1991

Design of one-dimensional and two-dimensional FIR digital filters with integer coefficients.

Hassan. Nivi
University of Windsor

DESIGN OF 1-D AND 2-D FIR DIGITAL FILTERS

WITH INTEGER COEFFICIENTS

by
Hassan Nivi

A THESIS
Submitted to the Faculty of Graduate Studies through the
Department of Electrical Engineering in Partial Fulfilment
of the Requirements for the Degree of
Master of Applied Science at the
University of Windsor

Windsor, Ontario
1991

ISBN 0-315-72784-5

To My Wife

And

My Daughter

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to my supervisor, Dr. M. Ahmadi for his guidance and encouragement over the course of this research. Also, the critical comments, suggestions and evaluations of Dr. M. Shridhar, Dr. J.J. Soltis and Dr. R. Lashkari are greatly appreciated. The help of faculty members and computing consultants at the Computer Centre are greatly appreciated.

ABSTRACT

In the process of designing a digital filter, the coefficients of the transfer function are normally evaluated with a high degree of accuracy. To implement the designed filter in hardware, finite word length registers have to be utilized. This may bring about truncation of the filter coefficients to the limited word length registers employed, which in turn may affect the total response of the digital filter. To remedy this, filters are designed with the finite word length coefficients.

In this thesis a review of existing techniques for the design of integer coefficients 1-D FIR filters is presented. Where possible extension of these techniques to 2-D are also presented. In this thesis two design methods for 1-D and 2-D FIR filter with integer coefficients based on the suboptimal approach of discretization and reoptimization techniques are presented. These two approaches are somewhat modified versions of the well known branch and bound optimization technique. This thesis also presents a technique for the design of 1-D and 2-D FIR filters with integer coefficients using Mixed integer linear programming. Delta modulation is also employed for a new filter structure for 1-D FIR filter with its coefficients being -1, 0, and +1. McClallan transformation is used for designing a class of 2-D FIR filters with integer coefficients and a suitable realization structure.

In this thesis, we also present a comparison of various techniques we have proposed in terms of complexity of the design issue, as well as the computational burden for each iteration.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1. INTRODUCTION

Digital signal processing, a field which has it roots in the 17th and 18th centuries, has become an important tool in a multitude of diverse fields of science and technology. The techniques and applications of this field are as old as"Newton and Gauss" and as new as digital computers and integrated circuits. Each year, as integrated circuits have become faster, cheaper and more compact, it has become more possible to find feasible solutions to problems of ever-increasing complexity.

Simply stated, a signal is any medium for conveying information and digital signal processing is concerned with the representation of signals by sequences of numbers or symbols, the processing of these sequences, and extraction of that information contained in these sequences. The purpose of such processing may be to estimate characteristic parameters of a signal or to transform a signal into a form which is in some sense more desirable.

In one dimension the most useful applications of signal processing are in speech communication, either to enhance the intelligibility or to reduce the noise, data communication, biomedical engineering acoustic, sonar, radar, seismology, nuclear science, and many others. Signal processing problems are not isolated to one dimensional signals. Many picture processing applications require the use of two dimensional signal processing techniques. This is the case in x-ray enhancement and analysis of aerial photographs for detection of forest fires or crop damage, the analysis of satellite weather photos, and enhancement of Television Transmissions from Lunar and deep-space probes.

1.2. DIGITAL FILTERING

A major subdivision of Digital Signal Processing is digital filtering. A digital filter in 1-D or 2-D is a computational process or an algorithm by which a 1-D or 2-D digital signal or sequence of numbers (acting as input) is transformed into a 1-D or 2-D sequence according to some desired specification. It may involve amplifying or attenuating a range of frequency components, rejecting or isolating one specific frequency component, etc.

Digital filters are characterized in terms of difference equations. Two types of digital filters can be identified: non-recursive and recursive filters.

1.2.1. NONRECURSIVE SYSTEM

In the case of a digital nonrecursive system, the output sequence is defined as the weighted sum of the input sequence over a number of preceding samples. Thus in one dimension, the output y(n) written as a function of the input x(n) is given by

$$y(n) = \sum_{i=0}^{N} a(i)x(n-i) \tag{1-1}$$

where a(i) is a weighting factor on the various inputs, and N is the order of the filter.

In two dimensions this becomes

$$y(m,n) = \sum_{i=0}^{M} \sum_{j=0}^{N} a(i,j)\, x(m-i, n-j) \tag{1-2}$$

Nonrecursive filter is also called FIR filter because the impulse response is of finite duration.

1.2.2. RECURSIVE SYSTEM

In this case the present output is not only a function of the present and past value of the input, but also depends on the past value of the output. This condition can be written as

$$y(n) = \sum_{i=0}^{N_1} a(i)\, x(n-i) - \sum_{i=1}^{N_2} b(i)\, y(n-i) \tag{1-3}$$

and in case of 2-D

$$y(m,n) = \sum_{i=0}^{M_1} \sum_{j=0}^{N_1} a(i,j)\, x(m-i,n-j) - \sum_{i=0}^{M_2} \sum_{j=0\; i+j\neq 0}^{N_2} b(i,j)\, y(m-i,n-j) \tag{1-4}$$

A recursive filter is also called IIR (Infinite Impulse Response) because the impulse response is of infinite duration.

1.2.3. COMPARISON BETWEEN FIR AND IIR FILTER

The choice between an FIR filter and an IIR filter is the first question in the filter design process which has to be answered. The answer depends upon the relative weight of the advantages and disadvantages of each type of filter. In recursive filters the poles of the transfer function can be placed anywhere inside the unit circle. This degree of freedom permits us to meet the specification with the low order filter whereas, in the case of FIR filters, all the poles are fixed at the origin. This restriction would force the designer to use a higher order filter compared to the IIR filter in order to satisfy the same specifications.

A FIR filter has a finite number of nonzero samples and hence the impulse responses are always absolutely summable. Thus FIR filters are always stable whereas the stability condition in IIR filters is one of the most important issues in designing a filter. Finally, one of the important advantages of FIR filters over IIR filters is it is able to provide exact linear phase.

Stability and linear phase are two important characteristics of the FIR filter which account for the great advantage that FIR filters over IIR filters in one and two dimensions.

1.3. CHARACTERISTICS OF 1-D AND 2-D FIR FILTERS

1.3.1. CHARACTERISTICS OF 1-D FIR FILTER

The transfer function of a causal FIR filter is as follows.

$$H(z) = \sum_{i=0}^{N-1} h(i)\, z^{-i} \tag{1-5}$$

where N is the order of the filter. The frequency response is given by

$$H(e^{j\omega}) = \sum_{i=0}^{N-1} h(i)\, e^{-j\omega i} \tag{1-6}$$

For a constant phase and group delay the following condition has to be satisfied

$$h(i) = h(N-1-i) \quad for \quad 0 \le i \le N \tag{1-7}$$

Then the frequency response for an odd filter is presented as follows:

$$H(e^{j\omega}) = a(0) + 2\sum_{i=1}^{\frac{N-1}{2}} a(i)\cos(\omega i) \tag{1-8}$$

1.3.2. CHARACTERISTICS OF THE 2-D FIR FILTER

The z transform of a causal 2-D FIR filter is as follows:

$$H(z_1, z_2) = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} h(i,j)\, z_1^{-i} z_2^{-j} \tag{1-9}$$

To reduce the number of coefficients which have to be determined, the following symmetries can be considered.

a) Quadrantal Symmetry [1]

In this kind of symmetry, the following equality in the frequency and spatial domain are established.

$$|H(e^{j\omega_1}, e^{j\omega_2})| = |H(e^{-j\omega_1}, e^{j\omega_2})| = |H(e^{j\omega_1}, e^{-j\omega_2})| = |H(e^{-j\omega_1}, e^{-j\omega_2})| \tag{1-10}$$

$$h(n_1,n_2)=h(n1,N_2-1-n_2)=h(N_1-1-n_1,n_2)=h(N_1-1-n_1,N_2-1-n_2) \quad (1\text{-}11)$$

The frequency response corresponding to these conditions is as follows:

$$H(e^{j\omega_1},e^{j\omega_2}) = \sum_{i=0}^{\frac{N_1-1}{2}} \sum_{j=0}^{\frac{N_2-1}{2}} a(i,j)\cos(\omega_1 i)\cos(\omega_2 j) \quad (1\text{-}12)$$

Equation (1-10) demonstrates the reduction of the number of grid points for error function evaluation roughly by half (compared to central symmetry) and equation (1-11) shows the reduction of the number of variables from $(N_1)(N_2+1)/2$ to $(N_1+1)(N_2+1)/4$.

b) Octagonal Symmetry [1]

For the quadrantal symmetric 2-D filter to be octagonal symmetric 2-D filter the following equality in the frequency responses is established.

$$|H(e^{j\omega_1},e^{j\omega_2})| = |H(e^{j\omega_2},e^{j\omega_1})| \quad (1\text{-}13)$$

and for its spatial response we have

$$h(n_1,n_2)=h(n_2,n_1) \quad (1\text{-}14)$$

The number of grid points for error function evaluation is roughly reduced by half compared to quadrantal symmetry. The number of variables which have to be determined is reduced from $(N^2+N)/2$ to $(N+1)(N+3)/8$.

The frequency response in eight folded symmetry is defined by:

$$H(e^{j\omega_1}, e^{j\omega_2}) = \sum_{n=0}^{N-1} a(n,n)\cos(\omega_1 n)\cos(\omega_2 n) + \sum_{n=1}^{N-1}\sum_{m=0}^{n-1} a(n,m)\,[\cos(\omega_1 n)\cos(\omega_2 m) + \cos(\omega_1 m)\cos(\omega_2 n)] \tag{1-15}$$

1.4. COEFFICIENT QUANTIZATION ERROR

The coefficients of the transfer function are evaluated with a high degree of accuracy. In hardware implementation of any designed filter this coefficient has to be presented in a limited word length. To reduce the length of the coefficients to the desired one, we truncate the coefficients. If the quantization step is coarse, the filter may actually fail to meet the desired specifications. Coefficients-quantization error introduces perturbations in the zeros of the transfer function of FIR filter, which causes error in the frequency response. In the study of arithmetic errors, we shall be concerned with the sensitivities of digital filter networks to variations in the multiplier constants. In the network characterized by

$$H(z) = F(a_1, a_2, \ldots, z) \tag{1-16}$$

where a_1, a_2, ... are multiplier constants or the coefficients of the filter and $H(z)$ is the transfer function of the filter, the sensitivities are defined by

$$S_{a_i}(z) = \frac{\partial H(z)}{\partial a_i} \tag{1-17}$$

In the study of arithmetic errors, the sensitivity analysis plays an important role. To reduce the quantization error, we can design a filter with finite word length in such a way that the prescribed specifications are satisfied. To achieve this goal, various techniques have been presented in the literature. In this study, we explore those techniques, modify them, introduce a new technique and finally extend these techniques for designing 2-D FIR filters.

1-5 DESIGN TECHNIQUES FOR 1-D FIR FILTERS

The impulse response of a FIR filter can be obtained from the frequency response through the inverse Fourier transformation. The impulse response obtained is of infinite duration. The most straightforward approach to FIR filter design is to truncate these infinite-duration impulse response sequences. This truncation eliminates the components of the impulse response beyond certain points in n_1 and n_2 directions. This truncation unfortunately introduces an unwanted oscillation in the passband and stopband of the filters known as Gibbs'oscillations.

Instead of simply truncating the infinite Fourier series, the technique of windowing seeks to reduce the Gibbs phenomenon by multiplying the coefficients of the Fourier series by a smooth time-limited window. Among the more popular windows are the Kaiser window [7], the Hamming window [9], the Hann window [9], and the Dolph-Chebyshev window [10]-[13]. One of the attributes of windowing is that it is an analytical technique, whereas most other FIR design techniques are iterative in nature.

Windowing techniques generally enjoys a short design time but the results are not necessary optimal. The application of the Kaiser window in case of sharp transition band results usually in filters not meeting the specifications. Ahmadi [8] proposed a modified technique based on the Kaiser window to improve the above short comings.

A second FIR design technique (the frequency sampling method) was originally proposed by Gold and Jordan [14] and further developed by Rabiner et al. [15]. In this technique, the desired frequency responses are specified at equally spaced frequencies, and solved for the unit sample response h(n) of the FIR filter from these equally spaced frequency specifications. It is also desirable to optimize the frequency specification in the transition band of the filter. This problem can be shown to be a linear programming problem with very few independent variables, but a large number of constraints.

Hermann [16] was the first to develop a method for designing optimal (in a Chebyshev sense) FIR filters. By assuming that the frequency response of the optimal low-pass filter was equiripple in both the passband and the stopband, and by fixing the number of ripples in each band, Herrmann was able to write down a set of nonlinear equations which completely described the filter. He then proceeded to solve these equations directly, using a steepest descent method. The length of filters designed in this manner was limited to about 40.

The interpolating technique proposed by Hofstetter et al. [17], [18] offers a more efficient algorithm based on the Remez exchange algorithm capable of designing higher order filters.

Parks and McClellan [19] formulated the lowpass approximation of the desired response on two disjoint intervals, the passband and the stopband with a transition band left unspecified. Necessary and sufficient condition for the best Chebyshev approximation were obtained from the classical alternation theorem, and the Remez exchange algorithm was demonstrated to be an effective tool for the computation of these optimal filters. Subsequently, this formulation was extended to include all types of linear phase FIR filters [22].

Rabiner [20], [21] showed that linear programming offers an alternative method for computing the best Chebyshev approximation. Although linear programming is very flexible and can be used to approximate a wide variety of desired filter shapes, it is comparatively slow and hence the length of the filters it can design is limited.

1-6 DESIGN TECHNIQUES FOR 2-D FIR FILTERS

Huang [23] proposed an extension of windowing technique in 1-D FIR filter design to 2-D FIR filter design. Perhaps the simplest technique with the shortest design time is the windowing technique, but it gives suboptimal results. There is an improved method for the design of 2-D FIR filters with a circular and rectangular cut-off boundary using the Kaiser technique presented by Ahmadi [24]. This method utilizes the well-known Fourier series method in conjunction with the Kaiser window. A nonlinear programming is employed to obtain the parameter α. The result shows that the proposed method requires a filter with a lower order than direct use of the Kaiser window. It is restricted only to the design of lowpass circular or rectangular cut-off boundary FIR digital filter.

Using the linear programming technique, both frequency sampling and optimal (in the sense of Chebyshev approximation) 2-D filters have been successfully designed by Hu and Rabinar [25]. Computational considerations have limited the filter impulse response durations by 25 x 25 in the frequency sampling case and to 9 x 9 in the optimal design case. This method also requires an extremely long design time.

The Remez exchange algorithm is extended for the design of 2-D FIR filter approximating circularly symmetrical low-pass specification according to a weighted Chebyshev error norm. Since the approximating function does not necessarily satisfy the Haar condition, the optimal solution is not necessarily unique and a straightforward extension of the 1-D exchange method may fail to converge. It is shown by Kamp and Thiran [26] how the algorithm has to be complemented with a perturbation technique in order to force convergence under all circumstances.

One of the most popular techniques for designing a 2-D FIR filter is the McClellan transformation which was first developed by McClellan [27], then generalized by Mersereau et al. [28] for the higher order of transformation equation and elliptical cut-off boundary. This method is based on the transformation of the 1-D FIR filter to 2-D FIR filter by means of variable transformation. Design process in this technique splits into two steps. First, the transformation parameters have to be determined [28], [29], [30]. Second, the coefficients for the 1-D FIR prototype have to be obtained by one of the design techniques of 1-D FIR filter. This technique enjoys a short design time. The efficient implementation exists in [31]. Also, this technique can not be used to approximate closely all magnitude functions. The result of this technique is not optimal in the minimax sense.

John H. Lodge and M. Fahmy proposed an iterative optimization technique, based on the method of parallel tangents (PARTAN) [32] coupled with an efficient line searching technique for designing a 2-D FIR filter. The technique developed from the observation that in a 2 parameter optimization problem the minimum point often lies along the lines that join the zigzags of the steepest descent path. PARTAN attempts to improve the convergence by adding an extra line search for each iteration. The gradient vector has to be calculated for every iteration. Optimization starts by finding an initial approximation using one of the available windowing techniques. The l_p norm is employed to approximate a prescribed magnitude. Larger values of p can be used to put more emphasis on reducing the maximum error. The design time and computational complexity increase very slowly as the filter size gets larger.

The Weighted Least Square (WLS) Design technique was proposed by V. Algazi et al. [33]. This technique draws attention to a formal relation of the WLS to the Chebyshev approximations and exploits this relation to develop efficient algorithms modifying Lawson's algorithm [34] for the design of minimax filters in one and two dimensions. In 1-D, the algorithm converges exactly to the Chebyshev (and equiripple) approximation and it provides an alternative to the Remez exchange algorithm. The extension to 2-D is straightforward and the implementation quite efficient because of the simplicity of the WLS design technique. However, the algorithm lacks the proof of convergence for optimal solution to the minimax design [35].

A new technique for the design of 2-D FIR and IIR filters is due to Lampropoulos and Fahmi [36]. The problem is formulated for FIR and IIR filters in such a way that the performance index is a convex function of the optimization vector. The design technique is based on Newton's method. Each iteration consists of N easily computed steps where N is the number of parameters to be optimized. The proposed method enjoys a very high rate of convergence and short design time compared with the most efficient techniques now available [36]. The number of operations per iteration for the proposed technique is less than the other methods of the same family (Newton's method). It is highly insensitive to the increase in the dimensionality of the optimization vector and because of that it can be used for designing very high order digital filters. It can be used also to design 2-D filters with the coefficients having a desired finite wordlength.

1.7. THESIS ORGANIZATION

The thesis is divided into five chapters. The second chapter consists of a brief introduction on implementation problems with infinite precision coefficients and a survey on possible design techniques for a 1-D FIR filter with integer coefficients. Three techniques are chosen for further investigation and modification. A new technique based on discretization and reoptimization is introduced. Each design technique is explained fully, and a number of examples at the end of each section are provided and the results of these examples are used for a comparative study.

Chapter three starts with the survey on the 2-D design technique with integer coefficients, and it is followed by a direct extension of the discretization & reoptimization (type 1&2) from 1-D to 2-D in filters with circular symmetry . The problems with direct extension are discussed and the necessity of having a more practical technique is considered. The McClellan transformation is considered as an alternative. The characteristics of this transformation and design steps for the original McClellan transformation are discussed. The original McClellan transformation with the circular cut-off boundary is considered for this design technique. The transformation parameters, t, are fixed to 1/2 for circular symmetry. By considering this fact, no multipliers are required for the implementation.

Chapter four investigates the possible realization technique for 1-D and 2-D FIR filters. The implementation of the 1-D prototype filter employed in the McClellan transformation and the 2-D designed filter using this technique are explained.

The final Chapter concludes the thesis by stating the results derived in the previous chapters. The flowchart of the design technique one, discretization and reoptimization (type 1), is provided in appendix A. In appendix B, the flowchart of design technique 2, discretization and reoptimization type(2) is provided. Appendix C shows the linear and mixed integer programming procedure.

CHAPTER TWO

DESIGN OF THE 1-D FIR FILTER WITH INTEGER COEFFICIENTS

2.1. INTRODUCTION

The implementation of digital filters is achieved by storing sequence values and coefficients in a binary format with finite-word-length register. The finite -word-length constraint is manifested in a variety of ways. The coefficients of a digital filter designed by one of the techniques explained in chapter 1 are obtained with high accuracy. When these coefficients are quantized, the frequency response of the resulting digital filter may fail to meet the prescribed specifications. Therefore, it is often desirable to design filters with coefficients requiring fewer bits. To achieve this goal several techniques have been presented in the literature, some of which will be briefly explained in this section.

One of the simplest ways to design a filter with integer coefficients is by rounding the infinite precision coefficients which satisfy the prescribed specification. Although in some cases the resulting response is satisfactory, there is abundant evidence [37], [38] [39] that the performance of a rounded solution can be quite inferior to the optimal one. Another possible method can be achieved by letting the coefficients select the best rounding solution for themselves. The most direct method of solving is to design an infinite precision coefficients prototype and then consider rounding each coefficient value either up or down. This simple method will produce a discrete solution which is an immediate neighbour to the continuous solution. If a simple search is performed through all the possible cases of rounding the continuous-solution coefficient values either up or down for a problem with 128 coefficients, a total of 2^{128} filter frequency responses will have to be computed.

In addition, the possibility that none of these immediate neighbours of the continuous solution is able to meet a given specification exists. Thus any procedure of practical interest must evaluate a very much smaller

fraction of the possible discrete coefficient filter. At the same time, the approach must have a good chance of detecting a feasible (not necessarily the optimum) discrete solution, even if all the feasible discrete solution are located at a considerable distance from the infinite precision solution. To achieve these desires several techniques have been suggested to reduce the fraction of possible discrete coefficients filter and at the same time, coefficients are allowed to be optimized for obtaining a feasible solution.

The branch and bound technique for nonlinear discrete optimization due to Dakin [40] has been utilized by Charalambous and Best [41]. The proposed technique is used for the design of the recursive filter with finite word length coefficients. The technique can be employed for a FIR filter by removing some constraints for establishing the stability condition. The algorithm of Best and Ritter [42], [43] is used in this technique which finds the minimum of a nonlinear function subject to linear constraints. The technique starts with solving the problem for the infinite precision coefficient and then by the branching technique, new non-overlapping bounds for the coefficients (i) are applied to the problem. If the continuous solution for the coefficients (i) is x_i where

$$x_i^- < x_i < x_i^+ \tag{2-1}$$

where x_i^- and x_i^+ are integer neighbours, the new constraint will be

$$x_i^- > x_i$$

$$x_i > x_i^+ \tag{2-2}$$

The new feasible solution by considering new constraints is provided. This process is implemented for all branching elements and the best solution is then selected from among all the feasible discrete solutions. This technique needed the enormous computations such that they can be applied only to the design of filters of small orders. There is no guarantee for obtaining the optimal solution.

Another technique for designing a IIR filter based on discretization and reoptimization was suggested by Jing and Fam [44]. The infinite precision coefficients are obtained at the first step of design and after that the most sensitive coefficient is rounded up and down to its neighbours. A nonlinear optimization algorithm is employed for each of this integer's neighbours to find the new sets for the rest of the variable and the values of the error function in these two cases. The minimum value for error function allows us to fix the most sensitive coefficients to its corresponding integer value. This process continues until the value for all the coefficients becomes integer. Compared to previous techniques, this technique is much faster but still there is no guarantee for optimal solution. This technique will be fully explained in the following sections.

Mixed integer linear programming is another option which is suggested by Lawrence and Salazar [37] to design a linear phase FIR filter. This mixed integer programming allows the unknown coefficients to take on integer values while the stop-band and the pass-band ripples are allowed to be non-integer. The branch and bound algorithm [45] has been used in this technique. The formulation of the problem will be discussed in the following section. They also showed how to use zero-one integer formulation to design FIR filter. Zero-one integer programming is a special case of the integer linear programming problem. The decision variables for zero-one programming are restricted to two values, 0 or 1. The decision variables are the actual binary bits of the filter coefficients and can be presented as follows:

$$h_i = \sum_{q=0}^{B-1} h_{iq} 2^q \tag{2-3}$$

where B is the number of specified bits for representing the coefficients and h_{iq} is the decision variable for this formulation. Another kind of formulation can be done for the filters with powers of two coefficients. In this formulation, the filter coefficients are constrained to be zero or powers of two by adding the following constraint

$$\sum_{q=0}^{B-1} h_{iq} = 1 \tag{2-4}$$

The computation time and storage requirements were considerably reduced by using zero-one integer programming with constraint on the binary bits [37]. Although mixed integer programming has a better chance of getting a optimal solution compared to other techniques [41], [44], the computation cost increases exponentially with filter length; this fact restricts the highest possible order for the applying filter to 40.

FIR digital filters with discrete coefficient values selected from the power-of-two coefficient space are designed using the methods of integer programming by Lim and Parker [46]. In this technique coefficient values are expressed as a sum or difference of two power-of-two, which allows for a simple multiplierless implementation. Linear programming and quadratic programming algorithms are employed for minimizing a linear objective function (minimax error) or (weighted least square error) subject to linear constraints, respectively. Coupling these mathematical programming packages with a suitable branch and bound algorithm, the depth-first branch and bound search approach is the base concept of this approach. This technique can guarantee global optimality in the minimax sense. Unfortunately, the very high cost in computation and time limits its application to the design of linear phase FIR filters with $N < 40$.

Another efficient method in least mean square criteria is proposed by Lim and Parker [47]. In this technique an efficient optimization routine is incorporated into a tree search algorithm and a suitable branching policy is employed to optimize the remaining unquantized coefficients of a FIR linear phase filter when one or more of the coefficients takes on discrete value. The design procedure is as follows [47]. The tree structure for the case where $L=3$ is displayed in Figure (2-1).

Step 1: Design a filter with infinite precision coefficients.

Step 2: Select a coefficient and fix it at L different discrete values in the vicinity of the continuous solution.

Step 3: Employ the optimization routine to optimize the remaining unquantized coefficients for L problems.

Step 4: Are all coefficients discrete? If yes, stop. If not, go to step 5.

Step 5: Select a second coefficient and fix it at L different discrete values.

Step 6: Employ the optimization routine to optimize the remaining unquantized coefficients for L^2 problems.

Step 7: Select the best L problem and return to step 4.

The least squares criterion algorithm is suitable for designing FIR filters with $N \leq 90$ if the discrete coefficient grid is the power of two [48]. However some skilful operations to approximate the minimax criteria using the LMS criteria are necessary. Even then good filter characteristics are not ensured [48].

Bateman and Bedeliu [49] proposed the technique based on the delta modulation concept. The basic idea behind this approach is to convert the filter coefficients into a sequence by a delta modulation like scheme where instead of allowing only the values -1 and +1 the value zero is also allowed. As in delta modulation, over-sampling is required to keep the error at an acceptable level. This corresponds to interpolating the original impulse defined factor. Unfortunately, a characteristic of this structure is the very high sampling rate with the consequence that the number of taps is very large. In the following section several algorithms for the design of integer coefficients 1-D FIR filters will be explained and modified.

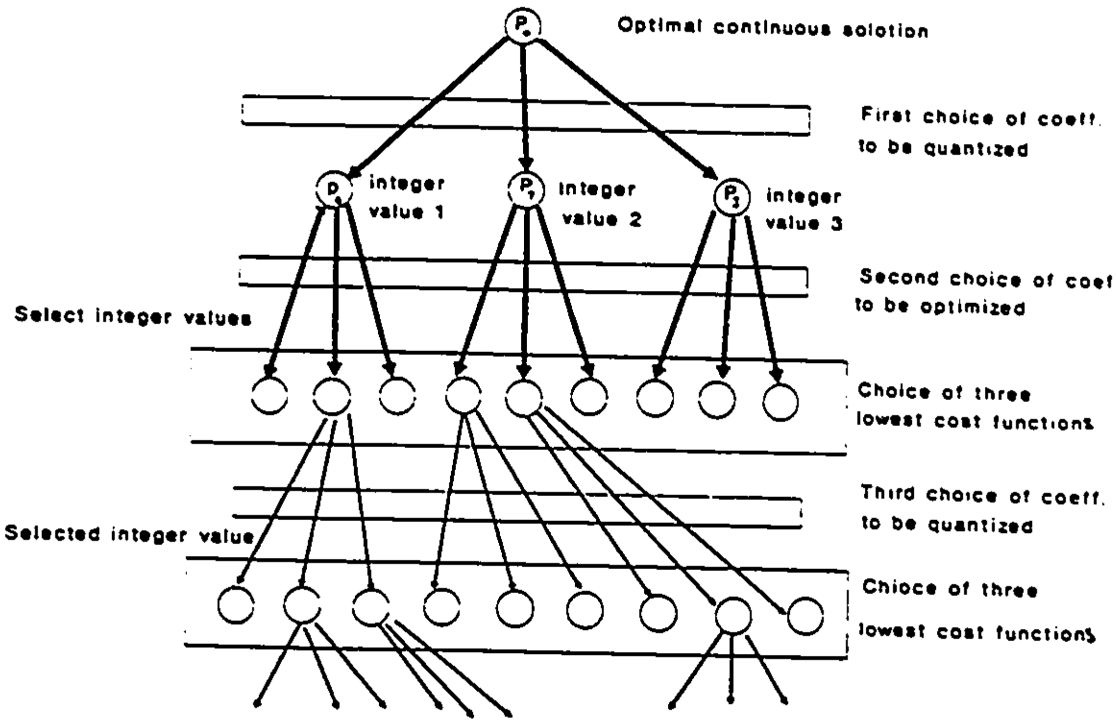


Figure (2-1) : Tree structure for the case where L=3

2.2. DISCRETIZATION AND REOPTIMIZATION (TYPE 1)

In this section, we introduce a technique to design a 1-D FIR filter with integer coefficients using nonlinear programming. The approach starts with an optimal design with infinite precision coefficients which will be called the Infinite Precision Coefficients design or IPC design throughout this chapter. The details of the proposed technique are as follows.

The whole process consists of an N times optimization process where N is the number of coefficients which has to be optimized through this optimization process. In each iteration the number of variables is reduced by one. Obviously, the total number of variables is equal to N-i where i is the iteration number. This technique for finding a fixed variable in each iteration is based on rounding up and down all the coefficients successively and searching for the one with the least effect on the cost function. Now, that specific variable can be fixed to its integer value and reoptimization can be done for the remaining variables.

In this approach, the least square error is the approximation criterion and nonlinear programming is the tool for solving this problem.

2.2.1. DESIGN STEPS

In this technique, the number of specified bits for representing the coefficients is the first data which has to be provided by the user. If the infinite precision coefficients of the designed filter are b_i, where $i=(N-1)/2$, N is the order of filter, if

$$-1 \leq b_i \leq +1 \tag{2-5}$$

then b_i can be shown by the following form

$$b_i = \frac{a_i}{2^{B-1}} \tag{2-6}$$

where B is the number of the bits including the sign bit and a_i is the integer coefficients for the prescribed filter. Equation 2-6 shows that for obtaining the integer coefficients the prescribed magnitude response has to be scaled by 2^{B-1}, but for arbitrary specification, other than equation (2-5), a scaling factor could be used in order to recover the desired range.

Various steps of the algorithm are as follows:

1) Find the infinite precision coefficients for the designed filter by nonlinear programming and the following least square error:

$$E = \sum_{\omega_j \in \Omega} [H_i(e^{j\omega_j}) - \sum_{i=0}^{\frac{N-1}{2}} a(i) \cos(\omega_j i)]^2 \qquad (2\text{-}7)$$

where a(i) is the coefficients of the designed filter, j is the number of frequency points for evaluation of the error function and $\omega \in [0, \pi]$

2) Set i=0

3) Set a (i) = a^+ (i) where a^+ (i) is the round up value for a(i) and find the value of error function, equation (2-5); the resulting value can be shown by E^+ (i)

4) Set a (i) = a^- (i) where a^- (i) is the round down value for a(i) and find the value of error function, equation (2-5); the resulting value can be shown by E^- (i)

5) Find E (i) = min{ E^+ (i), E^- (i)} and save the value

6) Set a (i) to its initial value and state i=i+1

7) Repeat steps 3-6 until i > N-1 / 2

8) Fix the value of the corresponding coefficient to the minimum error function calculated within steps 3-7 to a^+ (i) or a^- (i)

9) Go to step 1 and do the optimization routine to find a new set of values for the rest of the elements of vector A

$$A = (a(1), a(2), \ldots, a^{+}(i) \text{ or } a^{-}(i), \ldots, a(N-1/2)) \qquad (2\text{-}8)$$

The ith element of the vector A, a(i), has been fixed in step 8

10) repeat steps 1-9 until all components of the vector A convert to integer value.

In Appendix A, the flowchart of this algorithm is shown. In the following section several examples have been provided to illustrate the versatility of the design technique.

2.2.2. EXAMPLES

Example 1 : Design a low pass 1-D FIR filter with the order of 19, B=8, and the following specification:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 0 \leq \omega \leq 1 \\ 0 & 2 \leq \omega \leq \pi \end{cases}$$

Table 2-1 shows the infinite precision coefficients for example 1 using nonlinear programming. Table 2-2 shows the integer coefficients for designed filter using this technique and Table 2-3 shows the characteristics of design technique for example 1. Figures 2-2 and 2-3 show the magnitude response of IPC design and proposed technique 1, respectively.

a(0) = .4968E+00	a(3) = -.171E+00	a(6) = .2975E-02	a(9)
a(1) = .6223E+00	a(4) = -.411E-02	a(7) = -.254E-01	=.5939E-02
a(2) = .6199E-02	a(5) = .6906E-01	a(8) = -.147E-02	

Table 2-1: The infinite precision coefficients for example 1 using nonlinear programming

a(0) = 63	a(3) = -21	a(6) = 0	a(9) = 0
a(1) = 79	a(4) = 0	a(7) = -2	
a(2) = 0	a(5) = 8	a(8) = 0	

Table 2-2: The integer coefficients for example 1 using technique 1

DESIGN TYPE	#R.TIME(SEC)	M.M.P	M.M.S
IPC DESIGN	5	1.0015	.0014
D&R (TYPE 1)	150	.9961	.0126

Table 2-3: Characteristic of two design techniques in example 1

Example 2 : Design a high pass 1-D FIR filter with the order of 19, B=8, and the following specification:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 2 \le \omega \le \pi \\ 0 & 0 \le \omega \le 1.5 \end{cases}$$

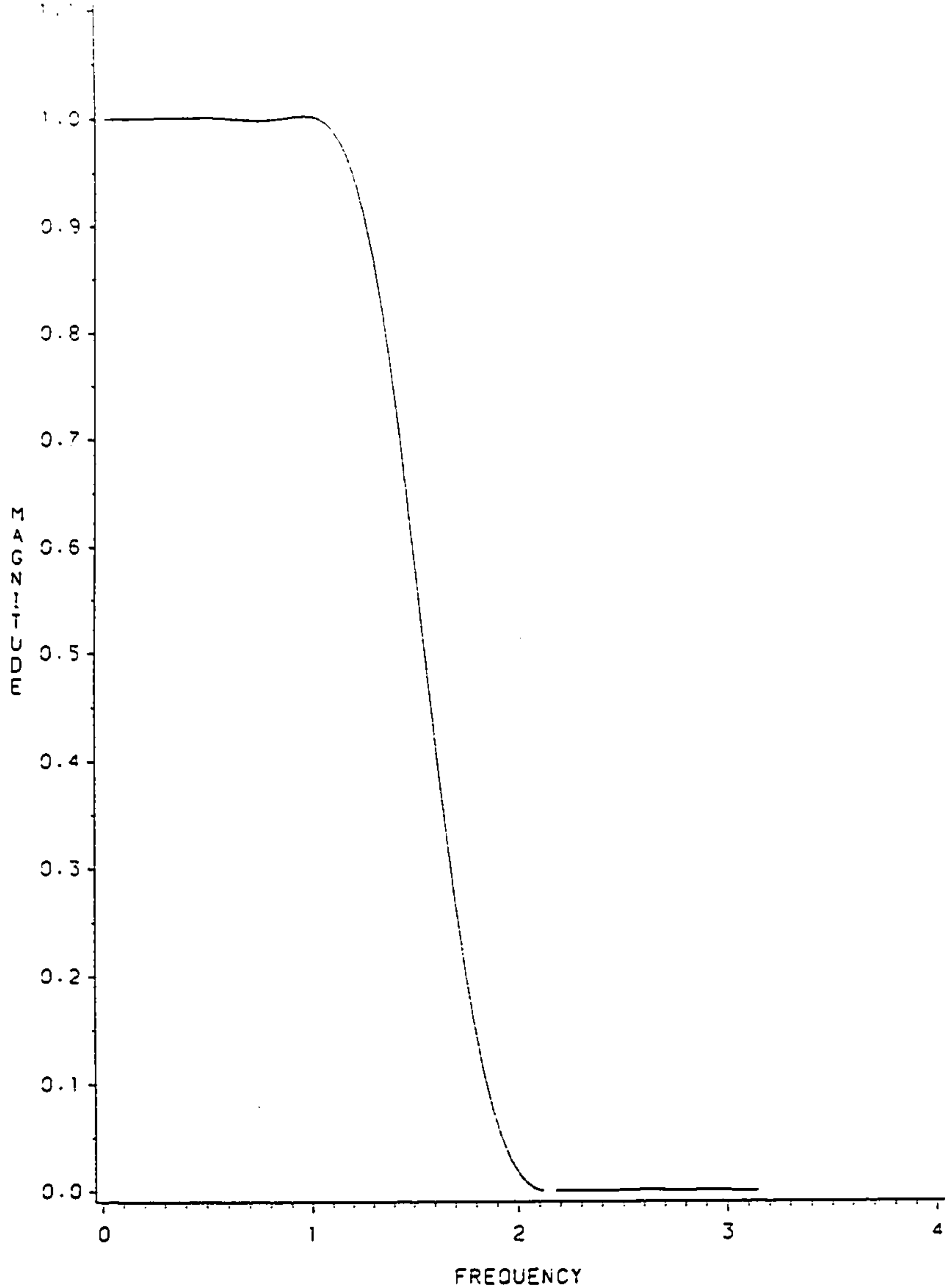


Figure (2-2) : The magnitude response for example 1 with infinite precision coefficients

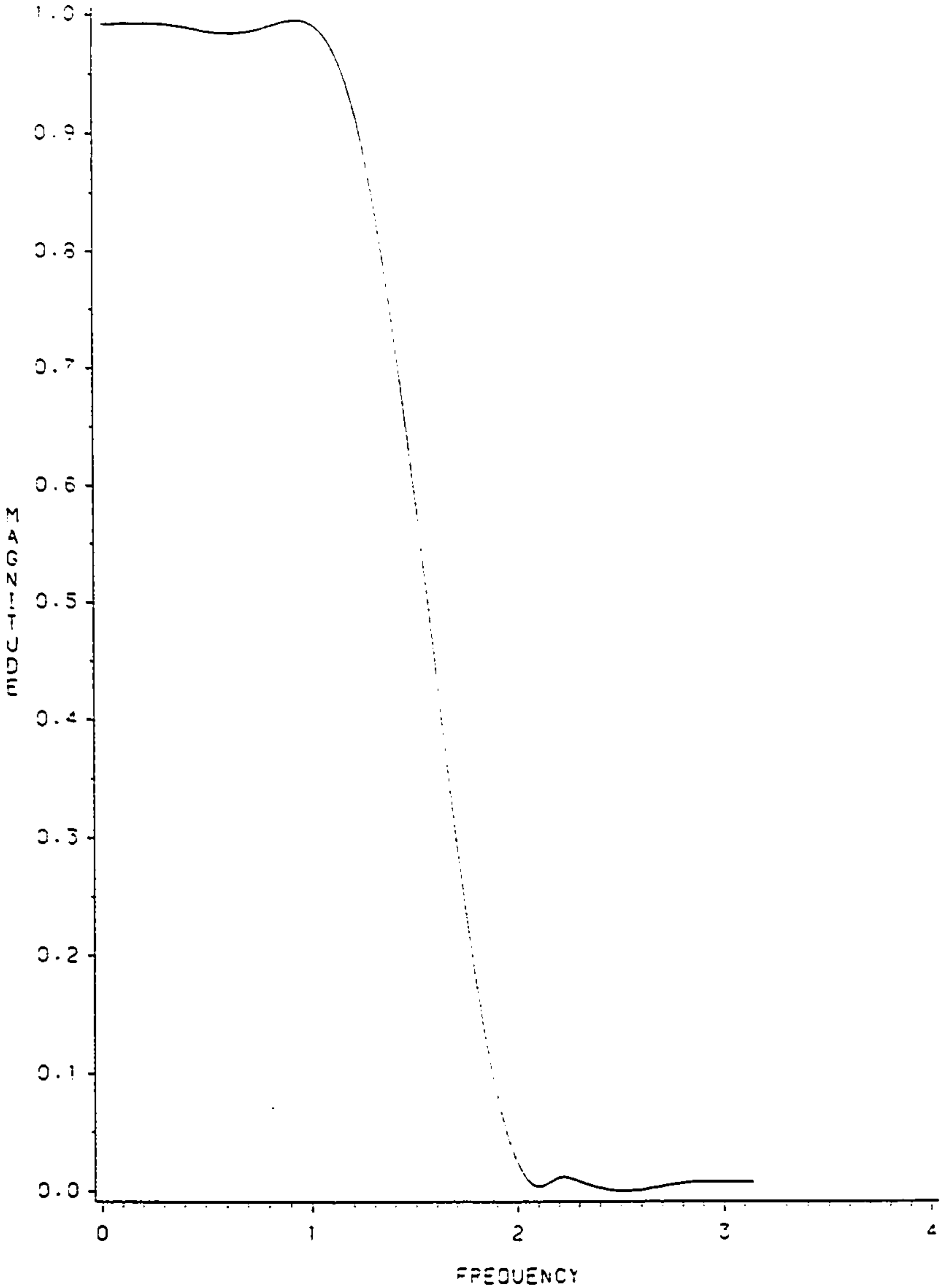


Figure (2-3) : The magnitude response for example 1 using Tech.1

Table 2-4 shows the infinite precision coefficients for example 2 using nonlinear programming. Table 2-5 shows the integer coefficients for designed filter using this technique and Table 2-6 shows the characteristics of design technique for example 2. Figures 2-4 and 2-5 show the magnitude response of IPC. design and technique 1, respectively.

a(0) =.4279E+00	a(3) =.1469E+00	a(6) = .6409E-01	a(9) =.9953E-02
a(1) =-.612E+00	a(4) = -.100E+00	a(7) = -.194E-03	
a(2) =.1313E+00	a(5) = -.389E-01	a(8) = -.289E-01	

Table 2-4 The infinite precision coefficients for example 2 using nonlinear programming

a(0) = 54	a(3) = 18	a(6) = 8	a(9) = 1
a(1) = -78	a(4) = -13	a(7) = 0	
a(2) = 17	a(5) = -4	a(8) = -3	

Table 2-5: The integer coefficients for example 2 using technique 1

DESIGN TYPE	#R.TIME(SEC.)	M.M.P	M.M.S
IPC DESIGN	5	1.0207	.0155
D&R (TYPE1)	60	1.0087	.0164

Table 2-6: Characteristics of two design techniques in example 2

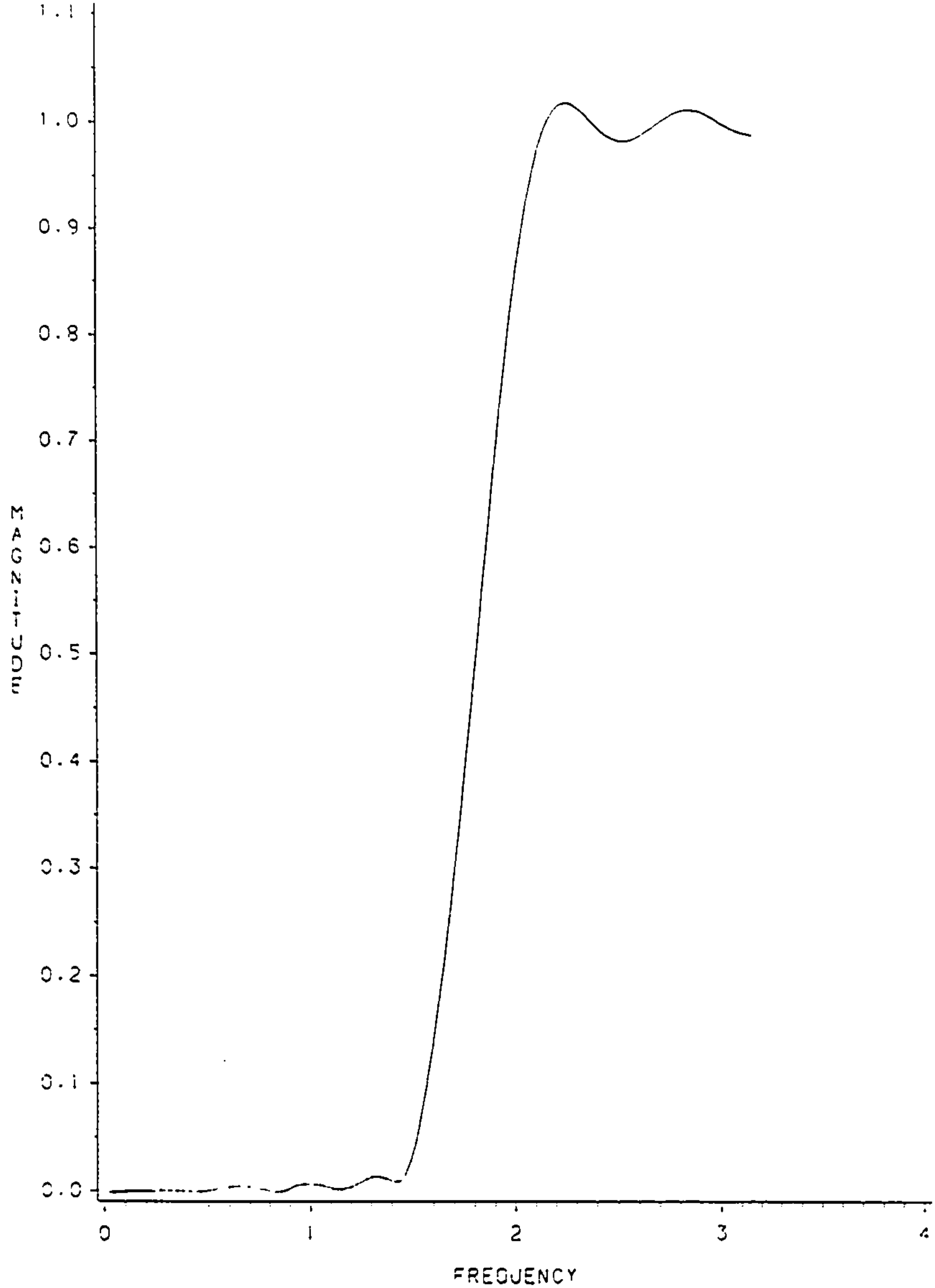


Figure (2-4) : The magnitude response for example 2 with infinite precision coefficients

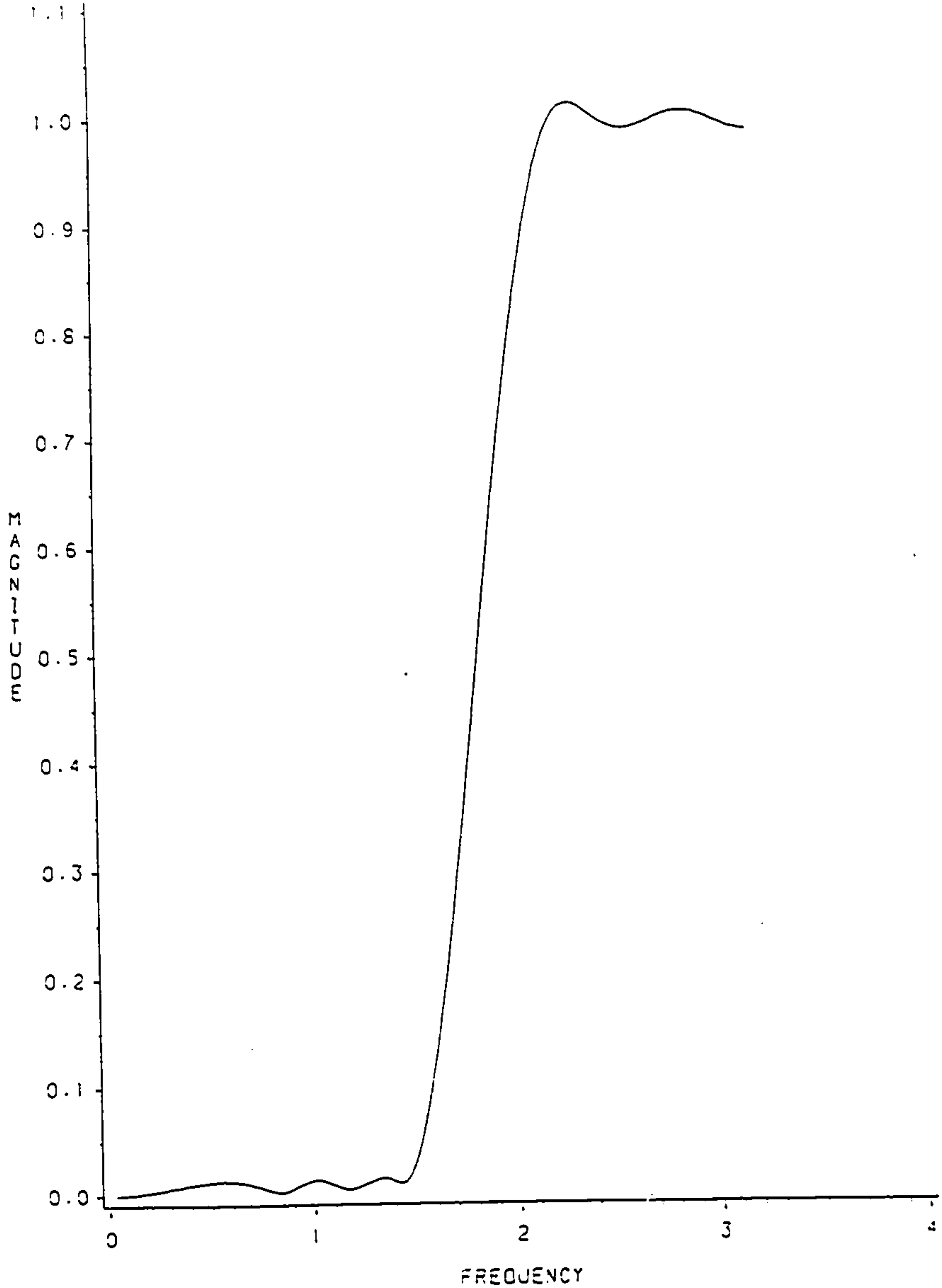


Figure (2-5) : The magnitude response for example 2 using Tech.1

Example 3 : Design a band pass 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 1.5 \le \omega \le 2 \\ 0 & 0 \le \omega \le 1 \\ & 2.5 \le \omega \le \pi \end{cases}$$

Table 2-7 shows the infinite precision coefficients for example 3 using nonlinear programming. Table 2-8 shows the integer coefficients of the designed filter using this technique and Table 2-9 shows the characteristics of the design technique for example 3. Figure 2-6 and 2-7 show the magnitude response of IPC design and technique I respectively.

a(0) =.3359E+00	a(3) =.1869E+00	a(6) = .7687E-02	a(9) =.5828E-01
a(1) = .104E+00	a(4) =.1825E+00	a(7) = -.559E-01	
a(2) = .504E+00	a(5) = -.723E-01	a(8) = -.210E-01	

Table 2-7 The infinite precision coefficients for example 3 using nonlinear programming

a(0) = 42	a(3) = 23	a(6) = 0	a(9) = 7
a(1) = -13	a(4) = 23	a(7) = -7	
a(2) = -64	a(5) = -9	a(8) = -2	

Table 2-8: The integer coefficients for example 3 using technique I

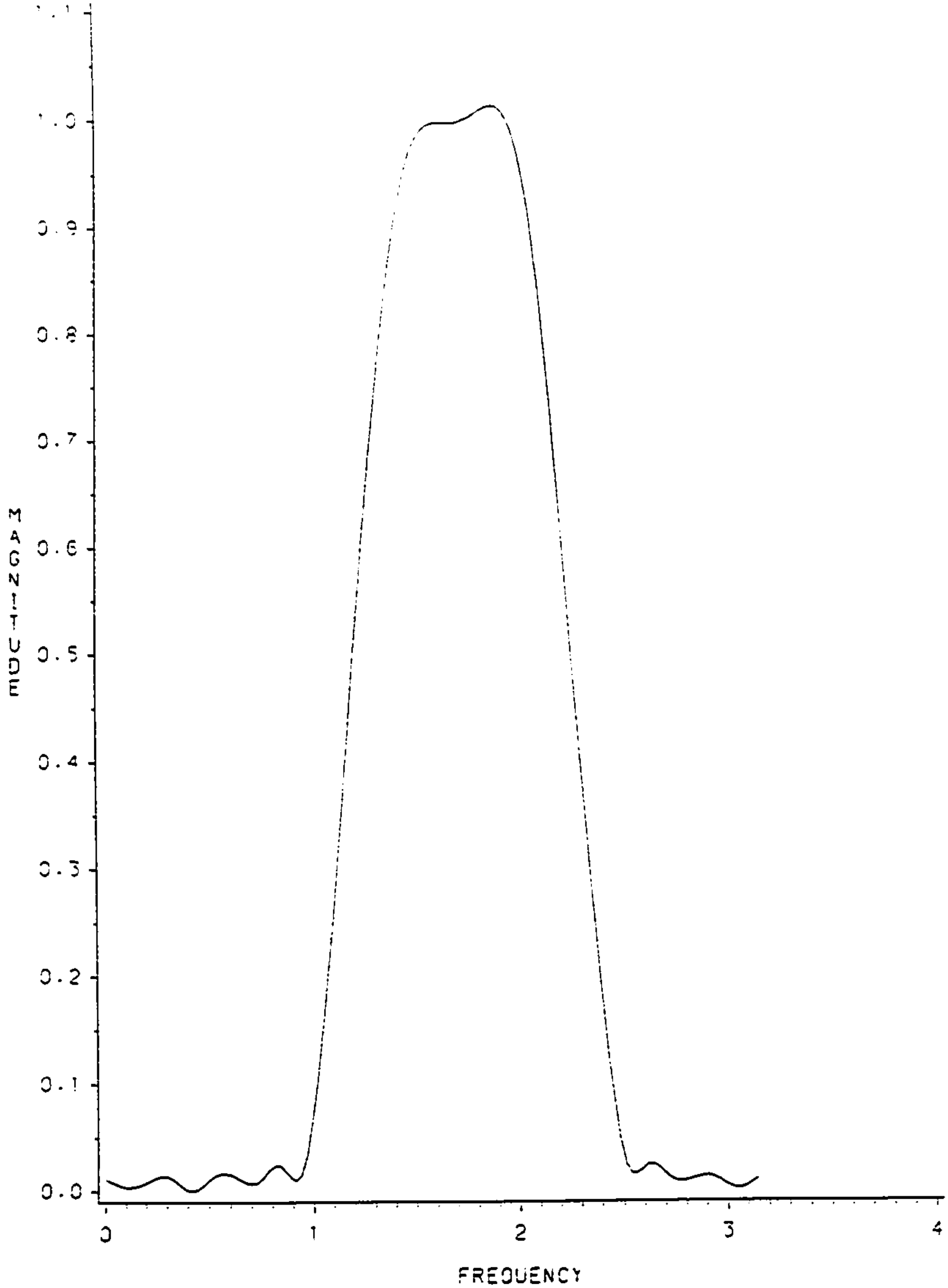


Figure (2-6) : The magnitude response for example 3 with infinite precision coefficients

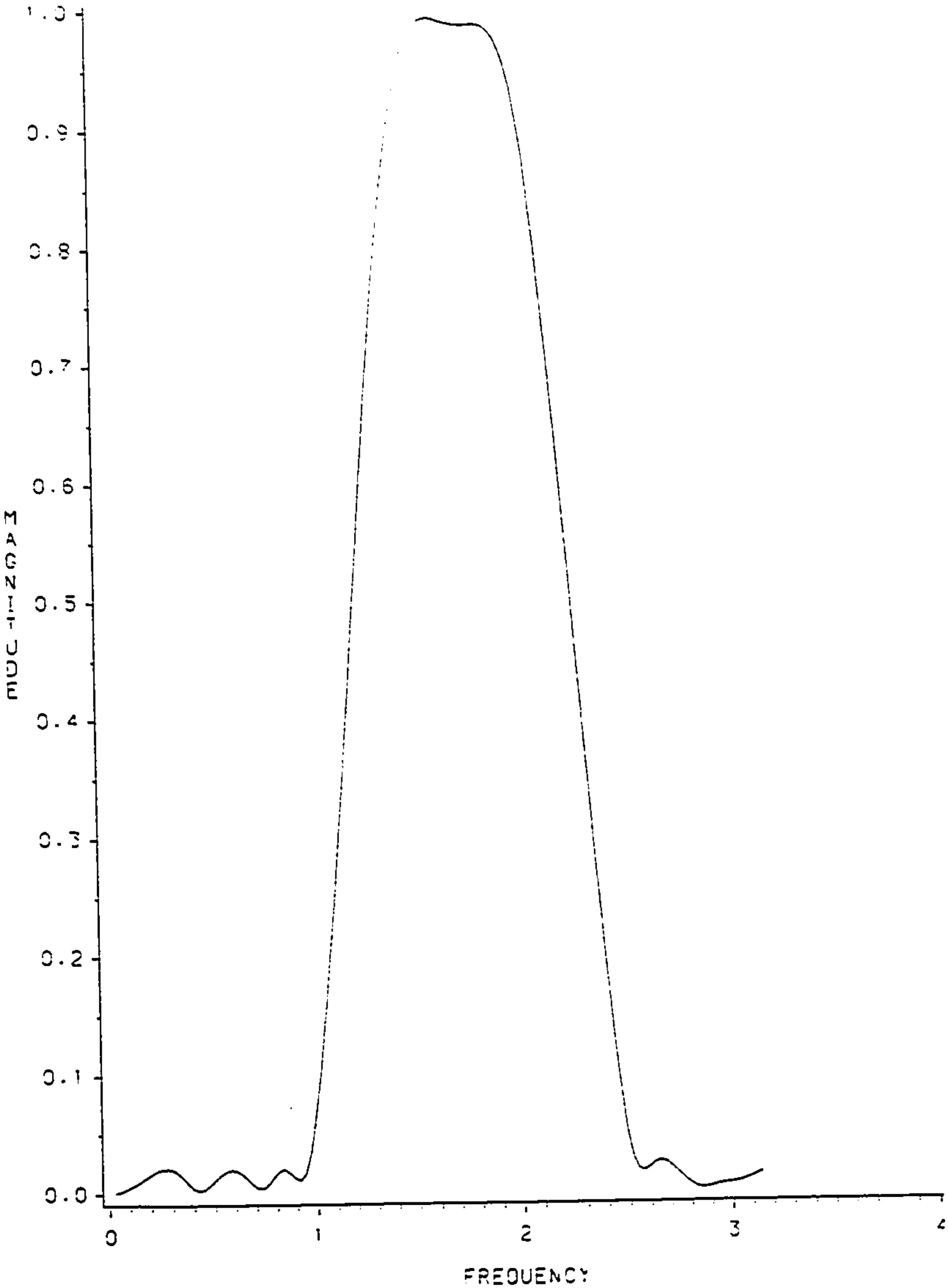


Figure (2-7) : The magnitude response for example 3 using Tech.1

DESIGN TYPE	#R.TIME(SEC)	M.M.P	M.M.S
IPC DESIGN	10	1.0128	.0254
D&R (TYPE1)	140	.99218	.0298

Table 2-9: Characteristics of two design techniques in example 3

Example 4 : Design a stop band 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$| H_i(e^{j\omega}) | = \begin{cases} 0 & 1.5 \le \omega \le 2 \\ 1 & 0 \le \omega \le 1 \\ & 2.5 \le \omega \le \pi \end{cases}$$

Table 2-10 shows the infinite precision coefficients for example 4 using nonlinear programming. Table 2-11 shows the integer coefficients of the designed filter using this technique and Table 2-12 shows the characteristics of the design technique for example 4. Figures 2-8 and 2-9 show the magnitude response of IPC design and technique 1, respectively.

a(0) =.6632E+00	a(3) = -.183E+00	a(6) = -.793E-02	a(9) =-.585E-01
a(1) =.1035E+00	a(4) = -.184E+00	a(7) = .5578E-01	
a(2) =.5066E+00	a(5) = .7137E-01	a(8) = .2310E-01	

Table 2-10: The infinite precision coefficients for example 4 using nonlinear programming

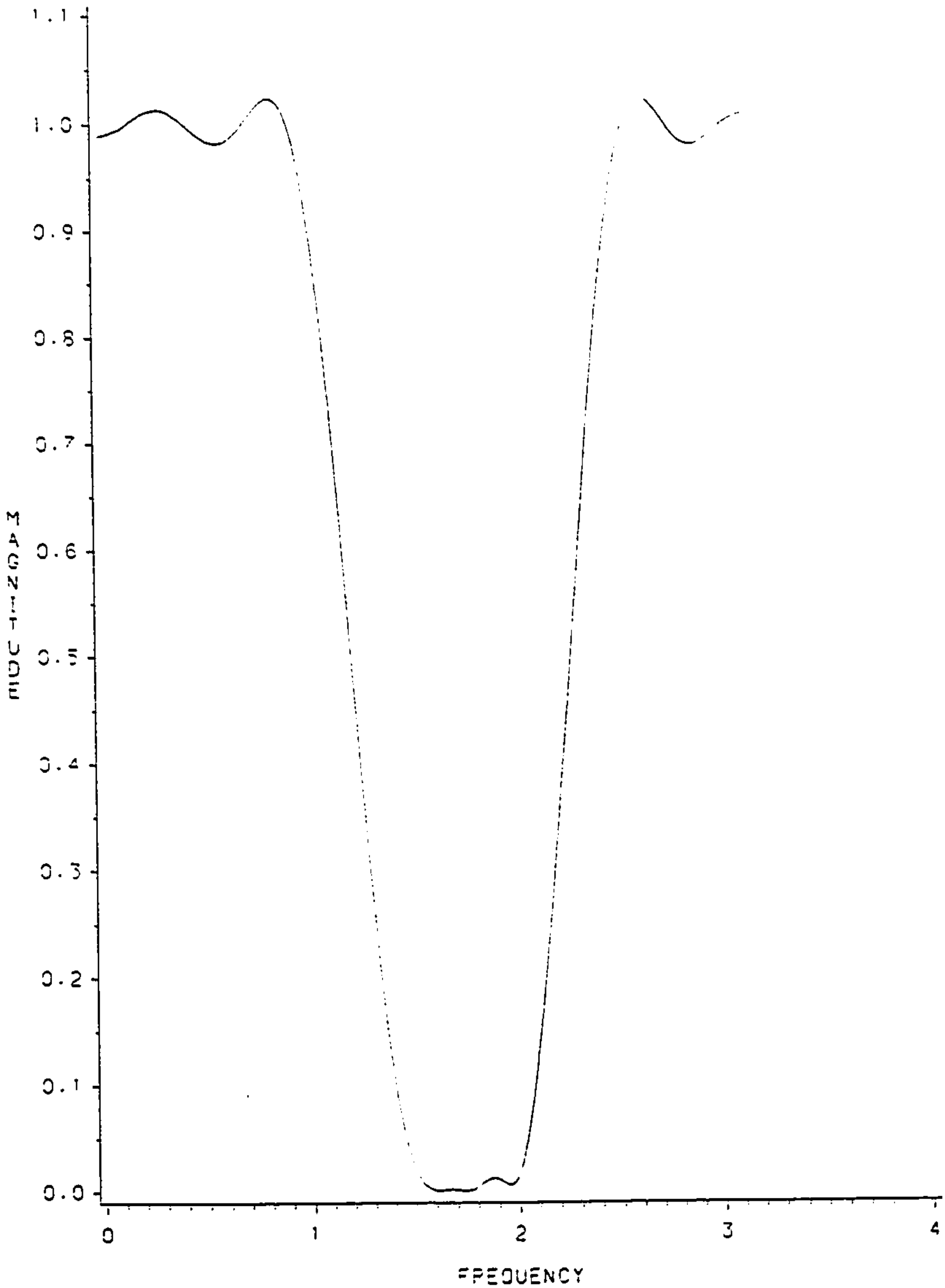


Figure (2-8) : The magnitude response for example 4 with infinite precision coefficients

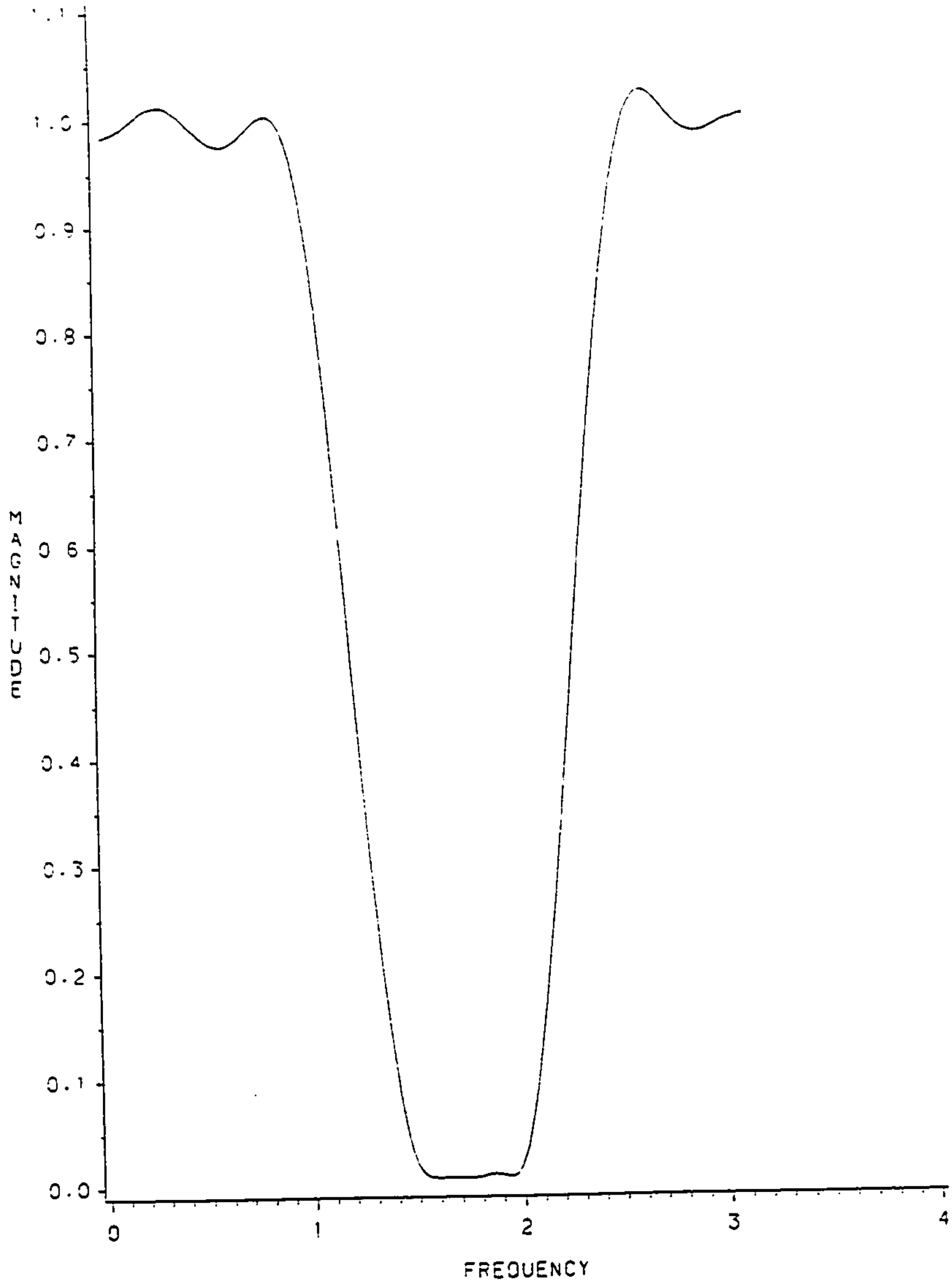


Figure (2-9) : The magnitude response for example 4 using Tech.1

a(0) = 84 a(1) = 13 a(2) = 65	a(3) = -23 a(4) = -23 a(5) = 9	a(6) = -1 a(7) = 7 a(8) = 2	a(9) = -7

Table 2-11: The integer coefficients for example 4 using technique 1

DESIGN TYPE	#R. TIME(SEC.)	M.M.P	M.A.S
IPC. DESIGN	10	1.0257	.0134
D&R (TYPE1)	70	1.0202	.0197

Table 2-12: Characteristics of two design techniques in example 4

Example 5 : Design a low pass 1-D FIR filter with the order of 49, B=8, and the following specification:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 0 \le \omega \le 0.3 \\ 0 & 0.6 \le \omega \le \pi \end{cases}$$

Table 2-13 shows the infinite precision coefficients for example 4 using nonlinear programming. Table 2-14 shows the integer coefficients of the designed filter using this technique and Table 2-15 shows the characteristics of the design technique for example 5. Figures 2-10 and 2-11 show the magnitude response of IPC design and technique 1 respectively.

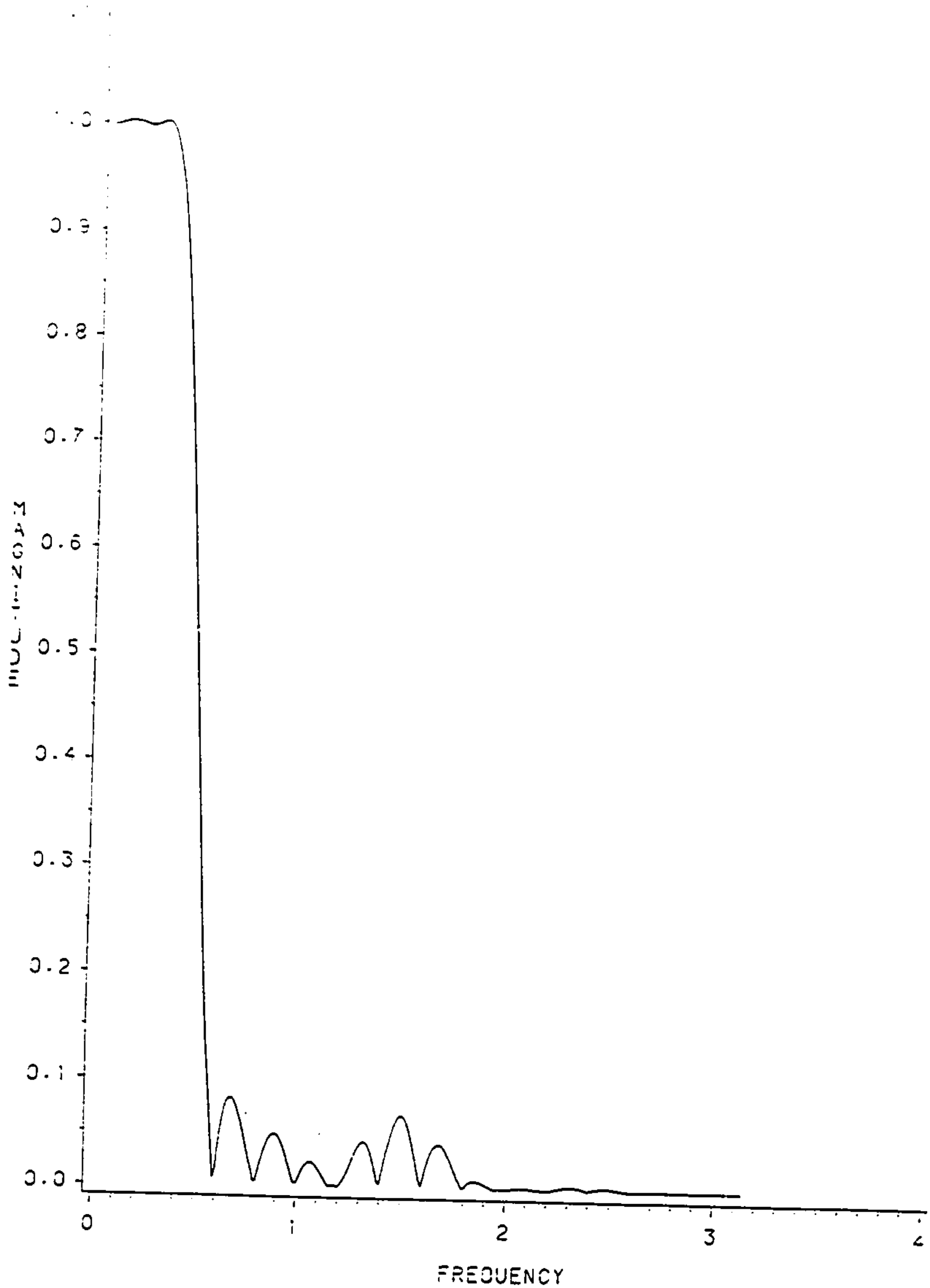


Figure (2-10) : The magnitude response for example 5 with infinite precision coefficients

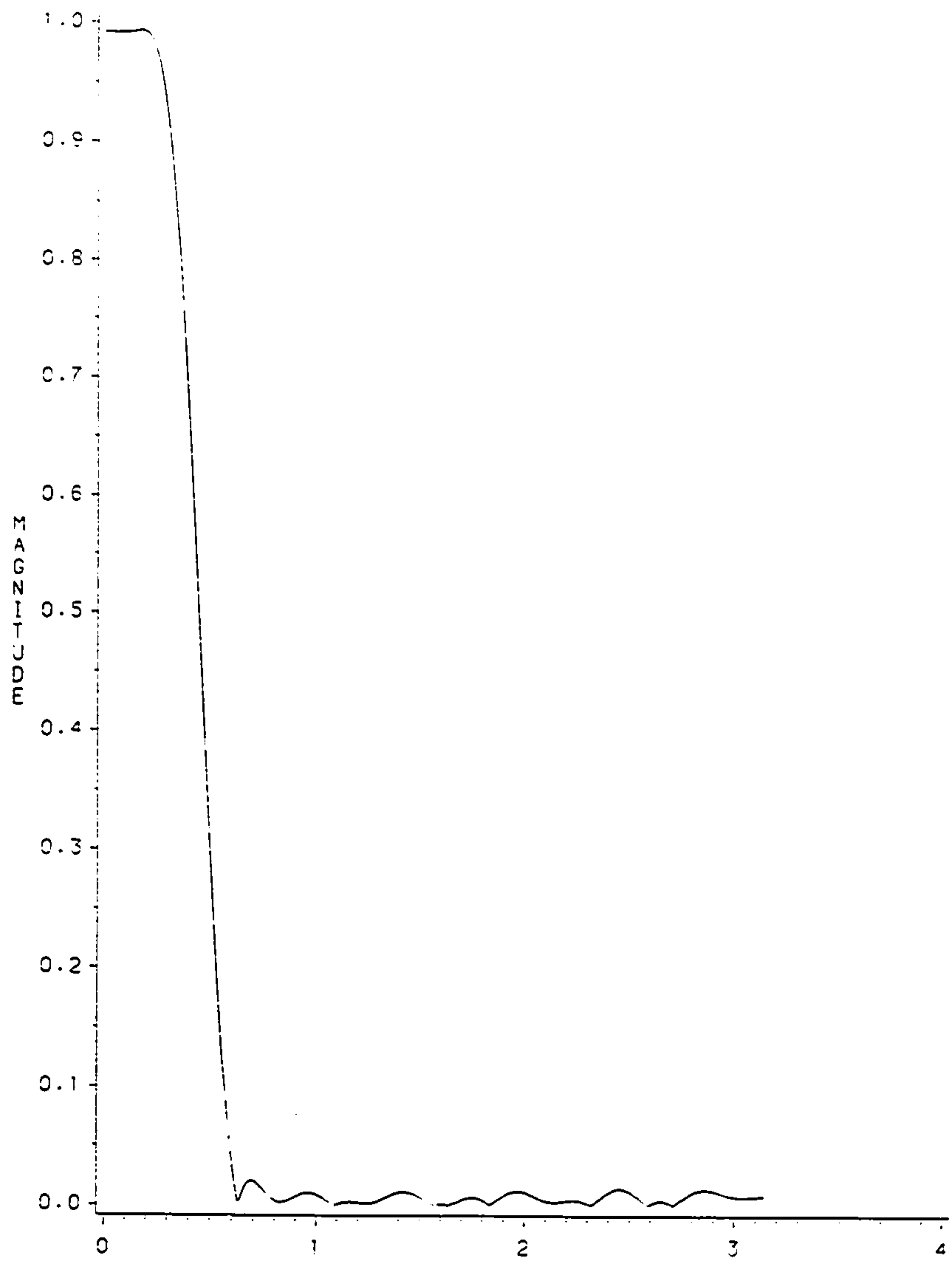


Figure (2-11) : The magnitude response for example 5 using tech.1

a(0) = .1417E+00	a(7) = .1997E-02	a(14) = -.137E-02	a(21)= .4880E-03
a(1) = .2733E+00	a(8) = -.268E-01	a(15) = .7842E-02	a(22)= -.163E-02
a(2) = .2445E+00	a(9) = -.422E-01	a(16) = .1279E-01	a(23)= -.244E-02
a(3) = .2009E+00	a(10)= -.454E-01	a(17) = .1374E-01	a(24)= -.227E-02
a(4) = .1485E+00	a(11)= -.393E-01	a(18) = .1166E-01	
a(5) = .9378E-01	a(12)= -.275E-01	a(19) = .7889E-02	
a(6) = .4131E-01	a(13)= -.137E-01	a(20) = .3878E-02	

Table 2-13: The infinite precision coefficients for example 5 using nonlinear programming

a(0) = 18	a(7) = 0	a(14) = 0	a(21)= 0
a(1) = 34	a(8) = -3	a(15) = 1	a(22)= 0
a(2) = 31	a(9) = -5	a(16) = 1	a(23)= 0
a(3) = 25	a(10)= -5	a(17) = 1	a(24)= 0
a(4) = 18	a(11)= -4	a(18) = 1	
a(5) = 11	a(12)= -3	a(19) = 0	
a(6) = 5	a(13)= -1	a(20) = 0	

Table 2-14: The integer coefficients for example 5 using technique 1

DESIGN TYPE	#R.TIME(SEC)	M.M.P	M.A.S
IPC. DESIGN	40	1.0007	.0015
D&R (TYPE1)	650	.98099	.0165

Table 2-15: Characteristic of two design techniques in example 5

2.3. DISCRETIZATION AND REOPTIMIZATION (TYPE 2)

One of the problems in discretization & reoptimization (type 1) appears when one of the very last remaining coefficients which has to be determined is also the most sensitive coefficient. In this case any small change in the value of this coefficient can drastically affect the total response of the filter in such a way that the desired specification fails to be met. To prevent this, the most sensitive coefficient can be determined among all the remaining coefficients at each iteration. The response degradation may be greatly reduced by readjusting other remaining coefficients.

In this section, we employ the technique proposed by Zhonogqi Jing and T.Fam [44] to design a FIR filter with integer coefficients. Similar to technique one, nonlinear programming will be utilized. The design steps for this technique are as follows.

$$E = \sum_{w_j \in W} [H_i(e^{jw}) - \sum_{i=0}^{\frac{N-1}{2}} a(i) \cos(w_j i)]^2 \qquad (3\text{-}3)$$

2.3.1. DESIGN STEPS

In this technique, the number of specified bits for representing the coefficients is (B) which includes the sign bit. If the infinite precision coefficients of the designed filter are b_i, where i=(N-1)/2 , N is the order of filter, if

$$-1 \leq b_i \leq +1 \tag{3-1}$$

then b_i can be shown by the following form

$$b_i = \frac{a_i}{2^{B-1}} \tag{3-2}$$

where a_i is the integer coefficients for the prescribed filter. Similar to technique 1, it is obvious that the proper scaling in the prescribed magnitude response in order to recover the desired has to be applied. The design steps with the required explanation at each step are as follows:

1) Set k=0

2) Find the infinite precision coefficients for n=(N-1/2); N is the order of the filter. The least square error for this formulation is as follows:

$$E = \sum_{\omega_j \in \Omega} [H_i(e^{j\omega}) - \sum_{i=0}^{\frac{N-1}{2}} a_i \cos(\omega_j i)]^2 \tag{3-3}$$

Where $H_i(e^{j\omega})$ is the ideal frequency response and a_i are the coefficients of designed filter. The j is the number of frequency points for evaluation of the error function and $\omega \in [0, \pi]$

3) Set $M = \frac{N-1}{2} - k$

4) Find the most sensitive coefficients among (M) coefficients. Based on equation (1-17) and the definition of a derivative, the new form can be written in the form of equation (3-4) .

$$S_{a_i}(z) = \left| \frac{H(x_1,x_2,\ldots,x_i,\ldots,x_n) - H(x_1,x_2,\ldots,x_i+\Delta x_i,\ldots,x_n}{H(x_1,x_2,\ldots,x_i,\ldots,x_n)} \right|$$

$$+ \left| \frac{H(x_1,x_2,\ldots,x_i,\ldots,x_n) - H(x_1,x_2,\ldots,x_i-\Delta x_i,\ldots,x_n}{H(x_1,x_2,\ldots,x_i,\ldots,x_n)} \right| \tag{3-4}$$

where Δx_i is the variation for x_i and equal to one. Equation (3-4) will be calculated for all the coefficients and finally the assumed coefficient a_i with the maximum sensitivity will be chosen.

5) Set $a_i = a_i^+$ where a_i^+ is the rounding up value for a_i and solve the following optimization problem

$$E^+ = E(a_i, A_{n-1}^+) \tag{3-5}$$

where E^+ is the value of error function and A_{n-1}^+ is the vector with n-1 remaining variables which are determined in the optimization process.

5) Set $a_i = a_i^-$ where a_i^- is the rounding down value for a_i and solve the following optimization problem:

$$E^- = E(a_i, A^-_{n-1}) \tag{3-6}$$

where E^- is the value of error function and A_{n-1}^- is the vector with n-1 remaining variables which are determined in the optimization process.

6) Find $E = \min\{E^+, E^-\}$ and fix the vector A (coefficient vector) to (a_i^+, A_{n-1}^+) or (a_i^-, A_{n-1}^-) 7) Repeat steps 3-6 until all components of the vector $A=(a_1, a_2, \ldots, a_n)$ convert to integer value.

The flowchart of this algorithm is provided in the Appendix B. In the following section several examples have been provided to illustrate the versatility of the design technique.

2.3.2. EXAMPLES

Example 1 : Design a low pass 1-D FIR filter with the order of 19, B=8, and the following specifications.

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 0 \le \omega \le 1 \\ 0 & 2 \le \omega \le \pi \end{cases}$$

Table 2-16 shows the integer coefficients of the designed filter using this technique and Table 2-17 shows the characteristics of the design technique for example 1. Figure 2-12

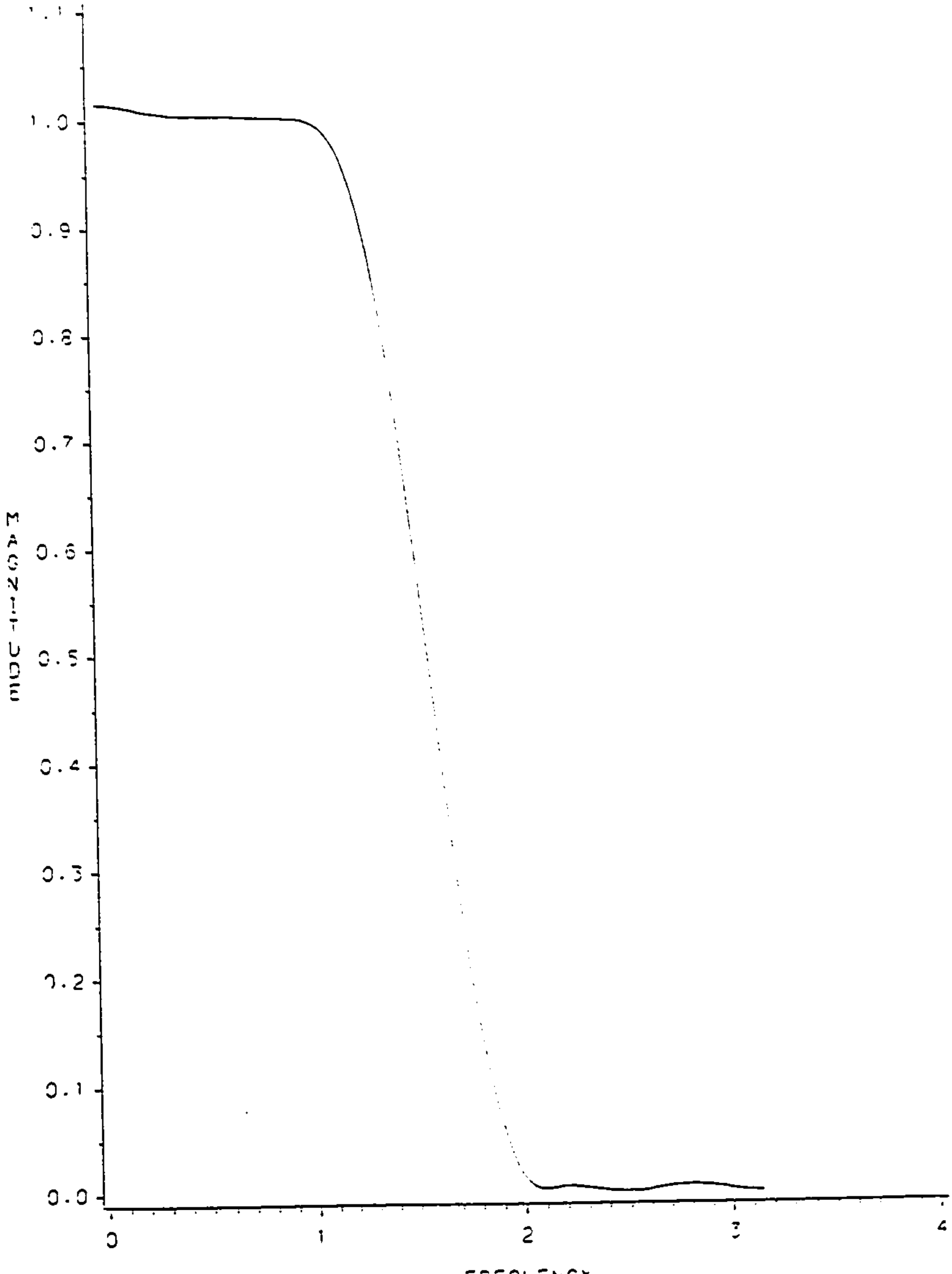


Figure (2-12) : The magnitude response for example 1 using tech.2

shows the magnitude response of the designed filter using technique 2.

a(0) = 64	a(3) = -22	a(6) = 0	a(9) = 1
a(1) = 80	a(4) = 0	a(7) = -3	
a(2) = 0	a(5) = 9	a(8) = 0	

Table 2-16: The integer coefficients for example 1 using technique 2

DESIGN TYPE	# R. TIME(sec.)	M.M.P	M.M.S
D & R TYPE 2	330	1.0078	.00781

Table 2-17: Characteristics of technique in example 1

Example 2 : Design a high pass 1-D FIR filter with the order of 19, B=8, and the following specifications.

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 2 \le \omega \le \pi \\ 0 & 0 \le \omega \le 1.5 \end{cases}$$

Table 2-18 shows the integer coefficients of the designed filter using this technique and Table 2-19 shows the characteristics of the design technique for example 2. Figure 2-13 shows the magnitude response of the design technique 2.

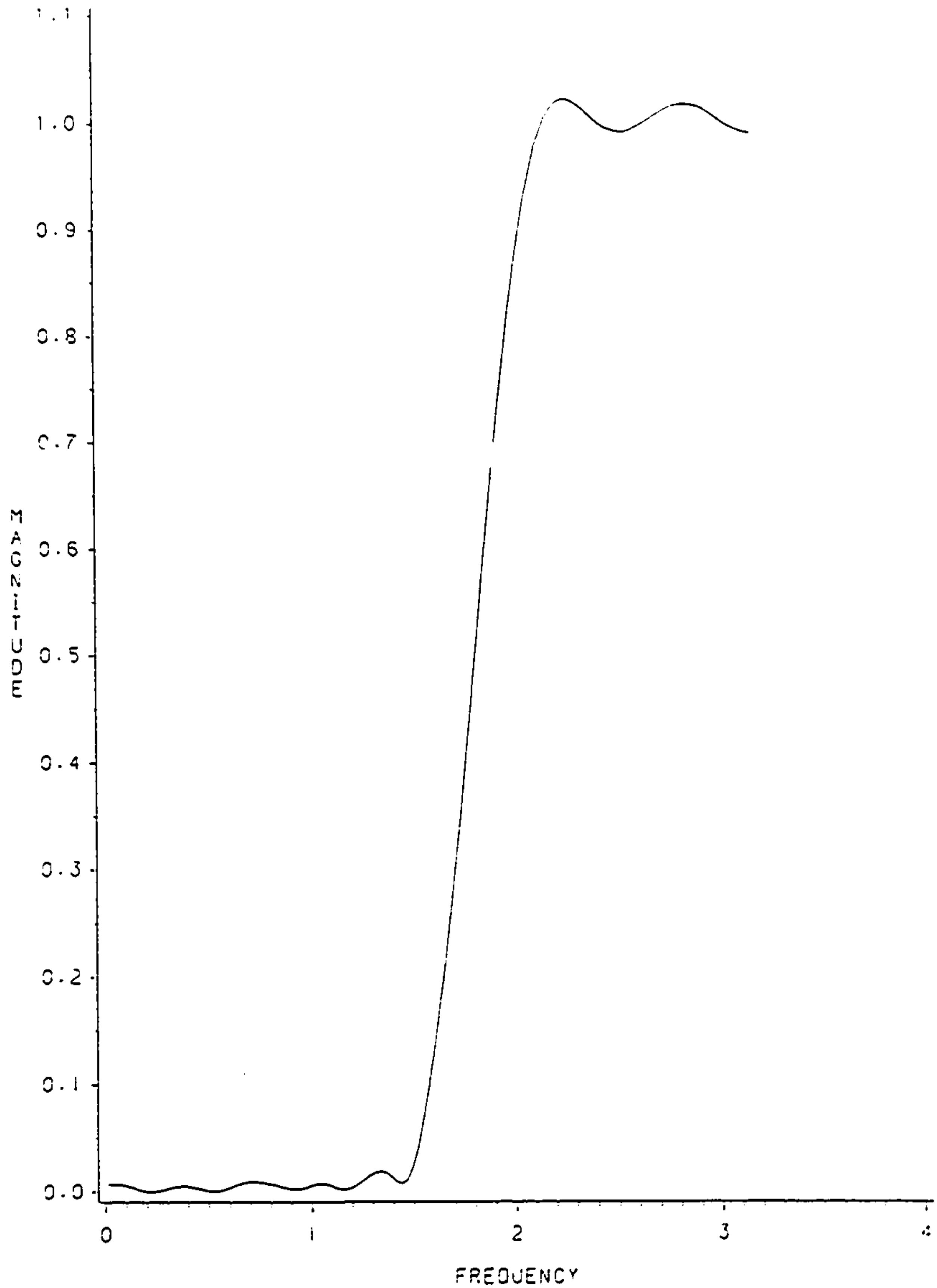


Figure (2-13) :The magnitude response for example 2 using Tech.2

a(0) = 55	a(3) = 19	a(6) = 8	a(9) = 1
a(1) = -79	a(4) = -13	a(7) = 0	
a(2) = 17	a(5) = -5	a(8) = -4	

Table 2-18: The integer coefficients for example 2 using technique 2

DESIGN TYPE	#R.TIME(S)	M.M.P.	M.M.S.
D & R TYPE 2	160	1.0156	.01599

Table 2-19 : Characteristics of design technique in example 2

Example 3 : Design a band pass 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 1.5 \le \omega \le 2 \\ 0 & 0 \le \omega \le 1 \\ & 2.5 \le \omega \le \pi \end{cases}$$

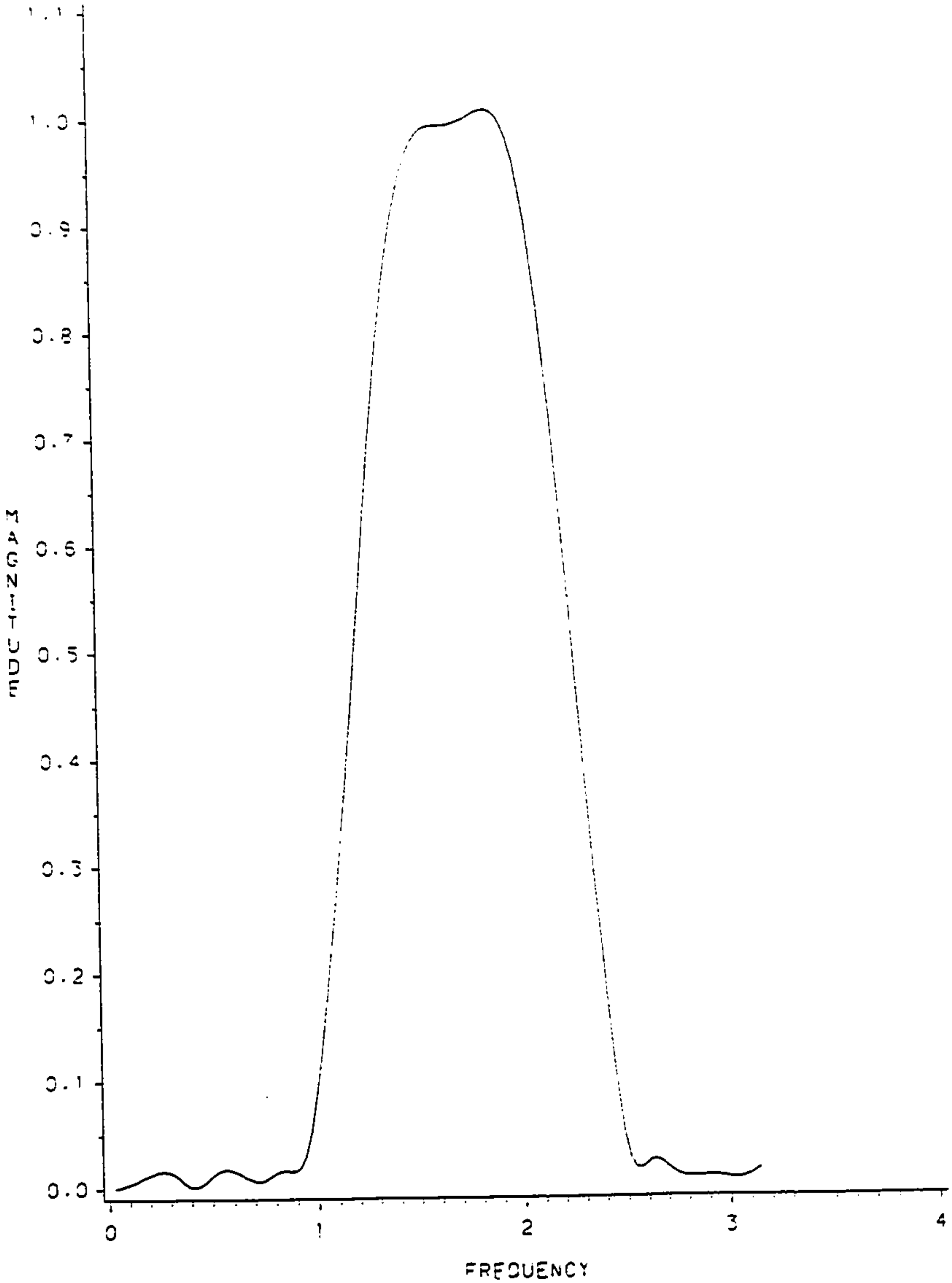


Figure (2-14) :The magnitude response for example 3 using Tech.2

Table 2-20 shows the integer coefficients of the designed filter using this technique and Table 2-21 shows the characteristics of the design technique for example 3. Figure 2-14 shows the magnitude response of the design technique.

a(0) = 43	a(3) = 23	a(6) = 1	a(9) = 7
a(1) = -13	a(4) = 23	a(7) = -7	
a(2) = -65	a(5) = -9	a(8) = -3	

Table 2-20: The integer coefficients for example 3 using technique 2

DESIGN TYPE	#R.TIME(S)	M.M.P	M.M.S
D & R type 2	400	1.0120	.0281

Table 2-21: Characteristics of design technique in example 3

Example 4 : Design a stop band 1-D FIR filter with the order of 19. B=8. and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 0 & 1.5 \le \omega \le 2 \\ 1 & \begin{matrix} 0 \le \omega \le 1 \\ 2.5 \le \omega \le \pi \end{matrix} \end{cases}$$

Table 2-22 shows the integer coefficients of the designed filter using this technique and Table 2-23 shows the characteristics of the design technique for example 4. Figure 2-15 shows the magnitude response of design technique.

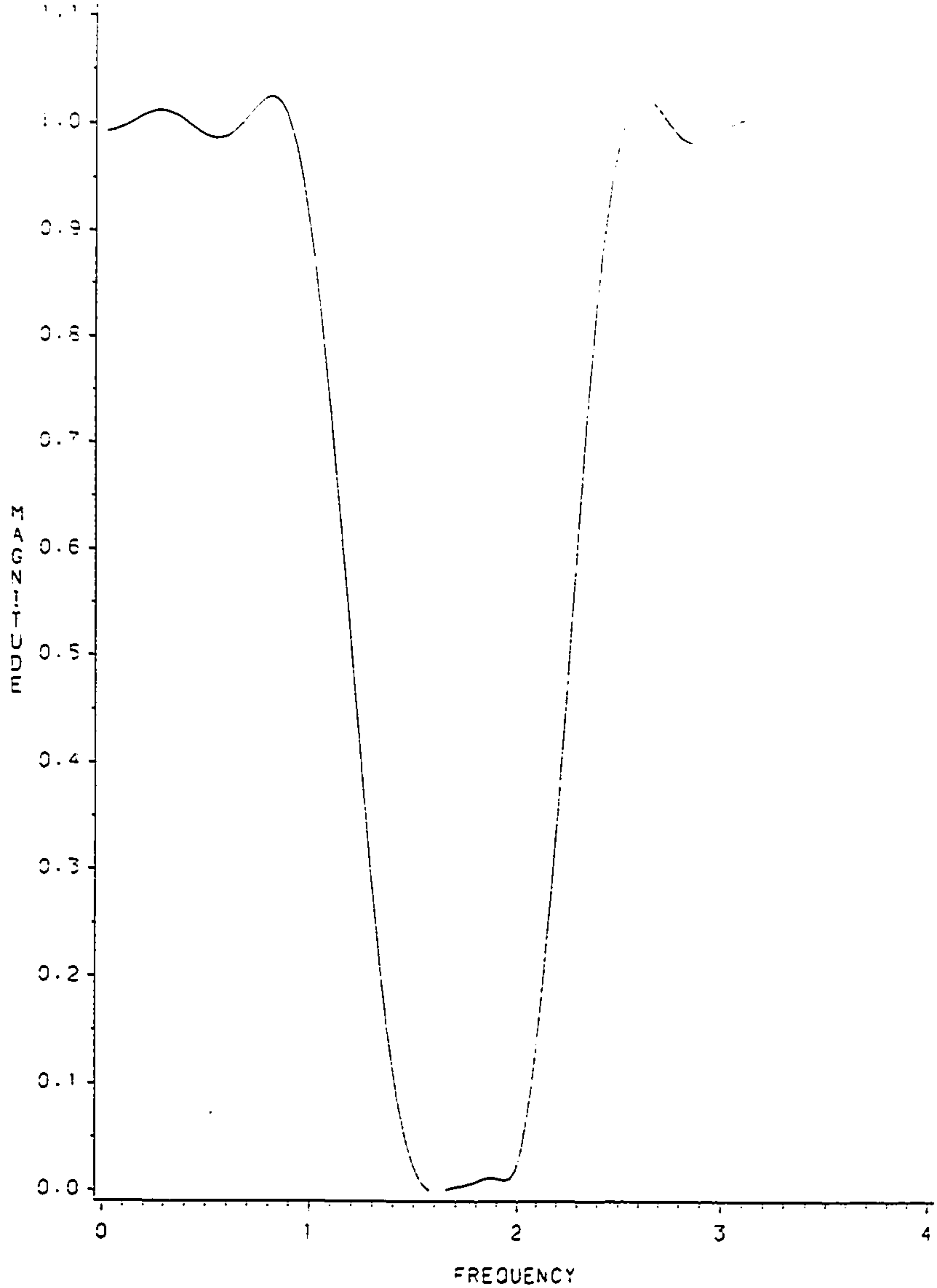


Figure (2-15):The magnitude response for example 4 using Tech.2

$a(0) = 85$	$a(3) = -23$	$a(6) = -1$	$a(9) = -7$
$a(1) = 13$	$a(4) = -24$	$a(7) = 7$	
$a(2) = 65$	$a(5) = 9$	$a(8) = 3$	

Table 2-22: The integer coefficients for example 4 using technique 2

DESIGN TYPE	#R.TIME(SEC)	M.M.P	M.A.S
D & R TYPE 2	120	1.0281	.0207

Table 2-23: Characteristics of two design techniques in example 4

Example 5 : Design a low pass 1-D FIR filter with the order of 49, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 0 \le \omega \le 0.3 \\ 0 & 0.6 \le \omega \le \pi \end{cases}$$

Table 2-24 shows the integer coefficients of the designed filter using this technique and

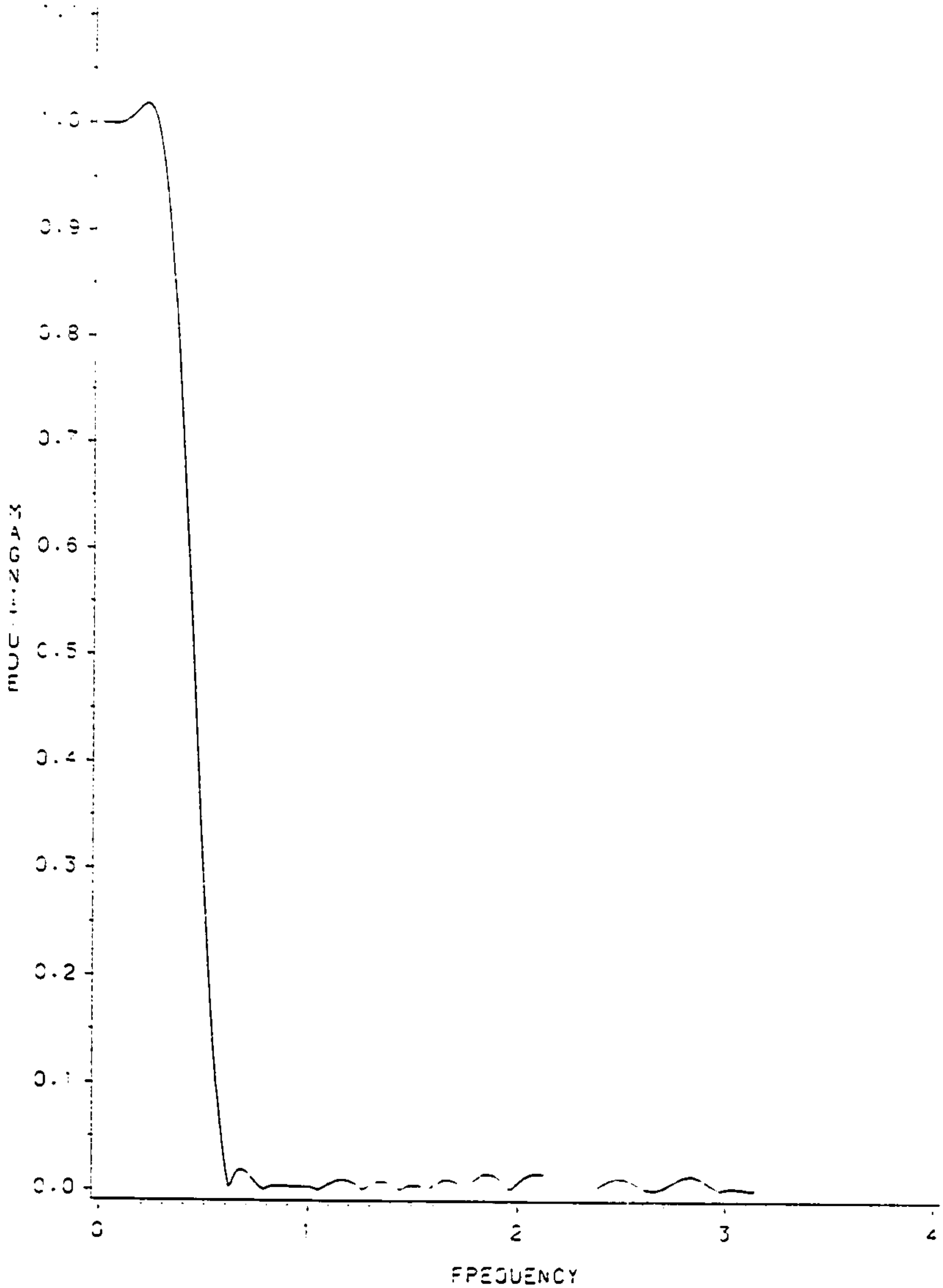


Figure (2-16):The magnitude response for example 5 using Tech.2

Table 2-25 shows the characteristics of the design technique for example 5. Figure 2-16 shows the magnitude response of the design technique.

a(0) = 18	a(7) = 0	a(14) = 0	a(21) = 0
a(1) = 35	a(8) = -3	a(15) = 1	a(22) = 0
a(2) = 31	a(9) = -5	a(16) = 2	a(23) = 0
a(3) = 26	a(10) = -6	a(17) = 2	a(24) = 0
a(4) = 19	a(11) = -5	a(18) = 1	
a(5) = 12	a(12) = -3	a(19) = 1	
a(6) = 5	a(13) = -2	a(20) = 0	

Table 2-24: The integer coefficients for example 5 using technique 2

DESIGN TYPE	#R. TIME(Sec)	M.M.P	M.A.S
D & R TYPE2	2280	1.0007	.0170

Table 2-25: Characteristics of design technique in example 5

2.4. MIXED INTEGER LINEAR PROGRAMMING

The techniques discussed in the previous section do not guarantee an optimal solution. To obtain an optimal solution, in this section, we employ the technique proposed by V.B.Lawrence and A.C.Salazar [37] to design a FIR filter with integer coefficients. In contrast to previous techniques linear programming will be utilized. A brief explanation of linear programming is provided in Appendix C. Formulation of the problem and several examples will be provided in this section.

There are two available programs for solving mixed linear integer programming which are used in this section and there both employ the branch and bound technique. The first one is known as Mint algorithm and second one is a commercial package named LINDO. The latter one is utilized in most of the examples but there are some examples which are solved by the former program. Comparison of these two techniques will be provided at the end of this chapter.

2.4.1. LINEAR PROGRAMMING FORMULATION

We now set up the filter design problem in such a way that it can be solved by linear programming. An optimum set of coefficients that gives the best approximation of $H(e^{jw})$ to a desired magnitude function $G(e^{jw})$ in the minimax sense will be the solution to our problem. The error in the approximation is defined by

$$E(e^{jw}) = G(e^{jw}) - H(e^{jw}) \tag{4-1}$$

The desired magnitude function $G(e^{jw})$ is a real valued function, which for our purposes will be defined only at a discrete set of frequencies (w_k) where $k=1, 2, \ldots, K$. The frequency response of the designed filter will be

$$H(e^{jw}) = a_0 + 2\sum_{i=1}^{\frac{N-1}{2}} a_i \cos(w_k) \tag{4-2}$$

where N is the order of the filter.

For clarity of illustration, the linear programming problem is formulated for a band-pass filter that best approximates the desired magnitude characteristics formulated, as follows

$$
\begin{aligned}
&G(e^{jw}) = 1 \qquad \textit{for the pass band region} \qquad w_{p_1} \le w_k \le w_{p_2} \\
&G(e^{jw}) = 0 \qquad \textit{for the stop band regions} \qquad 0 \le w_k \le w_{s_1} \qquad w_{s_2} \le w_K \le \pi
\end{aligned}
\tag{4-3}
$$

where the passband and stopband frequency regions are (w_{p1} -w_{p2}) and (0 -w_{s1}),(w_{s2}, π), respectively. From the above-mentioned characteristics, the following constraints for this design can be written as follows

$$
\begin{aligned}
&1-\delta_1 \le H(e^{jw_k}) \le 1+\delta_1 \qquad \textit{for the pass band region} \qquad w_{p_1} \le w_k \le w_{p_2} \\
&H(e^{jw_k}) \le \delta_2 \qquad \textit{for the stop band regions} \qquad 0 \le w_k \le w_{s_1} \qquad w_{s_2} \le w_k \le \pi
\end{aligned}
\tag{4-4}
$$

where δ_1, δ_2 are ripples in the passband and stopband regions, respectively.

The linear programming problem is to minimize

$$
f(\delta_1, \delta_2) = CX \tag{4-5}
$$

where C={c_1 , c_2 , 0 , 0 , ... , 0) with [(N-1)/2]+3 elements and c_1, c_2 are weights for δ_1, δ_2

subject to

$$AX < B \tag{4-6}$$

$$C \leq X \leq F \tag{4-7}$$

where X is an [(N-1)/2]+3 vector whose third to [(N-1)/2]+3 elements are the filter coefficients. The first and the second elements are the passband and stopband ripples.

$$X = \begin{bmatrix} \delta_1 \\ \delta_2 \\ a_0 \\ a_1 \\ \cdot \\ a_i \\ \cdot \\ a_{\frac{N-1}{2}} \end{bmatrix} \tag{4-8}$$

The matrix A is a 2K by {[(N-1)/2]+3} constraint matrix whose elements are either -1, 0, +1, or $\pm$[2cosw_k i], where i varies from 1 to [(N-1)/2]. B is a 2K vector whose elements are either -1, 0, or +1. A and B are [(N-1)/2+3] vectors that specify the upper and lower bounds on the vector X.

2.4.2. FORMULATION OF MIXED INTEGER LINEAR PROGRAMMING

The decision variables, a_i, obtained by the above formulation are the infinitely precise coefficients. When these variables are quantized to a fixed number of bits, the resulting solution is no longer optimal. To obtain an optimal solution, the effects of coefficients Quantization should be included in the formulation of the problem. Equation (4-4) can be written as

$$\hat{\delta}_1 \pm a_0 \pm 2\sum_{i=1}^{\frac{N-1}{2}} a_i \cos(w_k i) \le \pm 1 \qquad for \quad w_{p_1} \le w_k \le w_{p_2}$$

$$-\hat{\delta}_2 + a_0 \pm 2\sum_{i=1}^{N-1/2} a_i \cos(w_k i) \le 0 \qquad for \quad 0 \le w_k \le w_{s_1} \qquad w_{s_2} \le w_k \le \pi \tag{4-9}$$

In formulating a mixed integer programming problem, if the symbol B represents the number of bits including the sign bit then the following equation can be written:

$$a_i = \frac{b_i}{2^{B-b-1}} \tag{4-10}$$

where a_i is the integer solution to the problem and b is the number of necessary bits to represent the integer part of biggest infinite precision coefficients.

By substituting equation (4-10) into equation (4-9) and scaling the inequality by 2^{B-b-1} the final result will be

$$-\hat{\delta}_1 \pm a_0 \pm 2\sum_{i=1}^{\frac{N-1}{2}} a_i \cos(w_k i) \le \pm 2^{B-b-1} \qquad for \quad w_{p_1} \le w_k \le w_{p_2}$$

$$-\hat{\delta}_2 \pm a_0 \pm 2\sum_{i=1}^{N-1/2} a_i \cos(w_k i) \le 0 \qquad for \quad 0 \le w_k \le w_{s_1} \qquad w_{s_2} \le w_k \le \pi \tag{4-11}$$

where

$$\begin{cases} \alpha_1 \le \delta_1 \le \alpha_2 \\ \beta_1 \le \delta_2 \le \beta_2 \\ -2^{B-b-1} \le a_i \le 2^{B-b-1} \\ i = 0, 1, \cdots, \frac{N-1}{2} \end{cases} \tag{4-12}$$

Different values for α_1 ,α_2 , β_1 and β_2 in the following examples have been examined. The results show that in most of the cases to prevent an unbounded or a nonfeasible solution, values for these parameters should be greater than one. Although there is no limitation on the upper bound of these parameters, an unnecessarily high upper bound for these parameters enlarges the feasible region and increases computation cost without any further improvement in the final result.

2.4.3. EXAMPLES

Example 1: Design a low pass 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$
|H_i(e^{j\omega})| = \begin{cases} 1 & 0 \le \omega \le 1 \\ 0 & 2 \le \omega \le \pi \end{cases}
$$

Table 2-26 shows the integer coefficients for example 1 using mixed integer linear programming. Table 2-27 shows the characteristics of the design technique for example 1. Figure 2-17 shows the magnitude response of the design technique.

a(0) = 58	a(3) = -15	a(6) = 3	a(9) = -2
a(1) = 77	a(4) = -6	a(7) = 3	
a(2) = 10	a(5) = 0	a(8) = 0	

Table 2-26: The integer coefficients for example 1 using technique 3

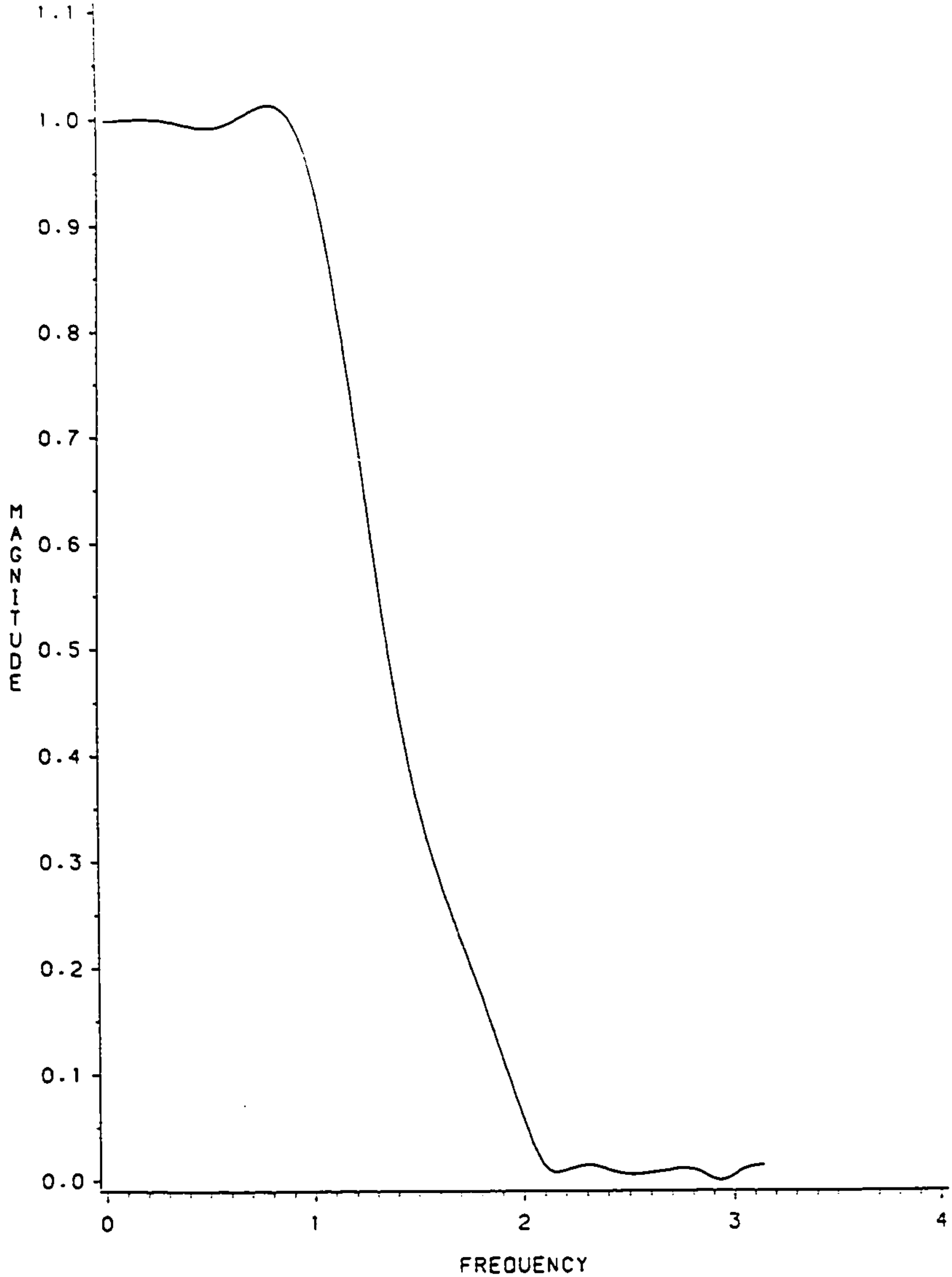


Figure (2-17) :The magnitude response for example 1 using linear programming

DESIGN TYPE	#R TIME(SEC.)	M.M.P	M.M.S
LINEAR.PROG.	40	1.0119	.01585

Table 2-27: Characteristics of design techniques 3 in example 1

Example 2 : Design a high pass 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 2 \le \omega \le \pi \\ 0 & 0 \le \omega \le 1.5 \end{cases}$$

Table 2-28 shows the integer coefficients of the designed filter using this technique and Table 2-29 shows the characteristics of the design technique for example 2. Figure 2-18 shows the magnitude response of the design technique 3.

a(0) = 57	a(3) = 23	a(6) = 11	a(9) = -3
a(1) = -80	a(4) = -11	a(7) = 6	
a(2) = 12	a(5) = -9	a(8) = -6	

Table 2-28: The integer coefficients for example 2 using technique 3

DESIGN TECH.	#R TIME(SEC.)	M.M.P	M.M.S
LINEAR PROG.	60	1.0312	.0644

Table 2-29: Characteristics of the design technique in example 2

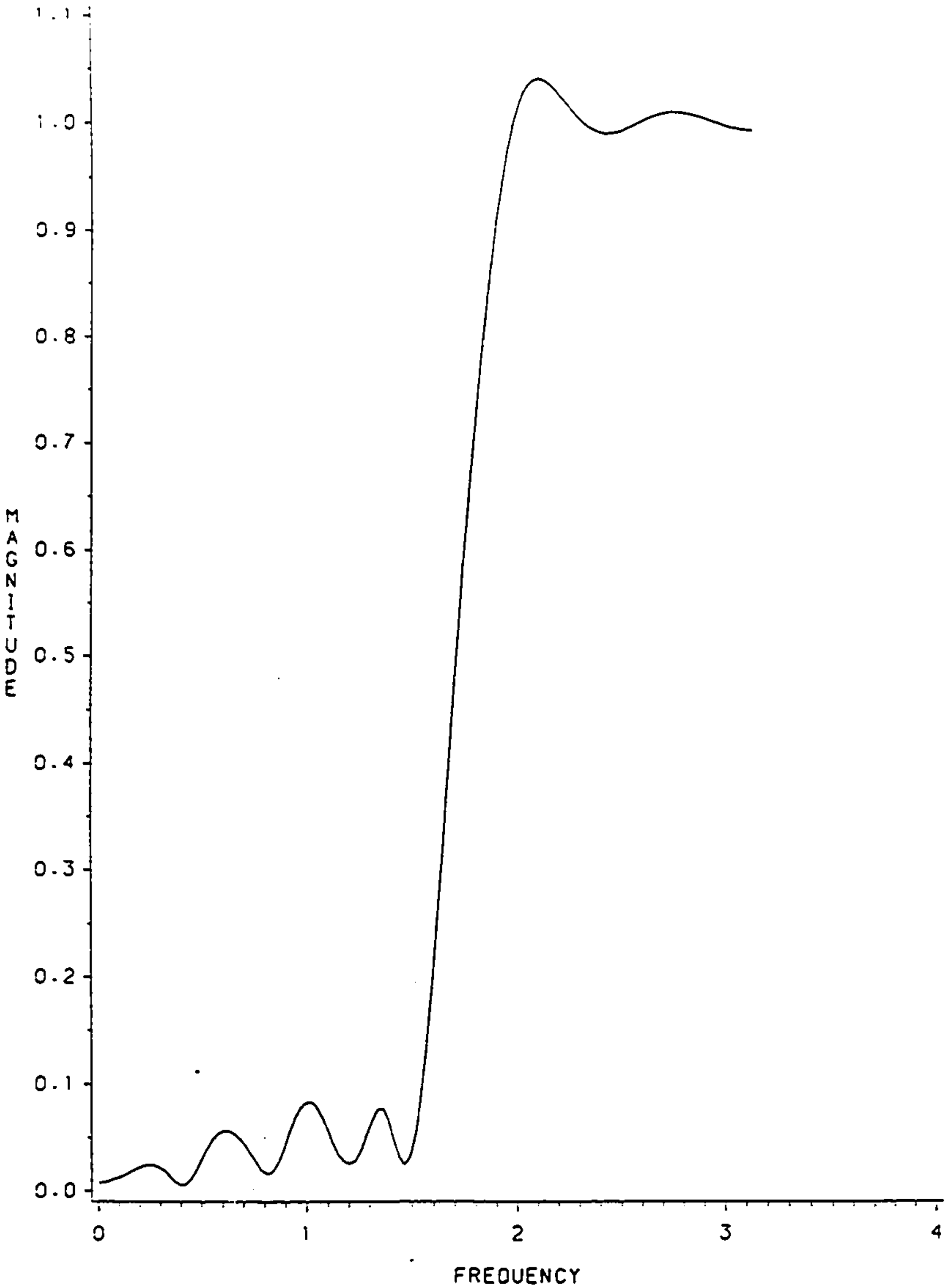


Figure (2-18) :The magnitude response for example 2 using linear programming

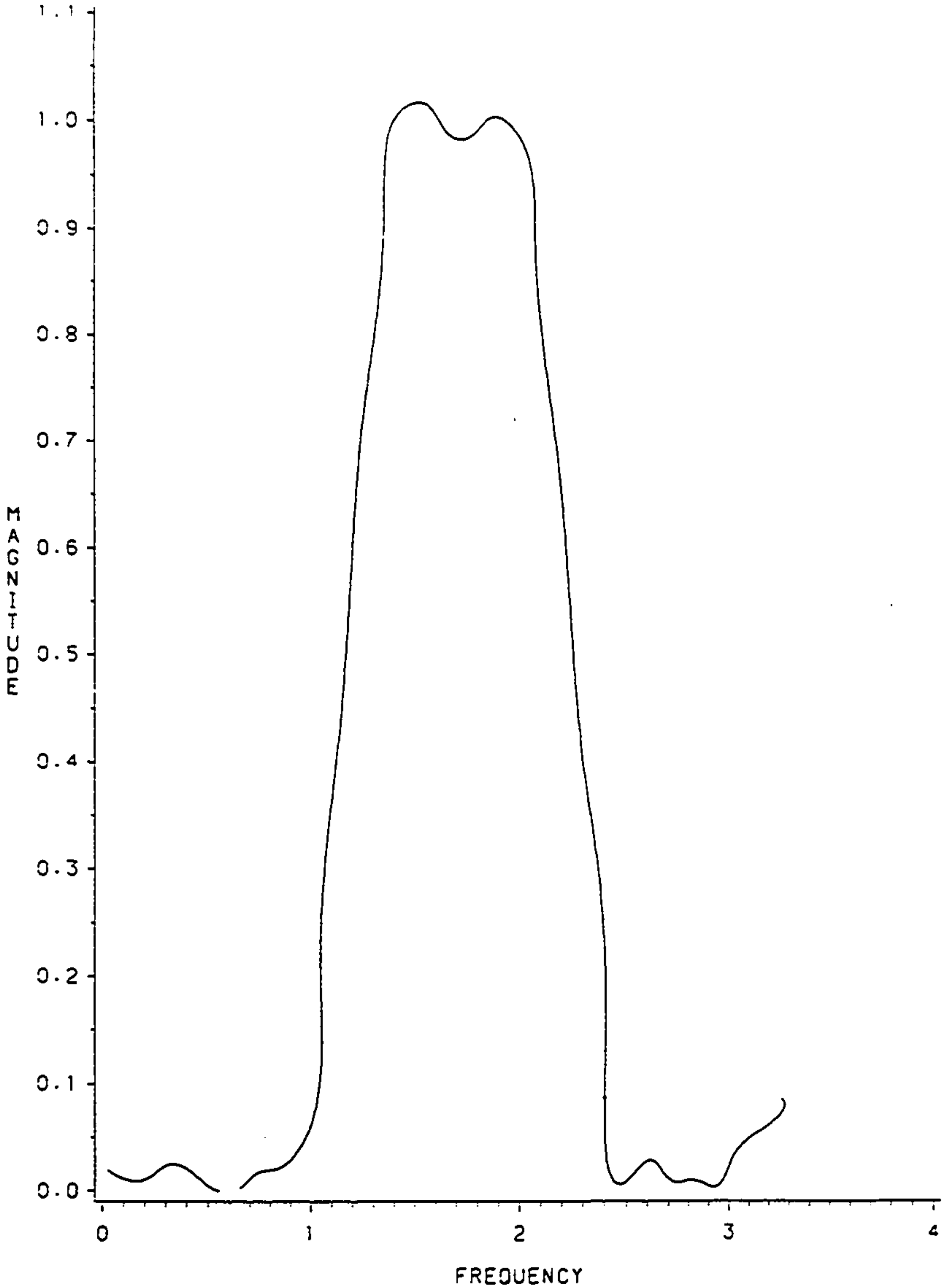


Figure (2-19) :The magnitude response for example 3 using linear programming

Example 3 : Design a band pass 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 1 & 1.5 \le \omega \le 2 \\ 0 & 0 \le \omega \le 1 \\ & 2.5 \le \omega \le \pi \end{cases}$$

Table 2-30 shows the integer coefficients of the designed filter using this technique and Table 2-31 shows the characteristics of the design technique for example 3. Figure 2-19 shows the magnitude response of the design filter.

a(0) = 44	a(3) = 22	a(6) = 2	a(9) = 8
a(1) = -12	a(4) = 22	a(7) = -8	
a(2) = -68	a(5) = -6	a(8) = -4	

Table 2-30: The integer coefficients for example 3 using technique 3

DESIGN TECH.	R. TIME(SEC.)	M.M.P.	M.M.S.
LINEAR PROG.	100	1.0156	.0546

Table 2-31: Characteristics of the design technique in example 3

Example 4: Design a stop band 1-D FIR filter with the order of 19, B=8, and the following specifications:

$$|H_i(e^{j\omega})| = \begin{cases} 0 & 1.5 \le \omega \le 2 \\ \\ 1 & \begin{array}{c} 0 \le \omega \le 1 \\ 2.5 \le \omega \le \pi \end{array} \end{cases}$$

Table 2-32 shows the integer coefficients of the designed filter using this technique and Table 2-33 shows the characteristics of the design technique for example 4. Figure 2-20 shows the magnitude response of the designed filter.

a(0) = 85	a(3) = -22	a(6) = -1	a(9) = 8
a(1) = 12	a(4) = -25	a(7) = 7	
a(2) = 66	a(5) = 8	a(8) = 4	

Table 2-32: The integer coefficients for example 4 using technique 3

DESIGN TECH.	R. TIME(SEC.)	M.M.P.	M.M.S.
LINEAR PROG.	100	1.0390	.0168

Table 2-33: Characteristics of the design technique in example 4

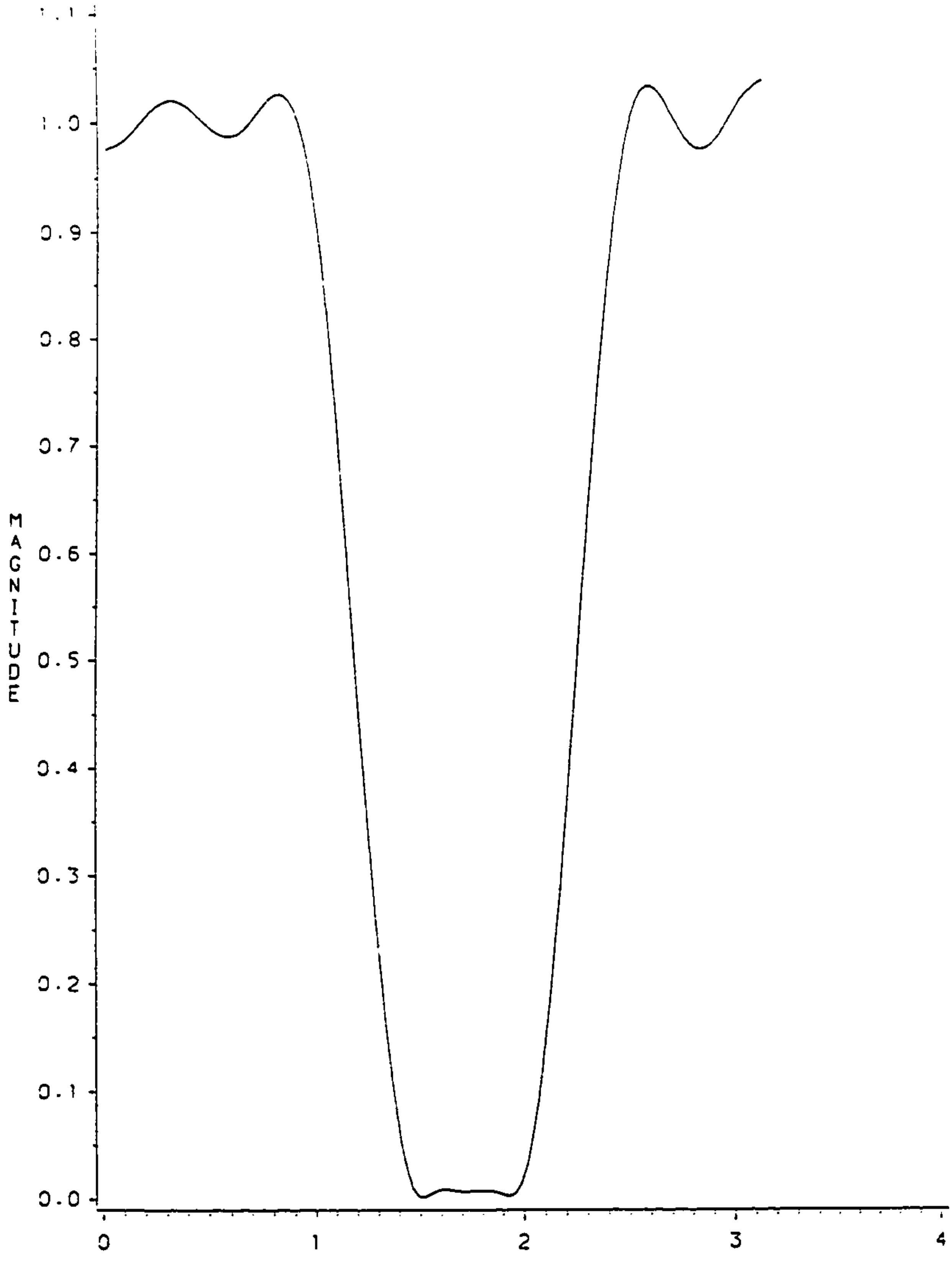


Figure (2-20) :The magnitude response for example 4 using linear programming (example 4)

2.5. DELTA MODULATION

In this section, we implement the technique proposed by Mark R. Bateman [49] to design FIR filters with the coefficients equal to 0, -1, and 1. The basic idea behind this approach is to convert the filter coefficients into a sequence of -1, +1 and 0 by a delta modulation-like scheme.

A delta modulator encodes an analogue input signal into binary pulses which are conveyed to the terminal equipment for transmission. These pulses are also locally decoded back into an analogue waveform by an integrator in the feedback loop and subtracted from the input signal to form an error which is quantized in one of two possible levels depending on its polarity. The output of the quantizer is sampled periodically to produce the output binary pulse. Figure (2-21) displays a linear delta modulation system. Figure (2-22) displays linear delta modulator waveforms when encoder is correctly tracking the input signal.

The structure of the design is based on the idea of delta modulation and consists of a transversal filter with tap coefficients restricted to -1, 0, 1, and cascaded with an accumulator. As in delta modulation the sampling rate must be sufficiently high to obtain acceptable performance with the consequence that useful transversal filters are very long (many taps).

In this approach the assumed delta modulator applies to the filter impulse response to yield a filter with only the coefficients 0, -1, +1. As in delta modulation over-sampling is required to keep the error at an acceptable level, it corresponds to interpolating the original impulse response h(n) by a factor k to get a new sequence ($h_k(n)$). The frequency response corresponding to h(n) and $h_k(n)$ is as follows:

$$H_k(\omega) = \sum_{n=0}^{\infty} h_k(n) e^{-j\omega n} \tag{5-1}$$

$$H(\omega) = \sum_{n=0}^{\infty} h(n) e^{-j\omega n} \tag{5-2}$$

The block diagram for the filter structure is shown in Figure (2-23).

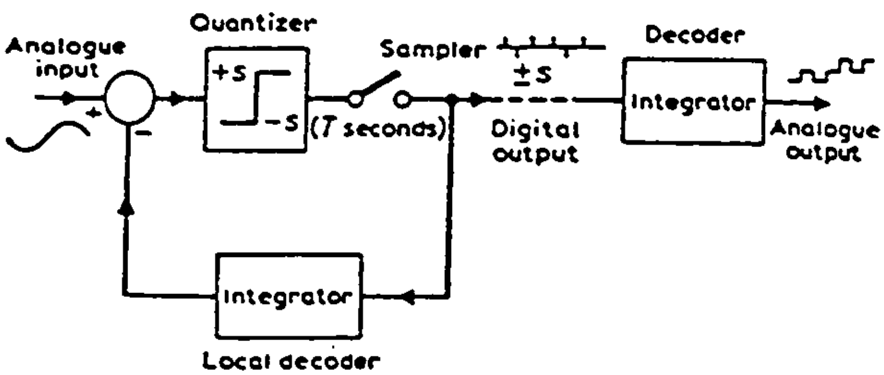
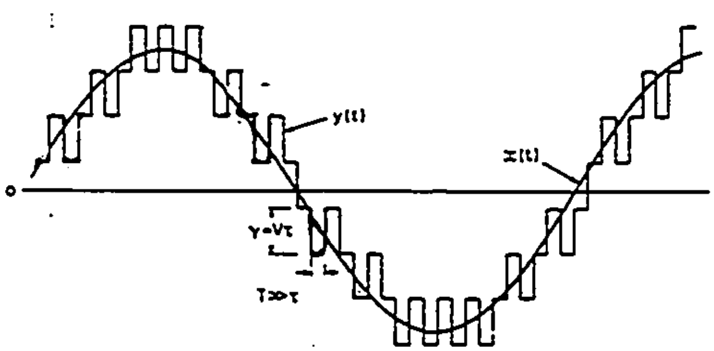


Figure (2-21) Delta modulator and decoder.



Figure(2-22) linear d.m. waveforms when encoder is correctly tracking the input signal

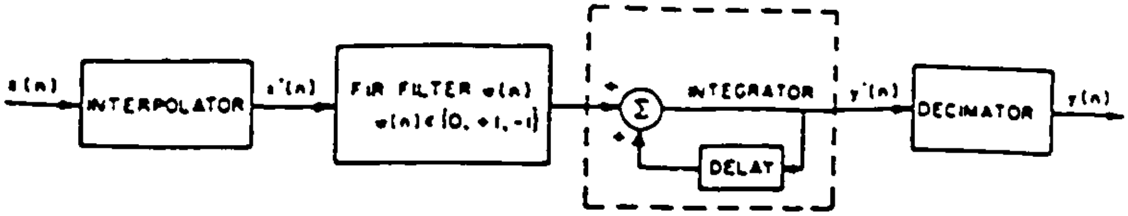


Figure (2-23): The block diagram for the filter structure.

Let r(n) be the sequence consisting of the +1, -1, and 0, we desired the running sum of the sequence r(n) to be a good approximation to the sequence $h_k(n)$

$$E_k = \sum_{n=0}^{2N} [h_k(n) - \Delta_k \sum_{p=0}^{n} r(n)]^2 \qquad (5\text{-}3)$$

E_k is within a prescribed tolerance. Δ_k is the scaling factor corresponding to the stepsize in delta modulation.

2.5.1. DESIGN STEPS

1) Specify a interpolation factor, K.

2) Determine the upper bound for the Δ_k by

$$\Delta_k \leq \frac{\gamma}{k^2} \qquad (5\text{-}4)$$

where

$$\gamma = 0.43616(b+a)\sqrt{b^2+a^2} \qquad (5\text{-}5)$$

a and b are passband and stopband edges respectively.

3) Determine the order of the filter by

$$N \approx 2\left[\frac{k}{(b-a)\pi\Delta_k}\right]^{\frac{1}{2}} \tag{5-6}$$

h_k (n) is truncated at $\pm$ N and delta modulated with the step size Δ_k. Therefore the resulting order for this filter is 2N.

4) Obtain h_k by any one of the optimization routines, and the following least square error

$$e = \sum_{\omega=0}^{\pi}\left[H_i - \sum_{n=0}^{2N} h_k(n) z^{j\omega n}\right]^2 \tag{5-7}$$

where H_i is the ideal specifications of the interpolated filter.

5) Set the high estimate for Δ_k as follows

$$\Delta_k = \max \mid h_k(n) - h_k(n-1) \mid \quad . \tag{5-8}$$

6) State r(n)=0, for $0 \leq n \leq 2N$.

7) Evaluate E_k in equation (5-3). Set E = E_k.

8) Set i = 0.

9) If r(i) = 1 go to step(17) , otherwise increment r(i) by a unit.

10) Set j = i+1.

11) Find a value for (j) in such away that r(j) $\neq$ -1.

12) Decrement r(j) by one and evaluate new E_k.

13) If $E_k < E$ set $E = E_k$, and go to step (25) otherwise reset r(j) to its former value.

14) Find a value for (k) in such away that $K > j$ and $r(k) \neq -1$.

15) Decrement r(k) by one and evaluate E_k.

16) If $E_k < E$ set $E = E_k$, and go to step (25) otherwise reset r(k) to its former value and go to step 17.

17) If $r(i) = -1$ go to step(25) , otherwise decrement r(i) by a unit.

18) Set $l = i+1$.

19) Find a value for (l) in such away that $r(l) \neq 1$.

20) Increment r(l) by one and evaluate new E_k.

21) If $E_k < E$ set $E = E_k$, and go to step (25) otherwise reset r(l) to its former value.

22) Find a value for (m) in such a way that $m > l$ and $r(m) \neq 1$.

23) Increment r(m) by one and evaluate E_k.

24) If $E_k < E$ set $E = E_k$, and go to step (25) otherwise reset r(k) to its former value and go to step 25.

25) If $i < 2N$, set $i = i + 1$, and go to step (9).

26) Repeat steps 8 - 25 until no more changes in the sequence r(i) are made.

27) Set

$$\frac{dE_k}{d\Delta_k} = 0 \tag{5-9}$$

and find the new value for Δ_k

$$\Delta_k = \frac{\sum_{n=0}^{2N} \left[\sum_{p=0}^{n} r(p) \right] h_k(n)}{\sum_{n=0}^{2N} \left[\sum_{p=0}^{n} r(p) \right]^2} \tag{5-10}$$

28) Repeat steps 6-27 until no more improvements in the value of E_k and no more changes in the sequence r(n) are made.

29) Perturb the value of Δ_k slightly and repeat steps (6-28) until E_k reaches to the prespecified minimum error.

30) Decimate the resulting output by following equation

$$H(\omega) = \frac{1}{k}\sum_{m_1=0}^{k-1} H_k(\omega') \qquad (5\text{-}11)$$

where

$$\omega' = \frac{\omega + 2\pi m_1}{k} \qquad (5\text{-}12)$$

and

$$H(\omega') = \Delta_k (1 - e^{-j\omega'})^{-1} \sum_{n=0}^{2N} r(n) e^{-j\omega' n} \qquad (5\text{-}13)$$

2.5.2. EXAMPLES

EXAMPLE 1: Design a low pass FIR filter with k = 7 and the following characteristics

$$H(e^{j\omega}) = \begin{cases} 1 & 0 \le \omega \le 1 \\ \\ 0 & 2 \le \omega \le \pi \end{cases}$$

$$\gamma = 0.43616(2+1)\sqrt{2^2+1^2} = 2.9258$$

$$\Delta_k \le \frac{\gamma}{k^2}$$

The maximum value for Δ_k

$$\Delta_k = 0.05971$$

$$N \approx 2\left[\frac{7}{(\frac{2}{7}-\frac{1}{7})\times\pi\times 0.05971}\right]^{\frac{1}{2}} \approx 32$$

$$2N = 64$$

The characteristics of this design technique are shown in table (2-34). The magnitude response of this filter is displayed in Figure (2-24).

DESIGN TECH.	# R TIME (SEC)	M.M.P.	M.M.S
DELTA MODULATION	60	1.1037	.0844

Table 2-34: The characteristics of the example 1 using Delta modulation technique

EXAMPLE 2: Design a low pass FIR filter with k = 3 and the following characteristics

$$H(e^{j\omega}) = \begin{cases} 1 & 0 \le \omega \le 0.3 \\ 0 & 0.6 \le \omega \le \pi \end{cases}$$

$$\gamma = 0.43616(0.6+0.3)\sqrt{0.6^2+0.3^2} = 0.2633$$

$$\Delta_k \le \frac{\gamma}{k^2}$$

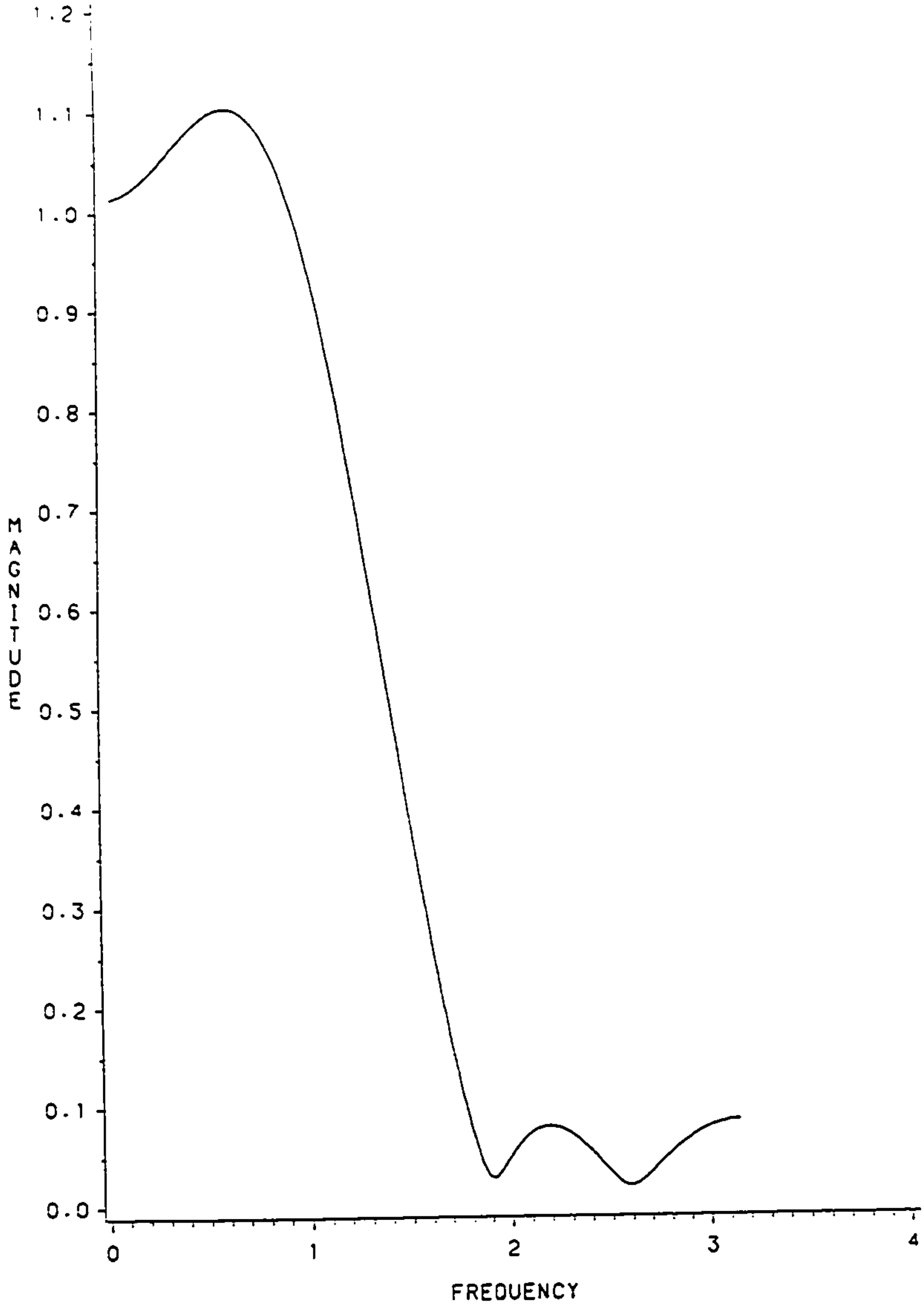


Figure (2-24) : The magnitude response for example 1 with the coefficients +1, 0, -1

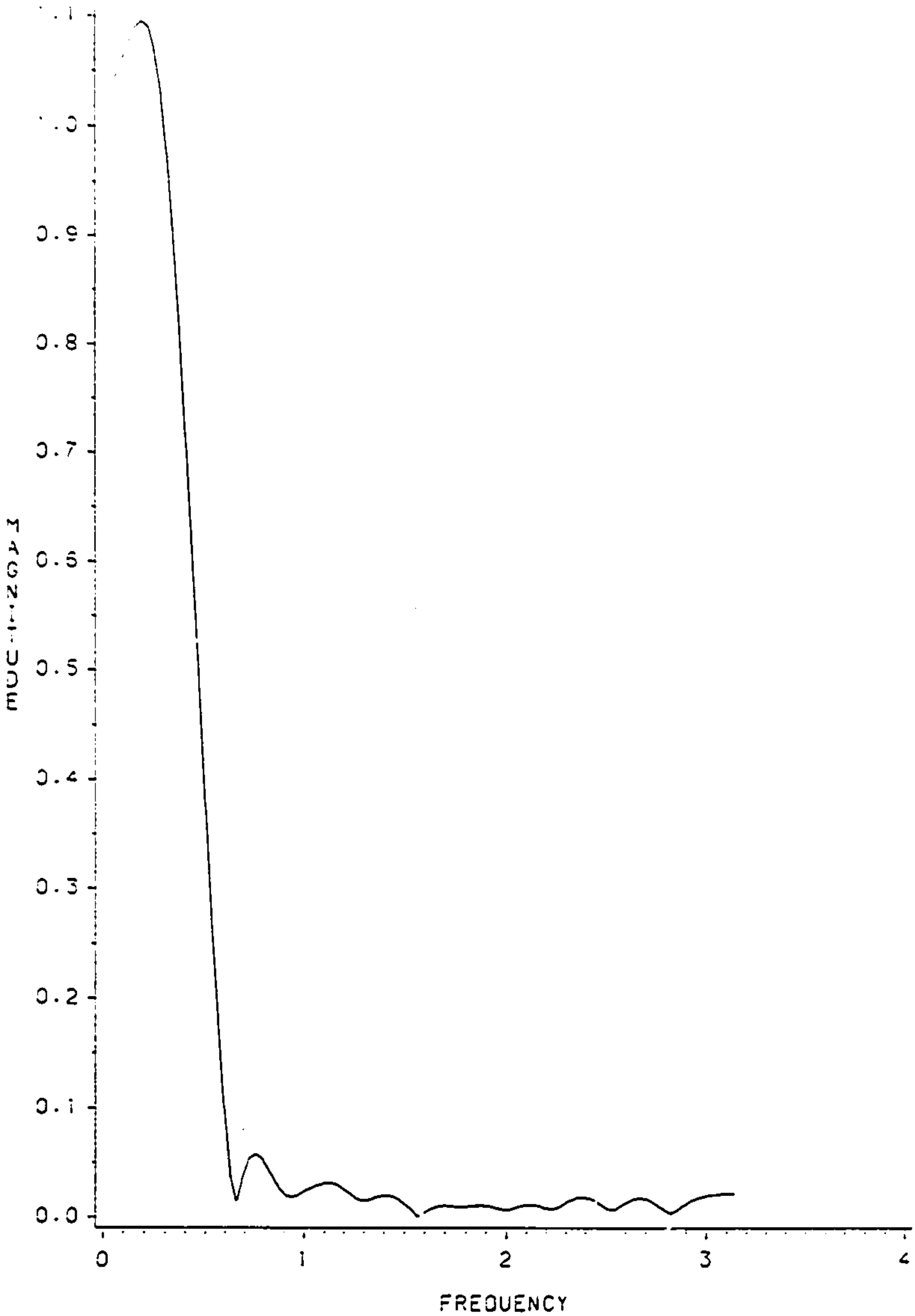


Figure (2-25) : The magnitude response with the coefficients +1, 0, -1 ,example 2

The maximum value for Δ_k

$\Delta_k = 0.02925$

$$N = 2\left[\frac{3}{(\frac{0.6}{3} - \frac{0.3}{3}) \times \pi \times 0.02925}\right]^{\frac{1}{2}} = 36$$

$2N = 72$

The characteristics of this design technique show in table (2-36). The magnitude response of this filter is displayed in Figure (2-25).

DESIGN TECH.	# R TIME (SEC)	M.M.P.	M.M.S
DELTA MODULATION	360	1.0948	.05721

Table 2-35: The characteristics of the example 2 using Delta modulation technique

DESIGN TECH.	R. TIME (SEC.)	M.M.P	M.M.S
I.P.C	5	1.0015	.0014
D & R (TYPE 1)	150	.99613	.0121
D & R (TYPE 2)	330	1.0078	.0078
MIXED INTEGER PROG	400 (60)	1.0119	.0158
DELTA MODULATION	60	1.1037	.0844

Table 2-36: The comparison table for example 1

DESIGN TECH.	# R. TIME (SEC.)	M.M.P	M.M.S
I.P.C	5	1.0207	.0155
D & R (TYPE 1)	60	1.0087	.0164
D & R (TYPE 2)	160	1.0156	.0159
MIXED INTEGER PROG.	400 (60)[1]	1.0312	.0644

Table 2-37: The comparison table for example 2

DESIGN TECH.	# R. TIME (SEC.)	M.M.P	M.M.S
I.P.C	10	1.0128	.0254
D & R (TYPE 1)	140	.99218	.0298
D & R (TYPE 2)	400	1.0132	.0281
MIXED INTEGER PROG.	500 (100)	1.0156	.0546

Table 2-38: The comparison table for example 3

DESIGN TECH.	# R. TIME (SEC.)	M.M.P	M.M.S
I.P.C	10	1.0257	.0134
D & R (TYPE 1)	70	1.0202	.0197
D & R (TYPE 2)	120	1.0281	.0207
MIXED INTEGER PROG.	400 (100)	1.0390	.0168

Table 2-39: The comparison table for example 4

DESIGN TECH.	# R. TIME (SEC.)	M.M.P	M.M.S
I.P.C	20	1.0007	.0001
D & R (TYPE 1)	900	.98099	.0165
D & R (TYPE 2)	1500	1.0078	.0170
DELTA MODULATION	180	1.0948	.0572

Table 2-40: The comparison table for example 5

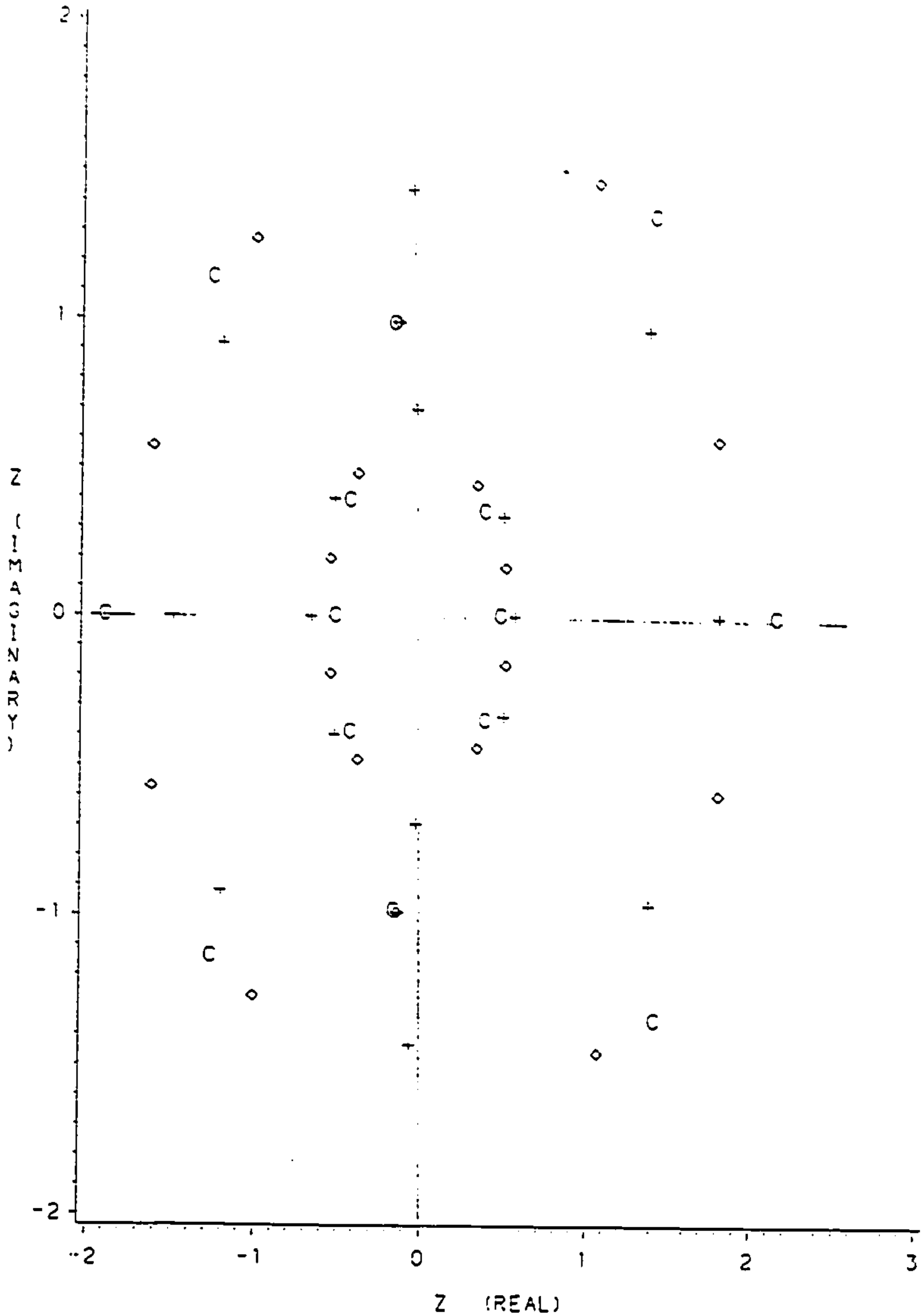


Figure (2-26): The location of zeros for example 1, (circle=type 1), (diamond= type 2), (plus type 3)

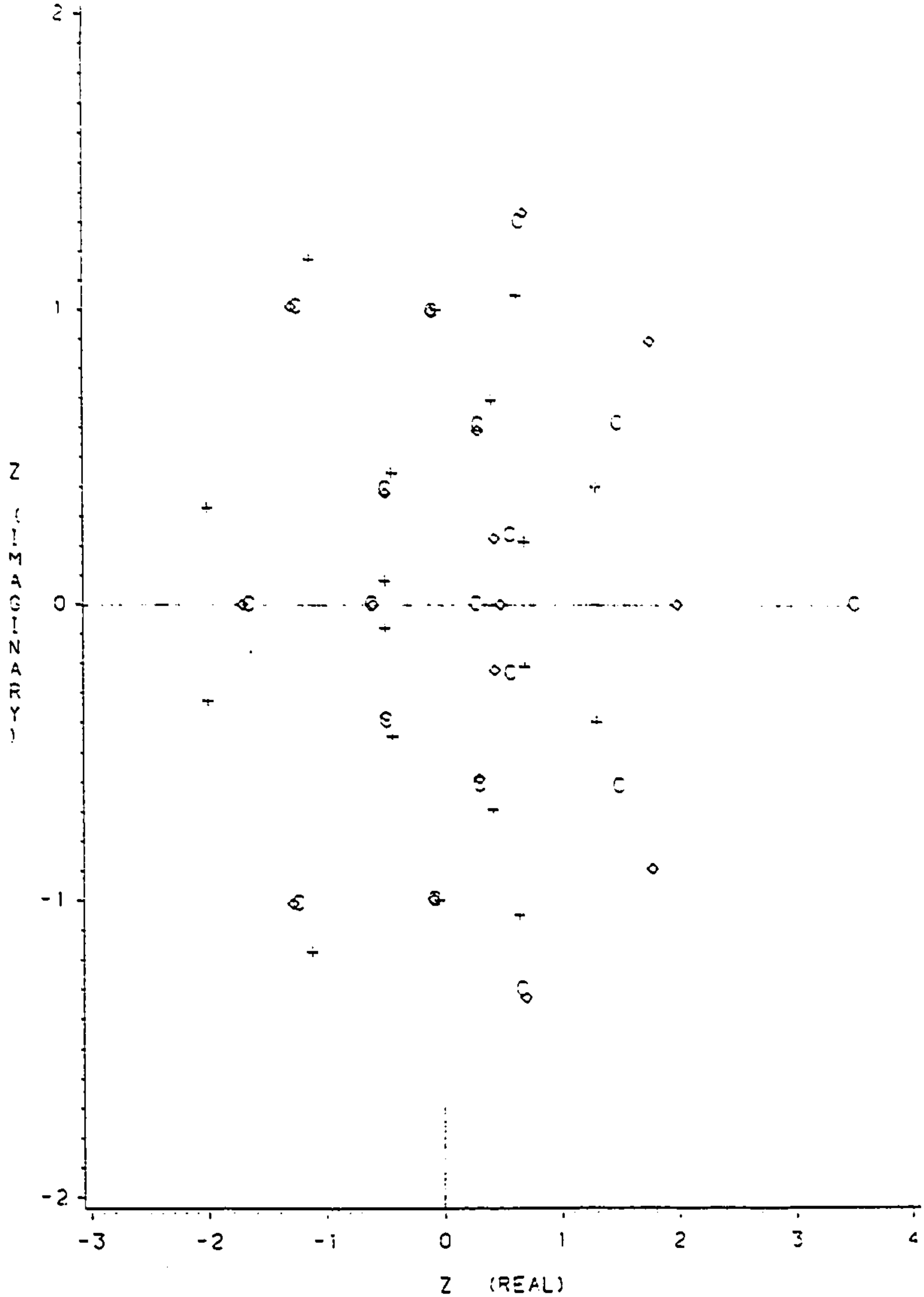


Figure (2-27): The location of zeros for example 2, (circle=type 1), (diamond= type 2) (plus=type 3)

2.6. SUMMARY

In this chapter the following subjects have been investigated:

1) The importance of the coefficient quantization error and its effect on the behaviour of the filter has been determined. To reduce this error several techniques for designing a FIR filter with integer coefficients have been presented, some of which were categorized and explored in the introduction of this chapter.

2) Four techniques have been considered for in-depth consideration.

2-1) Discretization and reoptimization (type 1):

A simple design approach is presented in which the whole process consists of an N times optimization process where N is the number of coefficients which has to be determined. At each iteration the least effective coefficient on error function is fixed to integer value and the remaining coefficients are determined through the optimization process.

2-2) Discretization and reoptimization (type 2)

A simple design approach is implemented in which the whole process consists of a 2N times optimization process in which N is the number of coefficients which has to be determined. At each iteration the most sensitive coefficient is discretized to integer values and the remaining coefficients are determined through the optimization process for two cases. The remaining coefficients are fixed to the set of values corresponding to minimum objective function. This process will apply until all the coefficients are converted to an integer values.

2-3) Mixed integer linear programming

The formulation of the problem for linear and integer programming has been explained.

Two packages named LINDO and MINT algorithm have been used for this technique. The comparison between these two packages and characteristics of this technique will be explained in chapter 5.

2-4) Delta modulation like filter

An attractive filter structure based on the delta modulation concept has been explained. The application of the design technique has been investigated by applying a few examples. The over-sampling factor" K" and the step size" Δ_k" play an important part in this technique.

To compare these techniques, Tables (2-36,37,38,40) display the characteristics of these techniques. The location of zeros in Z domain is one of the important investigations in filter design. By considering this fact, the zeros of the first two examples are displayed in Figures (2-26) and (2-27). The coefficients obtained by these techniques are different but the location of zeros for most of the coefficients are almost the same.

CHAPTER THREE

DESIGN OF THE 2-D FIR FILTER WITH INTEGER COEFFICIENTS

3.1. INTRODUCTION

During the past few years various design methods have been proposed for two dimensional finite impulse response digital filters; we have described some of those techniques in chapter one. The main advantages of such filters are inherent stability and being able to provide exact linear phase. This latter property is particularly important in many image processing applications.

Many reports in the literature concern the analysis of error resulting from finite word length in the 1-D or 2-D case. However, examples of design of 2-D Digital filters with integer coefficients are very rare, even though, in theory, most of the methods currently used in the 1-D case could be employed for the 2-D case. Some of the design techniques for 2-D FIR filters with integer coefficients are as follows.

An algorithm was presented by Siohan and Benslimane [51]. This algorithm associates linear programming and branch and bound technique. In the presented method the infinite precision design problem is solved before rounding the optimal continuous coefficients. Based on these infinite precision values, the discretization problem and the tree search strategies are used to obtain a set of integer coefficients for the design problem. The efficiency of the technique is limited for the design of 2-D FIR filters with different specifications and sizes, up to 9 x 9 in the case of circularly symmetric and up to 13 x 13 for diamond-shaped filters.

The second design technique was presented by Pei and Jaw [52]. This technique involves mapping of mulitiplierless 1-D FIR filters into 2-D filters by a change of variables. The proposed techniques by Lim and Parker [46], [47] have been used to design a 1-D multiplierless FIR filter and then the original McClellan transformation has been used for transforming the variables from 1-D to 2-D.

In this chapter, we present the extension of some of the techniques to 2-D filter design criteria. In this thesis several techniques are presented for the design of 2-D FIR filter with integer coefficients . These techniques are generally based on the extension of Discretization and Reoptimization (type 1&2) to 2-D FIR filters. Although the results confirm that these extensions are mathematically and practically possible, a great deal of computation and memory are required to implement these techniques. A third approach is also presented in this thesis which is based on the use of the McClellan transformation. This technique presents a computational efficient algorithm which is easy to implement and yields satisfactory results.

3.2. DISCRETIZATION AND REOPTIMIZATION (TYPE 1) IN 2-D FIR FILTER

In this section, we introduce a technique to design 2-D FIR filters with octagonal symmetry and with integer coefficients using discretization and reoptimization (type 1). The approach starts with an optimal design with infinite precision coefficients. The structure of the design technique for 1-D has been explained in chapter 2 and for the sake of brevity has been excluded in this chapter; however, the design steps for the 2-D case will be presented as follows.

In this technique, the number of specified bits for representing the coefficients is the first data which has to be provided by the user. If the infinite precision coefficients of the designed filter are b_i, where $i=(N-1)/2$, N is the order of filter, if

$$-1 \leq b_i \leq +1 \tag{3-1}$$

then b_i can be shown by the following form

$$b_i = \frac{a_i}{2^{B-1}} \tag{3-2}$$

where B is the number of the bits including the sign bit and a_i is the integer coefficients for the prescribed filter. Equation 3-2 shows that for obtaining the integer coefficients the prescribed magnitude response has to be scaled by 2^{B-1}, but for arbitrary specification, other than equation (3-1), a scaling factor could be used in order to recover the desired range. Various steps of the algorithm are as follows.

1) Find the infinite precision coefficients for the designed filter by nonlinear programming and the following least square error.

Equation (3-3) presents the error criterion for this design

$$E=\sum_{\omega_1=0}^{\pi}\sum_{\omega_2=0}^{\omega_1}\left[(|H_i|-|\sum_{n_1=0}^{\frac{N-1}{2}}\sum_{n_2=0}^{\frac{N-1}{2}}a(n_1,n_2)\cos(\omega_1 n_1)\cos(\omega_2 n_2)|\right]^2 \qquad (3\text{-}3)$$

where H_i is the specification of the desire filter and N is the order of the filter. Where a(i,j) is the coefficients of the designed filter.

2) Set i=0.

3) Set j=0.

4) Set a (i,j) = a^+ (i,j) where a^+ (i,j) is the round up value for a(i,j) and find the value of error function, equation (3-3); the resulting value can be shown by E^+ (i,j).

5) Set a (i,j) = a^- (i,j) where a^- (i,j) is the round down value for a(i,j) and find the value of error function, equation (3-3); the resulting value can be shown by E^- (i,j).

6) Find E (i,j) = min{ E^+ (i,j), E^- (i,j)} and save the value.

7) Set a (i,j) to its initial value and state j=j+1.

8) Repeat steps 4-7 until j > N-1 / 2.

9) Set i=i+1.

10) Repeat steps 3-9 until i > N-1 / 2.

11) Fix the value of the corresponding coefficient to the minimum error function calculated within steps 3-10 to a^+ (i,j) or a^- (i,j).

12) Go to step 1 and do the optimization routine to find a new set of values for the rest of the coefficients.

13) Repeat steps 1-12 until all coefficients convert to integer value.

In the following section an example of 2-D FIR filter with circular symmetry will be considered to illustrate the application of the design technique.

EXAMPLE 1: Design a 2-D FIR lowpass filter with the order 11 × 11 and the following specifications using discretization & reoptimization (type 1):

$$H_i = \begin{cases} 1 & 0 \le \sqrt{\omega_1^2+\omega_2^2} \le 1 \\ \\ 0 & 2 \le \sqrt{\omega_1^2+\omega_2^2} \le \pi \end{cases}$$

Based on the characteristics of octagonal symmetry the number of coefficients in the optimization process is reduced from 66 to 21. Table (3-1) shows the characteristics of this design technique. Table (3-2) presents the integer coefficients obtained by discretization and reoptimization (Type 1). The frequency response of this filter is displayed in Figure (3-1).

DESIGN TYPE	# R TIME(SEC.)	M.M.P.	M.M.S.
D & R (TYPE 1)	420	1.0230	.0566

Table (3-1): Characteristics of the design technique (type 1) in example 1

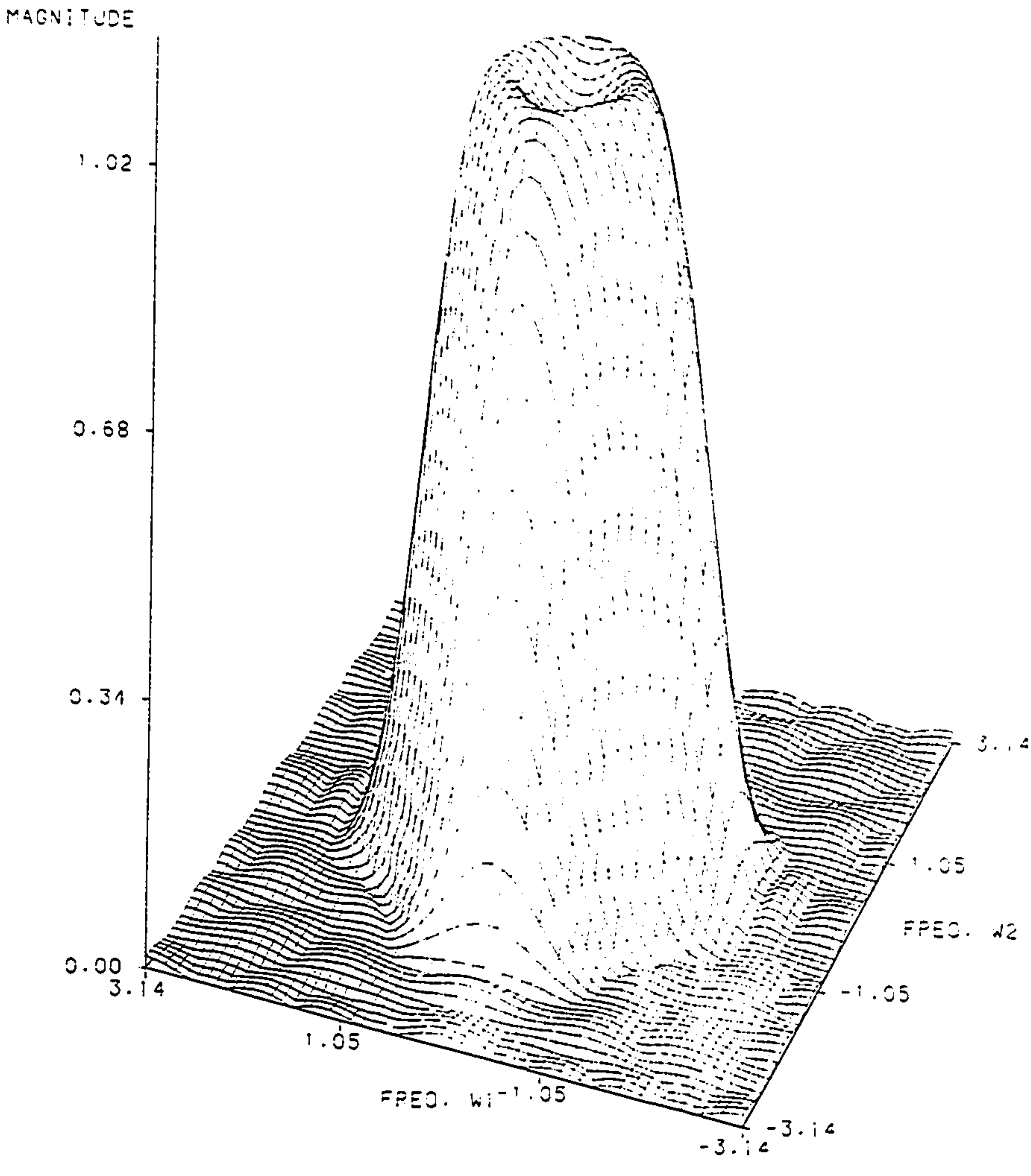


Figure (3-1) : The magnitude response of a 2-D FIR filter with integer coefficients

a(0,0) = 22	a(3,1) = -6	a(4,4) = 2
a(1,0) = 32	a(3,2) = -9	a(5,0) = 0
a(1,1) = 44	a(3,3) = -5	a(5,1) = 0
a(2,0) = 10	a(4,0) = -1	a(5,2) = 1
a(2,1) = 9	a(4,1) = -2	a(5,3) = 1
a(2,2) = -7	a(4,2) = -2	a(5,4) = 1
a(3,0) = -1	a(4,3) = 0	a(5,5) = 0

Table 3-': The integer coefficients for example 1 using type 1

3.3. DISCRETIZATION AND REOPTIMIZATION (TYPE 2) IN 2-D FIR FILTER

In this section, we employ the discretization and reoptimization (type 2) technique to design a 2-D FIR filter with integer coefficients and octagonal symmetry. Similar to technique one, nonlinear programming will be utilized. The design steps for this technique are as follows.

In this technique, the number of specified bits for representing the coefficients is (B) which includes the sign bit. Similar to technique 1, it is obvious that the proper scaling in the prescribed magnitude response in order to recover the desired range has to be applied. The design steps with the required explanation at each step are as follows:

1) Set M = the total number of coefficients which has to be determined. We have to mention that the problem is a two dimensional problem but for the simplicity of presenting the steps, the notations are shown in one dimension. For example the value M represents the total number of coefficients in two dimensions.

2) Set k=0.

3) Find the infinite precision coefficients for the designed filter by nonlinear programming and the following least square error. Equation (3-4) presents the error criterion for this design:

$$E=\sum_{\omega_1=0}^{\pi}\sum_{\omega_2=0}^{\omega_1}\left[(|H_i|-|\sum_{n_1=0}^{\frac{N-1}{2}}\sum_{n_2=0}^{\frac{N-1}{2}}a(n_1,n_2)\cos(\omega_1 n_1)\cos(\omega_2 n_2)|\right]^2 \qquad (3\text{-}4)$$

where H_i is the ideal frequency response and a_{ij} are the coefficients of designed filter.

4) Set $M = \frac{N-1}{2} - k$.

5) Find the most sensitive coefficients among (M) coefficients. Based on equation (1-17) and the definition of a derivative, the new form can be written in the form of equation (3-5).

$$S_{a_i}(z)=\left|\frac{H(x_1,x_2,\ldots,x_i,\ldots,x_n)-H(x_1,x_2,\ldots,x_i+\Delta x_i,\ldots,x_n}{H(x_1,x_2,\ldots x_i,\ldots,x_n)}\right|$$

$$+\left|\frac{H(x_1,x_2,\ldots,x_i,\ldots,x_n)-H(x_1,x_2,\ldots,x_i-\Delta x_i,\ldots,x_n}{H(x_1,x_2,\ldots,x_i,\ldots,x_n)}\right| \qquad (3\text{-}5)$$

where Δx_i is the variation for x_i and equal to one. Equation (3-5) will be calculated for all the coefficients and finally the assumed coefficient a_{ii} with the maximum sensitivity will be chosen.

6) Set $a_{ii} = a_{ii}^+$ where a_{ii}^+ is the rounding up value for a_{ii} and solve the following optimization problem:

$$E^+ = E(a_{ij}, A_{M-1}^+) \tag{3-6}$$

where E^+ is the value of error function and A_{M-1}^+ is the vector with M-1 remaining variables which are determined in the optimization process.

7) Set $a_{ii} = a_{ii}^-$ where a_{ii}^- is the rounding down value for a_{ii} and solve the following optimization problem:

$$E^- = E(a_{ij}, A_{M-1}^-) \tag{3-7}$$

where E^- is the value of the error function and A_{M-1}^- is the vector with M-1 remaining variables which are determined in the optimization process.

8) Find $E = \min\{E^+, E^-\}$ and fix the vector A to (a_i^+, A_{M-1}^+) or (a_i^-, A_{M-1}^-).

9) Repeat steps 4-8 until all components of the vector A convert to integer value.

In the following section several examples have been provided to illustrate the application of the design technique.

EXAMPLE 2: Design a 2-D FIR lowpass filter with the order 11 × 11 and the following specifications using discretization & reoptimization (type 1)

$$H_i = \begin{cases} 1 & 0 \le \sqrt{\omega_1^2+\omega_2^2} \le 1 \\ \\ 0 & 2 \le \sqrt{\omega_1^2+\omega_2^2} \le \pi \end{cases}$$

Based on the characteristics of octagonal symmetry the number of coefficients in the optimization process is reduced from 66 to 21. Table (3-3) presents the integer coefficients obtained by discretization and reoptimization (Type 2). Table (3-4) shows the characteristics of this design technique. The frequency response of this filter is displayed in Figure (3-2).

a(0,0)= 23	a(3,1)= -7	a(4,4)= 3
a(1,0)= 33	a(3,2)= -10	a(5,0)= 1
a(1,1)= 45	a(3,3)= -5	a(5,1)= 1
a(2,0)= 10	a(4,0)= -1	a(5,2)= 1
a(2,1)= 9	a(4,1)= -3	a(5,3)= 2
a(2,2)= -7	a(4,2)= -2	a(5,4)= 2
a(3,0)= -2	a(4,3)= 1	a(5,5)= 0

Table (3-3): the integer coefficients for example 1 using type 2

DESIGN TYPE	# R TIME(SEC.)	M.M.P.	M.M.P.
D & R (TYPE 2)	720	1.0385	.0337

Table (3-4): Characteristics of the design technique (type 2) in example 1

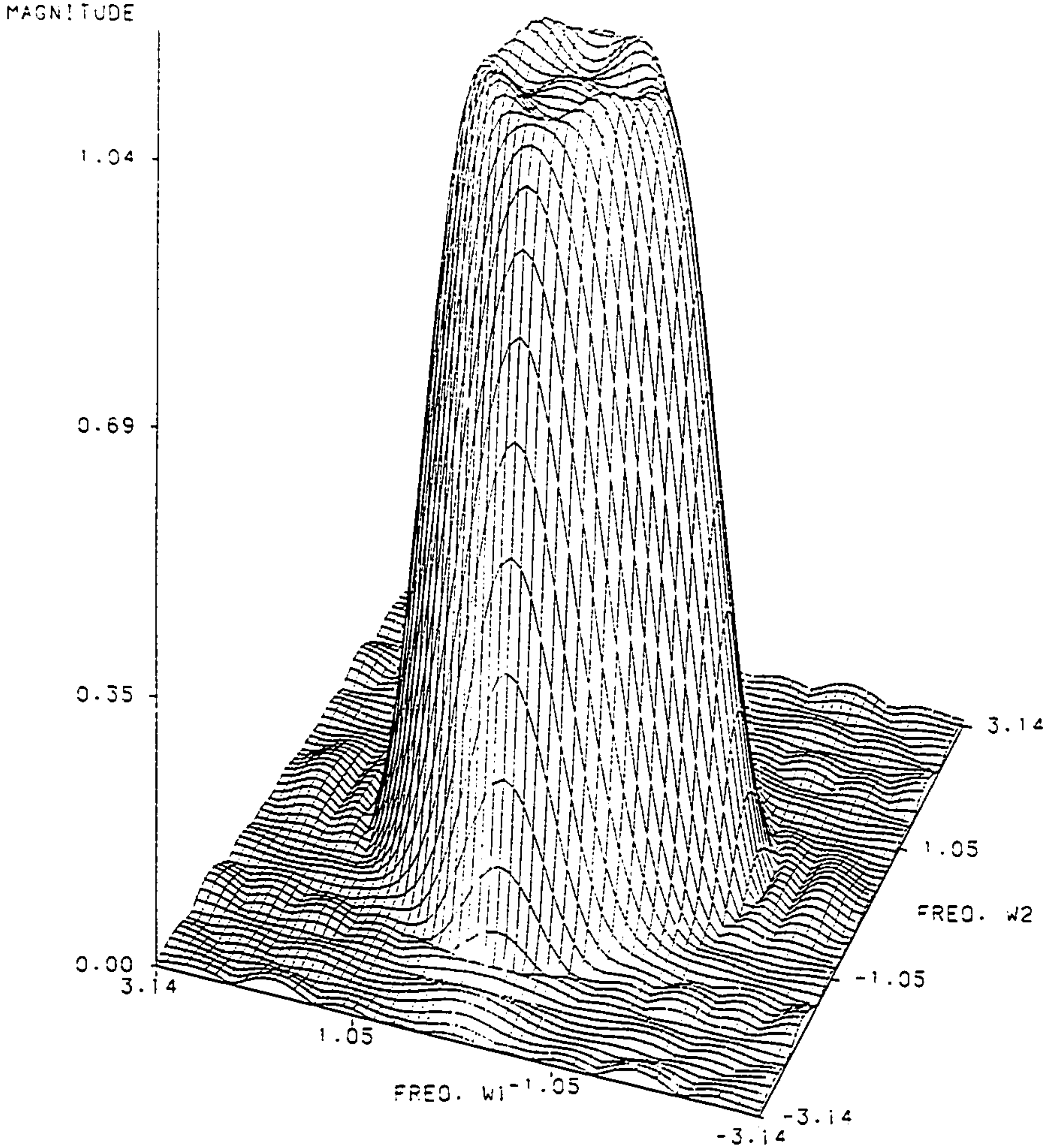


Figure (3-2) : The magnitude response of a 2-D FIR filter with integer coefficients

3.4. McCLELLAN TRANSFORMATION

A transformation for the design of 2-D FIR filters was first developed by James H. McClellan [27], then generalized by Mersereau and Mecklenbrauker [28]. The concept concerns itself with the transformation of a 1-D zero-phase FIR filter to 2-D zero-phase FIR filter by substitution of variables. It can be applied to 1-D filters of odd length and in one special case also to a filter of even length.

For a 1-D FIR filter of length 2M+1 to be zero-phase, its impulse response, h(n) must have Hermitian symmetry coefficients. Thus if h(n) is real, it must be also even. The frequency response $H(e^{j\omega})$ can thus be expressed in the form

$$H(e^{j\omega}) = h(o) + \sum_{n=1}^{M} h(n)[e^{-j\omega n} + e^{j\omega n}] \tag{3-8}$$

$$H(e^{j\omega}) = h(o) + \sum_{n=1}^{M} h(n)[\cos(\omega n)] \tag{3-9}$$

Equation (3-9) can be written in the following form

$$H(e^{j\omega}) = \sum_{n=0}^{M} b(n) \; [\; \cos(\omega)]^{n} \tag{3-10}$$

The McClellan transformation transforms a point in ω axis to a contour in the $(\omega_1\ \omega_2)$ plane by the following transformation

$$\cos(\omega) = \sum_{k=0}^{1} \sum_{l=0}^{1} t(k,l) \cos(\omega_1 k) \cos(\omega_2 l) \tag{3-11}$$

where t(k,l) are called the transformation parameters. By substitution of equation (3-11) into equation (3-10):

$$H(e^{j\omega_1},e^{j\omega_2})=\sum_{n=0}^{M} b(n)\left[\sum_{k=0}^{1}\sum_{l=0}^{1} t(k,l)\ \cos(\omega_1 k)\ \cos(\omega_2 l)\right]^{n} \tag{3-12}$$

Application of the recurrence formula for Chebyshev polynomials enables this latter relation to be expressed in the form of a 2-D zero-phase FIR filter. The resulting 2-D filter is of size (2M+1) × (2M+1).

It follows implicitly from equation (3-11) that points in the frequency response of the 1-D filter are mapped to contours in the (ω_1 ω_2) plane. Furthermore, the shape of these contours is determined only by the parameters t. On the other hand, the variation of the frequency response from contour to contour is controlled by the coefficients b(n).

By looking at equation (3-9) one can see that the design procedure for this technique can be split into two parts.

A) Design of the transformation parameters, t, such that the contours (ω=constant) produced by the transformation of the equation (3-11) in the (ω_1 ω_2) plane have some desired shape. Our desired shape in this technique is a circular contour.

B) Design of the 1-D zero-phase prototype filter with the characteristic of the equation (3-10).

We have already explained some of the techniques to design a 1-D FIR filter in chapter one. In the following sections two approaches for determination of the transformation coefficients will be discussed.

3.4.1. FORMULATION OF THE PROBLEM USING NONLINEAR PROGRAMMING

The original McClellan transformation can be shown as follows:

$$\cos(\omega) = F(\omega_1, \omega_2) = t(0,0) + t(1,0)\cos(\omega_1) + t(0,1)\cos(\omega_2) + t(1,1)\cos(\omega_1)\cos(\omega_2) \tag{3-13}$$

As a working example, the design of a 2-D low-pass filter whose passband is in the shape of a circle is considered. The contour in the frequency plane to which the passband edge of the 1-D filter should map is then described by the relation

$$\omega_1^2 + \omega_2^2 = R^2 \tag{3-14}$$

where R is the passband edge frequency of the 1-D prototype filter. For the transformation of a 1-D low-pass filter to a 2-D low-pass filter, 1-D origin will be mapped to the 2-D origin. This gives the constraint equation

$$t(0,0) + t(1,0) + t(0,1) + t(1,1) = 1 \tag{3-15}$$

We are able to reduce the number of independent variables from 4 to 3 by employing equation (3-12). To find values for these free variables the following equation should be solved for ω_2 in terms of ω and ω_1 and the free mapping parameters.

$$\cos(\omega) = F(\omega_1, \omega_2) \tag{3-16}$$

We can solve equations (3-14) and (3-11) for ω_2

$$\omega_2 = \sqrt{R^2 - \omega_1^2} \tag{3-17}$$

$$\omega_2 = G(\omega, \omega_1, t) = \arccos\left[\frac{\cos(\omega) - t(0,0) - t(1,0)\cos(\omega_1)}{t(0,1) + t(1,1)\cos(\omega_1)}\right] \tag{3-18}$$

An error function at the cut-off frequency can be defined as:

$$E(\omega_1) = G(\omega_c, \omega_1, t) - \sqrt{R^2 - \omega_1^2} \tag{3-19}$$

where ω_c is the cut-off frequency of the prototype. The parameters, t, can then be chosen to minimize some function of $E(\omega_1)$ such as least square error. Since the error function is a nonlinear function of unknown parameters, nonlinear optimization routines must be used for the minimization.

3.4.2. FORMULATION OF THE PROBLEM USING LINEAR PROGRAMMING

If the mapping were exact, then as the circular contour was traversed the value of $F(\omega_1, \omega_2)$ would be constant. This would result for $\omega = \omega_c$ in

$$\cos(\omega_c) = t(0,0) + t(1,0)\cos(\omega_1) + t(0,1)\cos(\sqrt{R^2 - \omega_1^2})$$
$$+ [1 - t(0,0) - t(1,0) - t(1,0)]\cos(\omega_1)\cos(\sqrt{R^2 - \omega_1^2}) \tag{3-20}$$

If the mapping is not exact, the equality in equation (3-20) will only be approximated and the error function will be defined by

$$E_1(\omega_1) = \cos(\omega_c) - t(0,0) - t(1,0)\cos(\omega_1) - t(0,1)\cos(\sqrt{R^2-\omega_1^2})$$

$$-[1 - t(0,0) - t(1,0) - t(1,0)]\cos(\omega_1)\cos(\sqrt{R^2-\omega_1^2}) \tag{3-21}$$

This error function is now a linear function of t(0,0), t(0,1), t(1,0), and thus linear programming optimization routines can be used to minimize the function.

NOTE : As a practical matter, a transformation will only be useful if the coefficients of the mapping t(l,k) satisfy the relation

$$| t(0,0) + t(1,0)\cos(\omega_1) + t(0,1)\cos(\omega_2) + t(1,1)\cos(\omega_1)\cos(\omega_2) | \le 1 \tag{3-22}$$

The preceding constraint can be incorporated directly as part of its input specification in linear programming formulation. In nonlinear programming formulation, it is possible to design the transformation parameters without regard to this constraint and then perform a simple linear scaling on the final parameters so that (3-22) is satisfied. The contour defined by F(ω_1 , ω_2) will not be changed by using

$$F'(\omega_1, \omega_2) = c_1 F(\omega_1, \omega_2) - c_2 \tag{3-23}$$

as a transformation for any c_2 and nonzero c_1. By choosing

$$c_1 = \frac{2}{F_{max} - F_{min}}$$

$$c_2 = c_1 F_{max} - 1 \tag{3-24}$$

where F_{max} denotes the maximum value of $F(\omega_1, \omega_2)$ and F_{min} denotes its minimum value. The condition (3-22) is now satisfied by this scaling. The shape of the contours is unchanged, but the 1-D frequency ω which is associated with each contour is changed. In particular, if $F(\omega_1, \omega_2)$ was originally associated with the value ω, $F'(\omega_1, \omega_2)$ defined by (3-23) will be associated with the value ω' where

$$\omega' = \cos^{-1}(c_1 \cos(\omega) - c_2) \qquad (3\text{-}25)$$

3.5. DESIGN OF 2-D FIR FILTER WITH INTEGER COEFFICIENTS USING McCLELLAN TRANSFORMATION

Based on the mentioned design steps in previous sections, we should maintain the same procedure for our design. That means, firstly, we have to obtain the transformation parameter by satisfying the circular cut-off boundary, and secondly the coefficients for the 1-D prototype filter should be determined.

3.5.1. TRANSFORMATION PARAMETERS FOR THE CIRCULAR SYMMETRY

Based on the original McClellan transformation, the following value for transformation parameters for the circular cutoff boundary are suggested:

$$t(1,0) = t(0,1) = t(1,1) = -t(0,0) = 0.5 \qquad (3\text{-}26)$$

Since the multiplication by 1/2 can be performed by shifting operation, no multipliers are required for implementation of this step.

3.5.2. DESIGN OF 1-D PROTOTYPE FILTERS WITH INTEGER COEFFICIENTS

For the second step, a 1-D FIR filter with the frequency response formed in equation (3-10) has to be designed. Since the design procedure can entirely be fulfilled in 1-D domain, the following approaches which have been explained in chapter 2 can be utilized for this design process.

1) Discretization and reoptimization (type 1) combined with nonlinear programming

2) Discretization and reoptimization (type 2) combined with nonlinear programming

3) Mixed integer linear programming

3.6. EXAMPLES

EXAMPLES 1: Design a 2-D FIR low pass filter with the order 19x19 and the following specifications using three mentioned techniques.

$$H(e^{j\omega_1}, e^{j\omega_2}) = \begin{cases} 1 & 0 \le \sqrt{\omega_1^2+\omega_2^2} \le 1 \\ \\ 0 & 2 \le \sqrt{\omega_1^2+\omega_2^2} \le \pi \end{cases}$$

The specifications for the 1-D prototype are as follows:

$$H(e^{j\omega}) = \begin{cases} 1 & 0 \leq \omega \leq 1 \\ 0 & 2 \leq \omega \leq \pi \end{cases}$$

Tables (3-5),(3-6), and (3-7) show the integer coefficients for the three design techniques. Figure (3-3)-(3-8) show the magnitude responses of the design techniques for example 1

b(0) = 43	b(3) = -18	b(6) = -34	b(9) = 53
b(1) = 123	b(4) = -56	b(7) = 0	
b(2) = 72	b(5) = -93	b(8) = 39	

Table 3-5: Integer coefficients for example 1 using D & R (type 1)

b(0) = 44	b(3) = -19	b(6) = -35	b(9) = 54
b(1) = 124	b(4) = -57	b(7) = 0	
b(2) = 72	b(5) = -94	b(8) = 40	

Table 3-6: Integer coefficients for example 1 using D & R (type 2)

b(0) = 10	b(3) = -68	b(6) = 63	b(9) = -3
b(1) = 44	b(4) = -66	b(7) = -12	
b(2) = 31	b(5) = -55	b(8) = -22	

Table 3-7: Integer coefficients for example 1 using Mixed Integer Prog.

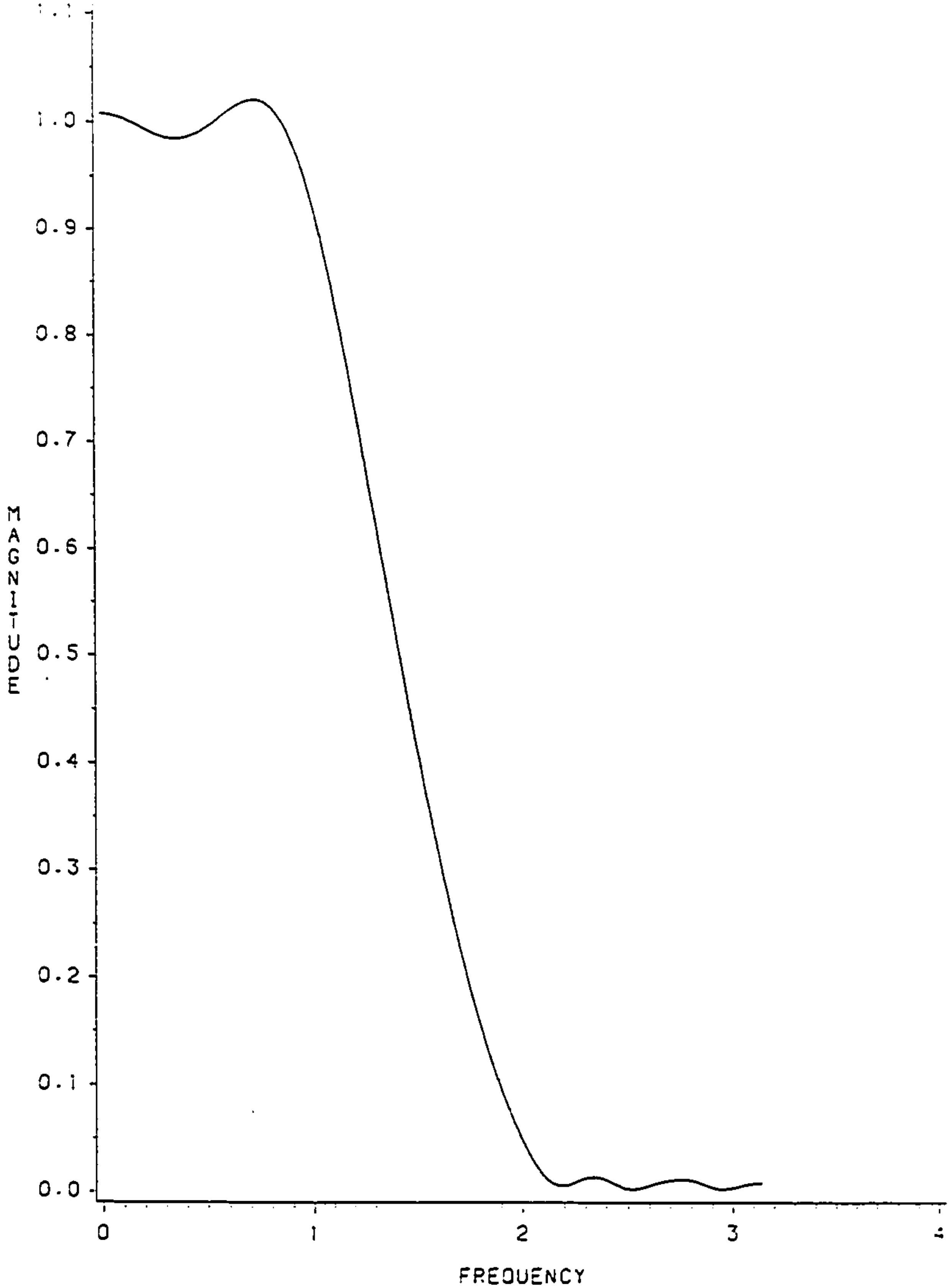


Figure (3-3): The magnitude response for 1-D prototype in example 1 using D&R (type1)

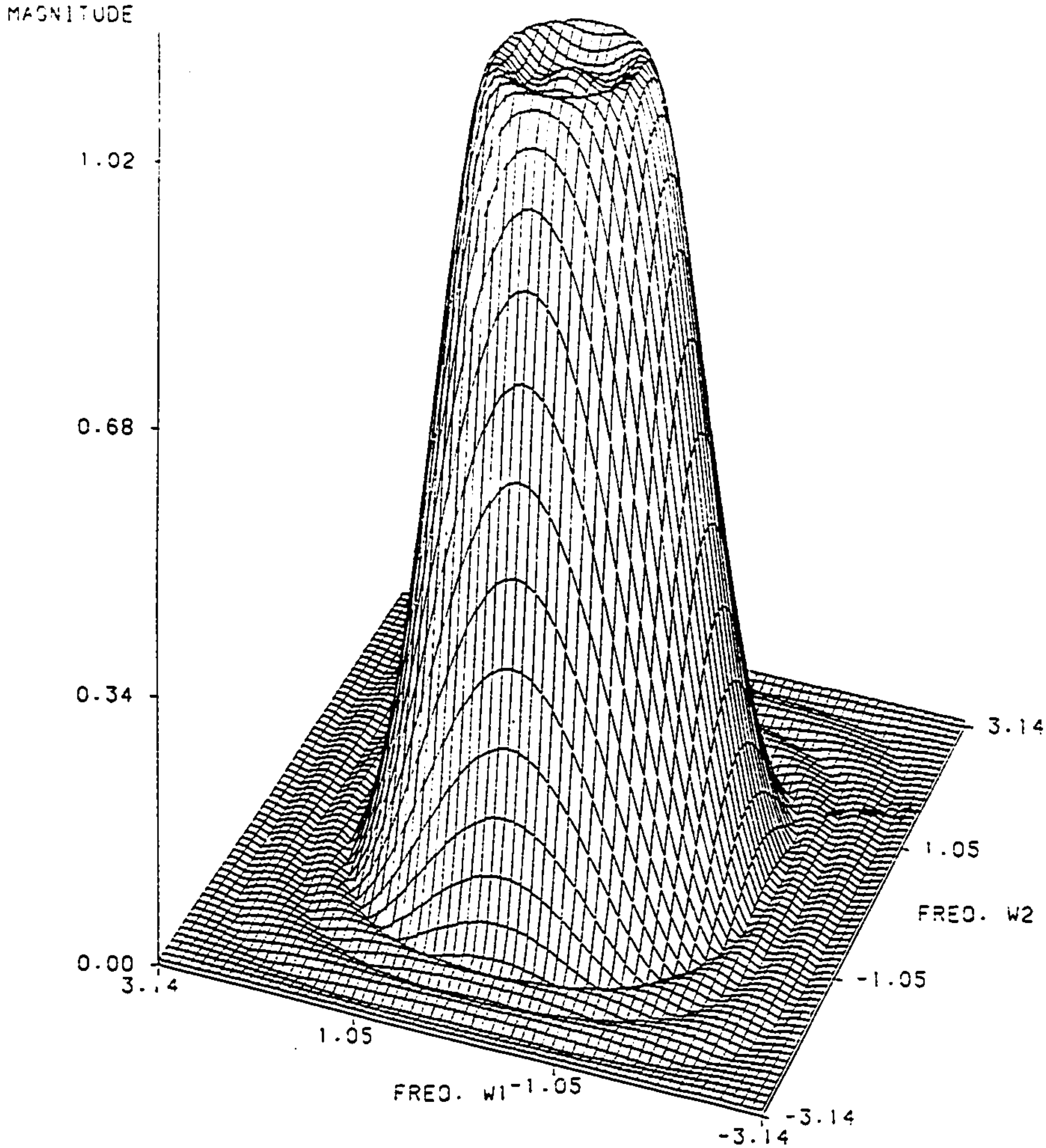


Figure (3-4): The magnitude response for 2-D FIR filter in example 1 using D&R (type1)

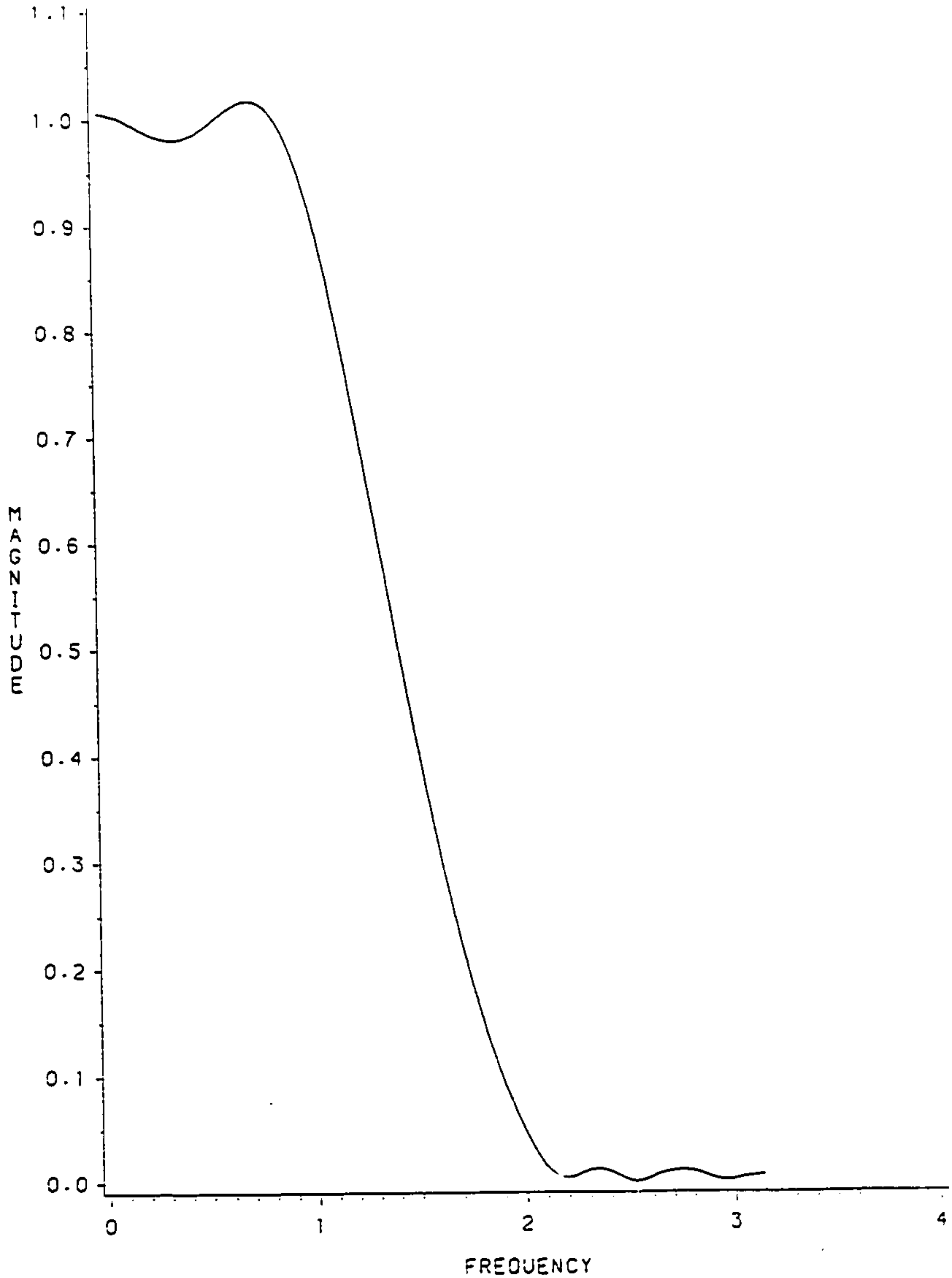


Figure (3-5): The magnitude response for 1-D prototype in example 1 using D&R (type2)

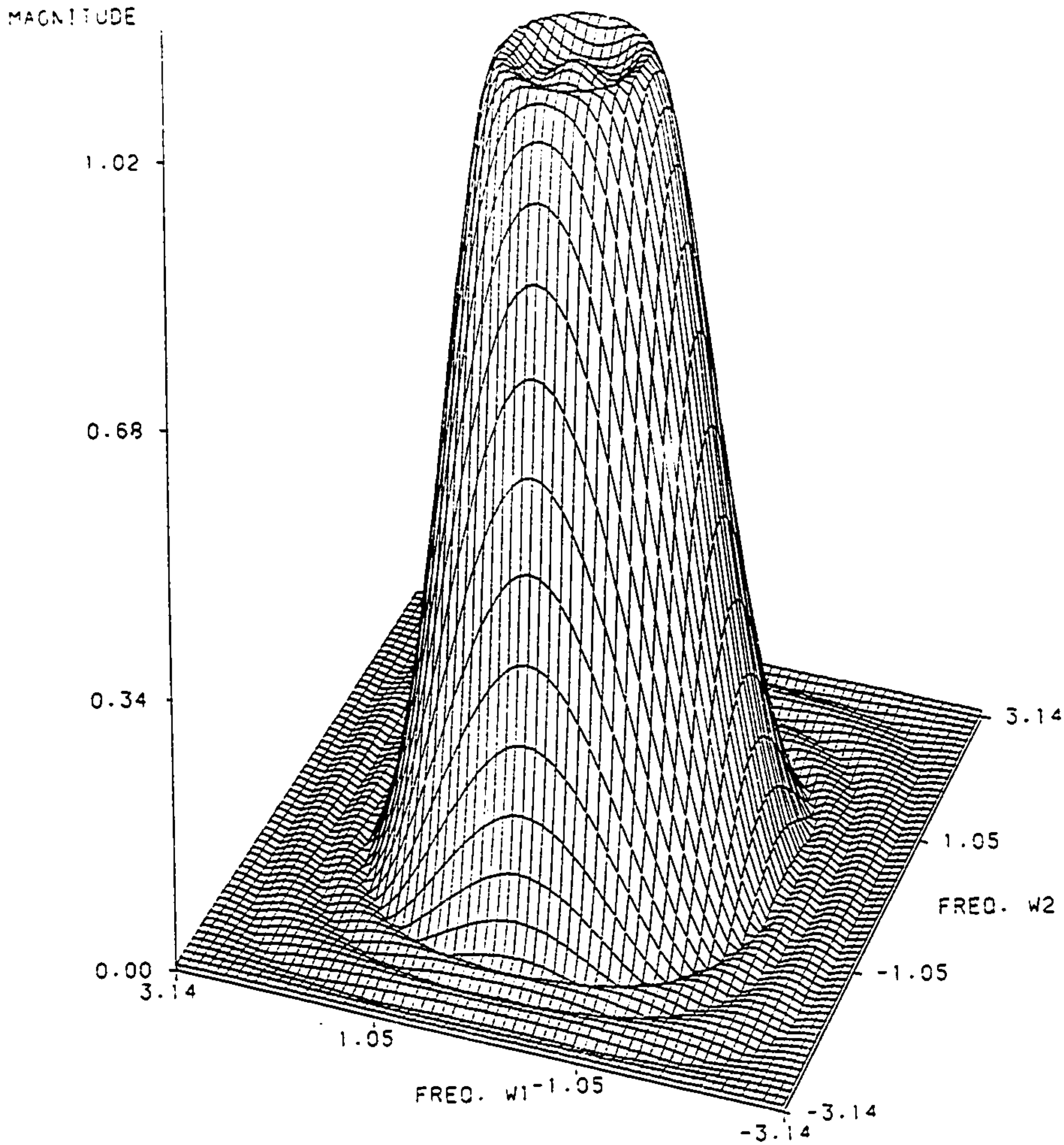


Figure (3-6): The magnitude response for 2-D FIR filter in example 1 using D&R (type2)

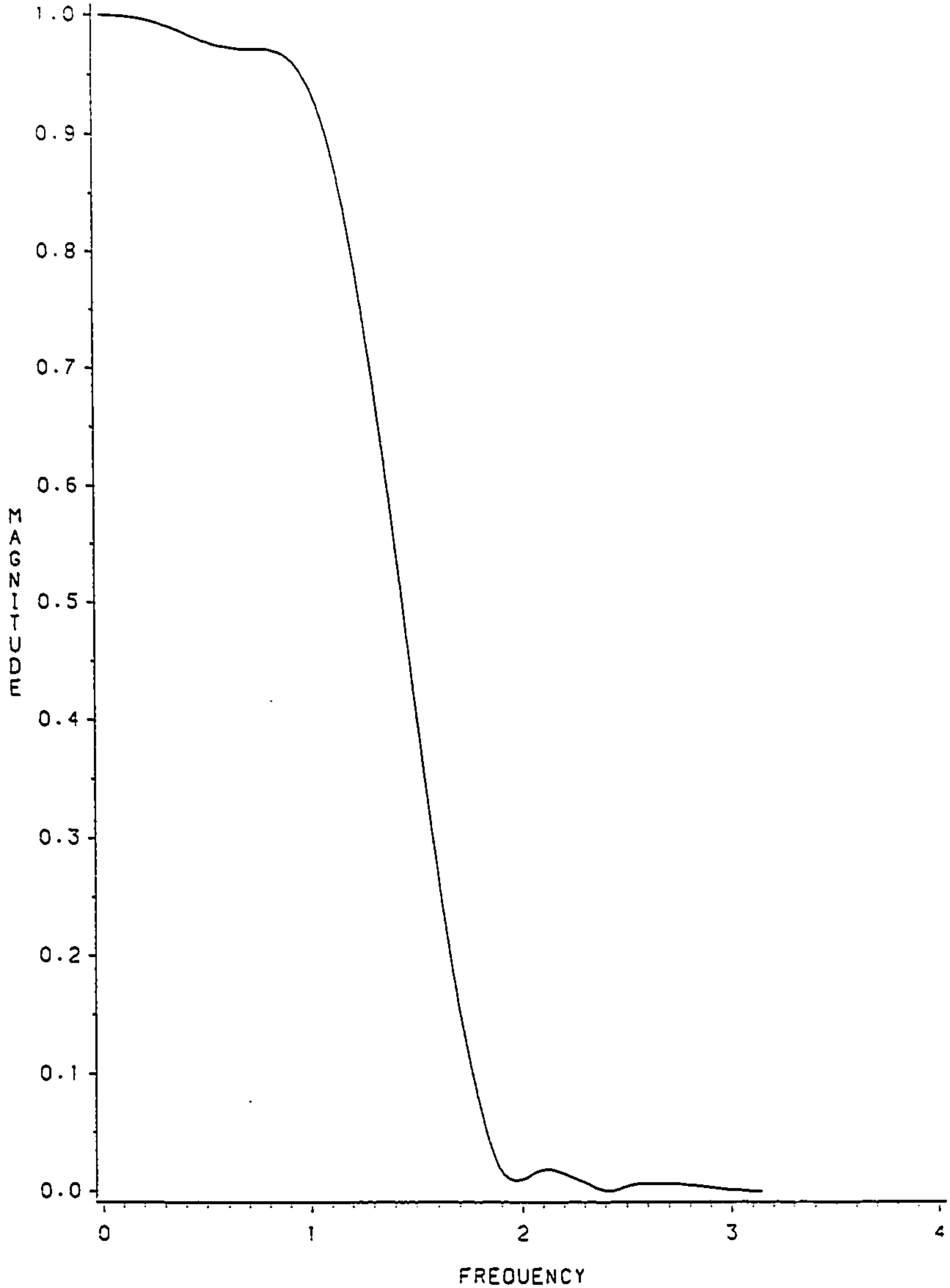


Figure (3-7): The magnitude response for 1-D prototype in example 1 using LINDO

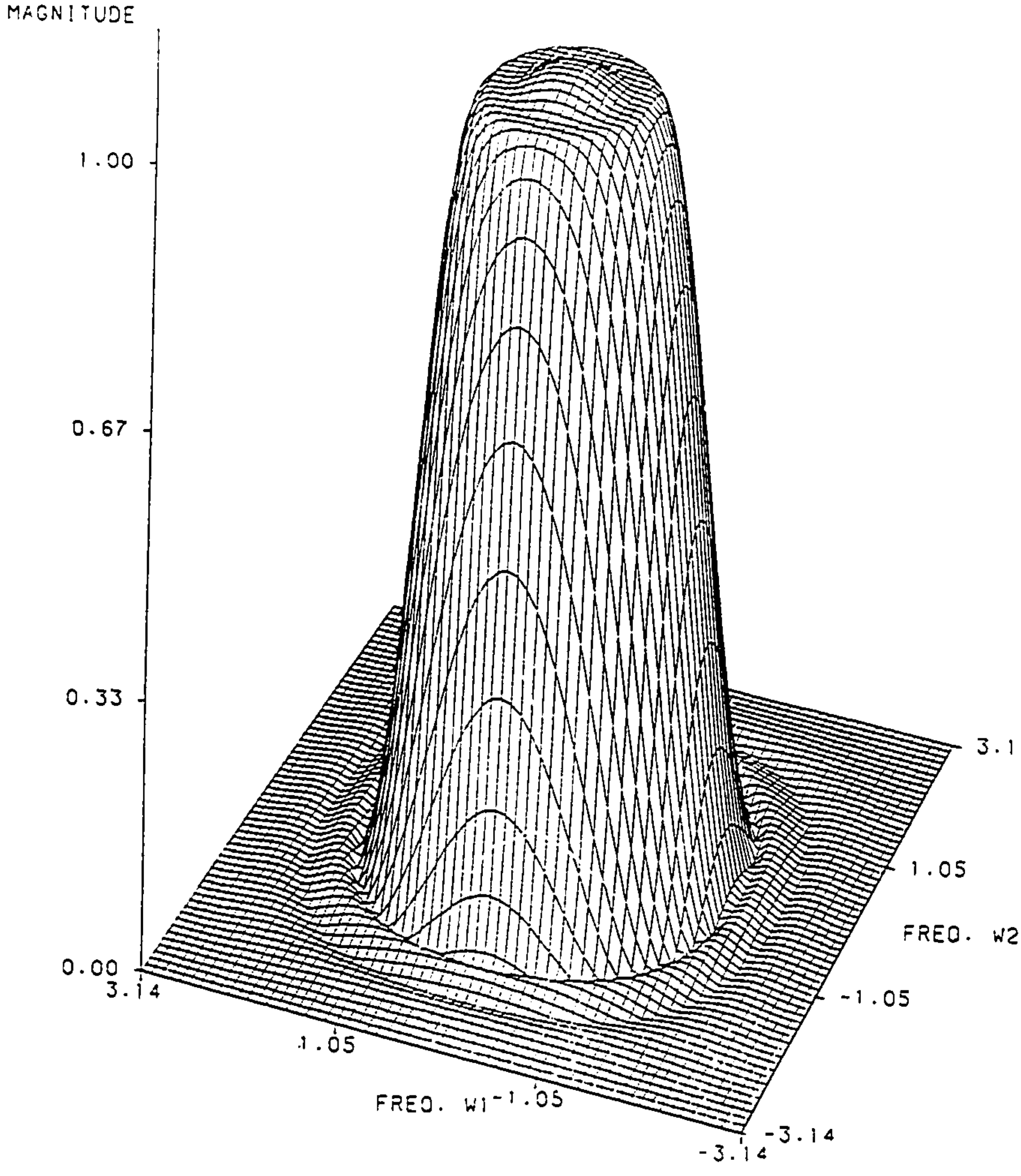


Figure (3-8): The magnitude response for 2-D FIR filter in example 1 using LINDO

Table 3-8 shows the characteristics of the three design techniques.

DESIGN TECHNIQUE	#R TIME (SEC.)	M.M.P.		M.M.S.
D & R (TYPE 1)	173	1-D	1.0170	.02604
		2-D	1.0171	.08082
D & R (TYPE 2)	420	1-D	1.0233	.02223
		2-D	1.0234	.08624
MIXED INTEGER (LP)	240	1-D	1.0001	.01861
		2-D	1.0011	.00688

Table 3-8: The characteristics of the design techniques for example 1

EXAMPLES 2: Design a 2-D FIR highpass filter with the order 19x19 and the following specifications using three mentioned techniques:

$$H(e^{j\omega_1}, e^{j\omega_2}) = \begin{cases} 0 & 0 \le \sqrt{\omega_1^2 + \omega_2^2} \le 1.5 \\ 1 & 2 \le \sqrt{\omega_1^2 + \omega_2^2} \le \pi \end{cases}$$

The specifications for the 1-D prototype are as follows:

$$H(e^{j\omega}) = \begin{cases} 1 & 2 \leq \omega \leq \pi \\ 0 & 0 \leq \omega \leq 1.5 \end{cases}$$

Tables (3-9), (3-10), and (3-11) show the integer coefficients for the three design techniques.

b(0) = 11	b(3) = 60	b(6) = -22	b(9) = -11
b(1) = -71	b(4) = -119	b(7) = 1	
b(2) = 103	b(5) = -12	b(8) = 61	

Table 3-9: Integer coefficients for example 2 using D & R (type 1)

b(0) = 11	b(3) = 61	b(6) = -23	b(9) = -12
b(1) = -72	b(4) = -120	b(7) = 2	
b(2) = 103	b(5) = -12	b(8) = 62	

Table 3-10: Integer coefficients for example 2 using D & R (type 2)

b(0) = 1	b(3) = 0	b(6) = 65	b(9) = 26
b(1) = -10	b(4) = -62	b(7) = -61	
b(2) = 24	b(5) = 41	b(8) = -24	

Table 3-11: Integer coefficients for example 2 using Mixed integer prog.

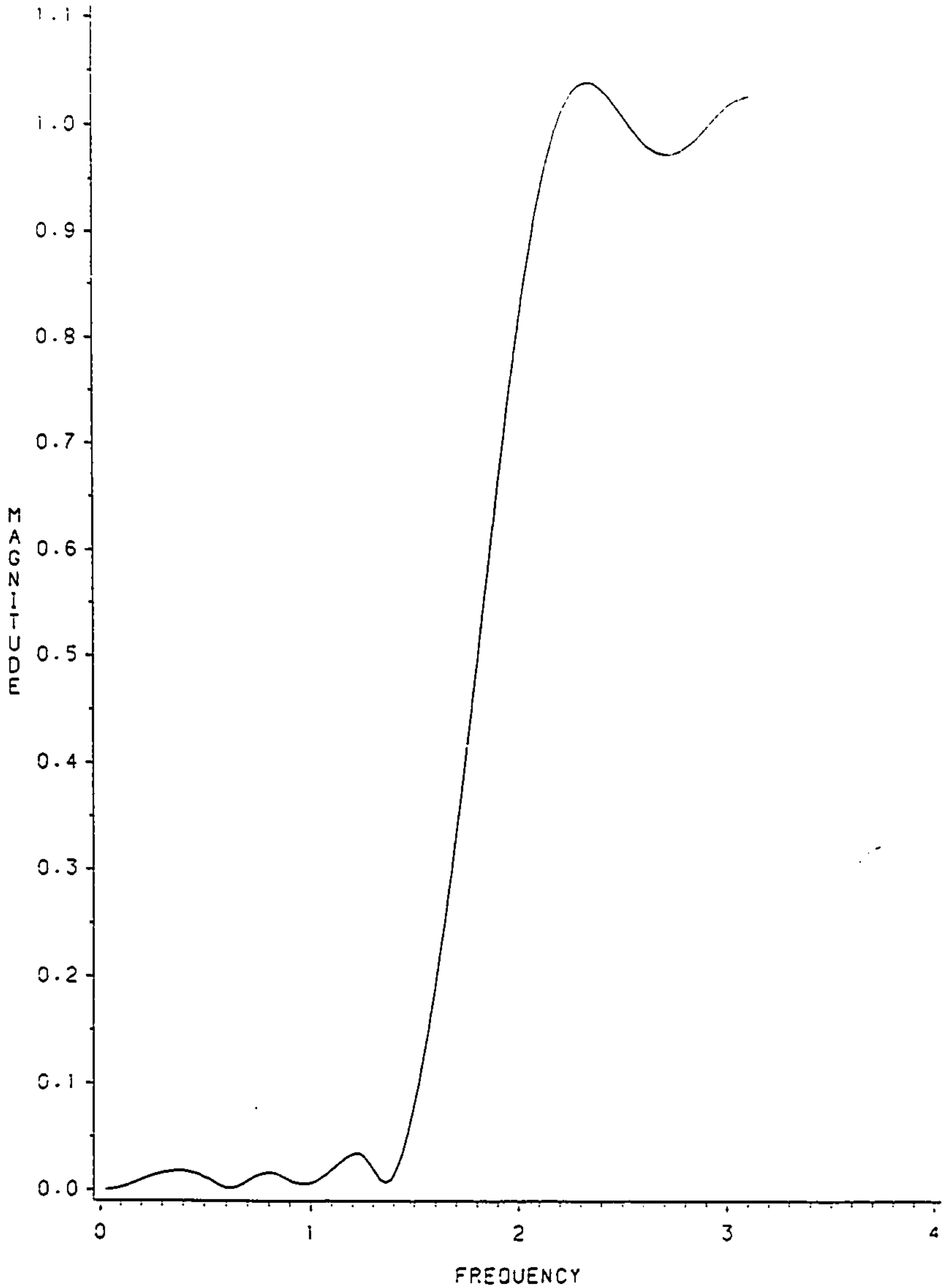


Figure (3-9): The magnitude response for 1-D prototype in example 2 using D&R (type1)

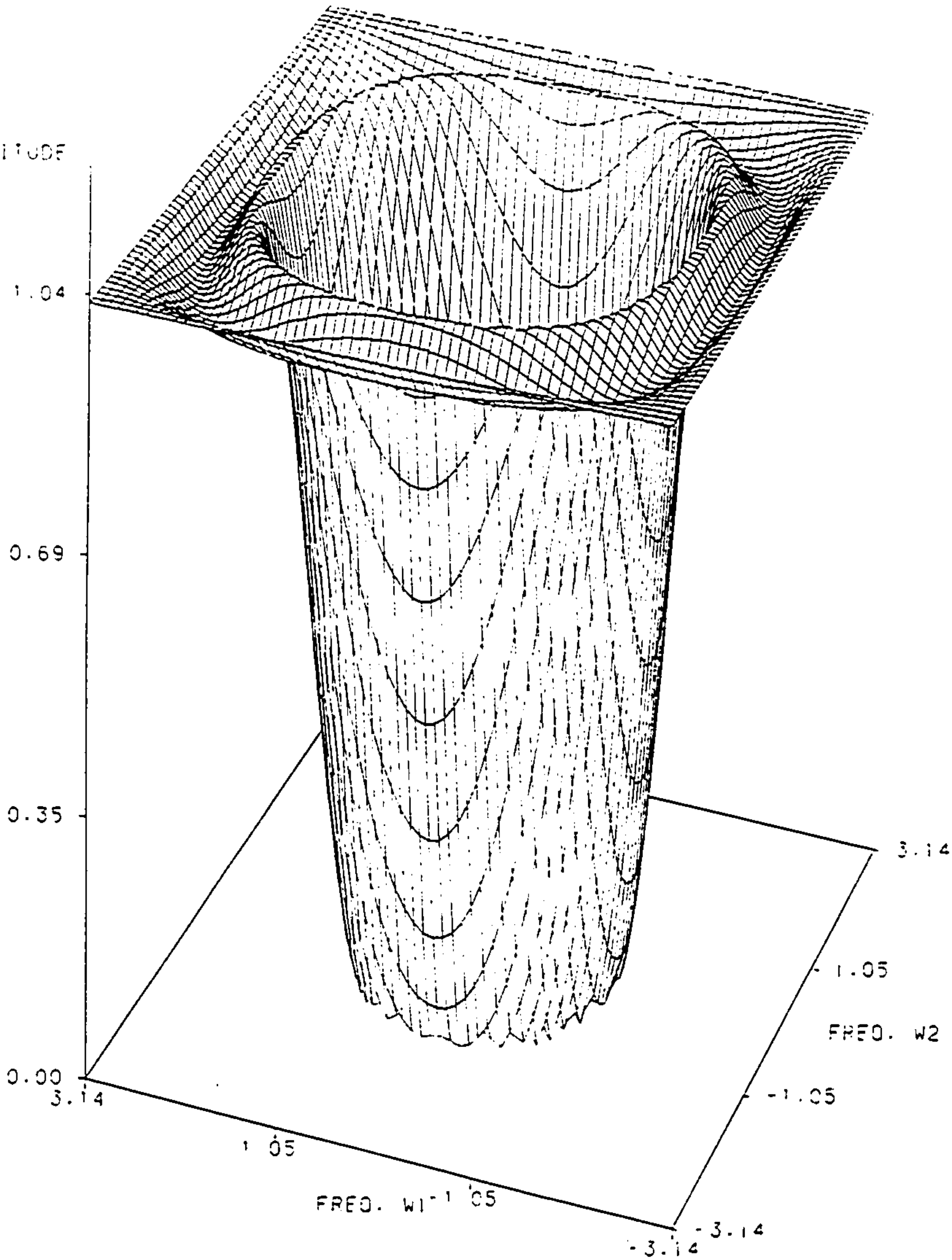


Figure (3-10): The magnitude response for 2-D FIR filter in example 2 using D&R (type1)

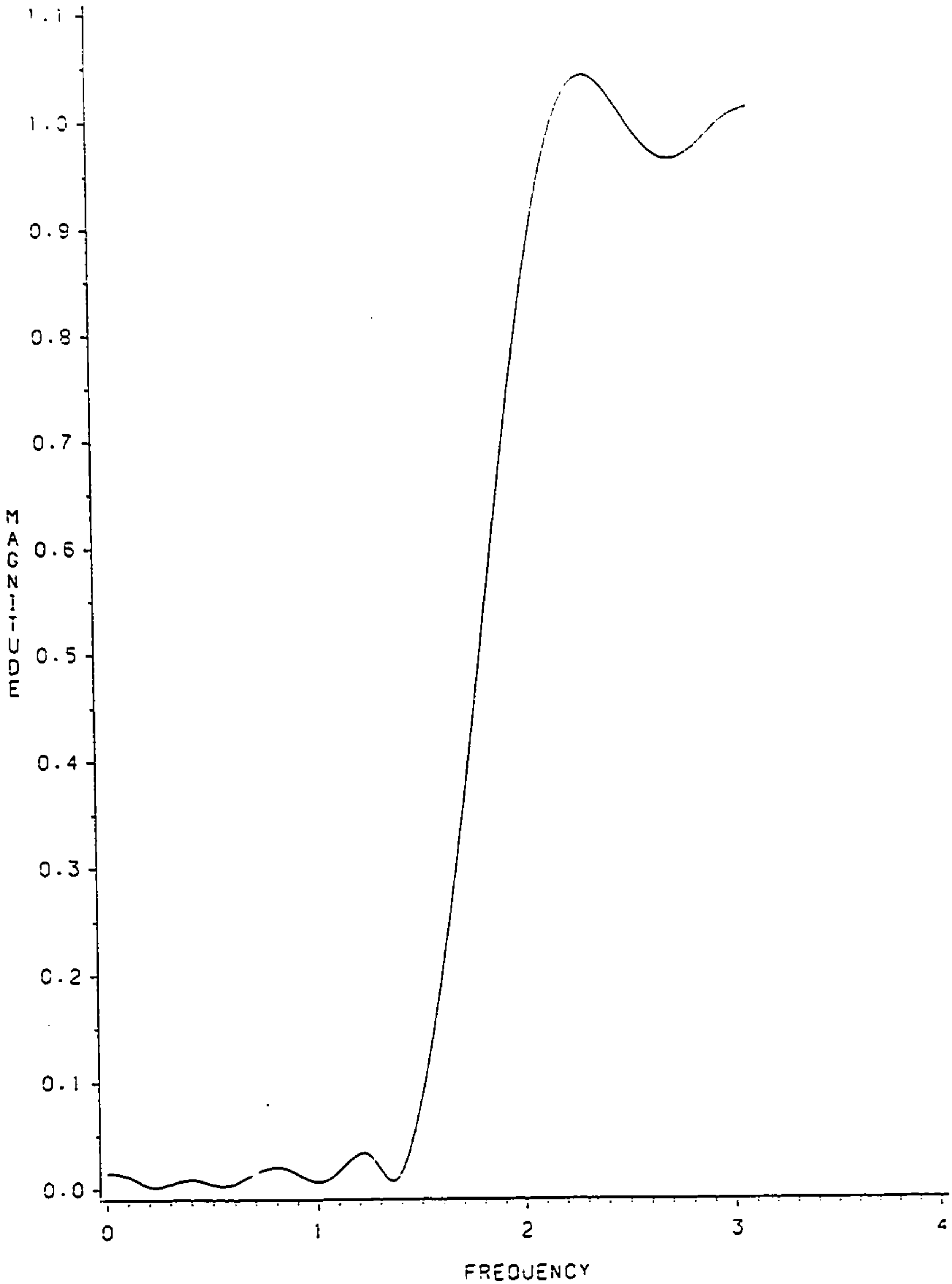


Figure (3-11): The magnitude response for 1-D prototype in example 2 using D&R (type2)

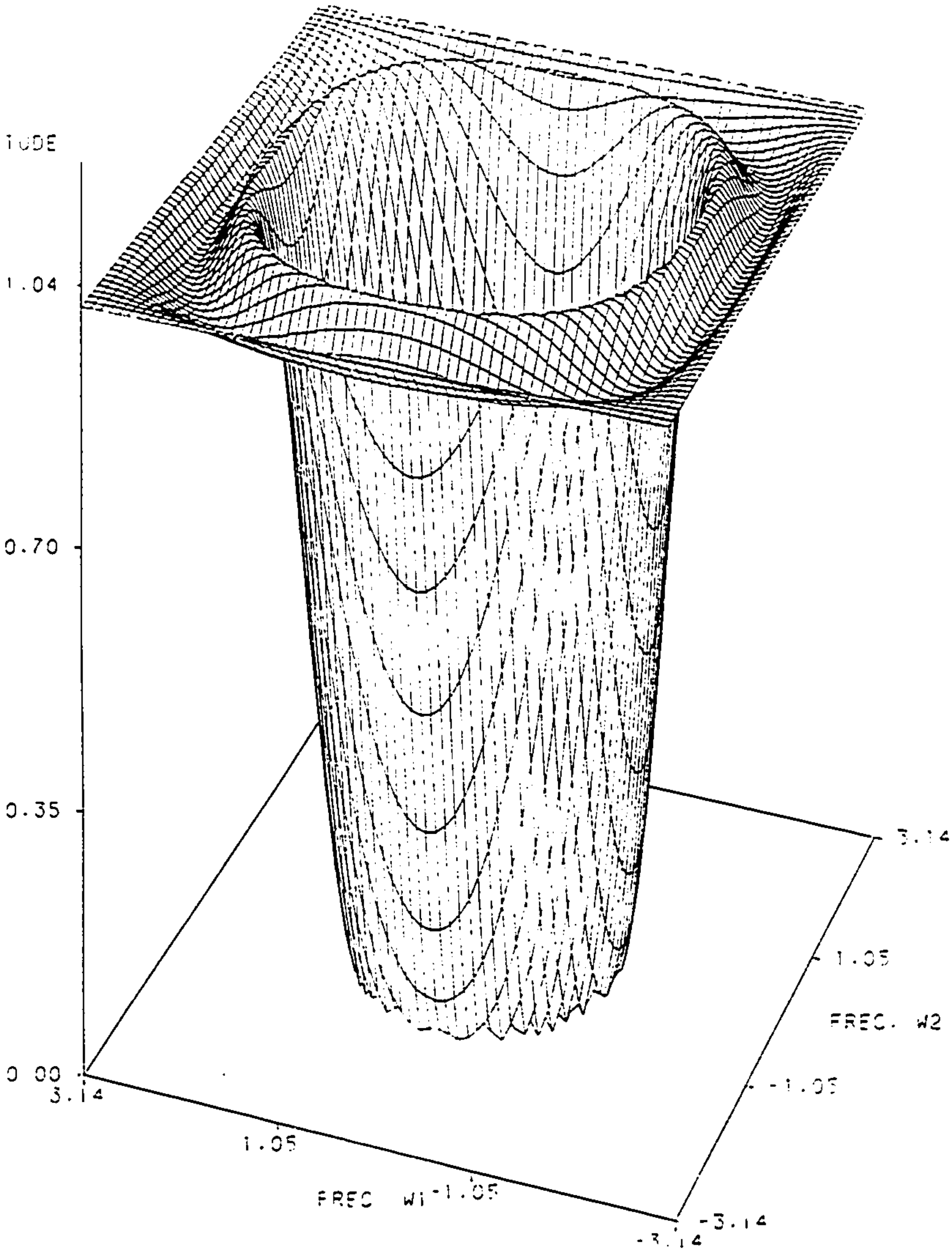


Figure (3-12): The magnitude response for 2-D FIR filter in example 2 using D&R (type2)

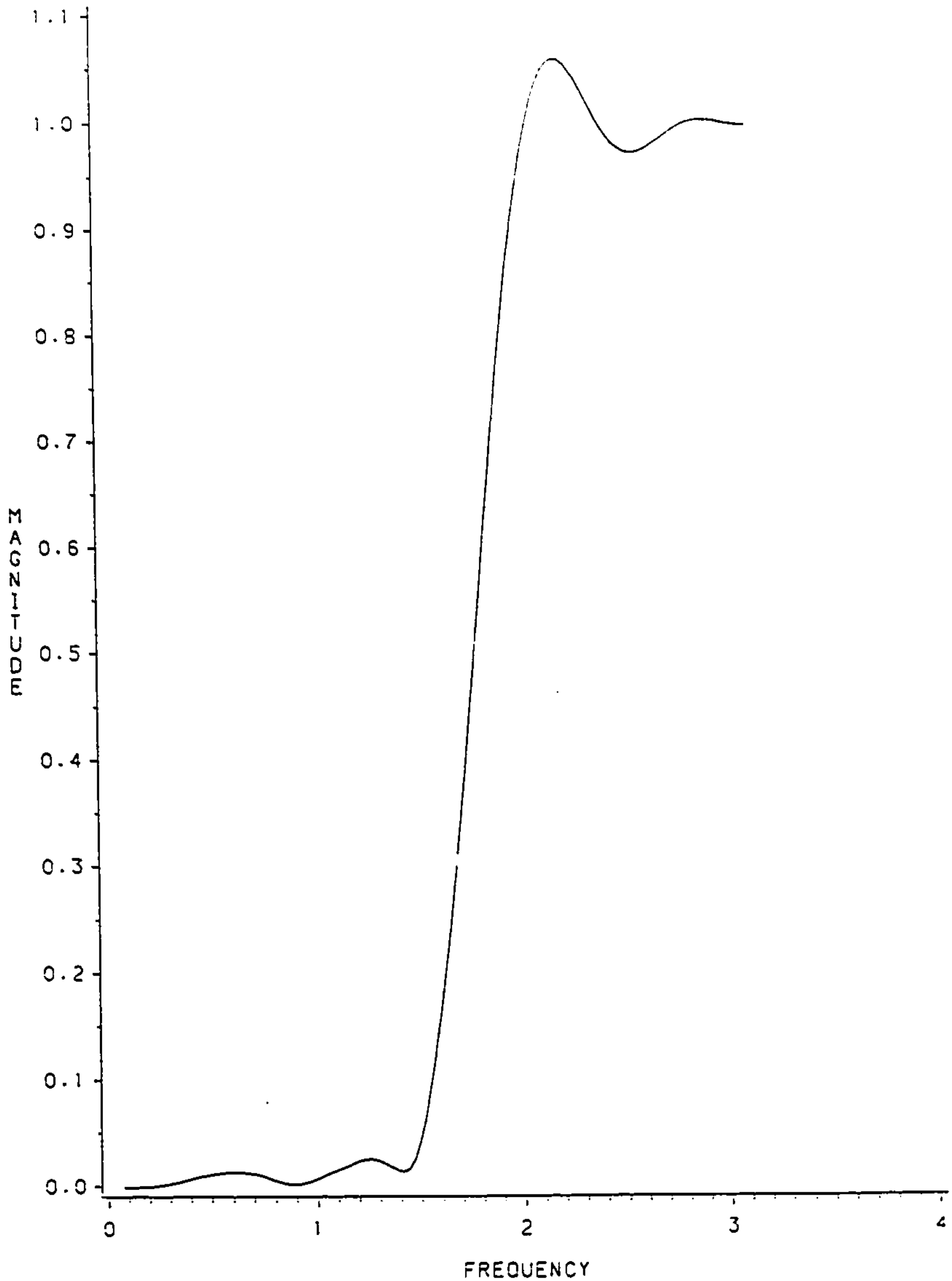


Figure (3-13): The magnitude response for 1-D prototype in example 2 using LINDO

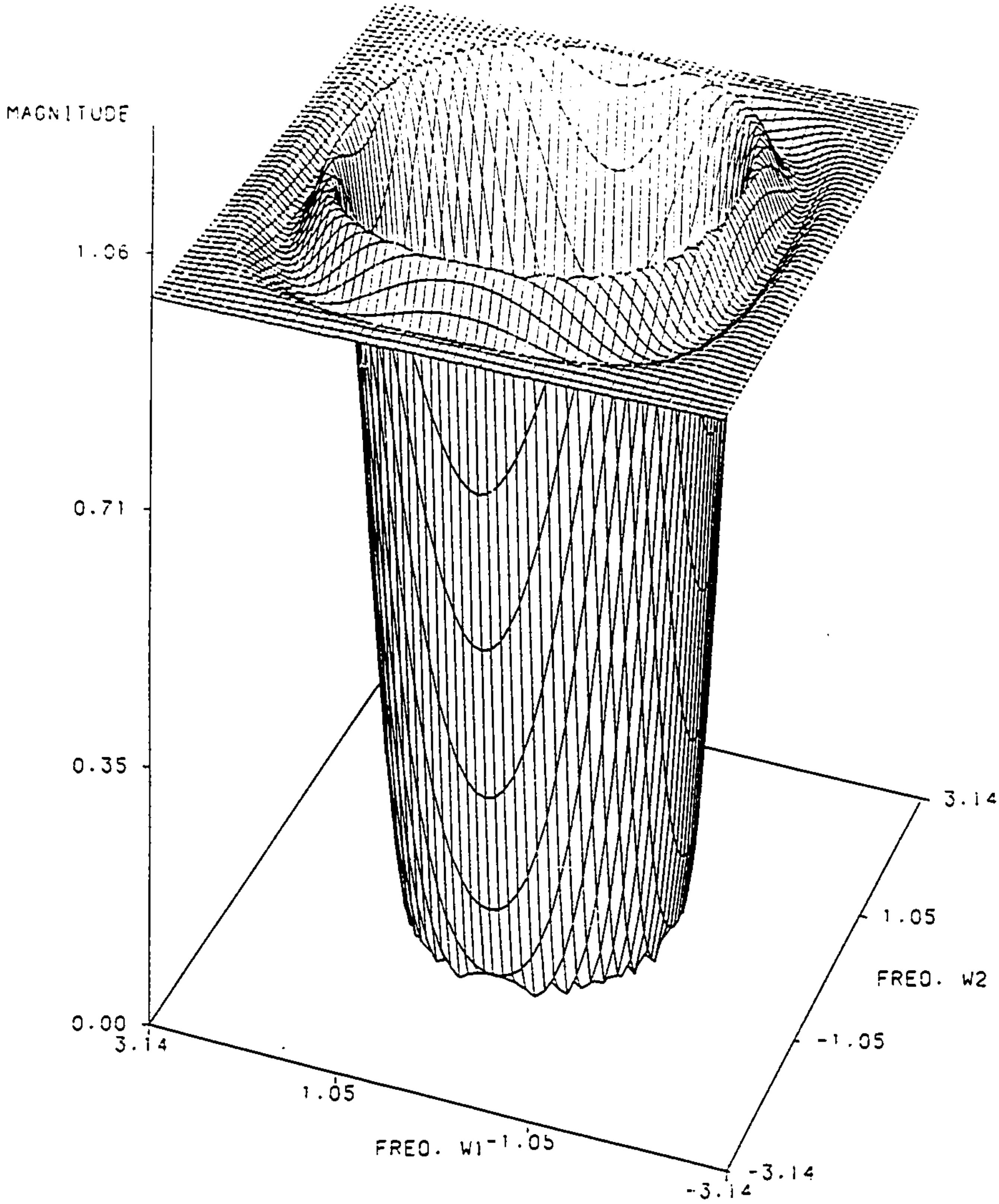


Figure (3-14): The magnitude response for 2-D FIR filter in example 1 using LINDO

Table 3-12 shows the characteristics of the three design techniques. Figures (3-9) to (3-14) present the magnitude responses of the 2-D and prototype 1-D filter.

DESIGN TECHNIQUE	#R TIME (SEC.)	M.M.P.		M.M.S.
D & R (TYPE 1)	100	1-D	1.0468	.07434
		2-D	1.0469	.09593
D & R (TYPE 2)	200	1-D	1.0312	.07234
		2-D	1.0312	.09472
MIXED INTEGER (LP)	720	1-D	1.0614	.02626
		2-D	1.0616	.04588

Table 3-12: The characteristics of the design techniques for example 2

EXAMPLES 3: Design a 2-D FIR band-pass filter with the order 39x39 and the following specifications using Discretization & Reoptimization (type 1&2).

$$H(e^{j\omega_1}, e^{j\omega_2}) = \begin{cases} 0 & 0 \leq \sqrt{\omega_1^2+\omega_2^2} \leq 0.3 \quad , \quad 1.3 \leq \sqrt{\omega_1^2+\omega_2^2} \leq \pi \\ 1 & 0.7 \leq \sqrt{\omega_1^2+\omega_2^2} \leq 0.9 \end{cases}$$

The specifications for the 1-D prototype are as follows:

$$
H(e^{j\omega}) = \begin{cases} 1 & 0.7 \leq \omega \leq 0.9 \\ 0 & 0 \leq \omega \leq 0.3 \quad , \quad 1.3 \leq \omega \leq \pi \end{cases}
$$

Tables (3-13), and (3-14) show the integer coefficients for the two design techniques.

b(0) = -1	b(6) = -38	b(12) = -1	b(18) = 31
b(1) = 0	b(7) = -66	b(13) = 1	b(19) = 12
b(2) = 42	b(8) = -38	b(14) = 15	
b(3) = 92	b(9) = -19	b(15) = 14	
b(4) = 12	b(10) = -9	b(16) = -12	
b(5) = -54	b(11) = 19	b(17) = 2	

Table 3-13: Integer coefficients for the example 3 using D & R (type 1)

b(0) = -1	b(6) = -39	b(12) = -1	b(18) = 32
b(1) = 0	b(7) = -67	b(13) = 2	b(19) = 12
b(2) = 43	b(8) = -38	b(14) = 15	
b(3) = 92	b(9) = -20	b(15) = 14	
b(4) = 13	b(10) = -10	b(16) = -13	
b(5) = -54	b(11) = 19	b(17) = 3	

Table 3-14: Integer coefficients for the example 3 using D & R (type 2)

Table 3-15 shows the characteristics of the three design techniques. Figure (3-15)-(3-18)

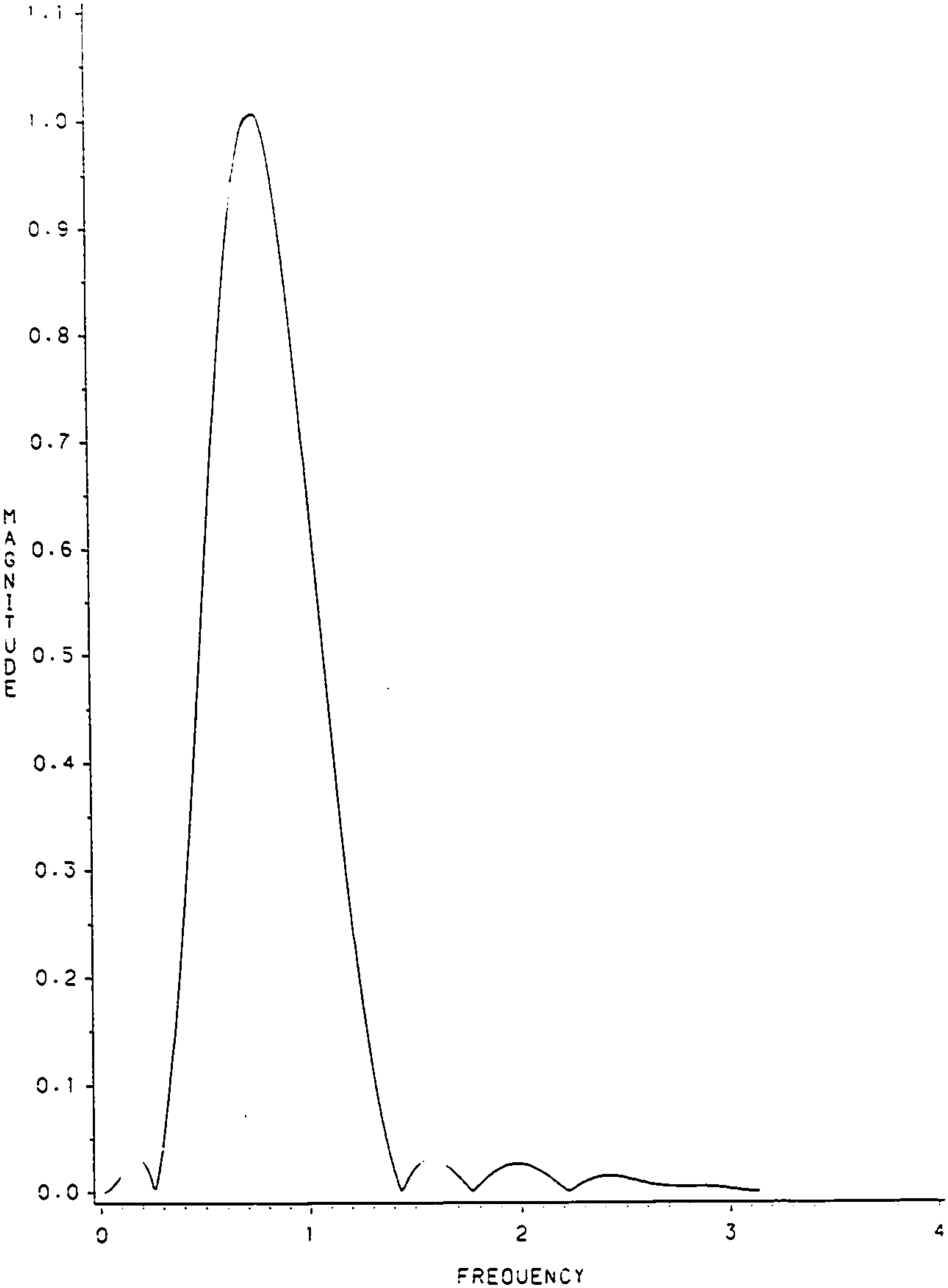


Figure (3-15): The magnitude response for 1-D prototype in example 3 using D&R (type1)

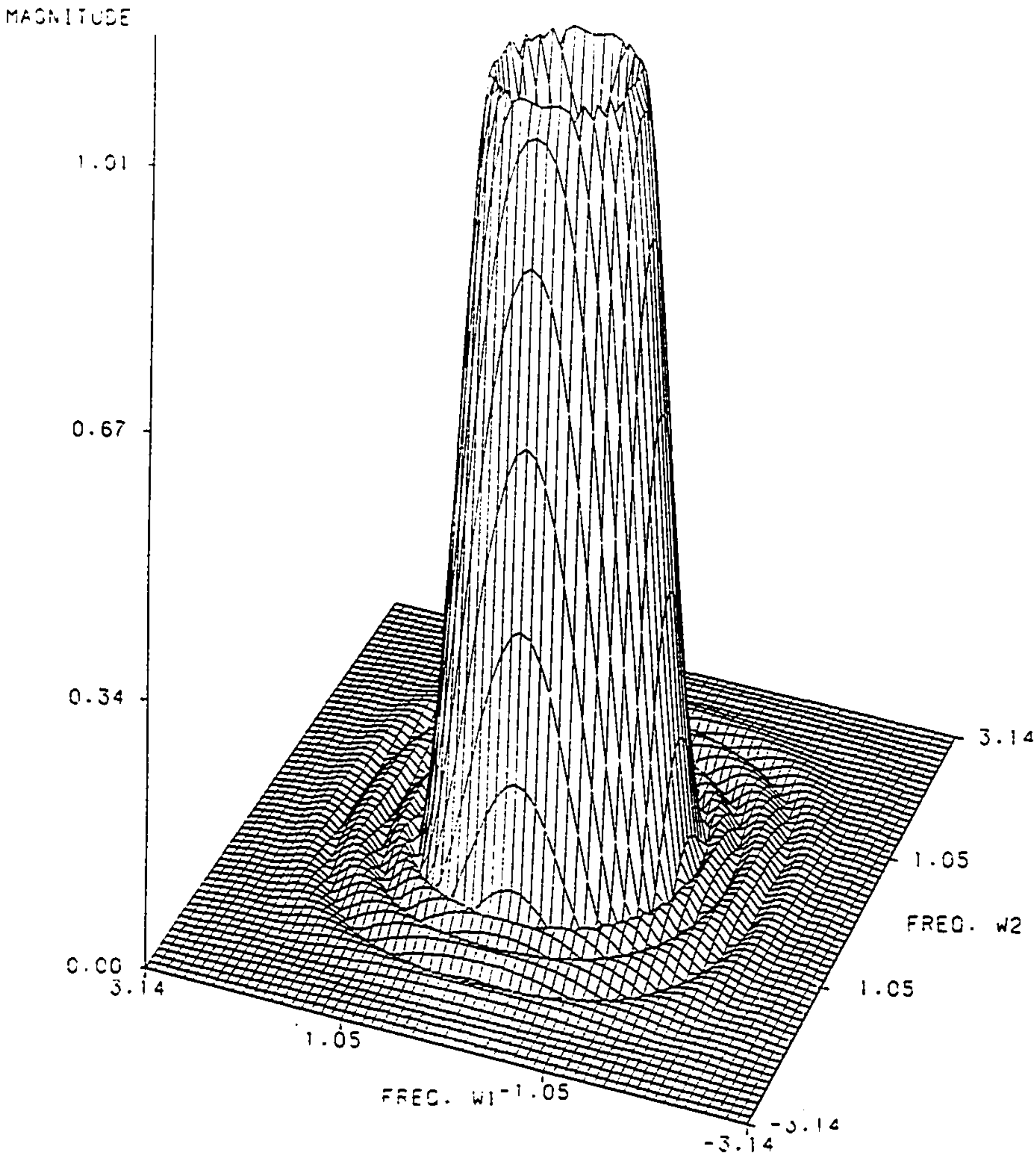


Figure (3-16): The magnitude response for 2-D FIR filter in example 3 using D&R (type1)

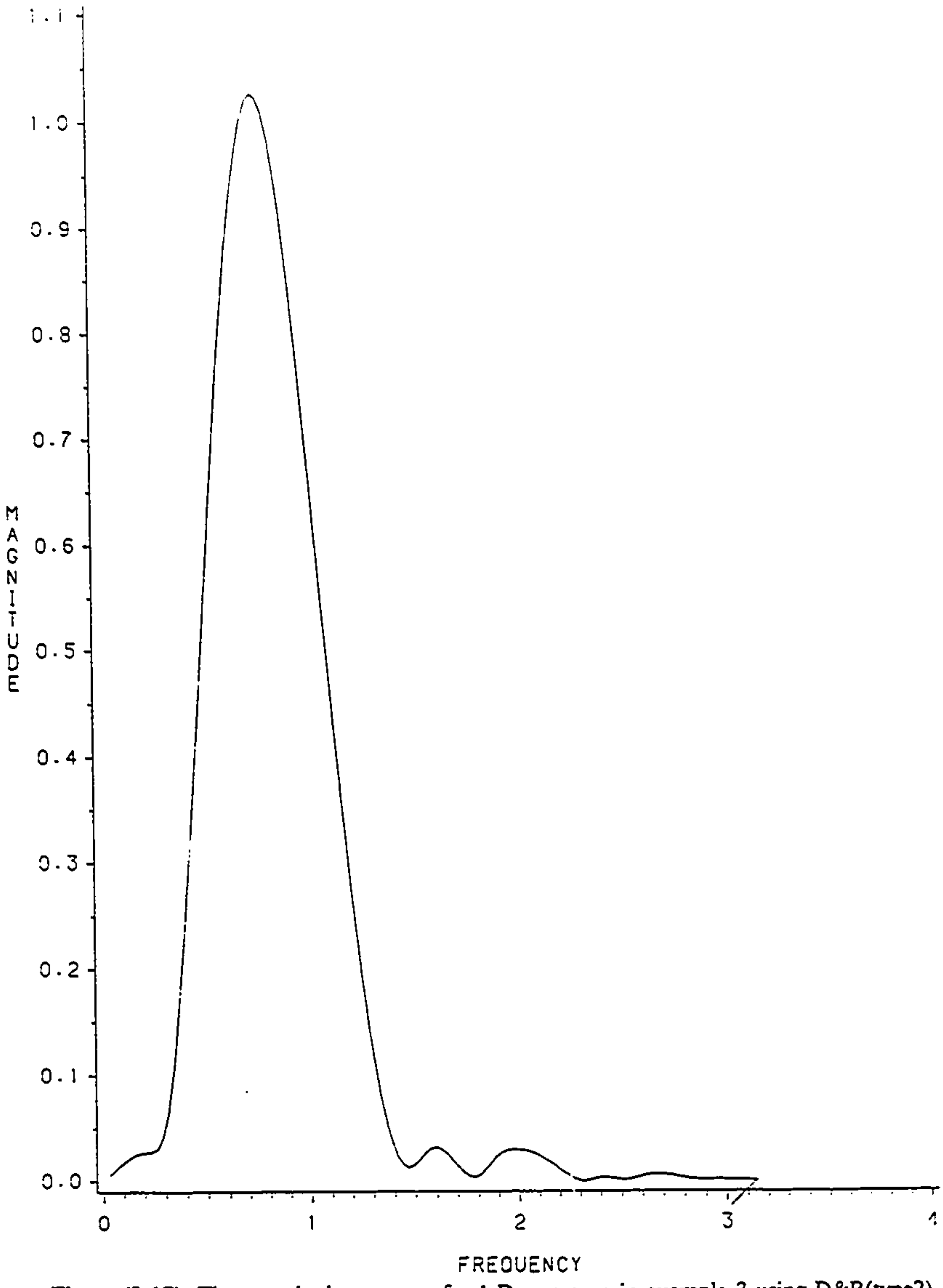


Figure (3-17): The magnitude response for 1-D prototype in example 3 using D&R(type2)

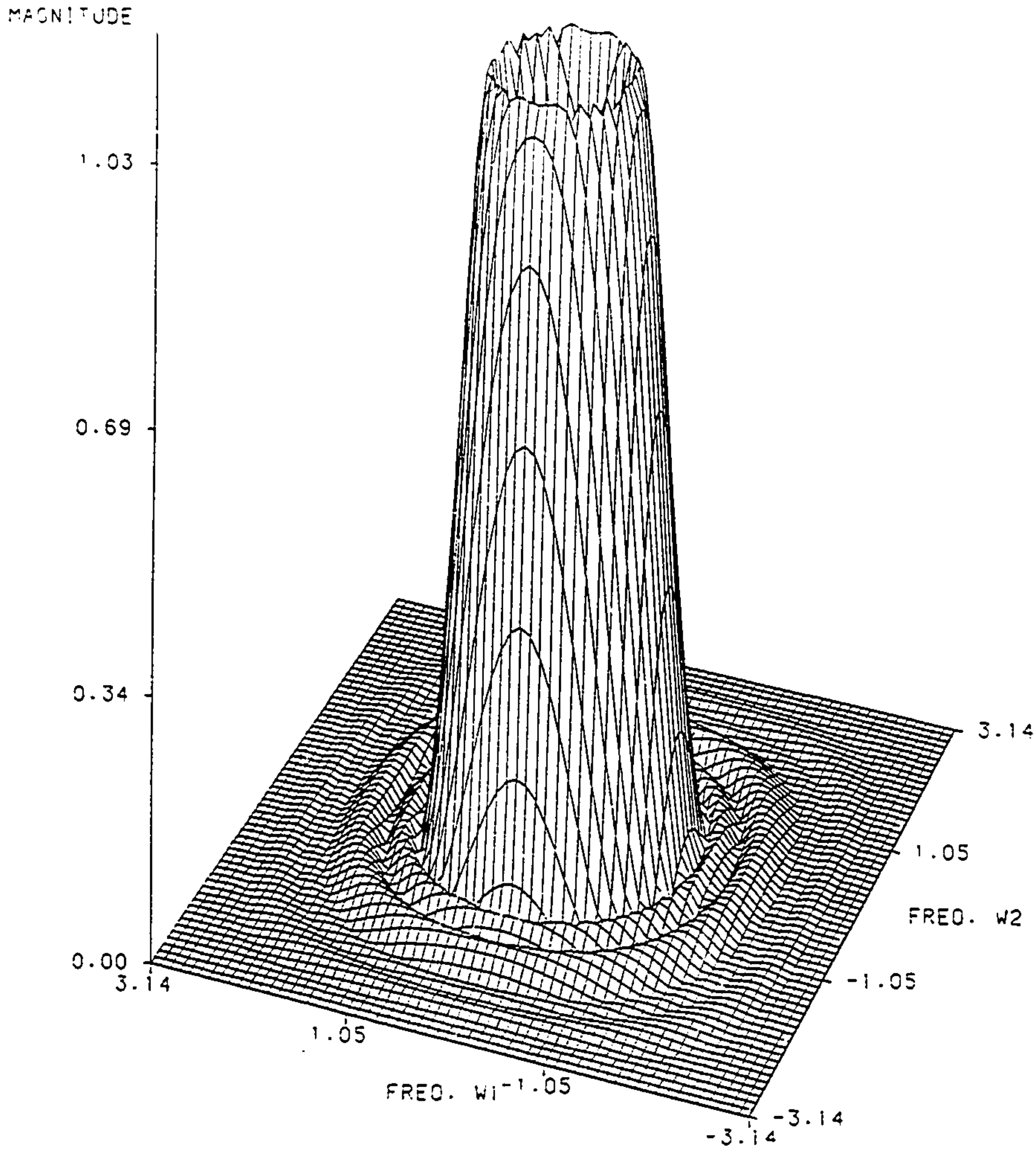


Figure (3-18):The magnitude response for 2-D FIR filter in example 3 using D&R (type2)

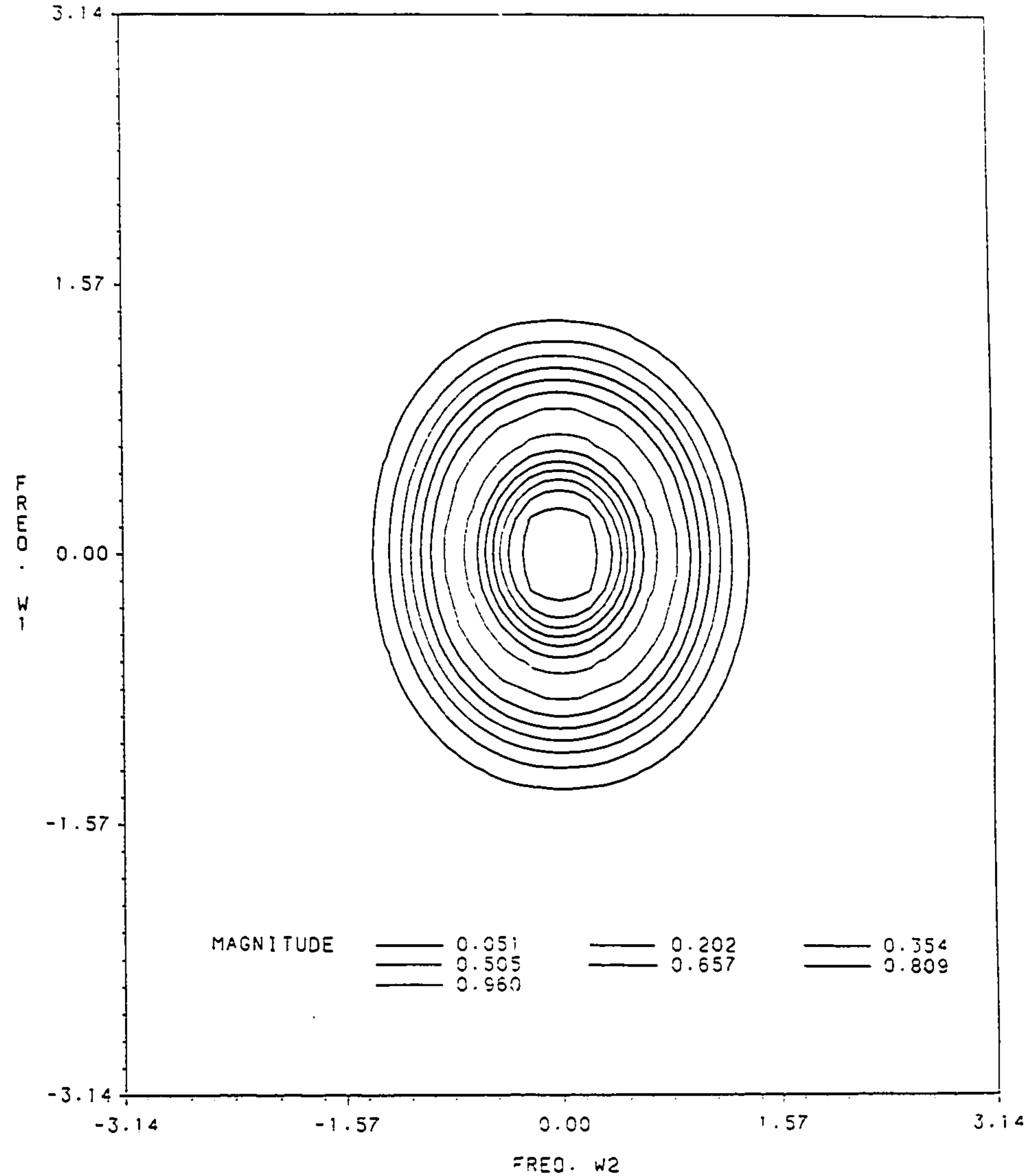


Figure (3-19): The contour plot of magnitude response of example 3 using D&R(type1)

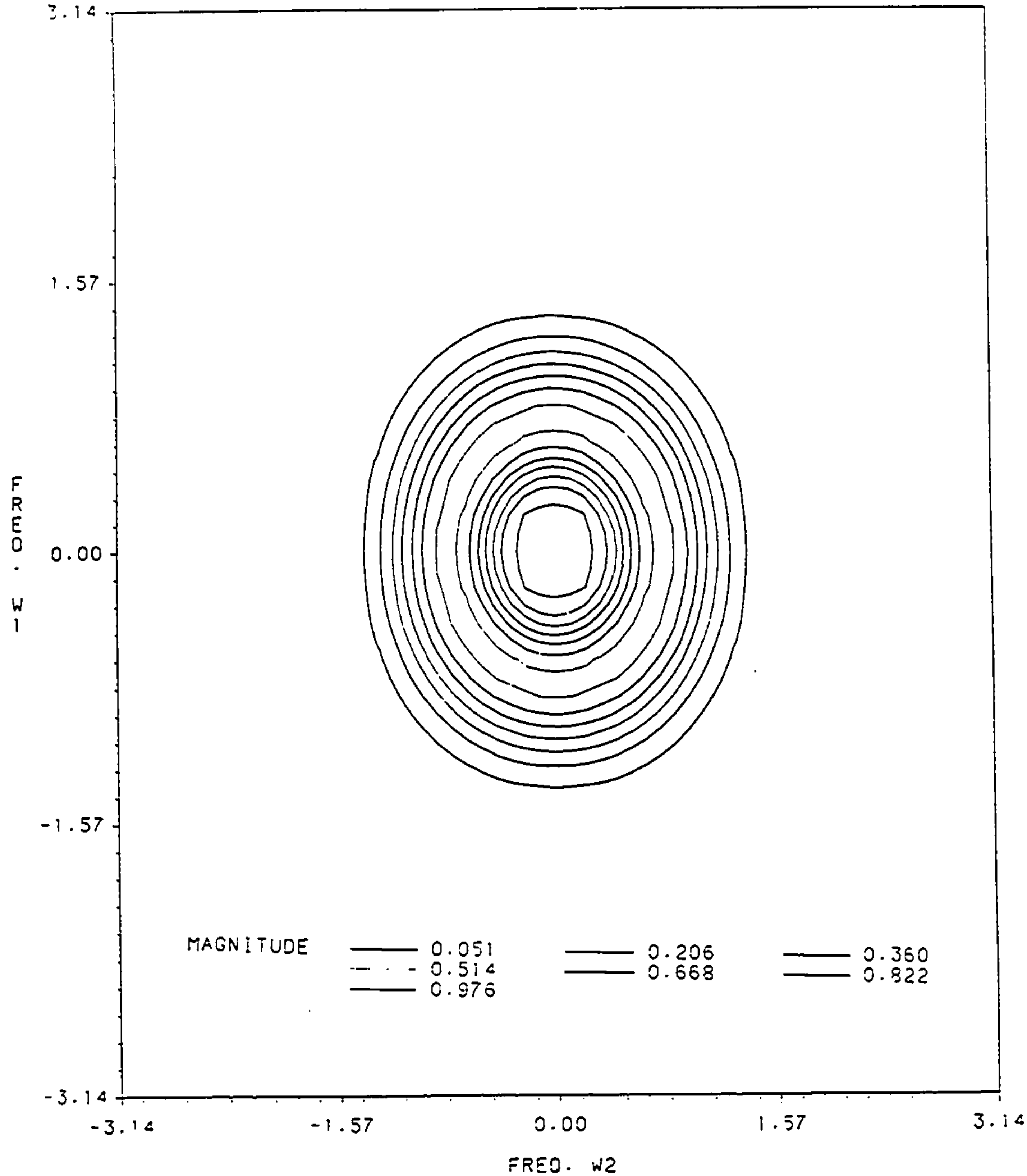


Figure (3-20): The contour plot of magnitude response of example 3 using D&R(type2)

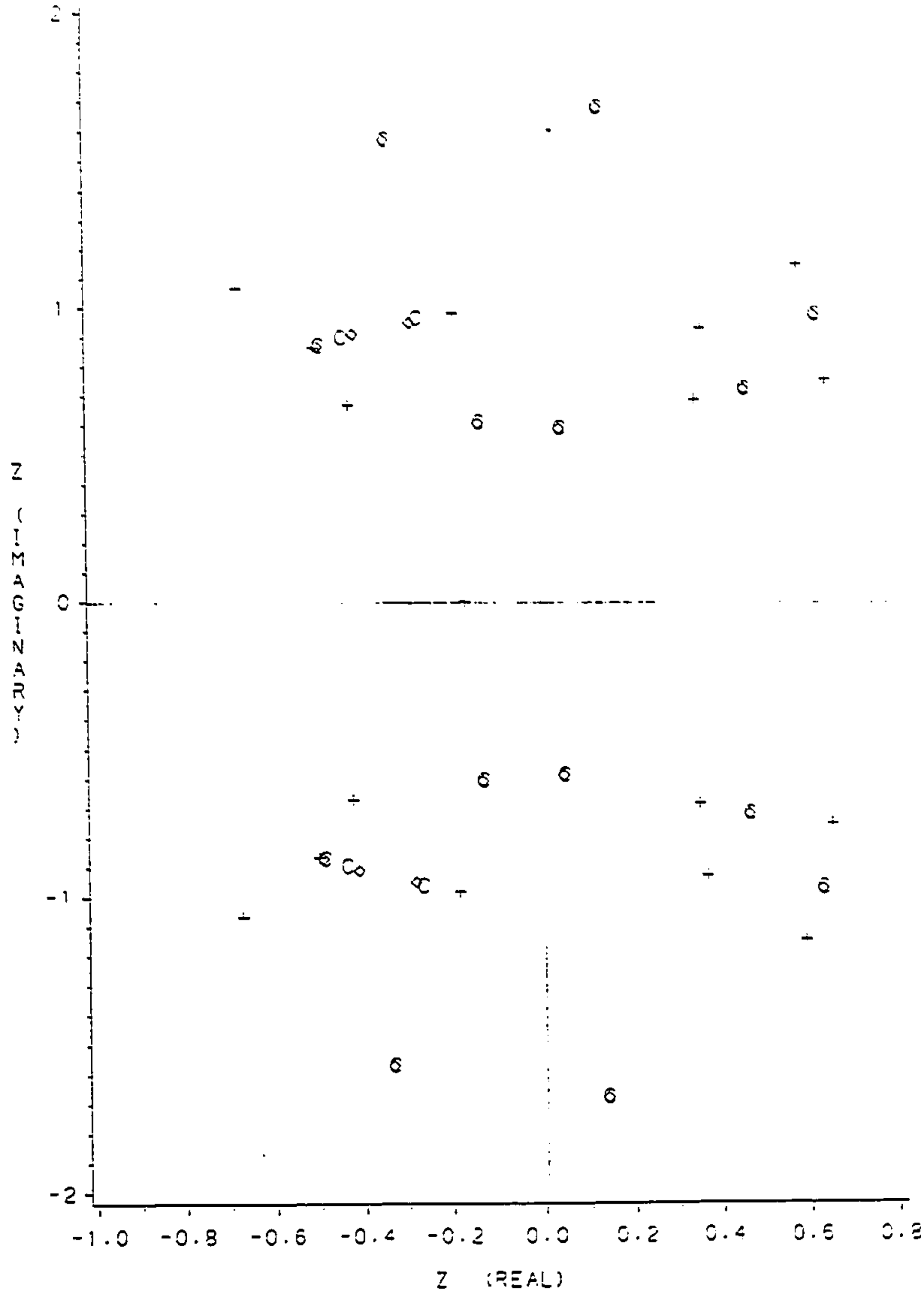


Figure (3-21) : The location of zeros for example 1, circle=type 1, Diamond=type 2, plus= lindo display the magnitude responses for example 3 using these techniques. Figures (3-17) and (3-19) and (3-20) display the contour plot of the magnitude response of this example using these two techniques. For further investigation, the position of zeros can be found by solving the transfer function of the filter. To achieve this an Nth order polynomial has to be solved. Figure 3-19 shows the positions of these zeros for example 1 using mentioned techniques.

DESIGN TECHNIQUE	#R TIME (MIN.)	M.M.P.		M.M.S.
D & R (TYPE 1)	10	1-D	1.0192	.06254
		2-D	1.0230	.12303
D & R (TYPE 2)	20	1-D	1.0337	.05264
		2-D	1.0377	.12551

Table 3-15: The characteristic of design techniques in example 3

3.7. SUMMARY

In this chapter we have investigated the design of a 2-D FIR filter with integer coefficients. To achieve our goal, some of the techniques in the literature have been investigated. The direct extension of Discretization & Reoptimization (type 1&2) has been examined for 2-D FIR filter with octagonal symmetry. Although the number of coefficients and function evaluation is substantially reduced, the memory requirement and computation burden are two troublesome points in this design.

As an alternative approach the McClellan transformation has been considered. In this technique a 1-D FIR filter is transformed to a 2-D FIR filter by means of the McClellan transformation. This design technique splits the process into two steps, firstly the transformation parameters are obtained by linear or nonlinear programming formulation. secondly the coefficients of the 1-D prototype filter are determined by any technique in 1-D FIR filter design. We showed that by choosing the proper values for transformation parameters no multipliers are required in this part. To determine the integer coefficients for the 1-D prototype filter Discretization & Reoptimization (type 1&2) and Mixed integer linear programming have been used.

Several examples demonstrate the versatility of this design technique. The magnitude responses and the zero mapping for example 1 have been included in this chapter.

CHAPTER FOUR

REALIZATION STRUCTURE FOR 1 AND 2-D FIR FILTERS

4.1. INTRODUCTION

One of the steps in designing a filter is a realization of that filter. The realization is a process which converts the transfer function into a filter network. There are various structures for the realization of any FIR filter. To choose a best possible structure for the desired filter specifications the computational complexity, and the required memory of that structure have to be carefully considered, and also the effect of finite-word-length in the response of the structure has to be carefully evaluated.

In this chapter, some of the realization structures for 1 and 2-D FIR filter will be reviewed. A specific realization is required to realize the technique presented in chapter 3 to design a 2-D FIR filter with integer coefficients. By considering this fact, the direct form structure for realization of the design technique will be implemented.

4.1.1. REALIZATION FORM FOR 1-D FIR FILTER

In this section three methods for implementing an FIR system will be briefly reviewed. The first technique is the simplest structure called direct form. A second structure is the cascade form realization. The third structure is the frequency sampling realization. Finally, a lattice structure will be reviewed.

a) Direct form

The direct form realization follows from the nonrecursive difference equation or equivalently by the convolution summation.

$$y(n)=\sum_{i=0}^{M-1} h(i)x(n-i) \tag{4-1}$$

This structure requires M-1 memory location, and has the complexity of M multiplication and M-1 addition per output point. Figure (4-1) shows the direct form structure for FIR filter. By considering required symmetry in impulse response of the system for linear phase condition, Figure (4-2) shows the direct form realization of linear-phase FIR system. This structure is usually avoided for high order filters because of its high coefficients sensitivity. It also exhibits very low A-D round-off noise.

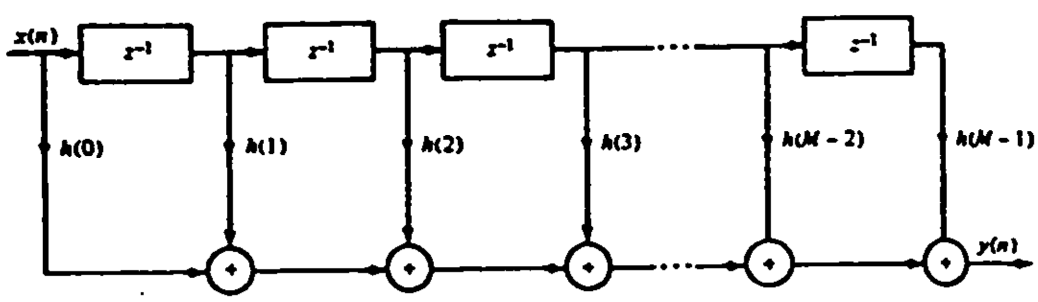


Figure (4-1) : Direct form structure for FIR filters

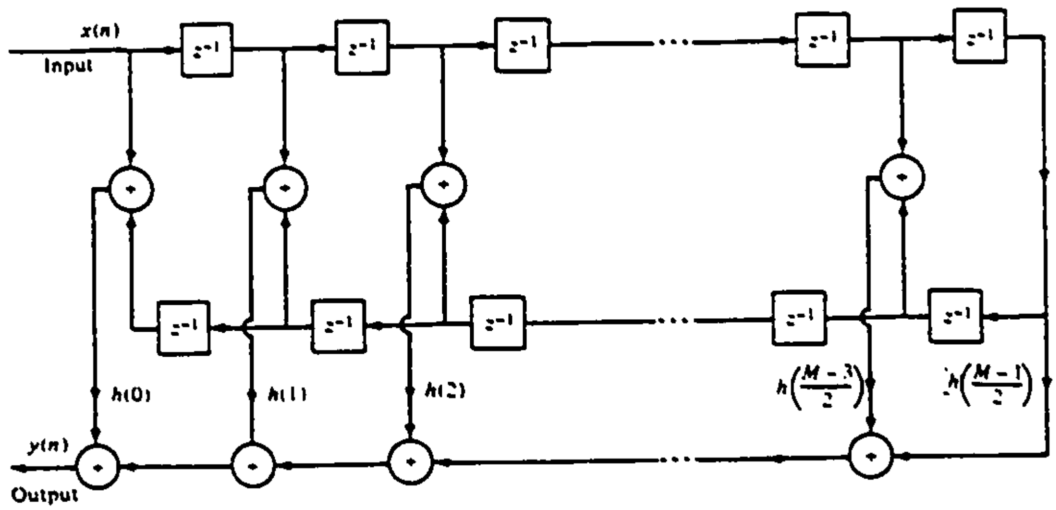


Figure (4-2): Direct form realization for linear-phase FIR filters

b) Cascade-form realization

The cascade realization follows by representing transfer function in terms of multiplication of a group of second-order FIR system. It can be shown that

$$H(z) = G\prod_{k=1}^{K} H_k(z) \qquad (4\text{-}2)$$

where

$$H_k(z) = 1 + b_{k1}z^{-1} + b_{k2}z^{-2} \qquad where\, k=1,2,\cdots,K \qquad (4\text{-}3)$$

and K is the integer part of$(M+1)/2$. The gain parameter G may be equally distributed among the K filter sections, such that $G=G_1, G_2, \ldots, G_K$. Figure (4-3) shows the cascade form realization along with the basic second order section. By imposing the symmetry condition for linear-phase FIR filter, the second order factors will be as follows:

$$1+az^{-1}+z^{-2} \tag{4-4}$$

and thus will require only one multiplication. The cascade and direct form require the same number of numerical operations.

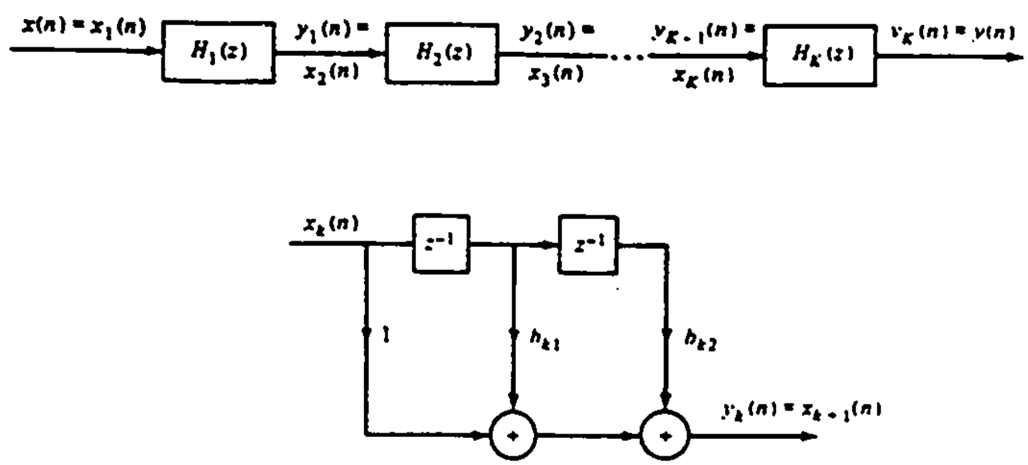


Figure (4-3) : Cascade realization of an FIR filter

c) Frequency-sampling structure [5]

The frequency-sampling realization is an alternative structure for an FIR filter in which the parameters that characterized the filter are the desired frequency response instead of the impulse response h(n). The desired frequency response at a set of equally spaced frequencies, namely

$$\omega_k = \frac{2\pi}{M}(k+\alpha) \tag{4-5}$$

where $k= 0, 1, \ldots,(M-1)/2$ M odd

$k= 0, 1, \ldots, (M/2)-1$ M even

$\alpha= 0$ or $1/2$

The value of the frequency response $H(\omega)$ at frequencies ω_k is simply

$$H(k+\alpha) = H(\frac{2\pi}{M}(k+\alpha)) = \sum_{n=0}^{M-1} h(n)e^{\frac{-j2\pi(k+\alpha)n}{M}} \tag{4-6}$$

It is a simpler matter to invert to (4-6) and express h(n) in terms of the frequency response

$$h(n) = \frac{1}{M}\sum_{k=0}^{M-1} H(k+\alpha)e^{2\pi(k+\alpha)\frac{n}{M}} \tag{4-7}$$

where $n=0,1,\ldots, M-1$

Now if we substitute (4-7) in Z-transform H(z), interchanging the order of two summation and performing the summation over the index n we obtain

$$H(z) = (1-z^{-N})\frac{1}{N}\sum_{k=0}^{N-1} \frac{\tilde{H}(k)}{1-W_N^{-k}z^{-1}} \tag{4-8}$$

$$W_N^{-k} = e^{j(2\pi k/N)} \tag{4-9}$$

$$\bar{H}(k) = H(W_N^{-k}) = |\bar{H}(k)|\, e^{j\theta(k)} \tag{4-10}$$

The quantities H(k) are called frequency samples. Equation (4-8) suggested that an FIR system can be realized as a cascade of a simple FIR system with an IIR system as shown in Figure (4-4). The frequency sampling realization of FIR is computationally more efficient than the direct form realization.

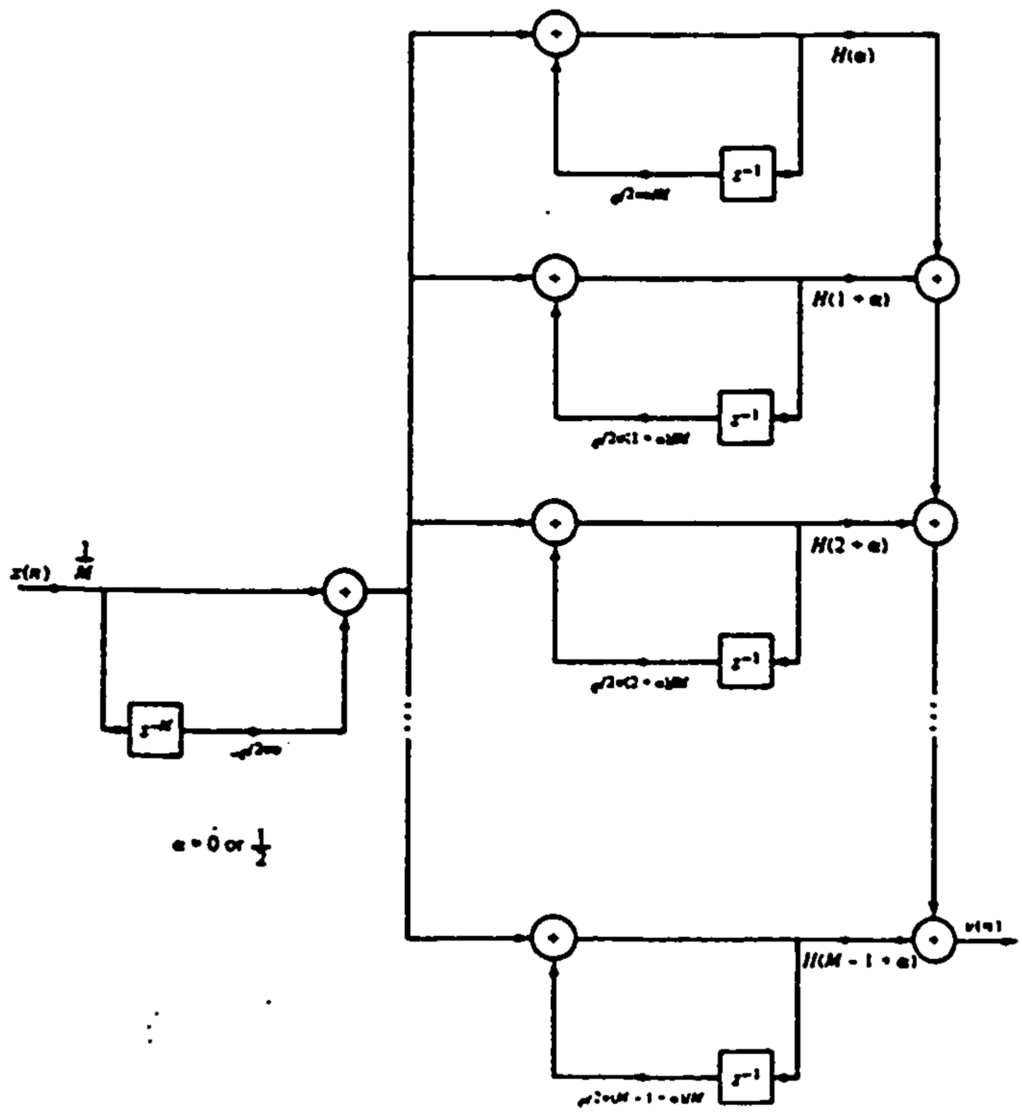


Figure (4-4) : Frequency-sampling realization of FIR filter.

d) Lattice realization [5]

Lattice filters are extensively used in digital speech processing and in the implementation of adaptive filters. It is desirable to view the FIR filters as linear predictors. The input data sequence x(n-1), x(n-2),.....,x(n-m) is used to predict the value of the signal x(n). hence we may express the linearly predicted value of x(n) as

$$x'(n) = -\sum_{k=1}^{m} \alpha_m(k) x(n-k) \tag{4-11}$$

where $\{-\alpha_m(k)\}$ represent the predication coefficients. The output sequence y(n) may be expressed as

$$y(n) = x(n) - x'(n) = x(n) + \sum_{k=1}^{m} \alpha_m(k) x(n-k) \tag{4-12}$$

Thus the FIR filter output given by (4-12) may be interpreted as the error between the true signal value x(n) and the predicted value x'(n). Suppose that we have a filter for which m=1. Clearly, the output of such a filter is

$$y(n) = x(n) + \alpha_1(1) x(n-1) \tag{4-13}$$

If we select $K_1 = \alpha_1(1)$, the parameter K_1 in the lattice is called a reflection coefficient. By cascading two lattice stages, it is possible to obtain the output for m=2. By continuing this process, one can easily demonstrate by induction the equivalence between an mth-order direct-form FIR filter and an m-order stage lattice filter. The lattice filter is generally described by the following set of order-recursive equations:

$$f_0(n)=g_0(n)=x(n) \tag{4-14}$$

$$f_m(n)=f_{m-1}(n)+K_m g_{m-1}(n-1) \qquad m=1,2,\ldots,M-1 \tag{4-15}$$

$$g_m(n)=K_m fm-1(n)+g_{m-1}(n-1) \qquad m=1,2,\ldots,M-1 \tag{4-16}$$

Then the output of the (M-1) stage filter corresponds to the output of an (M-1) order filter, that is

$$Y(n)=f_{m-1}(n) \tag{4-17}$$

Figure (4-5) illustrates an (M-1) stage lattice filter in block diagram along with a typical stage that shows the computations specified by (4-15) and (4-16).

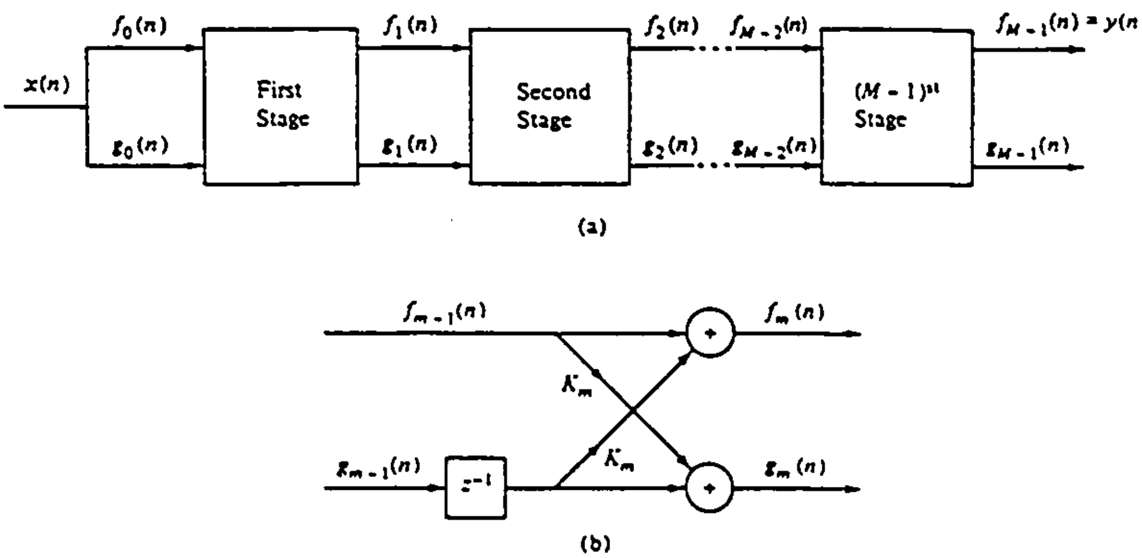


Figure (4-5): (M-1)-stage lattice filter

4.1.2. REALIZATION STRUCTURE FOR 2-D FIR FILTERS

Since 2-D FIR filters usually can not be factored, the realization structure for 2-D FIR filter is not as uncomplicated as for a 1-D FIR filter. It might seem unusual to classify the cascade structure as a specialized implementation but because of the formulation of a 2-D filter a possible method to implement a 2-D FIR filter is a direct form realization which is quite similar to 1-D structure. Consider a factorable 2-D FIR frequency response which can be written as

$$H(\omega_1, \omega_2) = F(\omega_1, \omega_2)\, G(\omega_1, \omega_2) \tag{4-18}$$

If $f(n_1,n_2)$ is an (MxM) points array and $g(n_1,n_2)$ is an [(N-M+1)x(N-M+1)] points array, then $h(n_1,n_2)$ will have an (NxN)-points region of support. To implement $h(n_1,n_2)$ as a direct convolution requires N^2 multiplication per output sample, whereas to implement the filter in the cascade form implied by (4-8) requires only $M^2 + (N-M+1)^2$ multiplications. To design a filter that is to be realized as a cascade, the frequency response must first be expressed in factored form:

$$H(z_1, z_2) = \prod_{k=1}^{L} \sum_{i=0}^{M} \sum_{j=0}^{N} [(a_{ij})_k z_1^{-i} z_2^{-j}] \tag{4-19}$$

If two FIR filters with impulse responses $h_1(n_1,n_2)$ and $h_2(n_1,n_2)$ are connected in parallel, they are equivalent to a single filter with the impulse response

$$h(n_1, n_2) = h_1(n_1, n_2) + h_2(n_1, n_2) \tag{4-20}$$

The frequency response for this kind of implementation must be expressed as follows:

$$H(z_1, z_2) = \sum_{i=1}^{k} H_i(z_1, z_2) \tag{4-21}$$

where
$$H_i(z_1, z_2) = \sum_{k=0}^{M} \sum_{j=0}^{N} (a_{kj})_i \, z_1^{-k} z_2^{-j} \qquad (4\text{-}21)$$

4.2. THE IMPLEMENTATION OF THE DESIGN TECHNIQUE

It can be recalled that the frequency response of a 2-D FIR zero-phase is as follows:

$$H(e^{j\omega_1}, e^{j\omega_2}) = \sum_{n=0}^{N} b(n)\,[F(\omega_1, \omega_2)]^n \qquad (4\text{-}22)$$

where N is the order of the filter and $F(\omega_1 , \omega_2)$ is defined by

$$F(\omega_1, \omega_2) = \sum_{k=0}^{1} \sum_{l=0}^{1} t(k,l)\cos(\omega_1 k)\cos(\omega_2 l) \qquad (4\text{-}23)$$

Although the impulse response of this filter is extended in $(2N+1) \times (2N+1)$ points, it can be noticed from equations (4-22) and (4-23) that the entire $(2N+1) \times (2N+1)$ point filter can be completely specified by $\{(2\times2) + (N+1)\}$ free parameters. It can be simply noted that the first term in the number of free parameters, (2×2), is derived from equation (4-23) and the second term , $(N + 1)$, is derived from equation (4-22). Now, it is reasonable to expect that there exist specific realizations for these filters which have an implementation complexity proportional to N rather than to N^2. To understand better this 2-D structure we first consider a related 1-D zero-phase filter of length $2N+1$ and the following frequency response:

$$H(e^{j\omega}) = \sum_{n=0}^{N} b(n)[\cos(\omega)]^n \tag{4-24}$$

If h_f (m) represents the impulse response of a linear shift-invariant system which has frequency response:

$$H_f(e^{j\omega}) = \cos(\omega) \tag{4-25}$$

The frequency response for $(\cos \omega)^n$ will be obtained by n times cascading of the system mentioned in equation (4-25). A tapped cascade of such systems with tap coefficients b(n) thus has the frequency response depicted in equation (4-24).

The system $h_f(l)$ corresponds to a non-causal FIR filter with an impulse response length of three;

$$h_f(l) = \begin{cases} \frac{1}{2} & l = \pm 1 \\ 0 & \textit{otherwise} \end{cases} \tag{4-26}$$

The transform function corresponding to this system is:

$$H_f(z) = \frac{1}{2}(z + z^{-1}) \tag{4-27}$$

Since this system is a non-causal system, it is not a realizable system. To fix this problem all impulse responses in the negative side have to be shifted to the right side of the axis. This action in the time domain corresponds to multiplication of the transfer function by $Z^{(N-1)/2}$, where N is the order of the filter. The overall network is diagrammed in Figure (4-6).

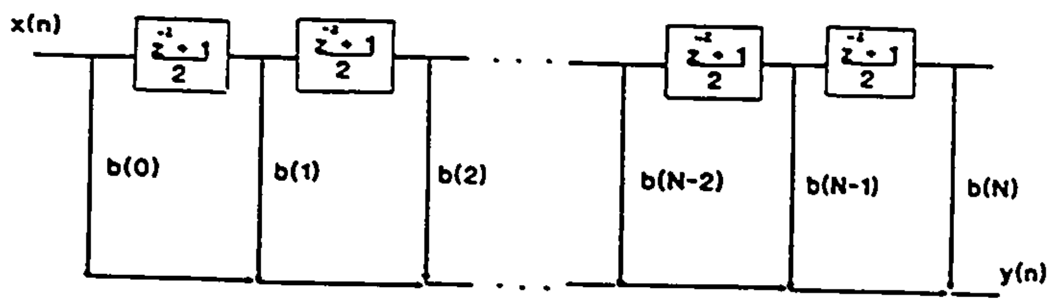


Figure (4-6): The overall network for the designed filter

It is known that a linear shift-invariant subnetwork can be replaced by another linear shift-invariant subnetwork. To implement the McClellan transformation a 1-D subnetwork

h_1 (l) is replaced by a 2-D subnetwork which has impulse response $h_t(m,n)$ and the following frequency response.

$$H_f(e^{j\omega_1}, e^{j\omega_2}) = F(\omega_1, \omega_2) = \sum_{k=0}^{1} \sum_{l=0}^{1} t(k,l) \cos(\omega_1 k) \cos(\omega_2 l) \tag{4-28}$$

The Z transform corresponding to this system is

$$H_f(z_1, z_2) = t(0,0) + t(0,1)\left(\frac{z_1 + z_1^{-1}}{2}\right) + t(1,0)\left(\frac{z_2 + z_2^{-1}}{2}\right) + t(1,1)\frac{(z_1 + z_1^{-1})(z_2 + z_2^{-1})}{4} \tag{4-29}$$

The filter h_f (m,n) is a zero-phase 2 $\times$ 2 point non-causal FIR filter. When the general form of the transfer function for a (2 x 2) noncausal FIR filter compares with the transfer function in equation (4-29), one can see that the relation between h_f (m,n) and transformation parameters, t, is as follows:

$$h_f(m,n) = \frac{t(|m|,|n|)}{(2-\delta_{m0})(2-\delta_{0n})} \tag{4-30}$$

where δ_{mn} is the Kronecker delta and $-1 \leq m, n \leq +1$.

To implement a transformed design, the prototype 1-D zero-phase filter h_f (l) has to be first realized then the transformation must be physically implemented by replacing the operators by transformation filter, h_f (m,n). Figure (4-7) shows the direct transformed implementation.

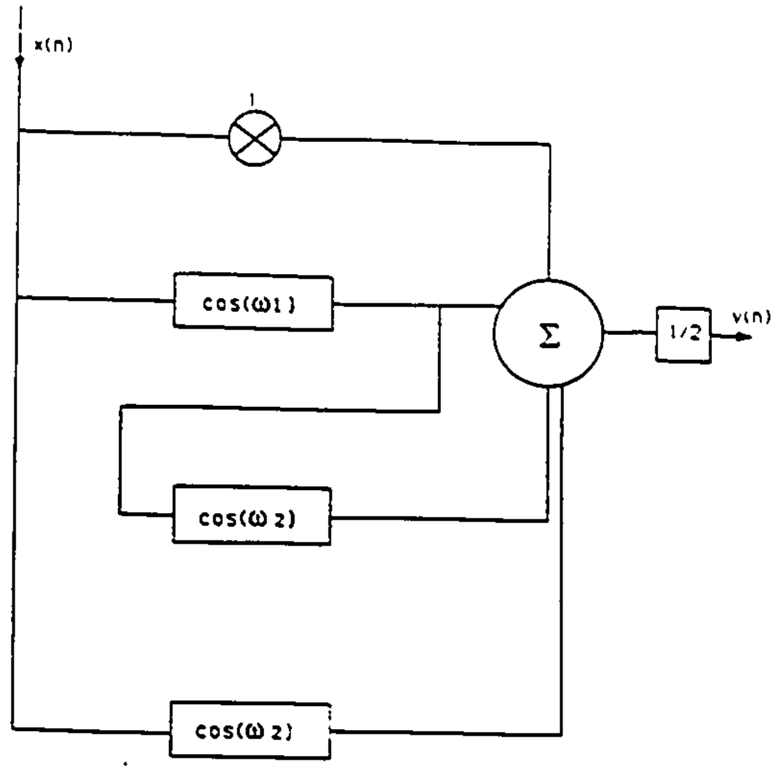


Figure (4-7) : The direct transform implementation

The tap coefficients in the filter implementation are the b(n)'s given by the 1-D prototype filter. The subnetwork $h_f(m,n)$ depends upon the specific transformation which is being used. In the case of a 2-D FIR filter with nearly circular symmetry, to change the cutoff frequency of the filter, the only change will be in the tap coefficients, b(n).

Other structures can also be considered for realization of these techniques. For example, the coefficients of direct form FIR filters can be converted to lattice coefficients. Suppose that we are given the FIR coefficients for the direct form realization or, equivalently, the polynomial $A_m(z)$ and we wish to determine the corresponding lattice filter parameters $\{K_i\}$. For the M-stage lattice $K_m = \alpha_m(m)$. K_m is obtained from the polynomial $A_m(z)$ for $m=M-1,M-2,\ldots,1$. Consequently, we need to compute the polynomial $A_m(z)$ starting from $m = M-1$ and stepping down successively to $m=1$.

4.3. SUMMARY

In this chapter we have recalled some of the techniques for realization of the one and two dimensional FIR filters. The McClellan transformation as a most popular technique can overcome the design difficulties in 2-D FIR filter design. It was also shown that filter design by this method also possesses efficient implementations which can partially overcome the difficulty with implementation.

We have shown that the implementation complexity of this design technique is proportional to N rather than N^2. The impulse responses of the transformation equation have been calculated in terms of the transformation parameters. Thus direct transformed implementation could be realized. We also considered all possible realization techniques such as lattice structure for implementing this design technique.

CHAPTER FIVE

SUMMARY AND CONCLUSION

SUMMARY AND CONCLUSION

In this thesis the following subjects have been investigated.

1) The importance of coefficient quantization error and its effect on the behaviour of the filter has been shown. To reduce this error several techniques for designing a FIR filter with integer coefficients have been presented in the literature some of which were categorized and explored in the introduction to chapter one.

The first design technique, which is the combination of the branch and bound technique and nonlinear programming, shows its limited ability to handle high order filters, and there is no guarantee for obtaining an optimal minimum. The second design technique does not guarantee the optimal solution but it can handle higher order filters. Mixed integer linear programming is also found to be very attractive for providing optimal results but the computational complexity limits the design for the filter with the order up to forty.

The next design technique is the power of two coefficients which require no multiplier in the structure. Linear or nonlinear programming is employed in this technique, the result of the design based on the type of program, linear or nonlinear, can be optimal or non-optimal. The computational cost limits a filter design order up to ninety.

The last design technique builds based on delta modulation concept. A structure consists of a transversal filter with tap coefficients restricted to -1, 0, +1, which is cascaded with an accumulator. As in delta modulation the sampling rate must be sufficiently high to obtain acceptable performance; as a result the useful transversal filters in this structure have a very high order.

2) Four techniques have been considered for in-depth consideration.

2-1) Discretization and Reoptimization (type 1):

A simple design approach is presented in which the whole process consists of an N times optimization process in which N is the number of coefficients which has to be determined. At each iteration the least effective coefficient on error function is fixed to integer value and the remaining coefficients are determined by the optimization process. This technique has been applied to design several filters. The following are the major characteristics of this technique:

a) The approximate running time is less than the other methods except delta modulation technique;

b) The method is capable of handling high order filters;

c) A feasible solution is guaranteed but local optimum may result; and

d) The method is sensitive to initial value in optimization process.

2-2) Discretization and reoptimization (type 2)

A simple design approach is implemented in which the whole process consists of a 2N times optimization process where N is the number of coefficients which need to be determined. At each iteration the most sensitive coefficient is discretized to integer values and the remaining coefficients are determined through the optimization process for two cases. The remaining coefficients are fixed to the set of values corresponding to minimum objective function. This process will apply until all the coefficients convert to integer value. This technique has been applied to design several filters. The following are the major characteristics of this technique:

a) The approximate running time is twice as long as type 1;

b) The method is capable of handling high order filters;

c) A feasible solution is guaranteed but local optimum may result;

d) The method is sensitive to initial value in optimization process; and

e) The method shows better response compared to type 1 in the least square error criteria.

2-3) Mixed integer linear programming

The formulation of the problem for linear and integer programming has been explained. Two packages named LINDO and another one known as MINT have been used for this technique. The following are the major characteristics of this technique and the comparison between these two packages.

a) The process of setting up the problem is more difficult in the LINDO in contrast to the MINT.

b) The approximate running time is much faster in the LINDO.

c) The LINDO has less sensitivity to the values of the frequency grid points. The integer linear programming approach for designing filters is very sensitive to the number and the value of the frequency grid points.

d) The LINDO is quite flexible to the upper bounds values for the coefficients, but the upper bounds in the MINT have a certain limit which depends on the number of variables.

e) The number of required iterations in the LINDO can be adjusted by the user where the MINT algorithm is terminated when the optimality is established.

f) No optimal solution for these examples has been established The mentioned time in comparison tables implies for 15000 iterations.

g) This technique is not very suitable for the filters with order higher than forty.

2-4) Delta modulation like filter

An attractive filter structure based on the delta modulation concept has been explained. The application of the design technique has been investigated by applying few examples. The oversampling factor "K" and the step size "Δ_k" play an important part in this technique. The accuracy of the design can be increased by increasing the value of K. Designing a high order filter with a very narrow passband and very sharp transition bound is the price for obtaining higher accuracy.

Tables (2-36,37,38,40) display the characteristics of these techniques compared to other techniques. The location of zeros in Z domain is one of the important investigations in filter design. Although the coefficients obtained by these techniques are different, most of the zeros are closely bunched together.

In chapter three, we have investigated the design of a 2-D FIR filter with integer coefficients. To achieve our goal, some of the techniques in the literature have been investigated. The direct extension of Discretization & Reoptimization (type 1&2) has been examined for 2-D FIR filter. Although the use of octagonal symmetry substantially reduces the number of coefficients which has to be determined and function evaluation, the memory requirement and computation burden are two inconvenient points in this design technique. It is worth mentioning that these techniques can design filters with a higher order than the one reported in [51].

As an alternative approach the McClellan transformation has been considered. In this technique a 1-D FIR filter is transformed to a 2-D FIR filter by means of the McClellan transformation. This design technique splits the process into two steps, firstly the transformation parameters are obtained by linear or nonlinear programming formulation, secondly the coefficients of the 1-D prototype filter are determined by any design technique in 1-D FIR filter. We showed that by selecting the proper values for transformation parameters no multipliers are required in this part. To determine the integer coefficients for the 1-D prototype filter Discretization & Reoptimization (type 1&2) and Mixed integer linear programming have been used. Several examples demonstrate the versatility of this design technique. Although the coefficients obtained by these techniques are different, most of the zeros are closely bunched together.

In chapter four, we have recalled some of the techniques for realization of one and two dimensional FIR filters. The direct and cascade form, frequency sampling structure, and lattice structure in 1-D and direct, cascade, and parallel form have been explained. The McClellan transformation as a most popular technique can overcome the design difficulties in 2-D FIR filter design. It was also shown that filter design by this method also possesses efficient implementations which can partially overcome the difficulty with implementation.

We have shown that the implementation complexity of this design technique is proportional to N rather than N^2. The impulse responses of the transformation equation have been calculated in terms of the transformation parameters. Thus direct transformed implementation could be realized. We also considered other possible realization techniques such as lattice structure for implementing this design technique.

REFERENCES:

[1] R. King, M. Ahmadi, R. Naguib, A. Kwabwe, and M. Azimi Sadjadi,"Digital filtering in one and two dimensions, Design and application,"Plenum Press, 1989.

[2] A. Antoniou,"Digital filters: Analysis and design," McGraw-Hill, NewYork, 1979.

[3] L.R. Rabinar and B. Gold," Theory and application of digital signal processing," Prentice-Hall, NewYork 1975.

[4] A.V. Openheim and R.W. Schafer," Digital signal processing," Prentice - Hall, New Delhi, 1989.

[5] J.G. Proakis and D.G. Manolakis," Introduction to digital signal processing," MacMillan Publishing, NewYork, 1988.

[6] D.E. Dudeon and R.M. Mersereau," Multidimensional digital signal processing," Prentic-Hall, NewYork, 1984.

[7] J.F. Kaiser," Digital filters" ch. 7 in System Analysis by Digital Computer, F.F. Kuo and J.F. Kaiser," Wiley, NewYork, 1966.

[8] A. Antoniou, M. Ahmadi, C. Charalambous, "Design of factorable lowpass 2-D digital filters satisfying prescribed specifications," Proc. IEE 128, Part G, No 2, 53-60 (1981).

[9] R.B. Blackman and J.W. Tukey, "The Measurement of power Spectra," Dover, NewYork, 1958.

[10] G.E Heyliger, "The scanning function approach to the design of numerical filters, " Martin Co., Denver, Colo,. Rep. R-63-2, Apt. 1963.

[11] ---, "Design of numerical filters: Scanning functions and equal ripple approximation," in Proc. 4th Allerton Conf. Circuit and System Theory, pp. 175-185, 1966.

[12] G.E. Heyliger and C.A. Halijak, "Topics in the design of moving average numerical filters," in Proc. 2nd Asilomar Conf. Circuits and Systems, pp. 214-220, 1968.

[13] H.D. Helms, "Nonrecursive digital filters: Design methods for achieving specifications on frequency response," IEEE Trans. Audio Electroacoust., vol. Au-16, pp. 336-342, Sep. 1968.

[14] ---,"A direct search procedure for designing finite duration impulse response filters," IEEE Trans. Audio Electroacoust., vol. Au-17, pp.33-36, Mar. 1969.

[15] L.R. Rabiner, B. Gold, and C.A. McGonegal, "An approach to the approximation problem for nonrecursive digital filters, "IEEE Trans. Audio Electroacoust., vol. Au-18,pp.83-106,June 1970.

[16] O.Herrmann, "Design of nonrecursive digital filters with linear phase,"Electron. Lett., vol.6,no.11,pp.328-329,1970.

[17] E.Hofstetter, A. Oppenheim, and J. Siegel, "A new technique for the design of nonrecursive digital filters," in Proc. 5th Annu.Princeton Conf. Information Sciences and Systems, pp.64-72, 1971.

[18] ---,"On optimum nonrecursive digital filters," in Proc. 9th Allerton Conf. Circuit and System Theory, Oct. 1971.

[19] T.E. Parks and J.H. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," IEEE Trans. Circuit Theory, vol.Ct-19,pp.189-194,Mar.1972.

[20] L.R. Rabiner,"The design of finite impulse response digital filters using linear programming techniques," Bell Syst. Tech.J., vol.51,pp.1117-1198,July-Aug. 1972.

[21] ---,"Linear program design of finite impulse response(FIR) digital filters,"IEEE Trans. Audio Electroacoust., vol. Au-20, pp.280-288,Oct.1972.

[22] J.H.McCellan and T.W. Parks,"A unified approach to the design of optimum FIR linear phase digital filters," IEEE Trans. Circuit Theory, vol. CT-20,PP.697, Nov. 1973.

[23] T.S. Huang,"Two-dimensional Windows," IEEE Trans., A_ 20,pp. 88-89, 1972.

[24] M.Ahmadi, A.Chottera, "An improved method for the design of 2-D FIR digital filters with circular and rectangular cut-off boundary using Kaiser window," "Can.Elec.Eng.J.Vo 18 No 1,1983.

[25] J.V. Hu, and L.R. Rabiner,"Design Techniques for Two-Dimensional Digital Filters,"IEEE Trans, Au-20,pp.249-257, 1972.

[26] Y. Kamp and J.P. Thiran,"Chebyshev Approximation for Two-Dimensional Non-recursive Digital Filters," CAS-22,PP. 208-218, 1975.

[27] J.H. McClellan,"The Design of Two-Dimensional Digital Filters by Transformation,"Proc. of 7th Annual Princeton Conference on Information

Sciences and Systems,pp.247-251,1973.

[28] Mersereau, R.M., Mecklenbrauker, W.F.G. and Quatieri,T.F.Jr., "McClellan Transformation for Two-Dimensional Digital Filtering I-Design," IEEE Trans,CAS-23, pp.405-413,1976.

[29] S.N. Hazra, and M. Sudhakara Reddy," Design of circularly symmetric low-pass 2-D FIR digital filter using transformation," IEEE tran. Circuits and systems, vol. CAS-33, No.10, pp 1022-1026, Oct.1986.

[30] D.Thong Nguyen and M. N. S. Swamy," Formulas for Parameters Scaling in the McClellan Transform." IEEE Trans, vol. CAS-33, No 1, pp.108-109, January 1986

[31] Mecklenbrauker, W.F.G. and Mersereau, R.M., "McClellan Transformation for Two-Dimensional Digital Filtering:II-Implementaion pp.414-422, 1974

[32] J. H. Lodge and M. M., Fahmy," An efficient l_p optimization technique for the design 2-D linear-phase FIR digital filter," IEEE Trans. Acoust., Speech, Signal Processing, Vol.AASP-28, pp. 308-313, June 1980.

[33] V. R. Algazi, M.Suk. and C. S. Rim," Design of almost minimax FIR filters in one and two dimensions by WLS techniques," IEEE. Trans. Circuits Syst., vol.CAS-33,pp. 590-596,June 1986.

[34] C. L. Lawson," Contributions to the theory of linear least maximum approximations,"Ph.D. dissertation, U.C.L.A., 1961.

[35] M. O. Ahmad and J-D. Wang," An analytical least square solution to the design problem of 2-D FIR filters with quadrantally symmetric or antisymmetric frequency response," IEEE Trans. Circuite Syst., vol. 36, NO. 7, pp. 968-979 July 1989.

[36] G. A. Lampropoulos and M. Fahmi," A new technique for the design of 2-D FIR and IIR filters," IEEE Trans. Acoust., Speech, vol. ASSP-33, No. 1, February, 1985.

[37] V. B. Lawrence and A. C. Salazar," Finite precision design of linear-phase FIR filters," Bell Syst. Tech. J., vol. 59, pp. 1575-1598, Nov. 1980.

[38] M. Kodek," Design of optimal finite wordlength FIR digital filters using integer programming techniques," IEEE. Trans. Acoust., Speech, vol. ASSP-28, NO.3, June 1980.

[39] J. P. Marques De Sa," A new design method of optimal finite wordlength linear phase FIR digital filters," IEEE. Trans. Acoust., Speech, vol. ASSP-31, NO. 4,

August, 1983.

[40] R. J. Dakin," A tree search algorithm for mixed integer propgramming problems,"Comput. J., vol. 8, pp. 250-255, 1966.

[41] C. Charalambous and M. J. Best," Optimization of recursive digital filters with finite word lengths," IEEE. Trans Circuits Syst., vol. CAS-22, pp.190-196, M: r. 1974.

[42] M. J. Best," A feasible conjugate direction method to solve linearly constrained optimization problems," J. Optimiz. Theory Appl., in press.

[43] M. J. Best and K. Ritter," An accelerated conjugate direction method to solve linearly constrained minimization problems," Univ. of Waterloo, Waterloo, Ont., Canada, Res. Rep. CORR 73-16, 1973.

[44] Z. Jing and T. Fam," A new scheme for designing IIR filters with finite word length coefficients," IEEE. Trans., Acoust. Speech , vol. ASSP-34, No. 5, October 1986.

[45] IBM Program Product Manual- Mathematical Pogramming System Extended (MPSX) Mixed Integer Programming (MIP) Program Description, Second edition August 1973.

[46] Y. Ching. Lim and R. Parker," FIR filter design over a discrete power-of-two coefficient space," IEEE. Trans., Acoust. Speech, vol. ASSP-31, NO. 3, June 1983.

[47] Y. Ching. Lim and R. Parker," Discrete coefficient FIR digital filter design based upon an LMS criteria," IEEE. Trans. Circuits Syst., vol. CAS-30, pp. 723-739, Oct. 1983.

[48] Q. Zhao and Y. Tadokoro," A simple design of FIR filters with powers-of-two coefficients," IEEE. Trans. Circuits Syst., vol. CAS-35, No. 5, May 1988.

[49] Mark R. Bateman and Bede Liu," An approach to programmable CTD filters using coefficients 0, +1, and -1," IEEE. Trans.,Circuits Syst., vol. CAS-27, NO.6, June,1980.

[50] R. Steele,"Delta Modulation Syystem," John Wiley & Sons, NewYork, 1975.

[51] P. Siohan and A. Benslimane," Finite precision design of optimal linear phase 2-D FIR digital filters," IEEE. Trans. Circuits Syst., vol. CAS-36, NO. 1,January 1989.

[52] S.C. Pei and S.-B. Jaw," Efficient design of 2-D multiplierless FIR filters by

transfomations," IEEE. Circuits Syst., vol. CAS-34, NO. 4, April 1987.

[53] S.L. Gass," Linear programming methods and applications," McGraw-Hill, NewYork, 1985.

[54] M.S. Bazaraa and J.J. Jarvis," Linear programming and network flow," John Wiley & Sons, NewYork, 1984.

[55] Stanley Zionts," Linear and integer programming," Prentice-Hall, NewJersey, 1980.

[56] Billy E. Gillett," Introduction to operation research," McGraw-Hill,NewYork, 1980.

[57] R. Fletcher,"A new approach to variable metric algorithms," The Computer Journal, vol.13, No. 3, August, 1970.

APPENDIX A

THE FLOW-CHART FOR DISCRETIZATION AND REOPTIMIZATION (TYPE 1)

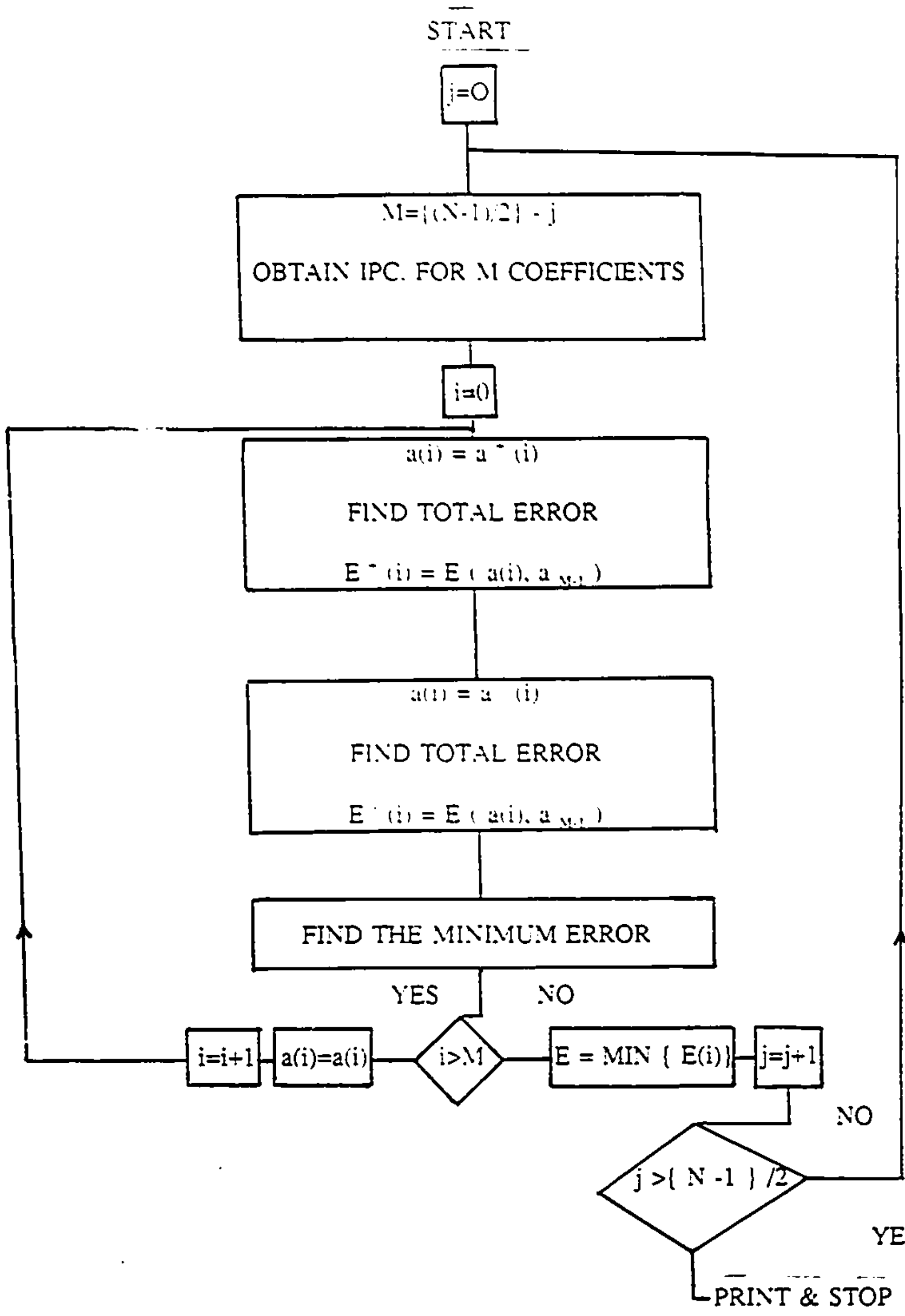


FIGURE A-1 : FLOW-CHART FOR THE D & R (TYPE 1)

APPENDIX B

THE FLOW-CHART FOR DISCRETIZATION AND REOPTIMIZATION (TYPE 2)

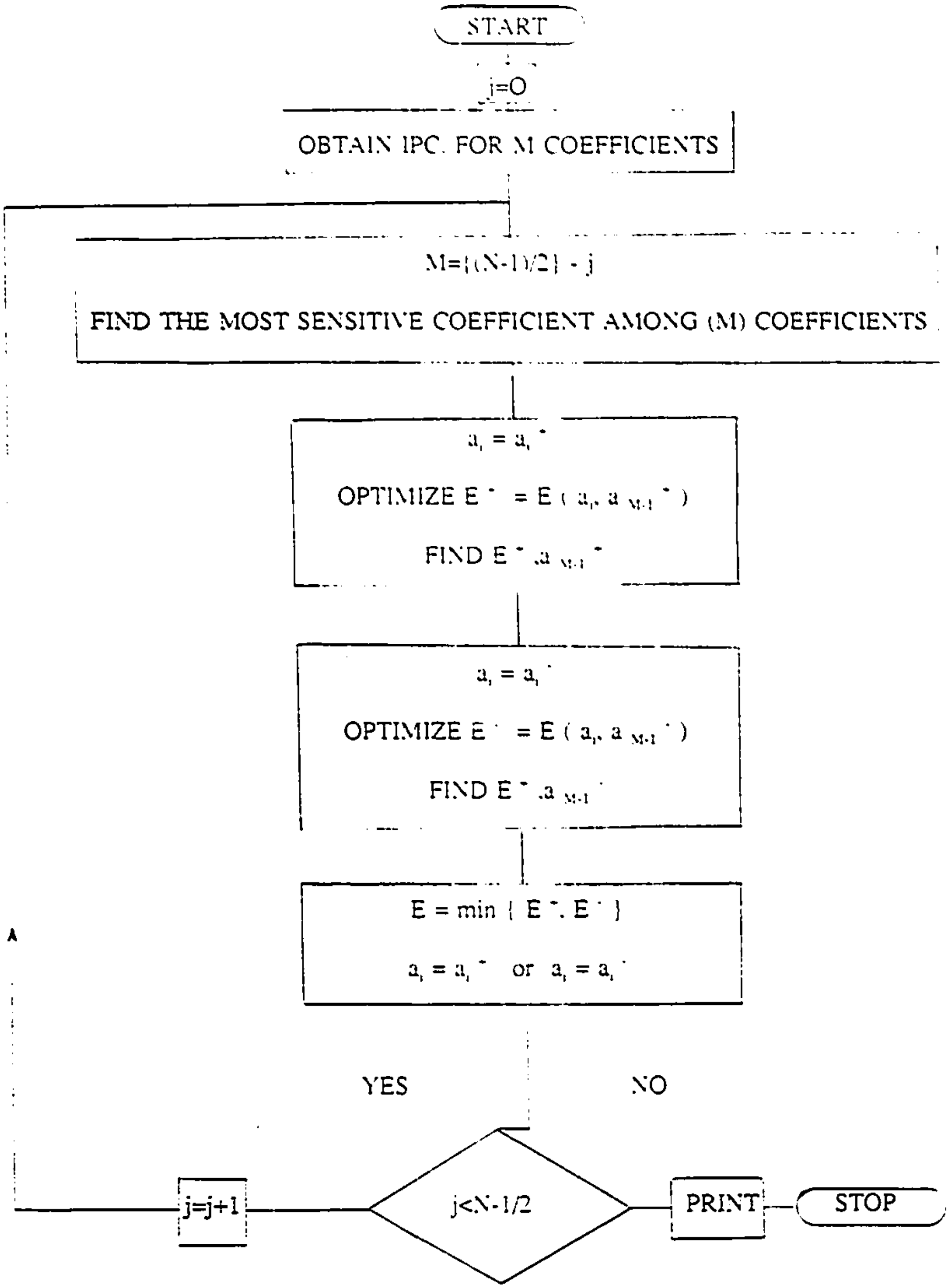


FIGURE B-1 : FLOW-CHART FOR THE D & R (TYPE 2)

APPENDIX C

NONLINEAR, LINEAR, AND MIXED INTEGER PROGRAMMING

C.1. NONLINEAR PROGRAMMING

A key assumption of linear programming is that all its functions (objective function and constraint functions) are linear. Although this assumption essentially holds for numerous practical problems, it frequently fails to be used for some problems. It is sometimes possible to reformulate non-linearities into a linear programming format. Nevertheless, it is often necessary to deal directly with nonlinear programming problems.

In one general form, the nonlinear programming problem is to find $X=(x_1,x_2,\ldots,x_n)$ so as to

$$\text{Maximize } F(X)$$

subject to

$$g_i(X) \leq b_i \qquad \textit{for} \quad i=1,2,\ldots,m \tag{C-1}$$

and

$$X \geq 0 \tag{C-2}$$

where F(X) and the G_i (X) are given functions of the n decision variables. No algorithm is available that will solve every specific problem fitting this format. However, by making various assumptions about these functions, substantial progress has been made for some important special cases of this problem. In our study, the rapidly convergent descent method for minimization by Fletcher and Powell has been used.

C.2. LINEAR PROGRAMMING

Historically, the initial mathematical statement of the general problem in linear programming along with the simplex method were first developed and applied in 1947 by George B. Dantzig, Marshal Wood, and their associates of the U.S. Department of the U.S. Air Force. Since then, linear programming has become an important tool of modern theoretical and applied mathematics. This remarkable growth can be traced to the pioneering efforts of many individuals and research organizations.

A linear programming problem differs from the general variety in that a mathematical model or description of the problem can be stated using relationships which are called straight-line or linear. The general mathematical model of the linear programming problem can be stated as follow

Minimize the objective function

$$c_1\, x_1 + c_2\, x_2 + \cdots + c_j\, x_j + \cdots + c_n\, x_n \qquad \text{(C-3)}$$

subject to the conditions

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1j}x_j + \cdots - a_{1n}x_n = b_1 \qquad \text{(C-4)}$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2j}x_j + \cdots + a_{2n}x_n = b_2$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mj}x_j + \cdots + a_{mn}x_n = b_m$$

and

$$x_1, x_2, \cdots, x_n \geq 0 \tag{C-5}$$

where $1 \leq i \leq n$ and $1 \leq j \leq m$ and a_{ij} are all constants. The c_i are called cost coefficients. The form of equations (C-4) and (C-5) are not always the same for different problems. For example instead of equality constraints we may have inequality constraints of both types and instead of having positive variables in equation (C-5) we may have unconstraint variables. In first case, slack variables can be used and in the second case any unconstraint variable can be defined as a difference of two positive variables.

The linear integer programming problem may be stated as follows

Minimize the objective function

$$CX \tag{C-6}$$

Subject to

$$AX=b \tag{C-7}$$

$$X \geq 0 \tag{C-8}$$

$$X \text{ integer}$$

If some elements of the vector X are allowed to be real then the program will be called Mixed integer linear programming. A natural method to solve this problem is to ignore the last condition" X integer " and solve the problem as a linear program. At optimality, if all of the variables have integer value, then we have the optimal solution to the original integer program. Otherwise two techniques are used for obtaining the integer solution

C.2.1 CUTTING PLANE TECHNIQUE: The cutting-plane technique for solving integer LP problems is simply a technique that squeezes down on the set of all feasible solutions of the corresponding non-integer LP problem by sequentially introducing additional constraints (cuts) without cutting-off any feasible integer solution. The dual simplex is applied to re-optimize the new linear programming until an optimal integer solution of the integer programming problem is reached.

C.2.2. BRANCH AND BOUND TECHNIQUE: A classic paper by A.H. Land and A.G. Doig using a branch and bound principle appeared in 1960. The method is elegant in its simplicity; it requires the use of the simplex method, and can solve both all-integer and mixed-integer problems. The branch and bound technique involves a well structured systematic search of the space of all feasible solutions of constrained optimization problems that has a finite number of feasible solutions.

VITA AUCTORIS

HASSAN NIVI

1955	Born on eleventh of June in Tehran Iran
1974	Completed high school education at Sokhan high school in Tehran, Iran
1978	Completed B.Sc in Physics from Physics Dept., Pars College in Tehran, Iran
1980	Completed M.Sc in Physics from Physics Dept., Pars College in Tehran, Iran
1992	Candidate for the degree of Master of Applied Science in Electrical Engineering at the University of Windsor, Windsor , Ontario, Canada