 - 2*a0*a1*a3*a4*a5 - 4*a0*a2*a3*a4*a5 + ...
    8*a1*a2*a3*a4*a5 + 12*a0*a1*a2*a3*a4*a5 - a2*a6 + a0*a1*a3*a6 + ...
    2*a2*a3*a6 + a1*a2*a3*a6 + 2*a2*a4*a6 + a0*a2*a4*a6 - ...
    2*a0*a1*a2*a4*a6 + a1*a3*a4*a6 - 2*a0*a1*a3*a4*a6 - 4*a2*a3*a4*a6 - ...
    4*a1*a2*a3*a4*a6 - a0*a1*a2*a3*a4*a6 + 2*a2*a5*a6 + a0*a3*a5*a6 - ...
    4*a0*a1*a3*a5*a6 - 4*a2*a3*a5*a6 - 2*a0*a2*a3*a5*a6 - ...
```

```
    2*a1*a2*a3*a5*a6 + 4*a0*a1*a2*a3*a5*a6 + a1*a4*a5*a6 - ...
    4*a2*a4*a5*a6 - 2*a0*a2*a4*a5*a6 - 2*a1*a2*a4*a5*a6 + ...
    3*a0*a1*a2*a4*a5*a6 - 4*a1*a3*a4*a5*a6 + 4*a0*a1*a3*a4*a5*a6 +...
    8*a2*a3*a4*a5*a6 - a0*a2*a3*a4*a5*a6 + 12*a1*a2*a3*a4*a5*a6 + ...
    6*a0*a1*a2*a3*a4*a5*a6);

p4=(-(a0*a3) + 2*a0*a1*a3 + 2*a0*a2*a3 - 4*a0*a1*a2*a3 - a1*a4 + ...
    2*a1*a2*a4 + a0*a1*a2*a4 + 2*a1*a3*a4 + a0*a2*a3*a4 - 4*a1*a2*a3*a4 - ...
    4*a0*a1*a2*a3*a4 - a2*a5 + a0*a1*a3*a5 + 2*a2*a3*a5 + a1*a2*a3*a5 + ...
    2*a2*a4*a5 + a0*a2*a4*a5 - 2*a0*a1*a2*a4*a5 + a1*a3*a4*a5 - ...
    2*a0*a1*a3*a4*a5 - 4*a2*a3*a4*a5 - 4*a1*a2*a3*a4*a5 - ...
    a0*a1*a2*a3*a4*a5 - a3*a6 + a0*a1*a4*a6 + a1*a2*a4*a6 + 2*a3*a4*a6 + ...
    a2*a3*a4*a6 + a0*a2*a5*a6 - 2*a0*a1*a2*a5*a6 + 2*a3*a5*a6 + ...
    a1*a3*a5*a6 - 2*a1*a2*a3*a5*a6 - a0*a1*a2*a3*a5*a6 - ...
    2*a0*a1*a4*a5*a6 + a2*a4*a5*a6 - 2*a1*a2*a4*a5*a6 - 4*a3*a4*a5*a6 - ...
    a0*a1*a3*a4*a5*a6 - 4*a2*a3*a4*a5*a6 - a1*a2*a3*a4*a5*a6) ;

p3=(-(a0*a2) + 2*a0*a1*a2 - a1*a3 + 2*a1*a2*a3 + a0*a1*a2*a3 - a2*a4 + ...
    a0*a1*a3*a4 + 2*a2*a3*a4 + a1*a2*a3*a4 - a3*a5 + a0*a1*a4*a5 + ...
    a1*a2*a4*a5 + 2*a3*a4*a5 + a2*a3*a4*a5 - a4*a6 + a0*a1*a5*a6 + ...
    a1*a2*a5*a6 + a2*a3*a5*a6 + 2*a4*a5*a6 + a3*a4*a5*a6);

p2=(-(a0*a1) - a1*a2 - a2*a3 - a3*a4 - a4*a5 - a5*a6) ;

p1=1;
        p(:,i)=roots([p1 p2 p3 p4 p5 p6 p7]);
end

%pole zero plot
f=[0:.01:1];
j=sqrt(-1);
clg
circle=exp(j*2*pi*f);
%axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on
plot(real(z),imag(z),'or')
plot(real(p),imag(p),'.r')
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.7 Miscallanous

A.7.1 impresp.m

```
function [out]=impresp(z,p,length)
% [out]=impresp(z,p,length)
%       Gives the impulse response of length [length] given the poles
%       and zeros

in=[1 zeros(size(1:length-1))]; %generates input impulse
out=filter(z,p,in);                     %generates output
plot(out,'o')
```

A.7.2 meshcc.m

```
function out=meshcc(m,alt)
%function out=meshcc(m,alt)
%MESHCC     Generates a plot using MESHC where the resulting contout plot has
%       lines which match the MESH plot.  The ALT variable determines the
%       altitude of the contour plot.
%
%       A similar effect can be achieved for SURFC and other similar plots
%       by changeing the first line of this function.
%
%       Calls MESHALT.M
%
%       Copied from User's Manual 2-111.
%       Integrates functions FINDLINE and CHANGELN

h=meshalt(m,alt);       %may change to surfc or other command
                        %the alt variable determines the altitude of the
                        %contour plot

l=[];
for jj=h'
        if strcmp(get(jj,'Type'),'line')
                l=[l;jj];
        end
end

h=l;

lh=length (h);
map=colormap;
lm=length(map);
for jj=1:lh
        set(h(jj),'Color',map(jj*lm/lh,:))
end
```

```
if nargout>0
        out=h;
end
```

A.7.3 meshalt.m

```
function h=meshalt(x,alt,y,z,c)
%MESHALT Combination MESH/CONTOUR plot.
%       MESHC(...) is the same as MESH(...) except that a contour plot
%       is drawn beneath the mesh.
%
%       Because CONTOUR does not handle irregularly spaced data, this
%       routine only works for surfaces defined on a rectangular grid.
%       The matrices or vectors X and Y define the axis limits only.
%
%       See also MESH.
%
%       Adopted from MESH.m
error(nargchk(1,5,nargin));

if nargin==1 | nargin==2,          % Generate x,y matrices for surface z.
  z = x;
  [m,n] = size(z);
  [x,y] = meshgrid(1:n,1:m);

elseif nargin==3,
  z = x; c = y;
  [m,n] = size(z);
  [x,y] = meshgrid(1:n,1:m);

end

if min(size(z))==1,
  error('The surface Z must contain more than one row or column.');
end

% Determine state of system
cax = newplot;
next = lower(get(cax,'NextPlot'));
hold_state = ishold;

% Plot mesh.
if nargin==3 | nargin==5,
  hm=mesh(x,y,z,c);
else
  hm=mesh(x,y,z);
end

hold on;

a = get(gca,'zlim');
```

```
zpos = alt; % Always put contour below the plot at an altitude of alt.



% Get D contour data
[cc,hh] = contour3(x,y,z,6);        %settings for optimal contour plot
                                    %sing: 5
                                    %asym: 6

%%% size zpos to match the data

for i = 1:length(hh)
        zz = get(hh(i),'Zdata');
        set(hh(i),'Zdata',zpos*ones(size(zz)));
end

if ~hold_state, set(cax,'NextPlot',next); end
if nargout > 0
        h = [hm; hh(:)];
end
```

A.7.4 zplane.m

```
function zplane(b,a)
%ZPLANE
%       zplane (b,a) b=numerator a=denominator

%plots the unit circle
f=[0:.01:1];
j=sqrt(-1);
hold off
circle=exp(j*2*pi*f);
axis(1.5*[-1 1 -1 1]);
axis('square')
plot(circle)
hold on

%plots the roots
z=roots(b);
p=roots(a);
plot(real(z),imag(z),'o')
plot(real(p),imag(p),'xr')

%labels axes
title('Pole Zero Plot of Filter')
ylabel('Im(z)')
xlabel('Re(z)')
hold off
```

A.7.5 movspeed.m

```
%generates a box to control movie speed and repetition
%calls upon UPDATE.M

figure('position',[200 250 260 100],'name','Movie Speed Control Box',...
        'numbertitle','off')        %sizes and places the figure win
                                    %[200 250 (location) 260 100 (size)]
                                    %also gives figure box a title

val=10; %sets the initial movie speed

rep=uicontrol('style','pushbutton','position',[150 30 90 20],...
        'callback','a=val','horizontalalignment','center','string','Repeat');
        %puts the button in for repeating
        %callback calls the movie command

h=uicontrol('style','slider','position',[20 30 100 20],'max',20,...
        'min',1,'Value',val,'callback','update');
        %puts the slider in for movie speed
        %callback call M-file "update.m" to update the text

val=fix(val);                       %determines the movie speed in integers
v=num2str(val);                     %converts val to a string

set (axes,'visible','off')   %makes axes invisible

text('units','pixels','position',[30 60],'string','Frames per Second:')
text('units','pixels','position',[150 60],'string',v) %# FPS

%places the text in the window in the proper position
```

A.7.6 update.m

```
%called by MOVSPEED.M
%update the text that denotes movie speed

a=get (gca,'children');   %gets the handle for the movie speed number
delete (a(1))             %deletes old handle

val=get(h,'value');       %gets the new movie speed from position of slider
val=fix(val);             %determines the movie speed in integers
v=num2str(val);           %converts val to a string

text('units','pixels','position',[150 60],'string',v)
                          %writes the new movie speed
```

VITA

David Sun was born in Washington, DC on October 16, 1970 to Jack and Teresa Sun. He graduated from Virginia Tech with a B.S. in electrical engineering in June 1992. Immediately following, he began as a full time graduate student with the Fiber and Electro• Optics Research Center where he had been working as an undergraduate. Mr. Sun received his M.S. in electrical engineering in May 1994.

Mr. Sun's hobbies include photography, aquariums and SCUBA diving. He is currently a SCUBA instructor and the President of David Sun, Inc., a business dedicated to training of SCUBA divers.

Mr. Sun's professional interests include optical fiber sensing, research management and business investing. He is also a member of the Optical Society of America.