DESIGN, STABILITY AND APPLICATIONS OF TWO-DIMENSIONAL RECURSIVE DIGITAL FILTERS

By

MADJID AHMADI

June, 1977

A Thesis submitted for the degree
of Doctor of Philosophy of the
University of London

Department of Electrical Engineering,
Imperial College of Science and Technology,
London, S.W.7.

ABSTRACT

Digital processing of two-dimensional signals is becoming increasingly important, and is finding applications covering various scientific disciplines.

Of the number of structures of two-dimensional digital filters possible, the recursive (IIR) is probably the mose efficient. The design of this kind of structure is, however, made complicated by stability considerations. This thesis will thus review past work on the stability problem and present some new methods for stabilising unstable filters.

Existing spatial and frequency domain design techniques for two-dimensional filters are briefly outlined, and a number of new methods suggested. In the frequency domain, a novel design technique for two-dimensional recursive (IIR) filters is described, along with another technique applicable to the more general N-dimensional case. One further frequency domain design technique suitable for N-dimensional non-recursive (FIR) filters is also given. While, for the spatial design techniques, a number of these are generalised and extended to the N-dimensional case.

ACKNOWLEDGEMENTS

I wish to express my gratitude to all members of the Communications Section who, in different manners, offered and provided help during my stay at Imperial College.

In particular I would like to thank Mr. R.A. King for his supervision and encouragement throughout the course of this research. I appreciate not only his suggestions, but also the warm and generous sentiments he has always shown towards me.

I would also like to express my indebtedness to Dr. A.G. Constantinides of Imperial College for having directed my attention initially to the field of digital signal processing, and for a number of contributions and suggestions that appear in this work.

I am very grateful to Mr. Y.A. Fakhro for the kindness and patience shown in checking through the manuscripts and for supplying many detailed corrections.

The work reported in this thesis has benefited in many ways from the numerous discussions that I had with several persons, most notable among whom are Dr. L.R. Rabiner of the Bell Laboratories, U.S.A., Dr. W. Saraga of Imperial College, Dr. D.J. Goodman of the Bell Laboratories, U.S.A., Professor V. Cappellini of the University of Florence, Italy, Professor S. Mitra of the University of California, Dr, Y. Kamp of M.B.L.E. Laboratories, Belgium, Dr. P.L. Emiliani of IROE, Italy, and Dr. F. Lotti of IROE, Italy.

The help and encouragement of Mr. P.M. Buckley, Dr. V.

Lawrence, Dr. A. A. Hashim, Dr. N.L. Esteves, Dr. M.M. Silva, Mr. Y.A. Fakhro, Mr. Ali Nazif and Mr. P.R. Pirnia is also very much appreciated.

Finally thanks are due to Mrs Shelagh Murdock for the conscientious and careful way in which she approached the typing of this thesis.

LIST OF SYMBOLS

$\sum_{i=1}^{N} q_i$	defined as $q_1+q_2 \ldots q_N$
$\prod_{i=1}^{N} q_i$	defined as $q_1 q_2 \ldots q_N$
u	angular frequency
T	sampling period
$\in$	belongs to
$\notin$	does not belong to
S	complex frequency variable
$z=e^{-juT}$	exponential frequency variable
a^*	complex conjugate of a
$*$	convolution sign
δ	Kronecker delta function
$i=1,n$	defined as $i=1,2,3,\ldots,n$

$\bigcap_{i=1}^{n} |z_i| = 1$ $\;(|z_1|=1) \cap (|z_2|=1) \;\ldots \cap (|z_n|=1)$

Im	imaginary part
Re	real part

$R_2 = \{(z_1,z_2) : |z_1| = 1 \cap |z_2| = 1\}$

$R_n = \{(z_i,\ i=1,2,\ldots,n) : \bigcap_{i=1}^{n} |z_i| = 1\}$

$D_1 = \{(z_i,\ i=1,2,\ldots,n) : \bigcap_{i=1}^{n} |z_i| \leqslant 1\}$

$D_2 = \{(z_i,\ i=1,2,\ldots,n) : \bigcap_{i=1}^{n} |z_i| < 1\}$

CONTENTS

Page

CHAPTER 1

INTRODUCTION

The last decade has witnessed some tremendous advances in the technology of digital systems, with a dramatic drop in the cost of basic hardware elements used in implementing such systems. As a result, many applications of digital signal processing have become feasible, and this in turn has stimulated the development of further theory. The trend, however, has been generally confined to one-dimensional (1-D) systems, and parallel development of two-dimensional (2-D) and the more general N-dimensional systems is yet to be seen.

Two-dimensional data, in the form of numbers regularly or irregularly spaced in a plane, are commonly encountered in a variety of engineering disciplines. In some cases, (1-D) signal processing techniques can be adapted to deal with such (2-D) data, but there are situations where such adaptations can prove to be inadequate. The need therefore arises for the full development of a general theory for (2-D) and (N-D) systems on the lines of the well-developed (1-D) systems theory.

The manipulation of (2-D) data is commonly referred to as image processing. This is because such data are usually displayed as spatial images for human evaluation or appreciation even though the two dimensions may not necessarily both be spatial. In signal processing a dimension can mean any physical domain in which a signal is defined. Time, space and frequency are examples of such domains, and any combination of these is allowed as a co-ordinate system. This provides some insight into how (N-D) signals can exist in spite of the human mind's inability to comprehend anything that is beyond 3-dimensional.

There are three general branches of image processing. The most important of these, digital filtering, is the main theme of this thesis. The remaining two aspects, image encoding and machine pattern recognition, will not be discussed in length. Analogies of the (2-D) theory in the more general (N-D) case will be given whenever possible.

(2-D) digital filters are computational algorithms that transform (2-D) input sequences of numbers into (2-D) output sequences according to pre-specified rules, hence yielding some desired modifications to the characteristics of the input sequences. The (N-D) filters can be similarly considered to be performing manipulation on (N-D) sequences of numbers.

Applications of (2-D) digital filters cover a wide spectrum, the object being usually either enhancement of an image to make it more acceptable to the human eye, or removal of the effects of some degradation mechanisms, or separation of features for easier identification and measurement by human or machine.

Many important applications of (2-D) digital filters have been in the field of space technology. Here digitally processed satellite images are used in monitoring environmental effects, earth resources and urban land use [1,2]. In such applications, (2-D) digital filters enhance or reduce boundaries, remove low-frequency shading effects, reduce noise and correct for distortions inherent to the imaging systems employed [3,4].

There are also applications in medicine and biology. (2-D) X-ray films are digitally processed to reduce the low spatial frequencies' contents, and by doing so fracture lines and other features with large high frequency components become easier to detect. This procedure is sometimes followed or preceded by contrast enhancement and noise reduction. Additional bio-medical uses include removal of scan lines in radio-isotope scanning, low frequency background noise reduction in photo-micrographs, and digital processing of acoustical holograms, the latter rapidly establishing themselves as replacements for X-rays [5,6].

Seismic prospecting is one area in which data is acquired as (2-D) sequences that are not images in the conventional sense. Seismic detectors are placed at intervals both along and across an area, and the digitized outputs of the detectors after an explosion form a (2-D) data array. This array is then processed to minimize the effects of multiple reflections and wind-induced noise, and information about the subsurface structure of the locality is then readily obtained from the output of the filtering operation [9,10].

Geophysics is yet another science where (2-D) digital signal processing is extensively employed. Atmospheric temperature and pressure data are digitally smoothed before plotting on weather maps [11]. Similarly, magnetic and gravity measurements are digitally processed to reduce the effects of surface anomalies, thus facilitating the identification of large subsurface features.

The above is only a brief review of some of the many possible applications of (2-D) filters. Such a discussion invariably leaves out a number of other applications, and does not attempt to speculate on the possible future additions to the list. However, it is evident from the above that any area in which (2-D) data is encountered is also a possible field of application of (2-D) digital filters.

1.1 THE Z-TRANSFORM OF (2-D) AND (N-D) SIGNALS

It is assumed that the reader is familiar with the (1-D) z-transform and its use as a convenient means of carrying out certain operations involving (1-D) digital signals. On the same lines, the z-transform of a regularly sampled (2-D) function

$$h = \{h_{m_1,m_2}\}$$

is given by

$$H(z_1,z_2) = \sum_{m_1=-\infty}^{\infty} \sum_{m_2=-\infty}^{\infty} h_{m_1,m_2} z_1^{m_1} z_2^{m_2} \tag{1.1}$$

where z_1 and z_2 are complex variables, and h_{m_1,m_2} is understood to be zero for subscript pairs (m_1,m_2) that do not belong to the definition set of h. When h is defined for non-negative subscripts only, it is known as a first quadrant function and is then denoted by:

$${}^1h = \left\{{}^1h_{m_1,m_2}\right\} \begin{matrix} m_1 \geqslant 0 \\ m_2 \geqslant 0 \end{matrix}$$

Similarly, second, third and fourth quadrant functions can be defined:

$${}^2h = \left\{{}^2h_{m_1,m_2}\right\} \begin{matrix} m_1 \leqslant 0 \\ m_2 \geqslant 0 \end{matrix}$$

$${}^3h = \left\{{}^3h_{m_1,m_2}\right\} \begin{matrix} m_1 \leqslant 0 \\ m_2 \leqslant 0 \end{matrix}$$

$${}^4h = \left\{{}^4h_{m_1,m_2}\right\} \begin{matrix} m_1 \geqslant 0 \\ m_2 \leqslant 0 \end{matrix}$$

As would be expected, the above definitions with some simple extensions apply to the (N-D) case. Thus, for a regularly sampled (N-D) signal

$$h = \left\{h_{m_1,\ldots,m_n}\right\}$$

the z-transform is defined as:

$$H(z_1,\ldots,z_n) = \sum_{m_1=-\infty}^{\infty} \cdots \sum_{m_n=-\infty}^{\infty} h_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n} \quad (1.2)$$

where $z_1, z_2, \ldots, z_n$ are again complex variables, and $h_{m_1,\ldots,m_n}$ is zero for values of $(m_1,\ldots,m_n)$ that are outside the definition set of h. When h is defined for non-negative subscripts only, it is called a first-quadrant function and is denoted by a left superscript 1:

$$^1h = \left\{ {}^1h_{m_1,\ldots,m_n} \right\} \qquad \bigcap_{i=1}^{n} m_i \geqslant 0$$

This is also the definition of a causal function, and therefore none of the other possible quadrant functions (for which any or some of the subscripts $m_1,\ldots,m_n$ are negative) can be causal.

1.2 CHARACTERIZATION OF (2-D) AND (N-D) DIGITAL FILTERS

Digital filters are in general either linear or non-linear. For example, contrast enhancement, as carried out by (2-D) digital filters, is a non-linear operation, whereas, on the other hand, spatial frequency filtering and numerous other operations are linear. This thesis will be mainly concerned with linear filtering, and in particular with that which is shift invariant where the input and output satisfy a (2-D) linear constant coefficients difference equation of the form [19, 20]:

$$o(m_1,m_2) = \sum_{j_1=1}^{M_1+1} \sum_{j_2=1}^{M_2+1} a_{j_1,j_2} \cdot i_{m_1-j_1+1,m_2-j_2+1} - \sum_{k_1=1}^{N_1+1} \sum_{k_2=1}^{N_2+1} b_{k_1,k_2} \cdot o_{m_1-k_1+1,m_2-k_2+1} \quad (1.3)$$

where M_1, M_2, N_1 and N_2 are arbitrary numbers defining the order of the filter and $k_1 . k_2 \neq 1$ simultaneously. This can be generalised for the (N-D) recursive filter:

$$o(m_1,\ldots,m_n) = \sum_{j_1=1}^{M_1+1} \cdots \sum_{j_n=1}^{M_n+1} a_{j_1,\ldots,j_n} \cdot i_{m_1-j_1+1,\ldots,m_n-j_n+1}$$

$$- \sum_{j_1=1}^{N_1+1} \cdots \sum_{j_n=1}^{N_n+1} b_{j_1,\ldots,j_n} \cdot o_{m_1-j_1+1,\ldots,m_n-j_n+1}$$

$$\prod_{\ell=1}^{n} j_\ell \neq 1 \qquad (1.4)$$

where, in both (1.3) and (1.4) above, $\{i\}$ and $\{o\}$ are the input and output sequences respectively.

Furthermore, linear digital filters can be classified either as those with infinite impulse response (IIR), which are usually recursive, or as those with finite impulse response (FIR), usually non-recursive. The transfer functions for these two types of (2-D) and (N-D) digital filters are given below.

1.2.1 The Recursive Filter

Firstly, for the recursive (IIR) filter, as from (1.3)

$$(2\text{-D}):\ H(z_1,z_2) = \frac{A(z_1,z_2)}{B(z_1,z_2)} = \frac{\sum_{j_1=1}^{M_1+1} \sum_{j_2=1}^{M_2+1} a_{j_1,j_2}\, z_1^{j_1-1}\, z_2^{j_2-1}}{\sum_{j_1=1}^{N_1+1} \sum_{j_2=1}^{N_2+1} b_{j_1,j_2}\, z_1^{j_1-1}\, z_2^{j_2-1}} \qquad (1.5)$$

and:

$$(\text{N-D}):\ H(z_1,\ldots,z_n) = \frac{A(z_1,\ldots,z_n)}{B(z_1,\ldots,z_n)} = \frac{\sum\limits_{j_1=1}^{M_1+1}\cdots\sum\limits_{j_n=1}^{M_n+1} a_{j_1,\ldots,j_n} z_1^{j_1-1}\cdots z_n^{j_n-1}}{\sum\limits_{j_1=1}^{N_1+1}\cdots\sum\limits_{j_n=1}^{N_n+1} b_{j_1,\ldots,j_n} z_1^{j_1-1}\cdots z_n^{j_n-1}} \quad (1.6)$$

And secondly, for the non-recursive (FIR) digital filter:

$$(\text{2-D}):\quad H(z_1,z_2) = \sum_{j_1=1}^{M_1+1}\sum_{j_2=1}^{M_2+1} a_{j_1,j_2} z_1^{j_1-1} z_2^{j_2-1} \quad (1.7)$$

$$(\text{N-D}):\quad H(z_1,\ldots,z_n) = \sum_{j_1=1}^{M_1+1}\cdots\sum_{j_n=1}^{M_n+1} a_{j_1,\ldots,j_n} z_1^{j_1-1}\cdots z_n^{j_n-1} \quad (1.8)$$

1.2.2 THE CONVOLUTIONAL FILTER

For a convolutional filter f the relation between the input sequence i and the output sequence o is:

$$(\text{2-D}):\quad o_{p_1,p_2} = \sum_{m_1=-\infty}^{\infty}\sum_{m_2=-\infty}^{\infty} i_{m_1,m_2}.f_{p_1-m_1,p_2-m_2} \quad (1.9)$$

$$(\text{N-D}):\quad o_{p_1,\ldots,p_n} = \sum_{m_1=-\infty}^{\infty}\cdots\sum_{m_n=-\infty}^{\infty} i_{m_1,\ldots,m_n}.f_{p_1-m_1,\ldots,p_n-m_n} \quad (1.10)$$

and in terms of the z-transforms this is:

$$(\text{2-D}):\quad O(z_1,z_2) = I(z_1,z_2).F(z_1,z_2) \quad (1.11)$$

$$(\text{N-D}):\quad O(z_1,\ldots,z_n) = I(z_1,\ldots,z_n).F(z_1,\ldots,z_n) \quad (1.12)$$

1.2.3 ZERO PHASE FILTER

A zero phase filter is defined as a filter whose amplitude spectrum has the following property:

$$(2\text{-D}): \quad \left|H(e^{ju_1}, e^{ju_2})\right| = \left|H(e^{-ju_1}, e^{ju_2})\right| = \left|H(e^{-ju_1}, e^{-ju_2})\right|$$
$$= \left|H(e^{ju_1}, e^{-ju_2})\right| \qquad (1.13)$$

$$(N\text{-D}): \quad \left|H(e^{ju_1}, \ldots, e^{ju_n})\right| = \left|H(e^{-ju_1}, \ldots, e^{ju_n})\right| = \left|H(e^{ju_1}, e^{-ju_2}, \ldots, e^{ju_n})\right|$$
$$= \ldots = \ldots = \ldots = \left|H(e^{ju_1}, e^{ju_2}, \ldots, e^{-ju_n})\right|$$
$$\vdots$$
$$= \left|H(e^{-ju_1}, e^{-ju_2}, \ldots, e^{-ju_n})\right| \qquad (1.14)$$

The above equations clearly illustrate that the (N-D) case is simply an extension of the (2-D) case.

1.3 SURVEY OF PREVIOUS WORK

The theory of (2-D) digital filters stems from classical network synthesis and (1-D) digital filter theory. Both of these are very well-developed, while, by comparison, (2-D) digital filter theory itself remains in its infancy.

The introduction of the Fast Fourier Transform (FFT) algorithm in 1965 was a turning point in the short history of (2-D) digital filters' design and realisation [12]. Up to that time, most implementations made use of direct convolution, which is the basic form of digital filtering realisation. However, this suffered from poor computational efficiency and difficult design procedures, and the Fourier Transformation realisation made possible through the FFT was thus universally adopted in preference, since it offered better computational efficiency and reduced the frequency domain design problem to the simple task of specifying an array of frequency-weighting coefficients.

Alas, the Fourier Transformation realisation was limited to applications that did not involve abnormally large arrays of data, as the question of computer storage capacity then came into play. Also, processing on small machines and in real-time became the order of the day, and interest began to be shown in the third basic realisation philosophy, that of <u>recursion</u>, since this was the obvious answer to both real-time and small machine processing. Recursion filters are nowadays initially specified by either the frequency characteristics or the impulse response, and the design problem that remains, in the frequency domain or the space domain respectively, is to determine the filter coefficients, that would correspond to the given specification.

Darby and Davies [13] in 1968 published one of the first papers dealing with possible realisations for these filters specified by frequency characteristics. They proposed the use of the inverse (2-D) discrete Fourier transform for obtaining the impulse response coefficients used later in a convolution realisation; they also explored the desirability of choosing an appropriate window function for possible reduction of the Gibbs phenomenon resulting from using a truncated description of the filter frequency characteristics.

Huang [14] contributed to this by further showing that (2-D) window functions can be obtained from (1-D) window functions.

In 1972, McClellan [15,16] introduced a frequency domain method which transforms a (1-D) FIR filter into a (2-D) FIR filter by a change of variables. The method is very simple and useful but is limited to the class of filters whose frequency response is constant over large areas of the (2-D) frequency plane.

Also in 1972 Hu and Rabiner [17,18] extended the (1-D) frequency sampling technique [42]. This reduced the design of a (2-D) FIR filter to a linear programming problem, thus offering flexibility and the possibility of optimal solutions. But the method was expensive in terms of the computation required, and designs published using this approach were limited to the eighth degree in z_1 and z_2 for optimal designs and the twenty-fourth degree for sub-optimal designs.

Mersereau and Dudgeon [19] introduced a method in 1974 for representing (2-D) sequences as (1-D) sequences. This resulted in an FIR design method whereby the (2-D) design problem is recast as a (1-D) design problem with multiple stop and pass regions. This approach offers promise for efficient (2-D) FIR filter design since it allows the application of highly efficient (1-D) design algorithms. Additional work is necessary, however, because the technique does not produce good designs.

Recursive (IIR) (2-D) digital filters were not seriously investigated until in 1972 the stability condition of such filters was first derived. In that year, Shanks et al. [20] and independently Huang [21] published equivalent conditions and tests for stability of a (2-D) recursive digital filter. Anderson and Jury [22,23] and also Maria and Fahmy [24] simplified these results in 1973 by putting the stability tests in terms of the root locations or positivity of a set of polynomials in one variable. While this represented a considerable simplification, the procedure is still computationally difficult except for low-order cases [23,25].

An alternative test for stability was investigated by

Pistor [26] and Ekström [25,27] in 1974. The approach is to establish the stability criteria in terms of a transformation of the complex frequency response. The method still has unanswered questions concerning the approximations which must be made for a practical algorithm; however, it appears to provide not only a test for stability but a decomposition of an unstable filter into stable filters recursing in different directions.

E.L. Hall introduced the separable product technique [28,29] which enables the design of any two-dimensional rectangular cut-off boundary type filter by the use of two one-dimensional recursive filters in cascade. This was the first design method for (2-D) recursive digital filters.

The first transformation technique for (2-D) recursive digital filters was due to Shanks et al., in 1972 [20]. This technique takes a stable (1-D) continuous filter and uses a transformation to rotate the amplitude response in (2-D). A bi-linear transformation is then used on each variable to produce a (2-D) digital transfer function. Unfortunately there is no guarantee of stability and the approach suffers from warping effects of the bi-linear transformation on the frequency response. Costa and Venetsanopoulos [30] devised a method of using this approach to produce circularly symmetric filters while guaranteeing stability. Their technique produces filters of high order, however.

A design method used by Bernabo et al. [31] makes use of the transformation technique of McClellan [15] to a (1-D) zero phase recursive digital filter followed by a decomposition technique of Pistor [26] in order to obtain four one-quadrant recursive digital filters each recursing in a different direction. The obtained filter is of zero phase.

Shanks et al. [20] also extended a (1-D) time domain approximation technique to (2-D). A (2-D) recursive digital filter is obtained which approximates a desired (2-D) impulse response at a finite number of points. This has the advantage of not involving transcendental functions in the computation since the frequency domain is not used.

However, stability of the result cannot be guaranteed. An alternative spatial design technique for (2-D) recursive digital filters has been proposed by Bordner [32] whereby a given discrete, finite-time, impulse response is approximated by the response of a recursive filter. The approximation is derived on the basis of the minimum squared error between the desired impulse response and the impulse response of the recursive filter model; however, the model response is constrained to be square summable over $[0, \infty]$, thus guaranteeing the stability of the resulting filter.

M. Lal [33] has proposed another spatial design technique, where a given (2-D) impulse response over a specified sector is divided into N convenient groups and each group of samples is approximated by a second order (2-D) recursive filter in the least square sense with stability constraint on the coefficients of the second order filter in order to obtain stable filter.

In 1974, Maria and Fahmy [34,35] extended an iterative (1-D) IIR design technique due to Deczky [36]. They proposed an algorithm to obtain an approximation to a (2-D) frequency domain specification while giving some assurance that a stable filter would result [37]. This was achieved by checking stability at every iteration and reducing the step size if necessary to prevent crossing the stability boundary. Because of the difficulty in checking stability, the type of filter that can be designed is limited to a cascade of low order sections. In addition, the computation required at each iteration is large, so that the total order of a design is limited. The largest filter design published using this method was of second degree in each variable.

The interest shown in recent years in the stability of multivariable functions has not been paralleled by a similar interest in the design of such functions. However, these functions are finding increasing applications in the analysis and synthesis problems of various systems, and the time for some contributions in understanding and solving the design problems of such (N-D) functions has come.

Justice and Shanks [38] have studied the stability of (N-D) digital filters. Stability tests for such filters were advanced by Anderson and Jury [39], Bose and Jury [40] and Bose and Kamat [41].

1.4 OUTLINE OF THESIS

In Chapter 2, the concept of stability is defined for (2-D) recursive digital filters. Stability conditions for these filters are discussed and methods for determining stability are reviewed. The above discussion is then extended to stability conditions and tests for (N-D) recursive digital filters.

Several stabilisation procedures for unstable recursive digital filters are examined in Chapter 3 and one of these procedures is extended to (N-D).

In Chapter 4, frequency domain design techniques for (2-D) and (N-D) recursive filters are considered. A new (2-D) design technique for recursive filters is presented and two new design techniques for (N-D) filters are also proposed. The first of them is applicable to non-recursive, zero phase filters and the second to the recursive case.

Chapter 5 is concerned with spatial design techniques for (2-D) recursive digital filters. These are discussed in detail and some of the existing methods for (2-D) are extended to (N-D).

The final chapter summarizes the work of the thesis and a number of avenues are suggested in which further research may be conducted.

CHAPTER TWO

STABILITY

2.1 INTRODUCTION

The term stability is generally used to indicate that convolving a filter with some bounded input sequences always yields a bounded output. As can be seen from equations (1.3) and (1.4), where past output values in recursive filters are used by the recursion algorithm in calculating the present one, such value can become arbitrarily large independent of the size of the input values. Recursive filters can therefore be unstable.

To understand the various known conditions and tests for stability of the (2-D) recursive filters, some definitions relating to signal-representing sequences must first be defined.

A function $i(m_1,m_2)$ in two variables m_1, m_2 is absolutely bounded when:

$$|i(m_1,m_2)| \leqslant M < \infty \tag{2.1}$$

and is absolutely summable when:

$$\sum_{m_1} , \sum_{m_2} |i(m_1,m_2)| \leqslant N < \infty \tag{2.2}$$

for all integer pairs (m_1,m_2), where M and N are positive real numbers.

One kind of stability to consider is bounded input bounded output stability. With the input array $i(m_1,m_2)$ absolutely bounded, the question of what restrictions are to be placed on the filter's impulse response $f(m_1,m_2)$ to ensure that the output $o(m_1,m_2)$ is also absolutely bounded, arises. Remembering that the output of a system $o(m_1,m_2)$ may be expressed as the convolution of the input $i(m_1,m_2)$ and the impulse response of the network function:

$$o(m_1,m_2) = \sum_{p_1}\sum_{p_2} f(p_1,p_2)\ i(m_1-p_1+1,\ m_2-p_2+1) \qquad (2.3)$$

and applying Schwarz's inequality, gives:

$$|o(m_1,m_2)| \leqslant \sum_{p_1}\sum_{p_2} |f(p_1,p_2)|\ |i(m_1-p_1+1,\ m_2-p_2+1)| \qquad (2.4)$$

but since $|i| \leqslant M$ for all integer values of its arguments, the inequality reduces to:

$$|o(m_1,m_2)| \leqslant M \sum_{p_1}\sum_{p_2} |f(p_1,p_2)| \qquad (2.5)$$

which restricts the impulse response to an absolutely summable function. This has been shown to be a necessary and sufficient condition for stability [49]. So that, if the impulse response $f(p_1,p_2)$ is absolutely summable, and the input array $i(m_1,m_2)$ is absolutely bounded, then the output $o(m_1,m_2)$ is also absolutely bounded.

Another kind of stability is summable input, summable output stability. Here again the restrictions on the filter's impulse response $f(m_1,m_2)$, that ensure the summability of the output $o(m_1,m_2)$ with summable input $i(m_1,m_2)$ are derived through applying Schwarz's inequality to the convolutional sum of equation (2.3):

$$o(m_1,m_2) = \sum_{p_1}\sum_{p_2} f(p_1,p_2)i(m_1-p_1+1,\ m_2-p_2+1)$$

$$\sum_{m_1}\sum_{m_2}\left|o(m_1,m_2)\right| = \sum_{m_1}\sum_{m_2}\left|\sum_{p_1}\sum_{p_2} f(p_1,p_2)i(m_1-p_1+1,m_2-p_2+1)\right|$$

$$\leqslant \sum_{m_1}\sum_{m_2}\sum_{p_1}\sum_{p_2}\left|f(p_1,p_2)\right|\left|i(m_1-p_1+1,\ m_2-p_2+1)\right|$$

$$\leqslant \sum_{p_1}\sum_{p_2}\left|f(p_1,p_2)\right|\sum_{m_1}\sum_{m_2}\left|i(m_1-p_1+1,\ m_2-p_2+1)\right|$$

$$\leqslant N\sum_{p_1}\sum_{p_2}\left|f(p_1,p_2)\right| \qquad (2.6)$$

and therefore a sufficient condition for the filter to be considered stable is for its impulse response to be absolutely summable. Generally, the stability of a (2-D) recursive digital filter is determined by the coefficient of the denominator $B(z_1,z_2)$ of the z-transfer function of the filter when such filter is expressed in the form of equation (1.5). However, testing the stability then is difficult because the fundamental theorem of algebra is not applicable to two-variable functions: namely, denominator factorisation is not always possible. Testing the stability by finding the poles of the z-transfer function, as in the (1-D) case, is hence not possible, nor is filter stabilisation by the replacement of the poles in the instability region by poles in conjugate reciprocal positions with respect to the unit circle. A direct extension of the condition for stability in the (1-D) case to the (2-D) case was proposed by Shanks [20]. It can be stated as follows:

2.2 SHANKS THEOREM [20]

Theorem 2.1: Given that $B(z_1,z_2)$ is a polynomial in (z_1,z_2), a necessary and sufficient condition for the coefficients of the expansion of $F(z_1,z_2) = 1/B(z_1,z_2)$ in positive powers of z_1,z_2 to converge absolutely, and hence for $f(m_1,m_2)$ to be absolutely summable, is:

$$B(z_1,z_2) \neq 0 \qquad \text{for} \qquad |z_1| \leqslant 1 \cap |z_2| \leqslant 1$$

The above condition suggests a test procedure for checking the stability of the filter by finding the continuum of (z_1,z_2) values for which $B(z_1,z_2) = 0$. This is done by assigning values to the variable z_1 and finding the roots of $B(z_1,z_2) = 0$ as a function of z_2. For stability, it follows from theorem (2.1) that all roots of z_2 must be greater than one in magnitude when $|z_1|$ is less than one.

2.3 HUANG STABILITY TEST [21]

Shanks' condition for stability as stated above is very tedious to apply, since it requires the mapping of an infinite number of points from the z_1-plane into the z_2-plane. However, a considerably simplified version has been arrived at by Huang:

Theorem 2.2 (Huang): A causal recursive filter with a z-transfer function:

$$H(z_1,z_2) = \frac{A(z_1,z_2)}{B(z_1,z_2)}$$

is stable if and only if (iff):

(1) the map of $R1 \equiv (z_1, \; |z_1| = 1)$ in the z_2-plane according

to $B(z_1,z_2) = 0$ lies outside

$$d2 \equiv (z_2; |z_2| \leqslant 1); \text{ and}$$

(2) no point in $d1 = (z_1, |z_1| \leqslant 1)$ maps into the point $z_2 = 0$ by the relation $B(z_1,z_2) = 0$.

To check the stability using theorem (2.2), R_1 is mapped into the z_2-plane according to $B(z_1,z_2) = 0$ and the resulting image tested to determine whether it lies outside d2. Also, $B(z_1,0) = 0$ must be solved to find whether there are any roots with magnitude less than 1.

2.4 ANSELL STABILITY THEOREM [43]

Theorem (2.2) can be reduced to a stability test involving only a finite number of steps. However, as will be shown, this can still be very tedious.

Using the change of variables:

$$s_1 = \frac{1-z_1}{1+z_1} \tag{2.7}$$

and:

$$s_2 = \frac{1-z_2}{1+z_2} \tag{2.8}$$

and letting:

$$H(z_1,z_2) = \frac{A(s_1,s_2)}{B(s_1,s_2)} \tag{2.9}$$

where A and B are polynomials in s_1 and s_2, theorem (2.2) can be restated as follows:

Theorem 2.3: (Ansell) [43]: the causal recursive filter $H(z_1,z_2)$ is stable if and only if:

(1) In all real finite u_1, the complex polynomial in s_2, $B(ju_1, s_2)$ has no zeros in Re $s_2 \geqslant 0$, and

(2) The real polynomial in s_1, $B(s_1, 1)$, has no zeros in Re $s_1 \geqslant 0$.

Moreover,

<u>Theorem 2.4</u> (Ansell): Condition (1) of theorem (2.3) can be derived in another form. Let $B(ju_1, ju_2)$ be expressed, for real u_1 and u_2, as:

$$B(ju_1, ju_2) = \beta_0(u_1)u_2^n + \beta_1(u_1)u_2^{n-1} + \dots + \beta_n(u_1)$$
$$+ j\,[\alpha_0(u_1)u_2^n + \alpha_1(u_1)u_2^{n-1} + \dots + \alpha_n(u_1)] \quad (2.10)$$

where $\alpha_i(u_1)$ and $\beta_i(u_1)$ are real polynomials in u_1 and where neither $\alpha_0(u_1)$ nor $\beta_0(u_1)$ is identically zero. Also, define $\gamma_{k,\ell}(u_1)$ as:

$$\gamma_{k,\ell} = \alpha_k\beta_\ell - \alpha_\ell\beta_k \quad (2.11)$$

for $0 \leqslant \ell,k \leqslant n$, setting α's and β's not present in $B(ju_1,ju_2)$ equal to zero. Then, with $D(u_1)$ denoting the n x n symmetrical polynomial matrix whose typical element $D_{ij}(u_1)$ $(1 \leqslant i,\ j \leqslant n)$ is the sum of all those $\gamma_{k,\ell}(u_1)$ $(0 \leqslant k,\ell \leqslant n)$ for which both

$$k + \ell = i + j - 1$$

and (2.12)

$$\ell - k \geqslant |i-j|$$

are satisfied, the n successive principal minors of $D(u_1)$ must be positive for all real u_1.

In the above, Sturm's [44] method can be used to test whether each minor of $D(u_1)$ is positive for all real u_1.

It can thus be seen that while Ansell's test requires only a finite number of mappings, the mathematics involved are still tedious.

2.5 THE ANDERSON AND JURY STABILITY TEST [22]

Huang's simplification of the stability test rests on the fact that the denominator of the z-transfer function $B(z_1,z_2) \neq 0$ for $|z_1| \leqslant 1 \cap |z_2| \leqslant 1$ iff the following two conditions hold:

$$B(z_1,0) \neq 0 \qquad |z_1| \leqslant 1 \tag{2.13}$$

$$B(z_1,z_2) \neq 0 \qquad |z_1| = 1 \cap |z_2| \leqslant 1 \tag{2.14}$$

Anderson and Jury's stability test is divided into two parts: first, checking condition (2.13) for the values of B with z_1 restricted, and, second, checking condition (2.14) for the values of B with both z_1 and z_2 restricted:

2.5.1 Denominator Examination with z_1 limited

Two methods for checking condition (2.13) have been proposed by Anderson and Jury [22]. The first of these is based on the use of the Schur-Cohn matrix [45]. This matrix is square, Hermitian, of size equal to the degree of $B(z_1,0)$, and with elements which are simple functions of the coefficients of $B(z_1,0)$. The matrix is negative definite if and only if $B(z_1,0)$ has all its zeros in $|z_1| > 1$. The negative definiteness can be established by examining the sign of the leading principal minors of the matrix.

The Schur-Cohn criterion for checking condition (2.13) will now be discussed in detail.

Suppose that:

$$f(z) = \sum_{i=0}^{n} a_i z^i \qquad (a_n \neq 0) \qquad (2.15)$$

where the associated n x n Hermitian matrix $C = (\gamma_{ij})$ is defined by:

$$\gamma_{ij} = \sum_{p=1}^{i} (a_{n-i+p}\, a^*_{n-j+p} - a^*_{i-p}\, a_{j-p}), \qquad i \leqslant j \qquad (2.16)$$

then:

(i) the number of zeros z_i of $f(z)$ for which $|z_i| < 1$ and for which z_i^{-1} is not also a zero is the number of positive eigen values of C;

(ii) the number of zeros z_i for which $|z_i| > 1$ and for which z_i^{-1} is not also a zero is the number of negative eigen values of C, and;

(iii) the number of zeros z_i for which either $|z_i| = 1$ or z_i^{-1} is also a zero (or both) is the nullity of C.

For condition (2.13) to hold, with $B(z_1,0)$ a real polynomial and C a real matrix, all eigen values of C must be negative definite. This is so if and only if odd-order leading principal minors are negative and even-order leading principal minors are positive.

An alternative procedure for checking condition (2.13) can be derived from the theorem of Jury [47], which is based on forming a sequence of polynomials.

Theorem (2.5):

Suppose that

$$f(z) = \sum_{i=0}^{n} a_i z^i \tag{2.15}$$

then, a sequence of polynomials can be defined as follows:

$$F_o(z) = f(z) = \sum_{i=0}^{n} a_i^{(o)} z^i, \quad F_1(z) = \sum_{i=1}^{n-1} a_i^{(1)} z^i,$$

$$F_2(z) = \sum_{i=0}^{n-2} a_i^{(2)} z^i$$

$$\vdots$$

$$F_j(z) = \sum_{i=0}^{n-j} a_i^{(j)} z^i \tag{2.17}$$

by $F_{j+1}(z) = a_o^{*(j)} F_j(z) - a_{n-j}^{(j)} F_j^*(z)$ (2.18)

where

$$F_j^*(z) = a_o^{*(j)} z^{n-j} + a_1^{*(j)} z^{n-j-1} + \ldots + a_{n-j}^*$$

$$= \sum_{i=0}^{n-j} a_i^{*(j)} z^{n-j-i} \tag{2.19}$$

(Thus $F_j^*(z)$ is obtained from $F_j(z)$ by coefficients reversal and conjugation.)

Setting $\delta_j = F_j(o)$ and $P_j = \delta_1 \delta_2 \ldots \delta_j$, then:

(i) all zeros of $f(z)$ lie inside $|Z| \leqslant 1$ iff $P_j < 0$ for all j;

(ii) all zeros of $f(z)$ lie outside $|Z| \leqslant 1$ iff $P_j > 0$ for all j, or equivalently, $\delta_j > 0$ for all j; and

(iii) if all P_j are non-zero, the number of negative P_j is the number of zeros of $f(z)$ inside $|Z|\leqslant 1$ and the number of positive P_j is the number of zeros outside $|Z|\leqslant 1$.

It should be noted that (ii) above covers the requirement for condition (2.13).

2.5.2 Denominator Examination with both z_1 and z_2 limited

Broadly, checking the above condition falls into two distinct steps. First, a Schur-Cohn test will be applied. Second, the positiveness of a number of polynomials on $|z_1| = 1$ will be checked.

Step 1: Applying the Schur-Cohn test:

To check condition (2.14), $f(z)$ in (2.15) is replaced by $B(z_1,z_2)$ written as a polynomial in z_2:

$$B(z_1,z_2) = \sum_{j=0}^{q}\Big(\sum_{i=0}^{p} b_{ij}z_1^i\Big)z_2^j \tag{2.20}$$

The coefficient a_j is therefore $\sum_{i=0}^{p} b_{ij}z_1^i$ and the matrix C is q x q, with entries which from (2.16) are seen to be polynomials in z_1 and z_1^* with real coefficient (matrix C is Hermitian).

The condition $B(z_1,z_2) \neq 0$ for $|z_1| = 1$, $|z_2|\leqslant 1$ holds if and only if for all $|z_1| = 1$, C is negative definite, i.e. if and only if the leading principal minors C have appropriate signs. These principal minors being linear combinations of products of the γ_{ij} are themselves polynomial in z_1 and z_1^* with real coefficients; they are also real, since C is Hermitian. By setting $z_1^* = z^{-1}$, on

$|z_1| = 1$, the polynomials will have the form $\sum_{j=0}^{N} C_j(z_1^j + z_1^{-j})$.

Such polynomials are termed self-inverse, i.e. if $z_1 = z_{1\alpha}$ is a zero so is $z_1 = z_{1\alpha}^{-1}$.

Step 2:

Step 2 involves checking the positiveness of a self inverse polynomial on $|Z_1| = 1$. Considering the self-inverse polynomial,

$$f(z_1) = \sum_{j=0}^{N} C_j(z_1^j + z_1^{-j}) \tag{2.21}$$

where the C_j are real constants, some immediate necessary conditions are obtained by putting $z_1 = 1, -1$:

$$\sum_{j=0}^{N} C_j > 0 \qquad \sum_{j=0}^{N} (-1)^j C_j > 0 \tag{2.22}$$

There are two techniques for checking sign definiteness of a self inverse polynomial.

The first approach makes use of using $z_i = \exp(i\theta)$ in (2.21) which results in:

$$f(\cos\theta) = 2\sum_{j=0}^{N} C_j \cos j\theta \tag{2.23}$$

Positiveness is to be checked for $0 \leqslant \theta < 2\pi$ (or $-\pi \leqslant \theta < \pi$). But since $\cos j\theta = \cos(-j\theta)$, it would be sufficient to examine the $0-\pi$ range. The following change of variable will be made:

$$X = \cos\theta \qquad T_k(X) = \cos k\theta \tag{2.24}$$

where $T_k(x)$ is the k^{th} Chebyshev polynomial of the first kind, defined recursively by

$$T_{k+1} = 2xT_k(x) - T_{k-1}(x), \quad T_1(x) = x \qquad T_o(x) = 1 \tag{2.25}$$

Then the requirement will be:

$$g(x) = \sum_{j=0}^{N} C_j T_j(x) > 0 \tag{2.26}$$

for $-1 \leqslant x \leqslant 1$. $T_j(x)$ is a polynomial in x of degree j; therefore g(x) has the form:

$$g(x) = \sum_{j=0}^{N} d_j x^j \tag{2.27}$$

for some real d_j. Positiveness can be checked by forming a Sturm chain. An example of this can be found in [44].

An alternative approach for checking positiveness is based on the determination of the zero distribution of $f(z_1)$ in equation (2.21). Because $f(z_1)$ is self-inverse, there are as many zeros of $z_1^N f(z_1)$ inside $|z_1| < 1$ as outside. Therefore, $f(z_1)$ is positive on $|z_1| = 1$ if and only if $f(1) > 0$ (or $f(z_1)$ is positive at any one point of $|z_1| = 1$) and $z_1^N f(z_1)$ has N zeros inside $|z_1| < 1$ (for then $z_1^N f(z_1)$ has N zeros outside, and thus no zeros on $|z_1| = 1$).

Unfortunately, the Schor-Cohn matrix for a self-inverse polynomial is zero, and the other procedure as mentioned before based on setting up a sequence of polynomials leads to a zero polynomial at the first recursion. Hence neither procedure, as it stands, is of help. However, the following result is of assistance:

<u>Theorem 2.6</u>: [45,47] Let $f(z_1)$ be as in equation (2.21). The number of zeros of $g(z_1) = z_1^N f(z_1)$ in $|z_1| < 1$ is the same as the number of zeros of $z_1^{2N-1} g'(1/z_1)$ in $|z_1| < 1$, where $g'(1/z_1)$ is obtained by differentiating $g(z_1)$ with respect to z_1 and substituting z_1^{-1} for z_1.

Assuming that $z_1^{2N-1} g'(1/z_1)$ has neither zeros reciprocal with respect to $|z_1| = 1$, nor zeros on $|z_1| = 1$, the Schur-Cohn criterion will yield the number of zeros inside $|z_1| < 1$; so may the procedure based on generating a recursive set of polynomials. If, on the other hand, $z_1^{2N-1} g'(1/z_1)$ does have reciprocal zeros or zeros on the unit circle, the situation must be analysed further. In view of the easily established relation

$$z_1 g'(z_1) + z_1^{2N-1} g'(1/z_1) = f(z_1) \tag{2.28}$$

it follows that such zeros of $z_1^{2N-1} g'(1/z_1)$ are also zeros of $f(z_1)$.

Accordingly the highest common factor of $z_1^{2N-1} g'(1/z_1)$ and $f(z_1)$ can be found, and then its zero properties can in turn be studied, it too being self-inverse. Proceeding in this way will result in determining the number of zeros of $z_1^{2N-1} g'(1/z_1)$ and therefore of $z_1^N f(z_1)$ inside $|z_1| < 1$, to conclude whether or not $f(z_1)$ is positive.

The Anderson and Jury method for testing the stability can be summarised as follows:

1) For checking condition (2.13):

(i) Use the bilinear transformation of $B(z_1,0) = 0$ and apply the Hurwitz method to the transformed polynomial.

(ii) An alternative test involves forming the Schur-Cohn matrix from $B(z_1,0) = 0$ and testing this for positiveness.

2) For checking condition (2.14) two successive tests as follows are needed:

(i) Apply a Schur-Cohn test to $B(z_1,z_2) = 0$ to get the self-inverse polynomials and check the positiveness of these polynomials.

(ii) Second, the positiveness of a number of polynomials of $B(z_1,z_2) = 0$ on $|Z_1| = 1$ should be checked.

2.6 MARIA AND FAHMY METHOD FOR TESTING STABILITY [24]

A method has been introduced by Maria and Fahmy [24] for checking condition (2.13) of Huang's criterion. This method, based on modification of Jury's table [46], is as stated below:

Theorem 2.7: Let $f(z)$ be the n^{th} degree polynomial given by:

$$f(z) = a_o + a_1 z + a_2 z^2 + \ldots + a_n z^n \tag{2.29}$$

where the coefficients a_i, $i = 0,1,\ldots,n$ are complex numbers. The roots of $f(z)$ are outside the unit circle if and only if:

$$b_o < 0, \quad c_o > 0, \quad d_o > 0, \ldots, \; g_o > 0, \ldots, \; t_o > 0 \tag{2.30}$$

where b_o, $c_o, \dots,$ t_o are obtained from the modified Jury table formed as follows:

$$
\begin{array}{llllll}
z^o & z^1 & z^2 \quad \cdots & z^{n-2} & z^{n-1} & z^n \\
a_o & a_1 & a_2 & a_{n-2} & a_{n-1} & a_n \\
a^*_n & a^*_{n-1} & a^*_{n-2} & a^*_2 & a^*_1 & a^*_o \\
b_1 & b_1 & b_2 & b_{n-2} & b_{n-1} & \\
b^*_{n-1} & b^*_{n-2} & b^*_{n-3} & b^*_1 & b^*_o & \\
c_o & c_1 & c_2 & c_{n-2} & & \\
c^*_{n-2} & c^*_{n-3} & & c^*_o & & \\
\vdots & & & & & \\
r_o & r_1 & & & & \\
r^*_1 & r^*_o & & & & \\
t_o & & & & &
\end{array}
\qquad (2.31)
$$

where:

$$
b_k = \begin{vmatrix} a_o & a_{n-k} \\ a^*_n & a^*_k \end{vmatrix}, \qquad c_k = \begin{vmatrix} b_o & b_{n-1-k} \\ b^*_{n-1} & b^*_k \end{vmatrix} \qquad (2.32)
$$

and a^*_k is the complex conjugate of a_k.

To check the first condition of Huang's Theorem (2.13) using the above results, it should be noted that that condition is satisfied if and only if the roots of z_2 in $B(z_1,z_2)\Big|_{|z_1|=1} = 0$ are outside the unit circle $|z_2| = 1$. To test this equivalent condition, $B(z_1,z_2)$ should be written in the form of

$$
B(z_1,z_2) = [a_j(z_1)]z_2^{m_2} + [a_{j-1}(z_1)]z_2^{m_2-1} + \dots + [a_o(z_1)] \qquad (2.33)
$$

where:

$$a_j(z_1) = \sum_{i=0}^{p} b_{ij} z_1^j \tag{2.34}$$

In such a form $B(z_1,z_2)$ is viewed as a polynomial in the single variable z_2 with the coefficients being functions in z_1. The above table should be constructed for this $B(z_1,z_2) = 0$.

The computation of the entries of the modified Jury table is considerably simplified because:

(i) all b_{ij}'s are real, and thus $a_j^*(z_1) = a_j(z_1^*)$, and

(ii) z_1 is restricted to the boundary $|z_1| = 1$, and thus $(z_1 z_1^*)^\ell = 1$ for all ℓ.

All the entries of the table will be functions of z_1 with b_o, c_o, ..., t_o taking the form

$$g_o = g_{oo} + g_{o1}(z_1^* + z_1) + g_{o2}(z_1^{*2} + z_1^2) + \ldots \tag{2.35}$$

Putting $z_1 = X + jY$ and noting that $|z_1| = 1$, the coefficients b_o, c_o, ..., g_o, ..., t_o can be expressed as functions of the real variable X by using the following substitutions for $(z_1^{*k} + z_1^k)$, k = 1, 2, ..., :

$$\begin{aligned}
(z_1^* + z_1) &= 2X \\
(z_1^{*2} + z_1^2) &= 4X^2 - 2 \\
(z_1^{*3} + z_1^3) &= 8X^3 - 6X \\
(z_1^{*4} + z_1^4) &= 16X^4 - 16X^2 + 2
\end{aligned} \tag{2.36}$$

Thus the test for condition (2.13) may be represented as a modified

Jury's criterion which takes the following form:

For $\quad -1 \leqslant X \leqslant +1,$

$$b_o(X) < 0$$

$$c_o(X) > 0, \quad d_o(X) > 0, \quad \ldots, \quad t_o(X) > 0$$

This criterion can be further simplified as:

(i) $\quad b_o(0) < 0, \quad c_o(0) > 0, \quad \ldots, \quad t_o(0) > 0,$ and

(ii) $\quad$ the polynomials $b_o(X)$, $c_o(X)$, ..., $t_o(X)$ have no real roots in the interval $|X| \leqslant 1$.

Sturm's test can be used to check condition (ii) above.

2.7 STABILITY OF (N-D) DIGITAL FILTERS

Stability of (N-D) filters has been studied by Justice and Shanks [38], Anderson and Jury [39], Bose and Jury [40] and Bose and Kamat [41].

Here the concept of the stability of (N-D) digital filters is defined and the conditions for stability of these systems are discussed.

First, though, some definitions and preliminary theorems that will be of later use are outlined.

A function $i(m_1, \ldots, m_n)$ of n variables $(m_1, \ldots, m_n)$ is called absolutely bounded when:

$$|i(m_1, \ldots, m_n)| \leqslant M < \infty \text{ for all } (m_i, \; i = 1, n) \qquad (2.37)$$

and absolutely summable when:

$$\sum_{m_1} \cdots \sum_{m_2} |i(m_1, \ldots, m_n)| \leqslant N < \infty \text{ for all } (m_i, i = 1, n) \quad (2.38)$$

where M and N are positive real numbers.

If the input $i(m_1, \ldots, m_n)$ is an absolutely bounded (N-D) function:

$$|i(m_1, \ldots, m_n)| \leqslant M < \infty \quad \text{for all } (m_i, i = 1, n) \qquad (2.39)$$

where M is a positive real number, then to find the restriction to be placed on the filter impulse response $f(m_1, \ldots, m_n)$ to ensure that the output $O(m_1, \ldots, m_n)$ is also absolutely bounded, the convolutional sum:

$$O(m_1, \ldots, m_n) = \sum_{p_1} \cdots \sum_{p_n} f(p_1, \ldots, p_n) i(m_1-p_1+1, \ldots, \\ \ldots, m_n-p_n+1) \qquad (2.40)$$

is considered. Applying Schwarz's inequality to this gives:

$$|o(m_1,\ldots,m_n)| \leqslant \sum_{p_1} \cdots \sum_{p_n} |f(p_1,\ldots,p_n)| \, |i(m_1-p_1+1,\ldots,m_n-p_n+1)| \qquad (2.41)$$

but since $|i| \leqslant M$ for all integer values of its arguments, it can be written as:

$$|o(m_1,\ldots,m_n)| \leqslant M \sum_{p_1} \cdots \sum_{p_n} |f(p_1,\ldots,p_n)| \qquad (2.42)$$

Thus, if the filter impulse response $f(m_1,\ldots,m_n)$ is absolutely summable, and the input array $i(m_1,\ldots,m_n)$ is absolutely bounded, then the output $o(m_1,\ldots,m_n)$ is also absolutely bounded. It has been proven [38] that this is a necessary and sufficient condition for stability. Further, it has been shown that when the input $i(m_1,\ldots,m_n)$ is absolutely summable, the output is also absolutely summable, only if the filter impulse response is summable.

Theorem 2.8:

A convolutional filter f as defined by equation (1.10) is stable iff:

$$\sum_{m_1=-\infty}^{\infty}\cdots\sum_{m_n=-\infty}^{\infty}\left|f_{m_1,\ldots,m_n}\right| < \infty \tag{2.43}$$

The above means that the z-transform of f is absolutely convergent for all $(z_i,\ i = 1,\ 2,\ \ldots,\ n) \in R_n$ where

$$R_n = \left\{(z_i,\ i = 1,2,\ldots,n):\ \bigcap_{i=1}^{n}|z_i| = 1\right\} \tag{2.44}$$

The proof of this can be found in [38] and [48].

2.7.1 Stability Constraints on (N-D) recursive filters

After discussing the concept of stability for (N-D) filters and deriving the constraints on such filters' impulse responses, the stability of (N-D) recursive filters is now briefly reviewed:

Theorem 2.9: [38]

A causal recursive filter $F(z_1,\ldots,z_n) = 1/B_1(z_1,\ldots,z_n)$ is stable if and only if there exists a stable filter ${}^1f$ such that ${}^1f * {}^1b = \delta$.

The above theorem can be re-stated as:

Theorem 2.10:[38,40]

The causal recursive filter

$$F(z_1,\dots,z_n) = 1/B(z_1,\dots,z_n) = \sum_{m_1=0}^{\infty}\cdots\sum_{m_n=0}^{\infty} f_{m_1,\dots,m_n} z_1^{m_1}\dots z_n^{m_n}$$

$$= 1/\left(\sum_{m_1=0}^{M_1}\cdots\sum_{m_n=0}^{M_n} b_{m_1,\dots,m_n} z_1^{m_1}\dots z_n^{m_n}\right) \qquad (2.45)$$

where:

$$\sum_{m_1=0}^{M_1}\cdots\sum_{m_n=0}^{M_n} \left| b_{m_1,\dots,m_n} \right| < \infty \qquad (2.46)$$

is stable iff:

$$B(z_1,\dots,z_n) \neq 0 \text{ for all } (z_i,\ i = 1,2,\dots,n) \in D_1 \qquad (2.47)$$

where:

$$D_1 = \left\{ (z_i,\ i = 1,2,\dots,n):\ \bigcap_{i=1}^{n} |z_i| \leqslant 1 \right\} \qquad (2.48)$$

It has also been proved [39] that the above condition is equivalent to the conditions:

$$B_j(z_1,z_2,\dots,z_j) \neq 0 \quad \left\{ \bigcap_{i=1}^{j-1} |z_i| = 1 \right\} \left\{ |z_j| \leqslant 1 \right\}$$

for j = 1, 2, ..., n where $B_j(z_1,\dots,z_j)$ is obtained by setting $z_{j+1} = z_{j+2} = \dots = z_n = 0$ in $B(z_1,\dots,z_n)$.

The test procedure is carried out by the repeated applications of the extended Jury's theorem [22] to (N-D) [39], to determine the content of a system of polynomial inequalities in a single variable. Details of this are omitted here for brevity, and are well-documented elsewhere [39,40,41].

CHAPTER THREE

STABILISATION METHODS

Having discussed, in the last chapter, several methods for testing the stability of (2-D) and (N-D) recursive digital filters, a number of techniques for stabilising unstable (2-D) recursive digital filters are now reviewed. One of these is then generalised to the (N-D) case by a somewhat thorough mathematical treatment.

3.1 SHANKS STABILISATION METHOD [20]

This method is based on a conjecture that, too, is due to Shanks. It is a direct extension of a well-established (1-D) method [51] based on some properties of the planar least square inverse of a matrix. Before reviewing the method in detail, some useful definitions and preliminary theorems relevant to its understanding are given.

Definition 3.1

A minimum-phase (1-D) discrete sequence is one with a z-transform having no zeros inside the z-plane unit circle.

A minimum-phase array $a(m_1,m_2)$ is defined as one that would satisfy both of the following two conditions:

(i) When the spectrum $A(u_1,u_2)$, of the (2-D) array $a(m_1,m_2)$, is evaluated at any real frequency $\hat{u}_1$, the resulting (1-D) function in frequency u_2 is minimum-phase.

(ii) The same spectrum $A(u_1,u_2)$ when evaluated at any real frequency $\hat{u}_2$ forms a (1-D) minimum-phase function in u_1.

The spectrum $A(u_1,u_2)$ is the (2-D) Fourier transform of the array $a(m_1,m_2)$. It can also be thought of as the z-transform $A(z_1,z_2)$ of $a(m_1,m_2)$ evaluated in $|z_1| = 1 \cap |z_2| = 1$. Therefore, conditions (i) and (ii) may be re-stated as follows:

Theorem 3.1 [20]

A two-dimensional array $a(m_1,m_2)$ with z-transform of $A(z_1,z_2)$ is minimum-phase iff:

$$A(z_1,z_2) \neq 0 \quad \text{for} \quad |z_1| \leqslant 1 \cap |z_2| \leqslant 1$$

From theorems 2.1 and 3.1 it can be concluded that the filter $F(z_1,z_2) = 1/B(z_1,z_2)$ is stable if $B(z_1,z_2)$ is minimum-phase. In (1-D) filter theory it is known that the planar least-squares inverse of a filter is minimum-phase [51]. The object here is to find how to make use of this property for stabilising an unstable (2-D) recursive digital filter without changing its amplitude spectrum.

Given an array C, an array P can be found such that the convolution of C and P is approximately equal to the unit impulse array δ. That is:

$$C * P \cong \delta \tag{3.1}$$

where the symbol $*$ denotes (2-D) convolution. In general, it is not possible to make $C*P$ exactly equal to δ. Let $C*P = G$; if P is now chosen such that the sum of the squares of the elements of $\delta - G$ is minimized, then, P is called a planar least square inverse (PLSI) of C.

The size of the array P is arbitrary. However, δ and G must have the same size, which depends on the size of the arrays C and P.

If C is an array of dimension (m_1,m_2) and P of (k_1,k_2), then G and δ must be (m_1+k_1-1) by (m_2+k_2-1). Since the size of P is arbitrary, there are many (PLSI) of the array C, one for each possible set of dimensions of P. However, once the dimensions k_1 and k_2 of the matrix P are fixed, there is one and only one array P that minimises the mean square difference between G and δ. This array P is determined from the matrices δ and C by the (2-D) Wiener technique [52].

As a group, PLSI's have some interesting properties. One particular such property, which has not been proved yet, is described by the following conjecture [20].

<u>Conjecture 3.1</u>

Given an arbitrary real finite array C, any planar least-squares inverse of C is minimum-phase. This is an important conjecture because it implies that the filter $F(z_1,z_2) = 1/B(z_1,z_2)$ is stable when B is a PLSI.

When a (2-D) recursive digital filter $F(z_1,z_2) = 1/B(z_1,z_2)$ is found to be unstable after completing the design, the question arises as to whether the coefficients of the filter's denominator $B(z_1,z_2)$ could be altered in order to produce a final stable filter. The denominator array of the unstable filter is denoted by B. B', a planar least-squares inverse of B, can be formed. Further, a planar least-squares inverse of B' can also be formed and called $\hat{B}$. Now, $\hat{B}$ is the inverse of the inverse, or the "double-inverse" of B. Intuitively, $\hat{B}$ and B will have some characteristics in common. Moreover, $\hat{B}$ is itself a PLSI. Hence it is minimum-phase, which B is not. Therefore, the filter $\hat{F}(z_1,z_2) = 1/\hat{B}(z_1,z_2)$ is stable.

The validity of assuming some degree of similarity between $\hat{B}$ and B is, of course, questionable. It has been so far taken for granted that $\hat{B}$, being the double-PLSI of B, is an approximate minimum-phase version of the latter. Thus the amplitude spectra of B and $\hat{B}$ would be expected to be roughly equal.

One of the factors governing the quality of the approximation is the size of the intermediate array B'. The larger the size of B', the better is the resemblance of B to the minimum-phase version array $\hat{B}$. Several examples, where the stabilised filters so obtained were a good approximation to the original unstable ones, can be found elsewhere [20,57]. They covered a number of cases involving filters of differing classes and degrees. However, only recently, Kamp [53] gave a counter-example that invalidated the method for certain cases.

3.2 READ AND TREITEL STABILISATION TECHNIQUE [54]

For a minimum-phase sequence, a(k), the phase, $\theta(e^{ju})$, and the log. of the amplitude spectrum, $\log|A(e^{ju})|$, are related by the Hilbert transform [55,56]:

$$\theta(e^{ju}) = -\frac{1}{2\pi}\int_0^{2\pi} \log|A(e^{j\Omega})| \cot\frac{u-\Omega}{2}\, d\Omega \qquad (3.2)$$

where the above equation results from a(k) (k = 0,1,...,M-1) being minimum-phase, and therefore causal, if and only if the inverse z-transform of $\log[A(z)]$ is causal. Since the Hilbert transform relates the real and imaginary parts of a minimum-phase function, its application to $\log|A(e^{ju})|$ in equation (3.2) gives the imaginary part of $\log|[A(z)]|$, which is the phase of $A(z)$. To implement the above integral on a digital computer the trapezoidal integration rule [58] has been employed to approximate the integral of equation (3.2) by the summation:

$$P_i(i) = \frac{1}{N}\sum_{k=0}^{N-1} P_r(k)\left[1-(-1)^{i-k}\right]\cot\frac{\pi}{N}(i-k) \qquad (3.3)$$

where $P_r(i)$ and $P_i(k)$ are respectively the real and imaginary part of the Fourier transform of $P(i)$.

This can be further simplified to:

$$\begin{aligned} P_i(i) &= \frac{2}{N}\sum_{k=1,3,5,..}^{N-1} P_r(k)\cot\frac{\pi}{N}(i-k), \text{ i even} \\ P_i(i) &= \frac{2}{N}\sum_{k=0,2,4,..}^{N-1} P_r(k)\cot\frac{\pi}{N}(i-k), \text{ i odd} \end{aligned} \qquad (3.4)$$

The above relations suggest a method for minimising the phase of a mixed or maximum-phase sequence. Firstly, the sequence is made causal and then,a new phase for the z-transform of the sequence is calculated using equation (3.2) to make it minimum-phase. This is the basis of the Read and Treitel stabilisation technique for (2-D) recursive filters. It is a direct extension of the (1-D) case already treated by the same authors. Further treatment of the technique necessitates the reader to be familiar with some alternative definitions of a few already defined functions, as given below.

A finite discrete impulse response $P(m_1,m_2)$ is causal if:

$$P(m_1,m_2) = 0 \qquad \text{for} \quad \begin{matrix} m_1 \geqslant M_1/2 \\ m_2 \geqslant M_2/2 \end{matrix} \tag{3.5}$$

where m_1 varies over the discrete set $\{0,1,\ldots,M_1-1\}$ and m_2 over the set $\{0,1,\ldots,M_2-1\}$. The even and odd parts of such a sequence are defined as:

$$P_e(m_1,m_2) = \tfrac{1}{2}\left[P(m_1,m_2) + P(M_1-m_1,\ M_2-m_2)\right] \tag{3.6}$$

and

$$P_o(m_1,m_2) = \tfrac{1}{2}\left[P(m_1,m_2) - P(M_1-m_1,\ M_2-m_2)\right] \tag{3.7}$$

respectively, and are related (as in the general case of a (2-D) minimum-phase sequence) by:

$$P_o(m_1,m_2) = \left[\mathrm{sgn}(m_1,m_2) + \mathrm{bdy}(m_1,m_2)\right] P_e(m_1,m_2) \tag{3.8}$$

where the sgn function is a finite (2-D) version of the (1-D) signum function. It is given by:

$$\mathrm{sgn}(m_1,m_2) = \begin{cases} 1 & 0 < m_1 < M_1/2 \quad \text{and} \quad 0 < m_2 < M_2/2 \\ -1 & M_1/2 < m_1 < M_1 \quad \text{and} \quad M_2/2 < m_2 < M_2 \\ 0 & \text{elsewhere} \end{cases} \tag{3.9}$$

The bdy function makes boundary adjustments and is defined by:

$$\mathrm{bdy}(m_1,m_2) = \begin{cases} 1 & m_2=0 \quad \text{and} \quad 0 < m_1 < M_1/2 \\ -1 & m_2=0 \quad \text{and} \quad M_1/2 < m_1 < M_1 \\ 1 & m_1=0 \quad \text{and} \quad 0 < m_2 < M_2/2 \\ -1 & m_1=0 \quad \text{and} \quad M_2/2 < m_2 < M_2 \\ 0 & \text{elsewhere} \end{cases} \tag{3.10}$$

The sequence $P(m_1,m_2)$ is the sum of its even and odd parts:

$$P(m_1,m_2) = P_e(m_1,m_2) + P_o(m_1,m_2) \tag{3.11}$$

Taking the Fourier transform of both sides yields:

$$\mathrm{DFT}[P(m_1,m_2)] = \mathrm{DFT}[P_e(m_1,m_2)] + \mathrm{DFT}[P_o(m_1,m_2)] \tag{3.12}$$

and, as is well-known, the DFT of a real and even function is real and even, and that of a real and odd function is odd and imaginary. Thus:

$$P_r(m_1,m_2) = \mathrm{DFT}[P_e(m_1,m_2)] \tag{3.13}$$

$$P_i(m_1,m_2) = -j\mathrm{DFT}[P_o(m_1,m_2)] \tag{3.14}$$

Taking the inverse discrete Fourier transform (IDFT) of both sides of (3.13) and substituting into (3.8) and using equation (3.14) will result in:

$$P_i(m_1,m_2) = -j\,\mathrm{DFT}(\{\mathrm{sgn}(m_1,m_2) + \mathrm{bdy}(m_1,m_2)\}.\mathrm{IDFT}[P_r(m_1,m_2)]) \tag{3.15}$$

This relation defines the (2-D) discrete Hilbert transform. (It clearly corresponds to the continuous transform given in equation (3.2).)

It will now be shown how to use the discrete Hilbert transform procedure (3.15) to obtain a minimum-phase version of a (2-D) array. The z-transform of such an array could be the denominator polynomial in the equation $F(z_1,z_2) = A(z_1,z_2)/B(z_1,z_2)$. Applying the scheme leads to a rational filter $\hat{F}(z_1,z_2)$ with very nearly the same amplitude spectrum as the original unstable filter $F(z_1,z_2)$. To arrive at a stable result, the phase of the (2-D) polynomial $B(z_1,z_2)$ should be minimized. The (2-D) discrete Hilbert transform can be applied in the same way as in (1-D) case [54] to obtain the minimum-phase version of a given array. Thus, given a (2-D) amplitude spectrum $B(m_1,m_2)$ of a causal sequence, the minimum-phase spectrum $\theta(m_1,m_2)$ is calculated from the equation:

$$\theta(m_1,m_2) = -j\mathrm{DFT}(\{\mathrm{sgn}(m_1,m_2) + \mathrm{bdy}(m_1,m_2)\}.\mathrm{IDFT}[\log B(m_1,m_2)]) \tag{3.16}$$

The derivation of the above expression can be found in [54]. Forming a minimum-phase version of an array by using equation (3.16) can be summarised by the few steps below.

<u>Step (1)</u>: Given a finite discrete (2-D) array, the coefficient array should be augmented with zeros to satisfy the condition for causality, equation (3.6). The added zeros increase the size of the array, so that it becomes amenable to Fast Fourier transform analysis.

<u>Step (2)</u>: The natural length of the amplitude spectrum of the augmented (2-D) array should be calculated.

<u>Step (3)</u>: The (2-D) discrete Hilbert transform must be applied to this (2-D) array. Thus the log. of the magnitude is treated as the real part and the discrete Hilbert transform then yields the imaginary part.

<u>Step (4)</u>: The imaginary part is used as the phase spectrum corresponding to the given amplitude spectrum. These two spectral characteristics completely describe the transform of the minimum-phase array.

<u>Step (5)</u>: After conversion from amplitude and phase to real and imaginary parts, the inverse transform is determined and truncated to obtain the same dimensions as the original array. This yields the minimum-phase version of the original array.

Although the Hilbert transform procedure works for most examples, it has been shown in [57] that there exist some cases where it proves to be of no value. One explanation of this may be that, if the discrete Hilbert transform stabilisation procedure was to yield the precise minimum-phase array, then all the elements in the inverse transform of the augmented array would be zero beyond the dimensions of the original array. In practice, though, it has been found that the element magnitudes of the augmented array beyond the boundaries defined by the original array are small compared to elements magnitudes within these boundaries, but are not zero. These elements are non-zero because the discrete Hilbert transform method of equation (3.3), implemented using the discrete Fourier transform, is an approximation to the original transform as given in equation (3.2).

Therefore, it can be concluded that the approximation of the integral associated with the truncation of the array has caused the failure of the Hilbert transform stabilisation procedure for certain arrays.

3.3 PISTOR STABILIZATION METHOD [26]

The Pistor method provides a procedure for decomposing an unstable (2-D) recursive filter having a non-zero, non-imaginary frequency response into four stable recursive filters, each of which recurses in a different direction.

For a real-valued discrete function c with a limited number of sample points, in which:

$$c = \left\{c_{m_1,m_2}\right\} \qquad \begin{array}{l} |m_1| \leqslant 2M_1 = \alpha_1 \\ |m_2| \leqslant 2M_2 = \alpha_2 \end{array} \tag{3.17}$$

the z-transform of which is of zero-phase and non-negative for all $(z_1, z_2) \in R_2$

$$R_2 = \left\{ (z_1, z_2) \,:\, |z_1| = 1 \cap |z_2| = 1 \right\} \tag{3.18}$$

satisfies

$$\mathrm{Im}[C'(u_1, u_2)] \equiv 0 \tag{3.19}$$

$$\mathrm{Re}[C'(u_1, u_2)] > 0 \tag{3.20}$$

where

$$C'(u_1, u_2) = \sum_{m_1=-\alpha_1}^{\alpha_1} \sum_{m_2=-\alpha_2}^{\alpha_2} c_{m_1, m_2} \, e^{-j2\pi u_1 m_1} \, e^{-j2\pi u_2 m_2}$$

$$= C(z_1, z_2) \text{ for } (z_1, z_2) \in R_2 \tag{3.21}$$

Equations (3.19) and (3.21) imply central symmetry of c:

$$c_{m_1, m_2} = c_{-m_1, -m_2} \tag{3.22}$$

Because of this symmetry, c is not a one-quadrant function [26]. However, because c has a limited number of sample points, it can be transformed by translation to any single quadrant function. Thus, the term $1/C(z_1, z_2)$ could be associated with four different recursive filters. None, however, would be stable. Proof of the above statement will be given in the generalized form for the N-dimensional case in the next section.

In the one-dimensional case, unstable recursive filters $1/B(z_1)$, with no poles on the unit circle, can be decomposed into two stable filters that recurse in opposite directions. In the (2-D) case it is analogously shown that, the unstable $1/C(z_1, z_2)$ can be decomposed into four stable filters that recurse in four different directions:

$$1/C(z_1,z_2) = \prod_{\ell=1}^{4} 1/K_\ell^{-1}(z_1,z_2) \qquad (3.23)$$

This is done by transforming equation (3.23) into the cepstrum domain [26]:

$$\hat{c} = \sum_{\ell=1}^{4} ({}^{\ell}\hat{k}) \qquad (3.24)$$

where $\hat{c}$ is the cepstrum of c, defined as a function whose z-transform $\hat{C}(z_1,z_2)$ is derived by the log. of the z-transform of c, $C(z_1,z_2)$.

The cepstrum $\hat{c}$ is readily obtained by evaluating the z-transform of c on R_2 as given by equation (3.18)

:

$$\hat{C}'(u_1,u_2) = \ln\left[\sum_{m_1=-2M_1}^{2M_1} \sum_{m_2=-2M_2}^{2M_2} c_{m_1,m_2} e^{-2\pi j(u_1m_1+u_2m_2)}\right] \quad (3.25)$$

Since $\hat{C}'$ is real and positive as in equations (3.19) and (3.20), the logarithm in equation (3.25) presents no problem. It is easily verified that all partial derivatives of $\hat{C}'(u_1,u_2)$ are continuous. Thus, for all u_1 and u_2, $\hat{C}'(u_1,u_2)$ is given by the Fourier series expansion [59]:

$$\hat{C}'(u_1,u_2) = \sum_{m_1=-\infty}^{\infty} \sum_{m_2=-\infty}^{\infty} \hat{c}_{m_1,m_2} e^{-2\pi j(u_1m_1+u_2m_2)} \qquad (3.26)$$

The coefficients $\hat{c}_{m_1,m_2}$ that make up the cepstrum $\hat{c}$ are obtained from:

$$\hat{c}_{m_1,m_2} = \int_0^1 \int_0^1 \hat{C}'(u_1,u_2) e^{2\pi j(u_1m_1+u_2m_2)} du_1 du_2 \qquad (3.27)$$

From the properties of $C'(u_1,u_2)$ and from equation (3.27) it follows that:

$$\hat{c}_{m_1,m_2} = \hat{c}_{-m_1,-m_2} \tag{3.28}$$

The procedure can be implemented by determining $\hat{c}$, from equations (3.21), (3.25) and (3.27).

This can be done by using the Fast Fourier Transform (FFT) [60] as indicated in Figure (3.1). The functions $\{c_{m_1,m_2}\}$ and $\{\hat{c}_{m_1,m_2}\}$ are IDFT of sampled version of $C'(u_1,u_2)$ and $\hat{C}'(u_1,u_2)$. $\hat{\varphi}(m_1,m_2)$ is defined as an aliased version of the desired result $\hat{c}$. The degree of aliasing can be controlled by the rate at which $C'(u_1,u_2)$ is sampled.

The function $\hat{\varphi}(m_1,m_2)$, truncated at the Nyquist subscripts I_{m_1Ny} and I_{m_2Ny}, is decomposed according to:

$$\hat{\varphi}_{tr} = {}^1\hat{k}' + {}^2\hat{k}' + {}^3\hat{k}' + {}^4\hat{k}' \tag{3.29}$$

where the $\hat{k}'$ are approximations of operator k in equation (3.24).

A more general treatment of the above technique is given by Ekstrom and Woods [27], where they have generalized the concept of spectral factorisation to (2-D); also, a family of canonical factorisations which are obtainable through the homomorphic transform [61] are found. In particular, the two, four and eight factor decomposition and their numerical implementation are treated in detail.

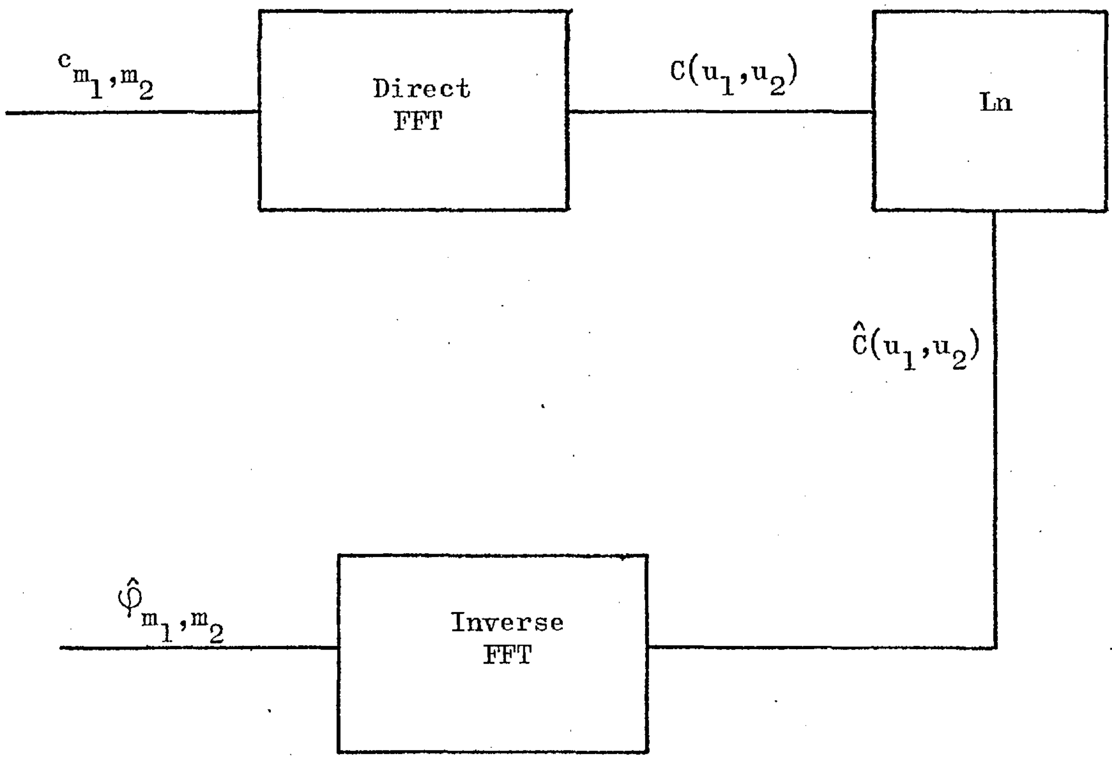

Fig. (3.1): Block diagram of determination of the approximate cepstrum transform for evaluation of equations (3.21), (3.25) and (3.27).

3.4 STABILISING (N-D) ZERO-PHASE RECURSIVE DIGITAL FILTERS [50,63]

This section presents an extension to (N-D) of the work of Pistor [26]. The stability of an (N-D) recursive digital filter is shown to be related to the properties of its cepstrum. A procedure is also given for the decomposition of unstable recursive digital filters having non-zero, non-imaginary frequency response into a set of stable single quadrant recursive filters. Some useful functions and generalised version of already stated theorem which will be used in developing the method will be given first.

Theorem (2.10), which is applicable to causal system, can be applied to non-causal filters by transformation from q^{th} quadrant to 1^{st} quadrant.

Corollary (3.1)

Let the q^{th} quadrant function ${}^{q}f$ $(q = 2,3,\ldots,2^n)$ be a stable convolutional filter. Then the non-causal recursive filter $1/B_1(z_1,\ldots,z_n)$ is stable if and only if $B_2(z_1,1/z_2,\ldots,z_n)$, $B_3(z_1,z_2,1/z_3,\ldots,z_n),\ldots,B_n(z_1,\ldots,1/z_n),\ldots,$ are non-zero for all $(z_i,\ i=1,n) \in D1$. The functions that meet the conditions of theorem (2.10) and corollary (3.1) are characterized by definition (3.2).

Definition (3.2)

A one-quadrant function ${}^{q}f$ is called recursively stable if and only if ${}^{q}f$ is a stable convolutional filter, i.e. $1/B_q(z_1,\ldots,z_n)$ is a stable recursive filter, either causal or non-causal.

3.4.1 N-Dimensional Cepstrum

Consider a function b and its z-transform:

$$\{b_{m_1,\ldots,m_n}\} \longleftrightarrow B(z_1,\ldots,z_n) \tag{3.30}$$

By taking the logarithm of B we obtain:

$$\hat{B}(z_1,\ldots,z_n) = \ln B(z_1,\ldots,z_n) \tag{3.31}$$

The cepstrum of b, denoted by $\hat{b}$, is a function whose z-transform is given by $\hat{B}(z_1,\ldots,z_n)$. Equations (3.30), (3.31) and (1.12) lead to the following input relation for a convolutional filter:

$$\hat{O}(z_1,\ldots,z_n) = \hat{I}(z_1,\ldots,z_n) + \hat{F}(z_1,\ldots,z_n) \tag{3.32}$$

3.4.2 Decomposition of unstable, recursive, zero-phase filters into stable recursive filters

Consider a real-valued discrete function c with a limited number of sample points:

$$c = \left\{c_{m_1,\ldots,m_n}\right\} \left(\bigcap_{i=1}^{n} |m_i| \leqslant 2M_i = \alpha_i\right) \tag{3.33}$$

Its Fourier transform is zero-phase and non-negative for all $(z_i,\ i = 1,n) \in R_n$, i.e.

$$\mathrm{Im}\left[C'(u_1,\ldots,u_n)\right] \equiv 0 \tag{3.34}$$

$$\mathrm{Re}\left[C'(u_1,\ldots,u_n)\right] > 0 \tag{3.35}$$

where:

$$C'(u_1,\ldots,u_n) = \sum_{m_1=-\alpha_1}^{\alpha_1} \cdots \sum_{m_n=-\alpha_n}^{\alpha_n} c_{m_1,\ldots,m_n} \cdot \exp\left(-2\pi j \sum_{i=1}^{n} u_i m_i\right)$$

$$= C(z_1,\ldots,z_n) \quad (z_i,\ i = 1,n) \in R_n \tag{3.36}$$

Equations (3.34) and (3.36) imply central symmetry of c:

$$c_{m_1,\ldots,m_n} = c_{-m_1,\ldots,-m_n} \tag{3.37}$$

Equation (3.37) shows that c is not a one-quadrant function. However, since c has a limited number of sample points, it can be transformed, by translation, to any single quadrant function. Thus the term $1/C(z_1,\ldots,z_n)$ could be associated with 2^N different recursive filters. However, none of these would be stable, as will now be proved. Consider a set of numbers:

$$\{\ell\} = \{\ell_1,\ell_2,\ldots,\ell_n\} = \{\pm 1, \pm 1,\ldots,\pm 1\}$$

Each quadrant function is obtained by assigning the values +1 or −1 to all the various elements of the subscript set $\{\ell\}$.

The following z-transforms can be obtained from c by translation:

$$\underset{(\ell_1,\ell_2,\ldots,\ell_n)}{C(z_1,\ldots,z_n)} = z_1^{\alpha_1\ell_1}.z_2^{\alpha_2\ell_2}\ldots z_n^{\alpha_n\ell_n}C(z_1,\ldots,z_n) \tag{3.38}$$

According to theorem (2.10) and corollary (3.1), the recursive filters associated with these z-transforms are unstable if a set $\{z_i,\ i = 1,n\} \in D1$ exists such that:

$$\bigcap_{q=1}^{2^n} {}^qC_{\ell_1,\ldots,\ell_n}(z_1^{\ell_1},\ldots,z_n^{\ell_n}) = 0 \tag{3.39}$$

By choosing $\bigcap_{i=2}^{n} z_i=1$ the existence of such a set is proved, if a $z_1 = w_1$, $|w_1| \leqslant 1$ can be found, which solves simultaneously the equations (3.40) and (3.41):

$$\begin{aligned} z_1^{\alpha_1}\,C(z_1, 1,\ldots,1) &= \sum_{m_1=-\alpha_1}^{\alpha_1}\cdots\sum_{m_n=-\alpha_n}^{\alpha_n} c_{m_1,\ldots,m_n} z_1^{\alpha_1+m_1} \\ &= z_1^{\alpha_1}\,G(z_1) = 0 \end{aligned} \tag{3.40}$$

$$z_1^{\alpha_1} C(1/z_1,1,\ldots,1) = z_1^{\alpha_1} G(1/z_1) = 0 \qquad (3.41)$$

Since it can be concluded from the central symmetry of c that:

$$G(z_1) = \sum_{m_1=-\alpha_1}^{\alpha_1} \cdots \sum_{m_n=-\alpha_n}^{\alpha_n} c_{m_1,\ldots,m_n} \, z_1^{m_1} = G(1/z_1) \qquad (3.42)$$

equations (3.40) and (3.41) are identical and will be satisfied either by $z_1 = 0$ or by a root of $G(z_1) = G(1/z_1)$. Since if $Z_1 = w_1$ is a root so is $z_1 = 1/w_1$, it is always possible to choose $|z_1| \leqslant 1$.

Thus a set $\{z_i, \ i=1,n\} \in D1$ can always be found, such that equation (3.39) is satisfied, and therefore no recursively stable operator can be obtained from c by translation. As is shown earlier in the one- (and two-) dimensional case, unstable recursive filters can always be decomposed into two (and four) stable recursive filters which recurse in two (and four) different directions [62,26].

Therefore the question arises whether an unstable (N-D) filter $1/C(z_1,\ldots,z_n)$ can be decomposed into 2^n stable filters that recurse in 2^n different directions. More precisely, we ask for recursively stable one-quadrant functions ${}^1k$, ${}^2k,\ldots,{}^{2^n}k$ such that:

$$1/C(z_1,\ldots,z_n) = \prod_{q=1}^{2^n} 1/k_q^{-1}(z_1,\ldots,z_n) \qquad (3.43)$$

We find a solution to this problem by transforming equation (3.43) to the cepstrum domain:

$$\hat{c} = \sum_{q=1}^{2^n} ({}^q\hat{k}) \qquad (3.44)$$

3.4.3 Stability Criterion for Recursive Filters based on (N-D) Cepstra

Let us assume that ${}^1f$ is a first quadrant function that is recursively stable. This assumption implies from theorem (2.8), (2.9), (2.10) and definition (3.2) that:

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} \left| {}^1f_{m_1,\ldots,m_n} \right| < \infty \tag{3.45}$$

$${}^1b * {}^1f = \delta_{m_1,\ldots,m_n} \tag{3.45a}$$

$$B_1(z_1,\ldots,z_n) \neq 0, \quad (z_i,\ i=1,n) \in D_1 \tag{3.46}$$

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} \left| {}^1b_{m_1,\ldots,m_n} \right| < \infty \tag{3.47}$$

We now wish to deduce that $\hat{B}_1(z_1,\ldots,z_n) = \ln B_1(z_1,\ldots,z_n)$ can be uniquely expanded into a power series

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} {}^1\hat{b}_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n} \tag{3.48}$$

for all $(z_i,\ i=1,n) \in D_1$. First we consider the case $(z_i,\ i=1,n) \in D_2$ where

$$D_2 = \left\{ (z_i,\ i=1,n) \ : \ \bigcap_{i=1}^{n} |z_i| < 1 \right\} \tag{3.49}$$

On this set $B_1(z_1,\ldots,z_n)$ is regular, due to equation (3.45), and consequently $\hat{B}_1(z_1,\ldots,z_n)$ is regular too, because $B_1(z_1,\ldots,z_n)$ is non-zero. Thus a unique expansion (3.48) of $\hat{B}_1$ exists in D_1. The coefficient ${}^1\hat{b}_{0,\ldots,0}$ is determined by evaluating

$$\hat{B}_1(z_1,\ldots,z_n) = \ln B_1(z_1,\ldots,z_n) \tag{3.50}$$

For $\left(\bigcap_{i=1}^{n} z_i = 0 \right)$

$$ {}^{1}\hat{b}_{0,\dots,0} = \ln {}^{1}b_{0,\dots,0} \tag{3.51} $$

To find the other coefficients we differentiate equation (3.50) with respect to each (z_i, i=1,n) and replace $1/B_1(z_1,\dots,z_n)$ by $F_1(z_1,\dots,z_n)$:

$$ z_i(\partial/\partial z_i)\hat{B}_1(z_1,\dots,z_n) = F_1(z_1,\dots,z_n) z_i(\partial/\partial z_i).B_1(z_1,\dots,z_n), \quad i = 1,n \tag{3.52} $$

The above relation corresponds to:

$$ \left\{ m_i\, {}^{1}\hat{b}_{m_1,\dots,m_n} \right\} = \left\{ {}^{1}f_{m_1,\dots,m_n} \right\} * \left\{ m_i\, {}^{1}b_{m_1,\dots,m_n} \right\}, \quad i = 1,n \tag{3.53} $$

or explicitly written:

$$ {}^{1}\hat{b}_{p_1,\dots,p_n} = \sum_{m_1=0}^{P_1} \cdots \sum_{m_n=0}^{P_n} \left(\frac{m_i}{p_i}\right) {}^{1}b_{m_1,\dots,m_n} . f_{p_1-m_1,\dots,p_n-m_n} $$

$$ i = 1,n, \quad p_i \neq 0 \tag{3.54} $$

We prove now that the power series (3.48) with coefficients defined by equations (3.51) and (3.53) is absolutely convergent and equal to $\hat{B}_1(z_1,\dots,z_n)$ for all (z_i, i = 1,n) $\in D_1 - D_2$.

$$ \sum_{p_1=0}^{N} \cdots \sum_{p_n=0}^{N} \left| {}^{1}\hat{b}_{p_1,\dots,p_n} \right| \left| z_1^{p_1} \right| \dots \left| z_n^{p_n} \right| \leqslant $$

$$ \sum_{p_n=1}^{N} \sum_{m_n=0}^{P_n} \left(\frac{m_n}{p_n}\right) \left| {}^{1}b_{0,0,\dots,m_n} \right| \left| {}^{1}f_{0,\dots,p_n-m_n} \right| + $$

$$ \sum_{p_1=1}^{N} \sum_{p_2=1}^{N} \cdots \sum_{p_n=0}^{N} \sum_{m_1=0}^{N} \cdots \sum_{m_n=0}^{N} \left(\frac{m_i}{p_i}\right) \left| {}^{1}b_{m_1,\dots,m_n} \right| . \left| {}^{1}f_{p_1-m_1,\dots,p_n-m_n} \right| $$

$$\leqslant \sum_{p_1=0}^{N} \sum_{p_2=0}^{N} \cdots \sum_{p_n=0}^{N} \sum_{m_1=0}^{N} \cdots \sum_{m_n=0}^{N} \left|{}^1b_{m_1,\ldots,m_n}\right| \left|{}^1f_{p_1-m_1,\ldots,p_n-m_n}\right|$$

$$\leqslant \sum_{m_1=0}^{N} \cdots \sum_{m_n=0}^{N} \left|{}^1b_{m_1,\ldots,m_n}\right| \sum_{p_1=0}^{N} \cdots \sum_{p_n=0}^{N} \left|f_{p_1,\ldots,p_n}\right|$$

and thus

$$\sum_{p_1=0}^{\infty} \cdots \sum_{p_n=0}^{\infty} \left|{}^1\hat{b}_{p_1,\ldots,p_n}\right| \left|z_1^{p_1}\right| \cdots \left|z_n^{p_n}\right| \leqslant$$

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} \left|{}^1b_{m_1,\ldots,m_n}\right| \sum_{p_1=0}^{\infty} \cdots \sum_{p_n=0}^{\infty} \left|{}^1f_{p_1,\ldots,p_n}\right| < \infty$$

$$(p_i,\ i = 1,n) \neq 0, \quad (z_i,\ i = 1,n) \in D_1 \qquad (3.55)$$

In equation (3.55) we have used the stability of ${}^1f$ and ${}^1b$, equations (3.45) and (3.47).

With the intermediate notation

$$E(z_1,\ldots,z_n) = \sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} {}^1\hat{b}_{m_1,\ldots,m_n}\, z_1^{m_1} \cdots z_n^{m_n} \qquad (3.56)$$

it remains to be proved that

$$E(z_1,\ldots,z_n) = \hat{B}_1(z_1,\ldots,z_n)$$

for all $(z_i,\ i = 1,n) \in D_1 - D_2$. Because $E(z_1,\ldots,z_n)$ and $B_1(z_1,\ldots,z_n)$ are given by power series that are absolutely convergent for all $(z_i,\ i = 1,n) \in D_1$, it follows that

$$\lim_{\zeta_1 \to 1} \left[B_1(e^{-2\pi j u_1}, z_2,\ldots,z_n) - B_1(\zeta_1 e^{-2\pi j u_1},\ldots,z_n)\right] = 0 \quad (3.57)$$

and

$$\lim_{\varsigma_1 \to 1} \left[E(e^{-2\pi j u_1}, \ldots, z_n) - E(\varsigma_1 e^{-2\pi j u_1}, \ldots, z_n) \right] = 0 \qquad (3.58)$$

for any fixed $(u_1, z_2, \ldots, z_n)$ with

$$(u_1, z_2, \ldots, z_n) \in \left\{ (u_1, z_i,\ i=2,n) : 0 \leqslant u_1 \leqslant 1 \quad \bigcap_{i=2}^{n} |z_i| < 1 \right\} = Q$$

From (3.57) we conclude that:

$$\lim_{\varsigma_1 \to 1} \left[\hat{B}_1(e^{-j2\pi u_1}, z_2, \ldots, z_n) - \hat{B}_1(\varsigma_1 e^{-2\pi j u_1}, z_2, \ldots, z_n) \right] = 0 \qquad (3.59)$$

for any $(u_1, z_i,\ i=2,n) \in Q$. When (3.59) is subtracted from (3.58)

$$\hat{B}_1(z_1, \ldots, z_n) = E(z_1, \ldots, z_n) \qquad (3.60)$$

for all $(z_i,\ i=1,n) \in \left\{ (z_i,\ i=1,n) : |z_1| \leqslant 1 \ \bigcap_{i=2}^{n} |z_i| < 1 \right\}$.

Repeating the argument for $z_1 = \exp(-2\pi j u_1)$ and $z_2 = \varsigma_2 \exp(-2\pi j u_2)$ or in general for $z_n = \varsigma_n \exp(-2\pi j u_n)$ where $u_1, \ldots, u_n$ are any fixed values, we finally find:

$$\hat{B}_1(z_1, \ldots, z_n) = E(z_1, \ldots, z_n), \ (z_i,\ i=1,n) \in D_1 \qquad (3.61)$$

Thus the convolutional stability of ${}^1b$ is a necessary condition for ${}^1f$ to be recursively stable. We prove now that this condition is also sufficient.

Again we first consider the set D_2. On D_2 $\hat{B}_1(z_1, \ldots, z_n)$ is regular because of (3.55), and since $B_1(z_1, \ldots, z_n) = \exp[\hat{B}_1(z_1, \ldots, z_n)]$ it is also regular; the identities

$$z_i \left(\frac{\partial}{\partial z_i}\right) B_1(z_1, \ldots, z_n) = B_1(z_1, \ldots, z_n) z_i \left(\frac{\partial}{\partial z_i}\right) \hat{B}_1(z_1, \ldots, z_n)$$

$$i = 1,n \qquad (3.62)$$

relate the absolutely convergent power series. These identities yield relations similar to equation (3.54):

$$^{1}b_{p_1,\ldots,p_n} = \sum_{m_1=0}^{P_1} \cdots \sum_{m_i=1}^{P_i} \cdots \sum_{m_n=0}^{P_n} \left(\frac{m_i}{p_i}\right) {}^{1}\hat{b}_{m_1,\ldots,m_n} \cdot {}^{1}b_{p_1-m_1,\ldots,p_n-m_n} \tag{3.63}$$

$$i = 1,n \quad \text{and} \quad P_i \neq 0$$

For $(\bigcap_{i=1}^{n} P_i) = 0$ we find from equation (3.51)

$$^{1}b_{0,\ldots,0} = \exp({}^{1}\hat{b}_{0,\ldots,0}) \tag{3.64}$$

with ^{1}b defined by equations (3.63) and (3.64). We have

$$B_1(z_1,\ldots,z_n) = \exp(\hat{B}_1(z_1,\ldots,z_n))$$

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} {}^{1}b_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n} \tag{3.65}$$

identically for all $(z_i,\ i=1,n) \in D_2$. We prove now that (3.65) is also valid for all $(z_i,\ i=1,n) \in D_1$. To do this it is sufficient to find a power series that is equal to $B_1(z_1,\ldots,z_n)$ and absolutely convergent for all $(z_i,\ i=1,n) \in D_1$.

Since $\hat{B}_1(z_1,\ldots,z_n)$ is given as a power series, that is, by assumption, absolutely convergent for all $(z_i,\ i=1,n) \in D_1$, and the expansion of $\exp(x)$ is absolutely convergent for all $|x| < \infty$, the expansion

$$\exp[\hat{B}_1(z_1,\ldots,z_n)] = \sum_{\ell=0}^{\infty} \frac{1}{\ell!} [\hat{B}_1(z_1,\ldots,z_n)]^{\ell}$$

$$= \sum_{\ell=0}^{\infty} \left(\frac{1}{\ell!}\right) \left(\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} {}^{1}\hat{b}_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n}\right)^{\ell} \tag{3.66}$$

is also absolutely convergent for all $(z_i,\ i=1,n) \in D_1$. Thus equation (3.66) can be regarded as

$$\exp[(\hat{B}_1(z_1,\ldots,z_n)] = \sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} a_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n} \tag{3.67}$$

where

$$a_{m_1,\ldots,m_n} = \sum_{\ell=0}^{\infty} a^{(\ell)}_{m_1,\ldots,m_n} \tag{3.68}$$

$$a^{(\ell)}_{m_1,\ldots,m_n} = \left(\frac{1}{\ell}\right) \sum_{p_1=0}^{m_1} \cdots \sum_{p_n=0}^{m_n} a^{(\ell-1)}_{p_1,\ldots,p_n} \hat{b}_{m_1-p_1,\ldots,m_n-p_n} \qquad \ell = 1,2,\ldots \tag{3.69}$$

$$a^{(o)}_{m_1,\ldots,m_n} = \delta_{m_1,\ldots,m_n} \tag{3.70}$$

Therefore the expansion (3.65) is valid and absolutely convergent for all $(z_i,\ i=1,n) \in D_1$. It is evident from the definition that $B(z_1,\ldots,z_n)$ is non-zero on D_1. Theorem 3.2 summarizes the previous results.

<u>Theorem 3.2</u>:

The sequence

$$\left\{{}^1b_{m_1,\ldots,m_n}\right\} \quad (m_i,\ i = 1,n) \geqslant 0$$

is recursively stable iff there exists a power series

$$\sum_{m_1=0}^{\infty} \cdots \sum_{m_n=0}^{\infty} {}^1\hat{b}_{m_1,\ldots,m_n} z_1^{m_1} \cdots z_n^{m_n} \tag{3.71}$$

that is absolutely convergent and equal to $\ln B_1(z_1,\ldots,z_n)$ for all $(z_i,\ i=1,n) \in D_1$.

Theorem 3.2 remains applicable to non-causal recursive operators ${}^{q}b$, $q = 2,3,\ldots,2^{n}$, if they are transformed to a first quadrant function. This leads to the following corollary:

Corollary 3.2

The q^{th} quadrant function ${}^{q}b$, in which $q = 2,\ldots,2^{n}$, is recursively stable iff:

$$\bigcap_{q=2}^{2^n} \ln {}^{q}B_{\ell_1,\ldots,\ell_n}(z_1^{\ell_1},\ldots,z_n^{\ell_n})$$

for each q is equal to a power series of the form (3.71), that is absolutely convergent for all $(z_i,\ i=1,n) \in D_1$. From theorem 3.2 and corollary 3.2 it is evident that the decomposition problem in the previous section can be solved if the cepstrum $\hat{c}$ of c can be decomposed (see equation (3.44)) into one-quadrant functions ${}^{q}\hat{k}$, in which $q = 1,\ldots,2^{n}$. These are stable convolutional filters. In the next section it is proved that the decomposition is possible in view of the properties of c.

3.4.4 Solution of the Decomposition Problem

Now we will prove the existence of a set of solutions. The cepstrum $\hat{c}$ is obtained by evaluating the z-transform of c on R. We first have:

$$\hat{C}'(u_1,\ldots,u_n) = \ln\Bigg[\sum_{m_1=-2M_1}^{2M_1}\cdots\sum_{m_n=-2M_n}^{2M_n} c_{m_1,\ldots,m_n}\cdot \exp(-2\pi j\sum_{i=1}^{n} u_i m_i)\Bigg] \qquad (3.72)$$

Since $\hat{C}'$ is real and positive (equations (3.34) and (3.35)), the logarithm in equation (3.72) presents no problem. It is easily verified that all partial derivatives of $\hat{C}'(u_1,\ldots,u_n)$ are continuous. Thus, for all $(u_i,\ i = 1,n)$, $\hat{C}'(u_1,\ldots,u_n)$ is given by the Fourier series expansion:

$$\hat{C}'(u_1,\ldots,u_n) = \sum_{m_1=-\infty}^{\infty} \cdots \sum_{m_n=-\infty}^{\infty} \hat{c}_{m_1,\ldots,m_n} \cdot \exp(-2\pi j \sum_{i=1}^{n} u_i m_i) \tag{3.73}$$

The coefficients $\hat{c}_{m_1,\ldots,m_n}$ that make up this cepstrum $\hat{c}$ are obtained from

$$\hat{c}_{m_1,\ldots,m_n} = \int_0^1\int \cdots \int_0^1 \hat{C}'(u_1,\ldots,u_n). \exp(2\pi j \sum_{i=1}^{n} u_i m_i) du_1,\ldots,du_n \tag{3.74}$$

From the properties of $C'(u_1,\ldots,u_n)$ and from equation (3.74) it follows that:

$$\hat{c}_{m_1,\ldots,m_n} = \hat{c}_{-m_1,\ldots,-m_n} \tag{3.75}$$

Repeated integration by parts will lead to the inequalities:

$$|\hat{c}_{0,0,\ldots,0}| \leqslant \alpha < \infty \tag{3.76}$$

$$|\hat{c}_{0,0,\ldots,m_n}| \leqslant \alpha/m_n^2 \qquad m_n \neq 0 \tag{3.77}$$

$$|\hat{c}_{m_1,0,\ldots,0}| \leqslant \alpha/m_1^2 \qquad m_1 \neq 0 \tag{3.78}$$

$$\vdots$$

$$|\hat{c}_{m_1,\ldots,m_n}| \leqslant \alpha/\prod_{i=1}^{n}(m_i)^2 \qquad \bigcap_{i=1}^{n} m_i \neq 0 \tag{3.79}$$

Thus any decomposition:

$$\hat{c} = \sum_{q=1}^{2^n} {}^q\hat{k} \tag{3.80}$$

of $\hat{c}$ into one-quadrant functions yields functions ${}^q\hat{k}$ that meet the condition of Theorem 3.2 or Corollary 3.2. Consequently each of these decompositions corresponds to a set $\{{}^qk: q = 1,\ldots,2^n\}$ of recursively stable functions.

3.4.5 Problem of an Optimal Solution

Ideallly we wish to find a decomposition of $\hat{c}$ into functions ${}^qk$ with a minimum number of non-zero sample values such that equation (3.43) holds exactly, or at least to a close approximation. Unfortunately such a decomposition cannot be achieved, but an approximation to this may be possible by truncation of ${}^qk$ into finite length functions $(q = 1,\ldots,2^n)$; since c is centrally symmetric and also symmetric with respect to the axis m_i $(i = 1,\ldots,n)$.

In general the decomposition of c may result in operators ${}^qk$ having an infinite number of sample points. If these operators are to be implemented numerically, some truncation of the operators becomes mandatory. This truncation means not only that the decomposition of equation (3.43) becomes an approximate one, but also that the recursive stability of the operator ${}^qk$ may be affected. It is shown in (2-D) case [26, 27] that the larger the size of the approximated array the more likely it is to be stable. Ekstrom et al. [27] has introduced the weighting sequences to smooth the truncated factors and remove the possible poles from the unstable domain in (2-D) case; this technique can also be used for (N-D) case. We can introduce the generalized version of the weighting sequences which Ekstrom used as follows:

1) $$W(m_1,\ldots,m_n) = \max\left\{(1 - \frac{|m_1|}{M_1})(1 - \frac{|m_2|}{M_2})\ldots(1 - \frac{|m_n|}{M_n}),\ 0\right\}$$

2) $$W(m_1,\ldots,m_n) = \exp - \sum_{i=1}^{n} \beta_i |m_i| \quad (\beta_i,\ i = 1,2,\ldots,n) \geqslant 0$$

and it is anticipated that these sequences may result in a stabilising effect on the truncated recursive quadrant filters.

CHAPTER FOUR

FREQUENCY DOMAIN DESIGN TECHNIQUES

INTRODUCTION

The problem of designing filters in the frequency domain is that of specifying the coefficients of the filter's frequency response as in equations (1.5) and (1.6), such that some aspects of this response would approximate some desired characteristics. The "approximating" design is determined by using any of a few well-known optimisation approaches, e.g. minimum-mean-square or minimax error.

This chapter reviews several frequency domain design techniques for (2-D) recursive digital filters, and proposes a new such technique. In addition, a novel design technique for (N-D) zero-phase recursive digital filters is introduced. Through the derivation of this evolves yet one more design technique, this time for the general (N-D) non-recursive digital filter case.

The earliest work in (2-D) filter design was carried out by Hall [28,29].

4.1 SEPARABLE PRODUCT TECHNIQUE [28,29]

E.L. Hall in 1970 published the first paper dealing with the design of stable (2-D) recursive digital filters. The proposed technique enables the design of any (2-D) rectangular cut-off boundary type filter by the use of two (1-D) recursive digital filters in cascade. This technique involves only the well-established principles of (1-D) filter design, hence stability is simply ensured.

4.2 SHANKS DESIGN TECHNIQUE [20]

The design technique proposed by Shanks consists of mapping (1-D) into (2-D) filters with arbitrary directivity in a (2-D) frequency response plane. These filters are called rotated filters because they are obtained by rotating (1-D) filters.

Suppose a (1-D) continuous filter whose impulse response is real, is given in its factored form:

$$H_1(s) = H_0\left[\prod_{i=1}^{m}(s-q_i)/\prod_{i=1}^{n}(s-p_i)\right] \tag{4.1}$$

where H_0 is a scalar gain constant. The zero locations q_i and the pole locations p_i may be complex, in which case their conjugates are also present in the corresponding product. The cut-off angular frequency for this filter is assumed to be unity.

The filter given in equation (4.1) can also be viewed as a (2-D) filter that varies in one-dimension only and could be written as follows:

$$H_2(s_1,s_2) = H_1(s_2) = H_0\left[\prod_{i=1}^{m}(s_2-q_i)/\prod_{i=1}^{n}(s_2-p_i)\right] \tag{4.2}$$

Rotating clockwise the (s_1,s_2) axis through an angle β by means of the transformation

$$s_1 = s_1'\cos\beta + s_2'\sin\beta \tag{4.3a}$$

$$s_2 = -s_1'\sin\beta + s_2'\cos\beta \tag{4.3b}$$

will result in a filter whose frequency response is rotated by an angle $-\beta$ with respect to the frequency response of (4.2):

$$H_2(s_1',s_2') = H_0 \frac{\prod_{i=1}^{m}\left[(s_2'\cos\beta - s_1'\sin\beta) - q_i\right]}{\prod_{i=1}^{n}\left[(s_2'\cos\beta - s_1'\sin\beta) - p_i\right]} \tag{4.4}$$

$H_2(s_1',s_2')$ describes a continuous (2-D) filter in the new co-ordinate system of s_1' and s_2'. The corresponding digital version of the above filter is obtained through the application of the (2-D) bilinear z-transform defined by the following two equations:

$$s_1' = \frac{2}{T}\frac{1-z_1}{1+z_1} \tag{4.5a}$$

$$s_2' = \frac{2}{T}\frac{1-z_2}{1+z_2} \tag{4.5b}$$

In the above, it is assumed that the sample interval T is the same in both directions. Substituting equation (4.5) into (4.4) will result in:

$$H(z_1,z_2) = A\prod_{i=1}^{M}\frac{a_{11}^i+a_{21}^i z_1+a_{12}^i z_2+a_{22}^i z_1 z_2}{b_{11}^i+b_{21}^i z_1+b_{12}^i z_2+b_{22}^i z_1 z_2} \tag{4.6}$$

where:

$$A = H_0(\tfrac{1}{2}T)^{n-m}$$

$$M = \max(m,n)$$

$$\left.\begin{aligned} a_{11}^i &= \cos\beta - \sin\beta - \tfrac{1}{2}Tq_i \\ a_{21}^i &= \cos\beta + \sin\beta - \tfrac{1}{2}Tq_i \\ a_{12}^i &= -\cos\beta - \sin\beta - \tfrac{1}{2}Tq_i \\ a_{22}^i &= -\cos\beta + \sin\beta - \tfrac{1}{2}Tq_i \end{aligned}\right] \quad \text{for } 1 \leqslant i \leqslant m \tag{4.7}$$

$$a_{11}^i = a_{21}^i = a_{12}^i = a_{22}^i = 1, \qquad \text{for } m < i \leqslant M$$

$$\left.\begin{aligned} b_{11}^i &= \cos\beta - \sin\beta - \tfrac{1}{2}Tp_i \\ b_{21}^i &= \cos\beta + \sin\beta - \tfrac{1}{2}Tp_i \end{aligned}\right] \quad \text{for } 1 \leqslant i \leqslant n$$

$$\left.\begin{array}{l} b^i_{12} = -\cos\beta - \sin\beta - \tfrac{1}{2}Tp_i \\ b^i_{22} = -\cos\beta + \sin\beta - \tfrac{1}{2}Tp_i \end{array}\right] \quad \text{for } 1 \leqslant i \leqslant n$$

$$b^i_{11} = b^i_{21} = b^i_{12} = b^i_{22} = 1 \qquad \text{for } n < i \leqslant M \tag{4.7}$$

Unfortunately this technique as it stands does not guarantee the stability of the designed filter and the approach suffers from warping effects of the bilinear transformation on the frequency response. A modification to this technique has been introduced by Costa and Venetsanopoulos.

4.3 COSTA AND VENETSANOPOULOS DESIGN TECHNIQUE [30]

In this paper it is shown that the rotated filters can be used in designing circularly symmetric (2-D) recursive filters. A number of rotated filters whose angles of rotation are uniformly distributed over 180° results in a filter having a magnitude response which approximates a circularly symmetric cut-off boundary by a polygon. This polygon has an even number of sides because each rotated filter contributes two sides of the polygon.

It has been proved that stability of the designed filter is ensured if the following two conditions hold:

(i) $270^\circ \leqslant \beta \leqslant 360^\circ$, where β is the angle of rotation, and

(ii) $C_i < 0$ for $i = 1,2,\ldots,M$, where $C_i = \mathrm{Re}\,[(T/2)p_i]$ and p_i represents the location of a pole.

This has been derived through the knowledge of the stability constraints on the coefficients of the denominator $B(z_1,z_2)$ of a second order (2-D) recursive digital filter [21] of the form of equation (4.6).

4.4 BERNABO DESIGN TECHNIQUE [31]

The Bernabo technique is based on the use of the transformation methods of McClellan [15,69] followed by the decomposition method of Pistor [26] in order to obtain four stable single quadrant filters which recurse in different directions which will now be discussed in detail.

4.4.1 McClellan Design Method for FIR Filters [15,69]

The McClellan transformation is a transformation of a (1-D) zero-phase FIR filter into a (2-D) zero-phase FIR filter by a substitution of variables. It can be applied to (1-D) filters of odd length and in one special case also to filters of even length. The latter will not be considered in this thesis.

For a (1-D) filter of length 2N+1 to be zero-phase its impulse response h(n) must have Hermitian symmetric coefficients. Thus if h(n) is real it must also be even. Denoting the frequency response by $H(e^{ju})$, it can thus be written:

$$H(e^{ju}) = h(0) + \sum_{n=1}^{N} h(n)\left[e^{-jun} + e^{jun}\right]$$

$$= h(0) + \sum_{n=1}^{N} 2h(n)\cos un \tag{4.8}$$

(2-D) digital filters which have frequency responses of the form:

$$H(e^{ju_1}, e^{ju_2}) = \sum_{m_1=0}^{M_1} \sum_{m_2=0}^{M_2} a(m_1,m_2)\cos m_1u_1 \cos m_2u_2 \tag{4.9}$$

are examples of (2-D) zero-phase filters. However, equation (4.9) does not represent the most general class of such filters, but many useful filters are in this class and it is the only class which will be considered here. The impulse response of such a system $h(m_1,m_2)$ is a real sequence which is an even function of its arguments.

The McClellan transformation converts (1-D) filters of the form (4.8) into (2-D) filters of the form (4.9) by means of the substitution:

$$\cos u = \sum_{p_1=0}^{1} \sum_{p_2=0}^{1} t(p_1,p_2)\cos p_1u_1 \cos p_2u_2 \tag{4.10}$$

The relation between $h(n)$ and $h(m_1,m_2)$ can be seen by rewriting equation (4.8) in the form:

$$H(e^{ju}) = \sum_{n=0}^{N} b(n) \Big[\cos u \Big]^n \tag{4.11}$$

Performing the substitution indicated in equation (4.10) gives:

$$H(e^{ju_1}, e^{ju_2}) = \sum_{n=0}^{N} b(n) \left[\sum_{p_1=0}^{1} \sum_{p_2=0}^{1} t(p_1,p_2)\cos p_1u_1 \cos p_2u_2 \right]^n \tag{4.12}$$

By exploiting the recurrence formula for Chebyshev polynomials, this can be written in the form of equation (4.9). The (2-D) filter which results is of size (2N+1)x(2N+1).

It follows implicitly from equation (4.10) that points in the frequency response of the (1-D) filter are mapped to contours in the (u_1,u_2) plane. Furthermore, the shape of these contours is determined only by the parameters t. The variation of the frequency response from contour to contour, on the other hand, is controlled by the impulse response of the (1-D) filter $h(n)$.

The problem of designing filters using this technique thus splits into two parts:

(i) the design of the contour parameters t such that the contours (u = constant) produced by the transformation of equation (3.10) in the (u_1,u_2) plane have some desired shape, and

(ii) the design of the (1-D) prototype filter $\acute{h}(n)$.

Due to the existence of several well-known efficient algorithms for designing (1-D) FIR filter (see, for example, Rabiner and Gold [70]), we shall only be concerned here with the first of these problems.

4.4.2 The Contour Approximation Problem

The transformation parameters $t(p_1,p_2)$ need to be chosen so that the contours (u = constant) produced by the transformation:

$$\begin{aligned}\cos u &= F(u_1,u_2) \\ &= \sum_{p_1=0}^{P_1}\sum_{p_2=0}^{P_2} t(p_1,p_2)\cos p_1u_1 \cos p_2u_2 \qquad (4.13)\end{aligned}$$

in the (u_1,u_2) plane have some desired shape. 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) prototype should map is then described by the relation:

$$u_1^2 + u_2^2 = R^2 \tag{4.14}$$

For simplicity the transformation considered here is first order of the form:

$$\cos u = F(u_1,u_2) = t(0,0) + t(1,0)\cos u_1 + t(0,1)\cos u_2 + t(1,1)\cos u_1 \cos u_2 \tag{4.15}$$

If the prototype is low pass and the (2-D) filter is to be low-pass, the (1-D) origin will usually map to the (2-D) origin. This gives the constraint equation:

$$t(0,0) + t(1,0) + t(0,1) + t(1,1) = 1 \tag{4.15}$$

One variable, say $t(1,1)$, is thus constrained to be a function of the other three, while those three are still unconstrained. To find values for these free variables equation (4.16) should be solved for u_2 in terms of u, u_1 and the free mapping parameters:

$$\cos u = F(u_1,u_2) \tag{4.16}$$

This yields the equation:

$$u_2 = G(u,u_1,t(0,0),\ t(1,0),\ t(0,1))$$

$$= \cos^{-1}\left[\frac{\cos u - t(0,0) - t(1,0)\cos u_1}{t(0,1) + t(1,1)\cos u_1}\right] \tag{4.17}$$

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

$$E_1(u_1) = G(u_0,u_1,\hat{t}) - \sqrt{R^2-u_1^2} \tag{4.18}$$

where u_0 is the cut-off frequency of the prototype. The parameters t can then be chosen to minimize some function of $E_1(u_1)$, such as its integral square value (L_2 approximation) or maximum absolute value (L_∞ or Chebyshev approximation). There are two difficulties with this formulation. First, the error function is not a linear function of known parameters; thus non-linear optimization routines must be used for the minimization. Second, for transformations other than those of first order, an explicit relation for G cannot be found.

A suboptimum approach reformulates the problem as a linear approximation problem. If the mapping were exact then as the circular contour was traversed, the value of $F(u_1,u_2)$ would be constant. This would result in:

$$\cos u_0 = t(0,0) + t(1,0)\cos u_1 + t(0,1)\cos\sqrt{R^2-u_1^2} + t(1,1)\cos u_1 \cos\sqrt{R^2-u_1^2} \tag{4.19}$$

where u_0 is the passband cut-off of the prototype. If the mapping is not exact, however, the equality in equation (4.19) will only be approximate and an error function can be defined as:

$$E_2(u_1) = \cos u_0 - t(0,0) - t(1,0)\cos u_1 - t(0,1)\cos\sqrt{R^2-u_1^2} - (1 - t(0,0) - t(0,1) - t(1,0))\cos u_1 \cos\sqrt{R^2-u_1^2} \tag{4.20}$$

This error is now a linear function of $t(0,0)$, $t(0,1)$ and $t(1,0)$, and thus linear optimization routines can be used to minimize the integral squared error or the maximum absolute error. The former problem can be solved trivially using a classical least squares formulation, and the parameters for minimizing the maximum absolute error can be found by linear programming.

The above transformation has been used by Bernabo et al. [31] in designing (2-D) zero-phase recursive digital filters.

The transfer function of a (2-D) zero-phase filter can be written as:

$$H(e^{ju_1},e^{ju_2}) = \frac{\sum_{m_1=0}^{M_1}\sum_{m_2=0}^{M_2} p(m_1,m_2)\cos m_1u_1 \cos m_2u_2}{\sum_{\ell_1=0}^{L_1}\sum_{\ell_2=0}^{L_2} q(\ell_1,\ell_2)\cos \ell_1u_1 \cos \ell_2u_2} \tag{4.21}$$

The procedure to be described here is based on transforming the squared magnitude frequency response of a (1-D) recursive digital filter

$$H(e^{ju}) = \frac{\sum_{n=0}^{N} a(n)\cos nu}{\sum_{m=0}^{M} b(m)\cos mu} \tag{4.22}$$

which can be written as

$$H(e^{ju}) = \frac{\sum_{n=0}^{N} a'(n)\left[\cos u\right]^n}{\sum_{m=0}^{M} b'(m)\left[\cos u\right]^m} \tag{4.23}$$

into a (2-D) function by the change of variable of the form

$$\cos u = \sum_{p_1=0}^{1}\sum_{p_2=0}^{1} t(p_1,p_2)\cos p_1u_1 \cos p_2u_2 \tag{4.24}$$

It can be easily seen by substituting equation (4.24) into equation (4.23) that a (2-D) zero-phase filter of the form of equation (4.21) will result.

Since the designed filter has zero-phase property, implying instability [26] , the decomposition technique of Pistor [26] should be applied to the obtained filter coefficients array p and q to find the coefficients of the single quadrant filters.

4.5 THE NEW DESIGN TECHNIQUE FOR (2-D) RECURSIVE FILTERS [64]

The proposed design technique now to be described is based on the use of a second order two-variable reactance function as a transformation applied to a one-dimensional low-pass filter.

Before describing the method in detail some useful definitions relevant to its understanding are given.

Definition 1 [66] : A function $f(s_1,s_2)$ is said to be 'analytic' at a point (s_{10},s_{20}) if f has a total differential at that point.

Definition 2 : A function $f(s_1,s_2)$ is said to be a positive function if in the domain $Re\ s_1 > 0$ and $R_e\ s_2 > 0$ f is analytic.

Definition 3 : A positive function $f(s_1,s_2)$ is said to be a two-variable positive real function, or to be positive real, if the positive function f is real for s_1 and s_2 real.

Definition 4 : A function $f(s_1,s_2)$ is said to be a two-variable reactance function [65] if f is positive real and if $f+f_*=0$. (The lower asterisk denotes the substitution of $-s_1$ and $-s_2$ for s_1 and s_2 respectively.)

A second order, two-variable reactance function can be written as:

$$f(s_1,s_2) = \frac{a_1 s_1+a_2 s_2}{1 + b s_1 s_2} \tag{4.25}$$

Definition 5 : The z-transform of a (2-D) zero-phase filter (1.13) which has symmetry with respect to both the u_1 and u_2 axis, may be obtained by cascading four one-quadrant recursive filters which recurse in different directions:

$$H(z_1,z_2) = F(z_1,z_2)F(z_1^{-1},z_2)F(z_1^{-1},z_2^{-1})F(z_1,z_2^{-1}) \tag{4.26}$$

The proposed method makes use of the second order two-variable reactance function, equation (4.25) as a transformation applied to a one-dimensional low-pass filter of the form:

$$H(s) = \sum_{i=1}^{N} \frac{A_i}{s+B_i} \tag{4.27}$$

to realize a first-quadrant, two-dimensional recursive filter. Second, third, and fourth quadrant filters are obtained by the same transformation but replacing s_1 by $-s_1$ for second, s_1 by $-s_1$ and s_2 by $-s_2$ for the third, and s_2 by $-s_2$ for the fourth quadrant filter. These four one-quadrant sections are cascaded and the bilinear transformation is used in order to obtain the digital version of the filter.

The cut-off boundary of the filter depends on the choice of the cut-off frequency of the one-dimensional filter, equation (4.27), and the coefficients of the two-variable reactance function equation (4.25). The resulting filter is a zero-phase recursive filter having symmetry with respect to the u_1 and u_2 axis. Since the z-transform of the filter has already been decomposed into four one-quadrant filters, the decomposition technique of Pistor [26] is avoided. All that remains is to prove that the resulting filter is stable.

The initial low-pass function equation (4.27) is positive real and so is the reactance function equation (4.25). It will now be proved that the use of a positive real function as a transformation applied to another positive real function leads to a positive real function. We will show that the two necessary and sufficient conditions that a positive real function must satisfy are preserved under this transformation.

(i) Positiveness

If $F(s_1,s_2)$ is positive real, that is $\text{Re } F(s_1,s_2) > 0$ for $\text{Re } s_1 > 0$ and $\text{Re } s_2 > 0$, and also $H(s)$ is positive real, namely $\text{Re } H(s) > 0$ for $\text{Re } s > 0$, it follows that $\text{Re } H[F(s_1,s_2)] > 0$ for $\text{Re } s_1 > 0$ and $\text{Re } s_2 > 0$.

(ii) Realness

If $F(s_1,s_2)$ is real for s_1 and s_2 real and $H(s)$ is real for s real, it follows that $H(F(s_1,s_2))$ is real for s_1 and s_2 real. Also if $F(s_1,s_2)$ is analytic in $\text{Re } s_1 \geqslant 0$ and $\text{Re } s_2 \geqslant 0$ and $H(s)$ is analytic in $\text{Re } s \geqslant 0$, then $H[F(s_1,s_2)]$ is also analytic in $\text{Re } s_1 \geqslant 0$ and $\text{Re } s_2 \geqslant 0$ [66]. Hence, the stability of the filter obtained by the new design technique is guaranteed.

Another way of justifying the use of the two-variable reactance function is that a digital filter with z-transform $H(z_1,z_2)$ obtained by performing a bilinear transformation on a network function $H(s_1,s_2)$ of a two-variable passive network is at worst marginally stable [67]. A two-variable reactance function has been shown [65,68] to be realizable asthe imittance of a finite passive network. Since the function obtained by using a two-variable reactance function, as a frequency transformation applied to another positive real function is also a positive real function as has been proved above, it will also be realizable with passive elements. Therefore filters with a transfer function derived in this way will be stable.

To illustrate the proposed design technique several low-pass (2-D) recursive digital filters will be designed using third order Butterworth, Chebyshev and elliptic filters as low-pass prototypes. Figs. (4.1)-(4.6) are amplitude spectrum and contour plot of (2-D) recursive digital filters obtained from a third order Butterworth low-pass prototype with a_1 and a_2 coefficients of the numerator of the two-variable reactance function set to unity and with different values for the denominator's coefficient, b. Fig. (4.7) is the amplitude spectrum of a (2-D) recursive digital filter obtained from a third order Chebyshev filter with a_1 and a_2 set to unity and b = 0.1. Fig. (4.8) is the amplitude spectrum of a (2-D) recursive digital filter resulting from a third order elliptic low-pass prototype with a_1 and a_2 set to unity and b = 0.2.

4.6 TRANSFORMATION TECHNIQUE FOR (N-D) FIR FILTERS [71]

This is a generalisation of McClellan Transformation [15,69] to (N-D). Considering a (1-D) FIR filter of odd length with zero-phase property as expressed in equation (4.8):

$$H(e^{ju}) = h(0) + \sum_{n=0}^{N} h(n)\left[e^{-ju_n} + e^{ju_n}\right]$$

$$= h(0) + \sum_{n=1}^{N} 2h(n)\cos un \qquad (4.28)$$

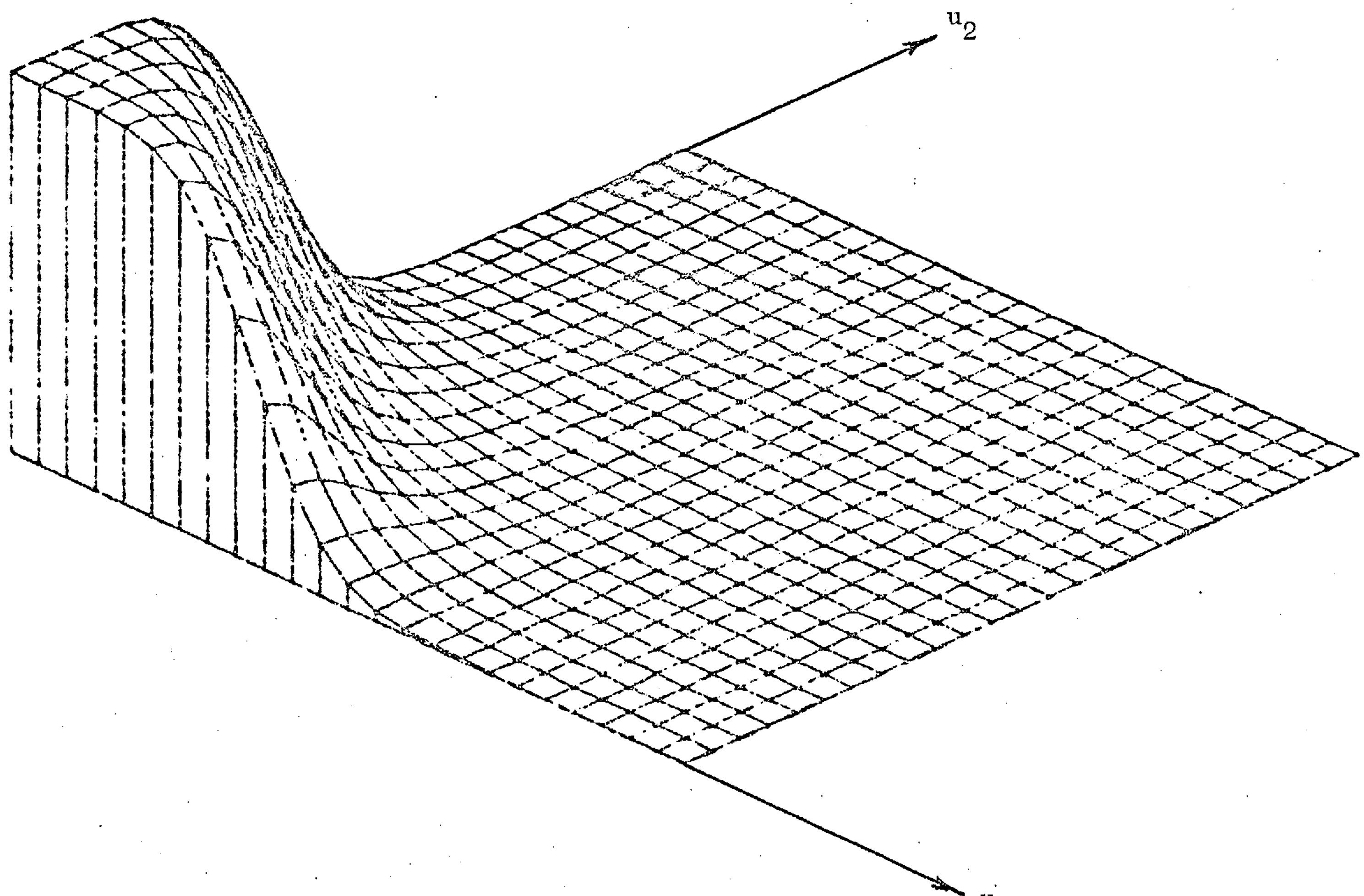

Fig. (4.1): Amplitude spectrum of a (2-D) recursive digital filter obtained from a third order Butterworth (1-D) filter with numerator coefficients of reactance transformation function $a_1=a_2=1$ and denominator coefficient $b = 0.1$.

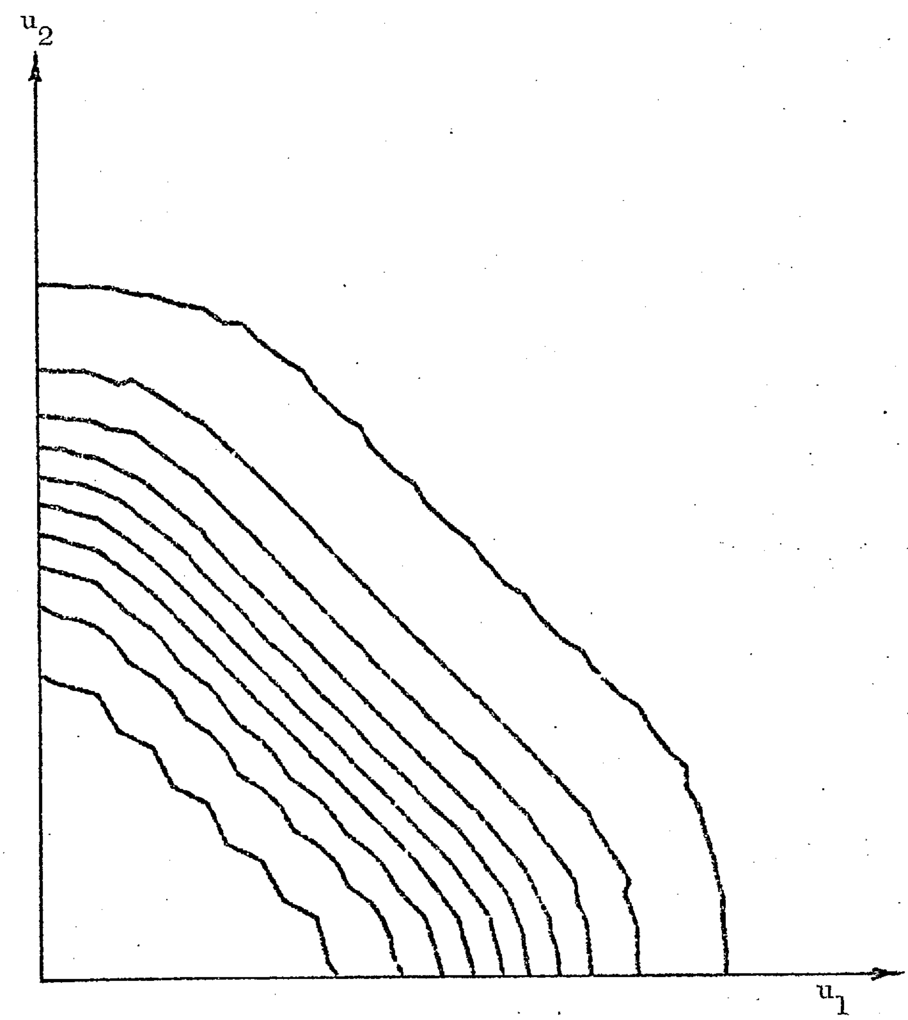

Fig. (4.2): Contour plot of the amplitude spectrum of a (2-D) recursive filter obtained from a third order Butterworth (1-D) filter with $a_1 = a_2 = 1$ and $b = 0.1$.

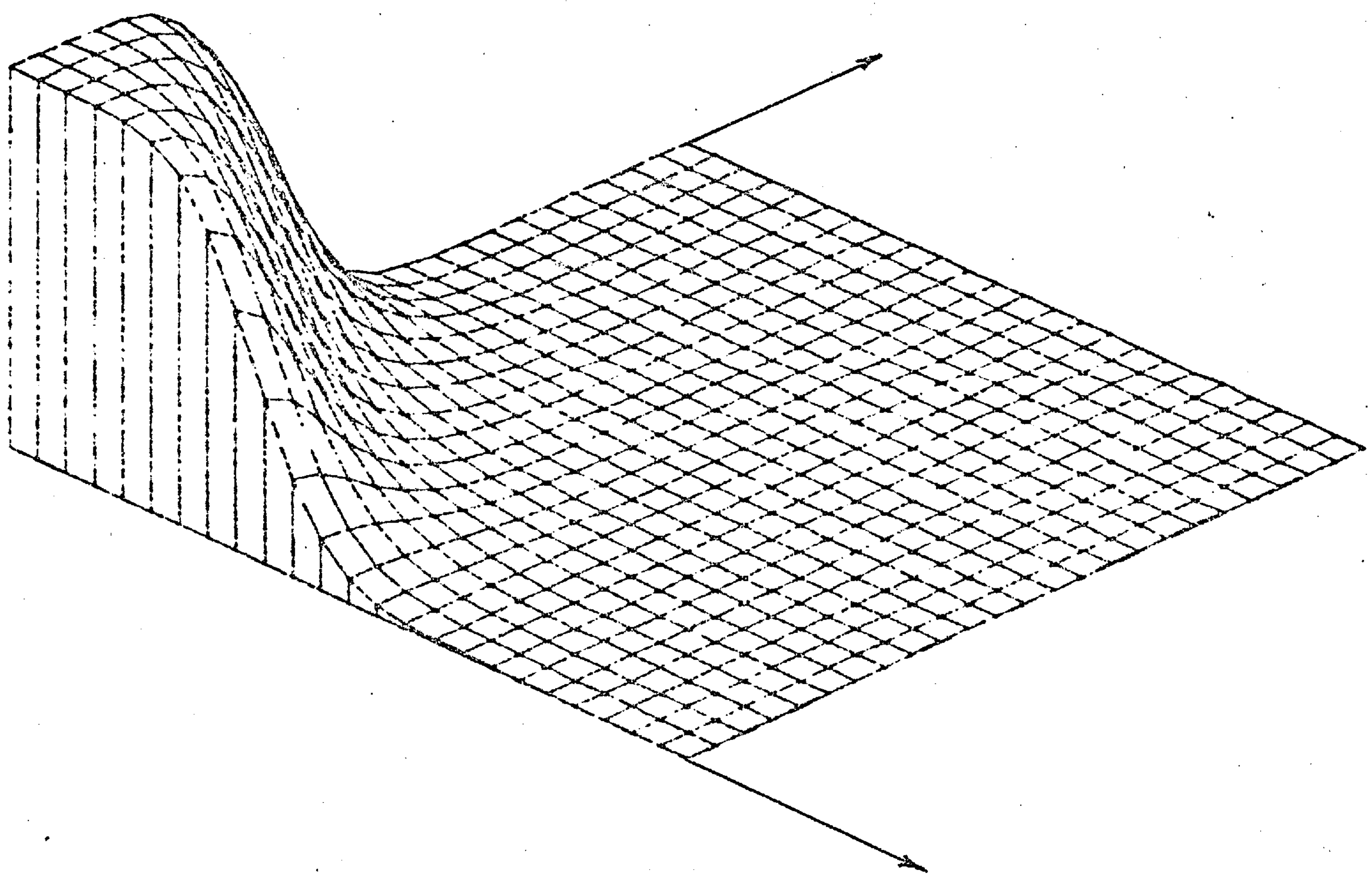

Fig. (4.3): Amplitude spectrum of a (2-D) recursive filter obtained from a (1-D) third order Butterworth filter with $a_1 = a_2 = 1$ and $b = 0.2$.

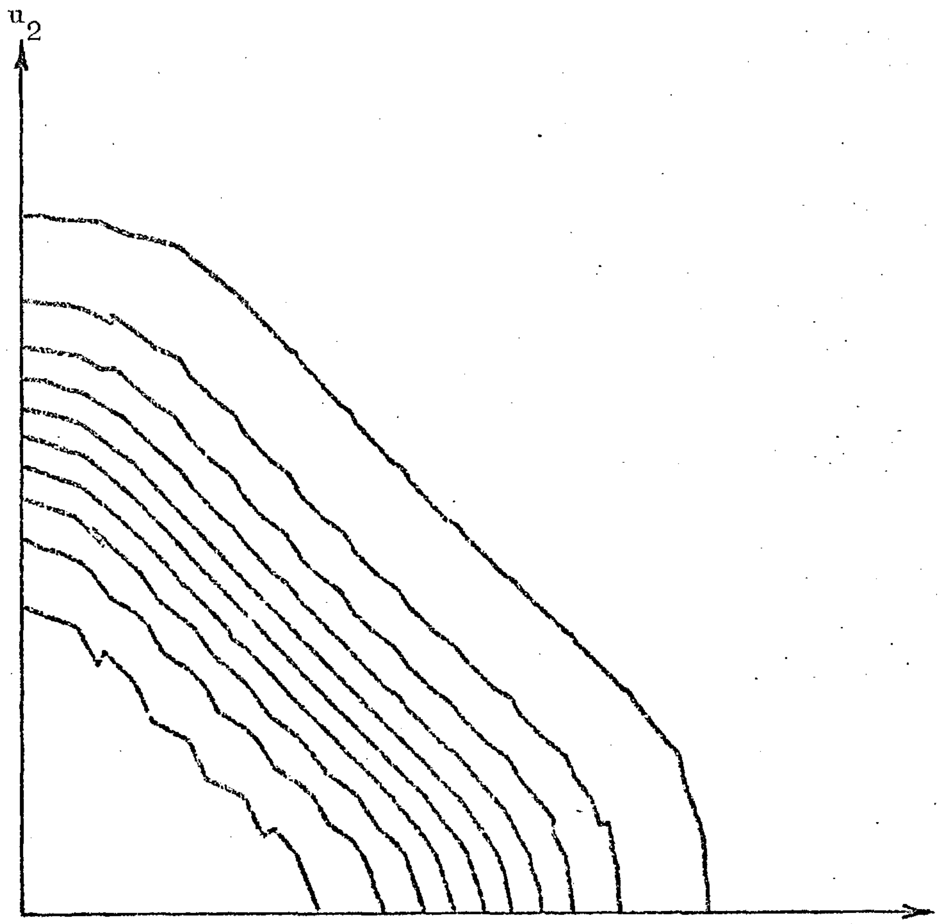

Fig. (4.4): Contour plot of a (2-D) recursive digital filter obtained from a third order Butterworth (1-D) filter with $a_1 = a_2 = 1$ and $b = 0.2$.

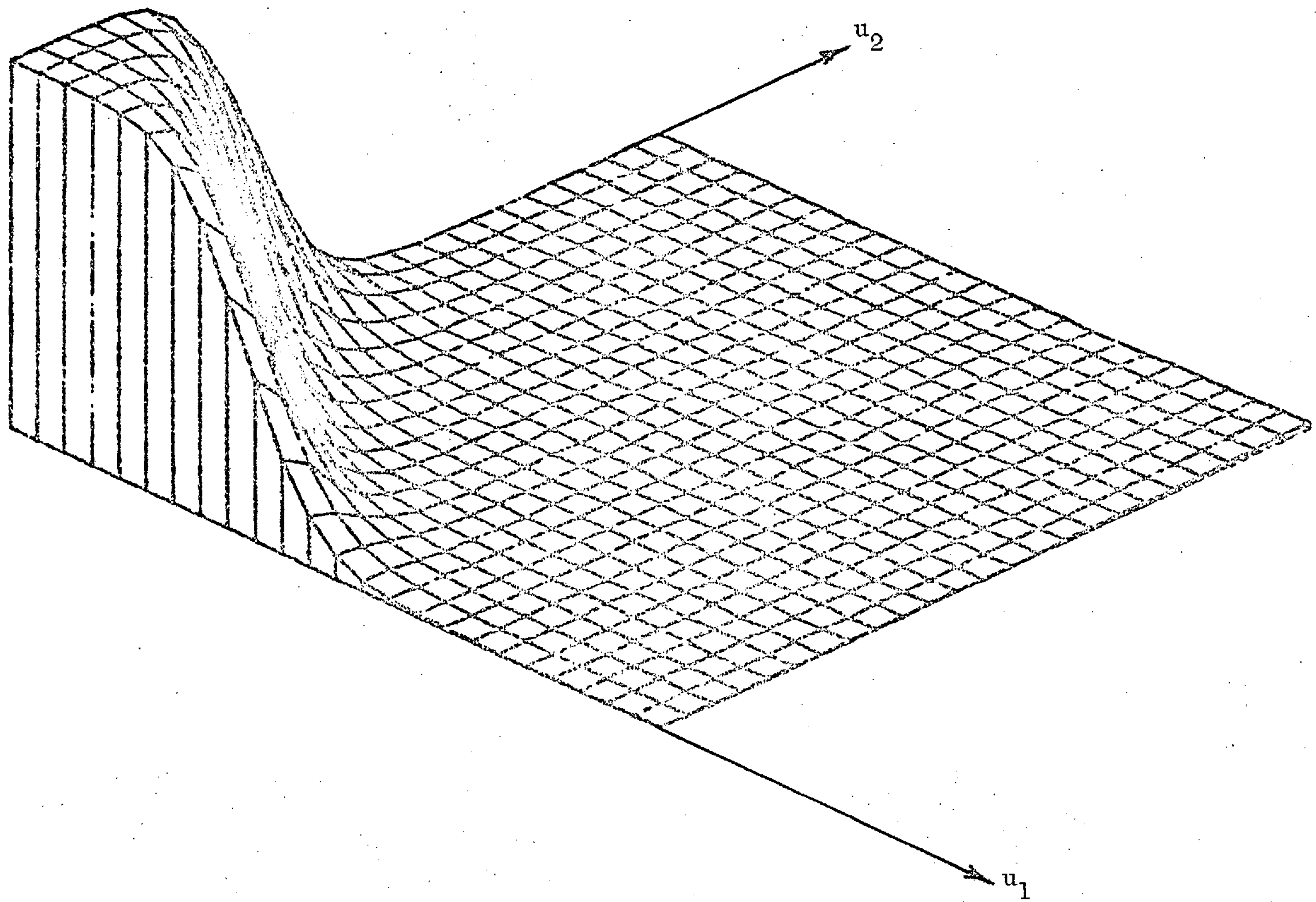

Fig. (4.5): Amplitude spectrum of a (2-D) recursive filter obtained from a third order Butterworth filter with $a_1 = a_2 = 1$ and $b = 0.6$.

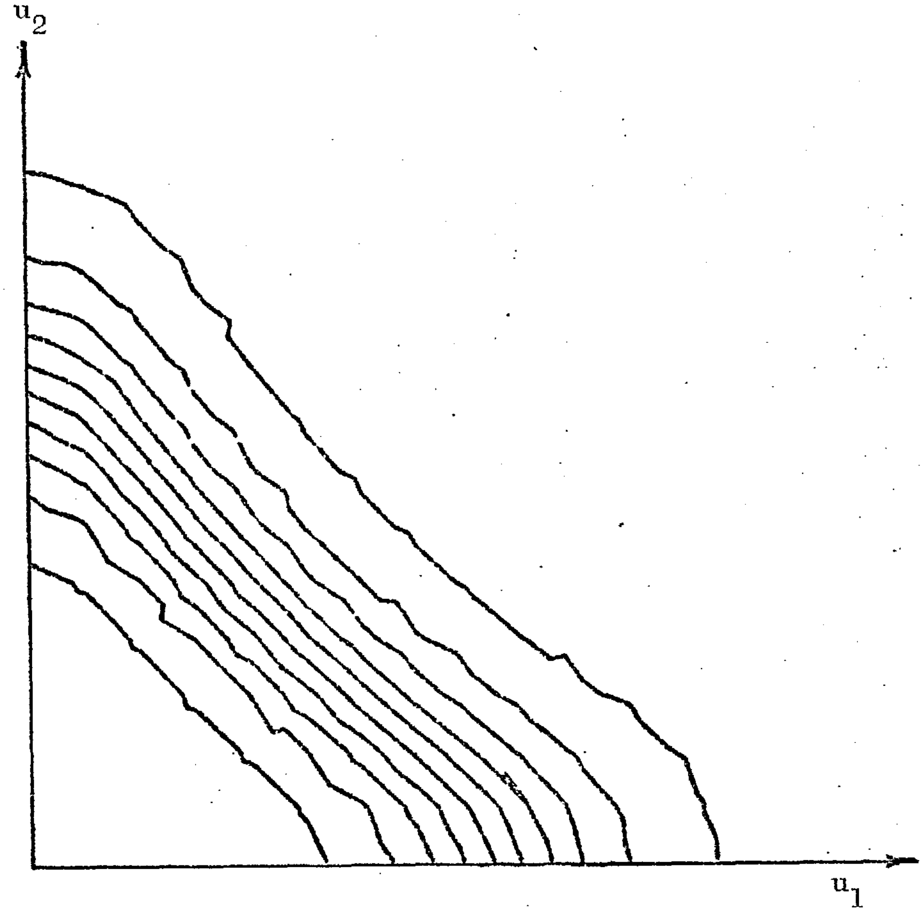

Fig. (4.6): Contour plot of the amplitude spectrum of a (2-D) recursive filter obtained from a third order Butterworth (1-D) filter with $a_1 = a_2 = 1$ and $b = 0.6$.

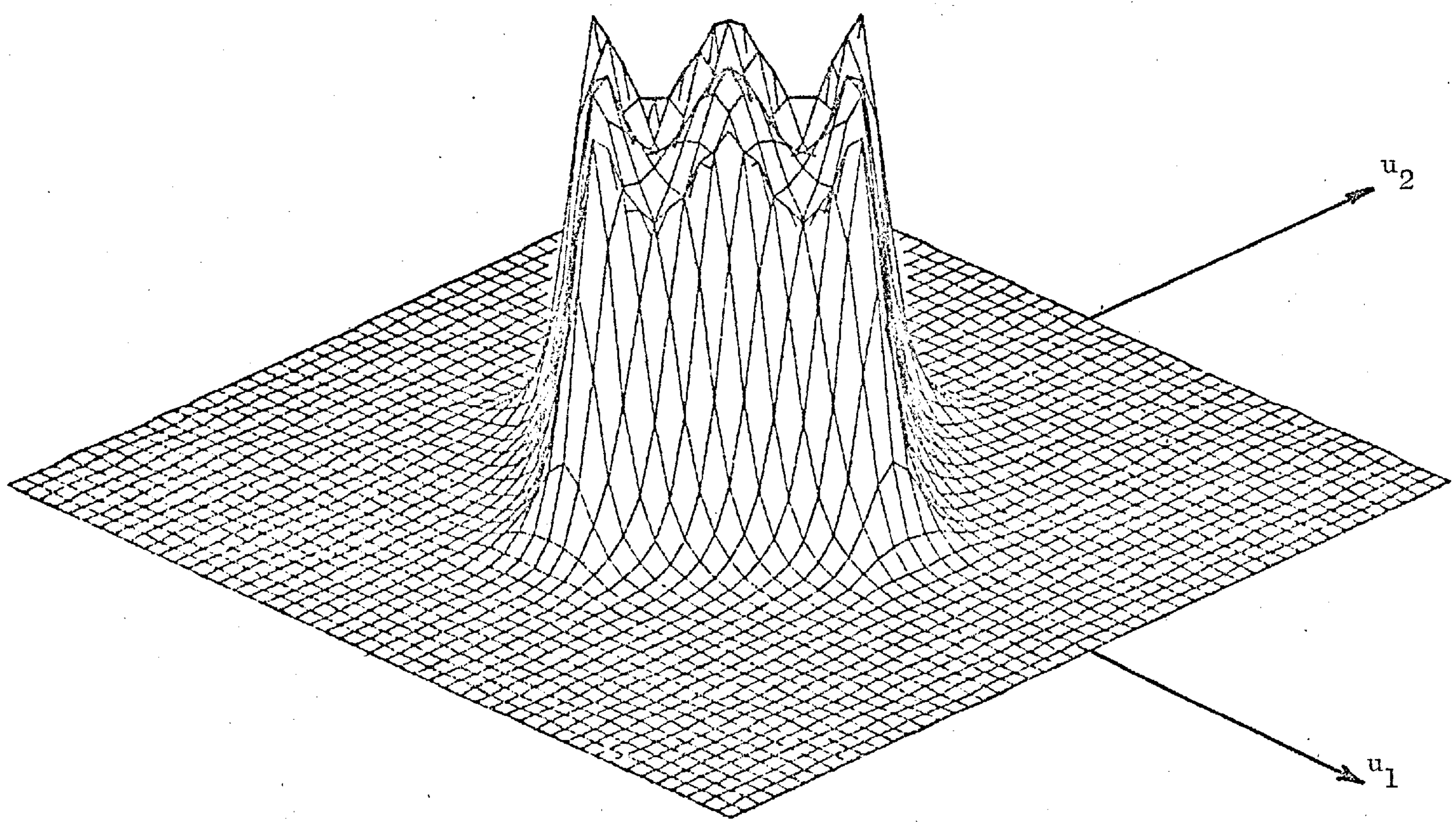

Fig. (4.7): Amplitude spectrum of a (2-D) recursive filter obtained from a (1-D) third order Chebyshev filter with a_1 and a_2 coefficients of the numerator of the 2-variable reactance function set to unity and $b = 0.1$.

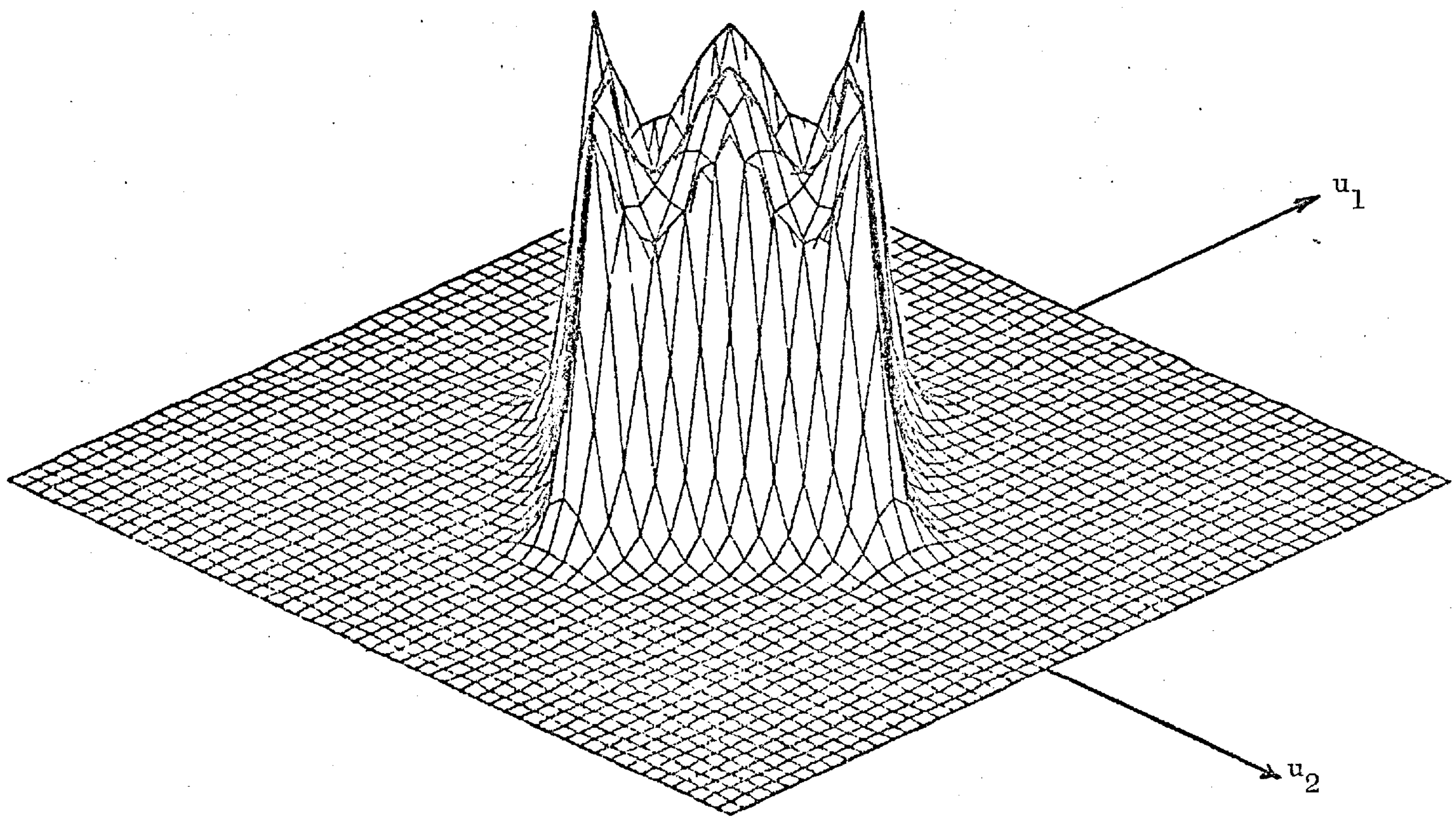

Fig. (4.8): Amplitude spectrum of a (2-D) recursive filter obtained from a (1-D) third order elliptic filter with a_1 and a_2 coefficients of the numerator of the two-variable reactance function set to unity, and $b = 0.2$.

Now, some (N-D) zero-phase FIR filters have a frequency response of the form:

$$H(e^{ju},\dots,e^{ju_n}) = \sum_{m_1=0}^{M_1}\dots\sum_{m_n=0}^{M_n} a(m_1,\dots,m_n)\cos m_1u_1 ,\dots,\cos m_nu_n \tag{4.29}$$

where equation (4.29) is not the most general class of such filters, but does cover many useful ones, and will be the only class to be considered here. The impulse response of such a system $h(m_1,\dots,m_n)$ is a real sequence which is an even function of each of its arguments.

The proposed transformation employs the substitution:

$$\cos u = \sum_{p_1=0}^{P_1}\dots\sum_{p_n=0}^{P_n} t(p_1,\dots,p_n)\cos p_1u_1,\dots,\cos p_nu_n \tag{4.30}$$

Hence, the relation between $h(n)$ and $h(m_1,\dots,m_n)$ can be seen by rewriting equation (4.28) as:

$$H(e^{ju}) = \sum_{n=0}^{N} b(n)\left[\cos u\right]^n \tag{4.31}$$

and performing the substitution indicated by equation (4.30) results in:

$$H(e^{ju_1},\dots,e^{ju_n}) = \sum_{n=0}^{N} b(n)\left[\sum_{p_1=0}^{P_1}\dots\sum_{p_n=0}^{P_n} t(p_1,\dots,p_n). \cos p_1u_1 ,\dots,\cos p_nu_n\right]^n \tag{4.32}$$

Furthermore, by exploiting the recurrence formula for Chebyshev polynomials, this can be rewritten in the form of equation (4.29). The (N-D) filter resulting from the transformation is of size

$(2NP_1+1) \times (2NP_2+1) \times \ldots \times (2NP_n+1)$, but only the case when $\bigcap_{i=1}^{n} P_i = 1$ is to be considered here.

As seen in equation (4.30), points on the frequency response of the (1-D) filter are mapped into contours in the $(u_1,\ldots,u_n)$ plane. The parameters $t(p_1,\ldots,p_n)$ determine the shape of these contours, while the impulse response $h(n)$ of the (1-D) filter controls the variation of the frequency response from one contour to another.

The problem of designing filters by this transformation thus divides into two parts: the determination of the contour parameters $\{t(p_1,\ldots,p_n)\}$, and the design of the (1-D) prototype filter $h(n)$. The former will be dealt with next.

4.6.1 The Contour Approximation Problem

The parameters $\{t(p_1,\ldots,p_n)\}$ must be chosen such that the transformation

$$\cos u = \sum_{p_1=0}^{P_1} \cdots \sum_{p_n=0}^{P_n} t(p_1,\ldots,p_n) \cos p_1 u_1, \ldots, \cos p_n u_n$$

$$= F(u_1,\ldots,u_n) \tag{4.33}$$

produces contours (constant u) having some desired shape in the $(u_1,\ldots,u_n)$ plane. As an example, consider the design of an (N-D) low-pass filter with an (N-D) spherical cut-off boundary. The contour in the frequency plane to which the pass-band edge of the (1-D) prototype should be mapped is then described by:

$$\sum_{i=1}^{N} u_i^2 = R^2 \tag{4.34}$$

and a possible first-order transformation is of the form

$$\cos u = \sum_{p_1=0}^{1} \cdots \sum_{p_n=0}^{1} t(p_1,\ldots,p_n) \cos p_1 u_1 \ldots \cos p_n u_n \tag{4.35}$$

If both the prototype and (N-D) filters are low-pass, then, depending on the shape of the cut-off boundary, it may be necessary to map the (1-D) origin into the (N-D) origin. In the case being considered, that of a filter with spherical boundary, this origin mapping is mandatory. Thus, the constraint equation:

$$\sum_{p_1=0}^{1} \cdots \sum_{p_n=0}^{1} t(p_1,\ldots,p_n) = 1 \tag{4.36}$$

results. This means that one variable, say $t(1,1,\ldots,1)$, is constrained to be a function of the rest, with the remainder of the variables unconstrained. To find values for these free variables, equation (4.33) can be solved for u_n in terms of $u, u_1,\ldots,u_{n-1}$ and the free mapping parameter. This yields:

$$u_n = G(u, u_1,\ldots, u_{n-1}, \hat{t}) \tag{4.37}$$

where $\hat{t}$ is the free variable parameter set, or:

$$u_n = \cos^{-1}\left(\frac{P(u_1,\ldots,u_{n-1},\hat{t})}{G(u_1,\ldots,u_{n-1},\hat{t})}\right) \tag{4.38}$$

An error function can be defined as:

$$E_1(u_1,\ldots,u_{n-1}) = G(u_0,u_1,\ldots,u_{n-1},\hat{t}) - \sqrt{R^2 - \sum_{i=1}^{N-1} u_i^2} \tag{4.39}$$

where u_0 is the cut-off frequency of the prototype. $\hat{t}$ can then be chosen to minimise some function of E_1 such as its integral square value (L_2 approximation) or its maximum absolute value (L_∞ or Chebychev approximation). However, this may prove to be difficult to carry out since the error function is not a linear function of the unknown parameters, and therefore non-linear optimization routines must be used for the minimization. Also, there can be no explicit expression for G in transformations other than first-order ones, i.e. other than those for which $(p_i;\ i=1,n) \geqslant 1$.

An alternative is a suboptimum approach that would reformulate the problem as that of a linear approximation. If the mapping were exact, then the value of $F(u_1,\ldots,u_n)$ would be constant as the spherical contour is traversed. This can be expressed as:

$$\begin{aligned} \cos u_o &= F_1(u_1,\ldots,u_{n-1},t) \\ &= t(0,\ldots,0) + t(1,\ldots,0)\cos u_1 + \ldots + \\ &\quad t(0,\ldots,1)\cos\sqrt{R^2-\sum_{i=1}^{N-1} u_i^2} + t(1,1,\ldots,0)\cos u_1 \cos u_2 + \ldots \\ &\quad \ldots + t(1,1,\ldots,1)\cos u_1 \cos u_2 \ldots \cos\sqrt{R^2-\sum_{i=1}^{N-1} u_i^2} \end{aligned} \tag{4.40}$$

where u_o is the pass-band cut-off of the prototype. On the other hand, the equality in equation (4.40) would be only approximate if the mapping were not exact. The error function could then be defined as:

$$E_2(u_1,\ldots,u_{n-1}) = \cos u_o - F_1(u_1,\ldots,u_{n-1},\hat{t})$$

$$= \cos u_o - t(0,0,\ldots,0) - t(1,0,\ldots,0)\cos u_1 \;\cdot\cdot$$

$$\cdot\cdot - t(0,0,\ldots,1)\cos\sqrt{R^2-\sum_{i=1}^{N-1} u_i^2} - t(1,1,\ldots,0)\cos u_1 \cos u_2 \;\cdot\cdot$$

$$\cdot\cdot - (1-t(0,0,\ldots,0) - t(0,1,\ldots,0) - \ldots - t(1,1,\ldots,0)).$$

$$.\cos u_1 \cos u_2 \cdot\cdot \cos\sqrt{R^2-\sum_{i=1}^{N-1} u_i^2} \tag{4.41}$$

The error has thus been made into a linear function of $\hat{t}$ enabling linear optimization routines to be used to minimize the integral square or the maximum absolute error. In the event of choosing the integral square minimization, a classical least squares formulation can be used. While for minimizing the maximum absolute error, the parameters can be determined through linear programming.

Intuitively, a transformation is only useful if the coefficients $t(p_1,\ldots,p_n)$ of the mapping were to satisfy the relation:

$$\left|\sum_{p_1=0}^{1}\cdots\sum_{p_n=0}^{1} t(p_1,\ldots,p_n)\cos p_1u_1\ldots\cos p_nu_n\right| \leqslant 1 \qquad 0 \leqslant u_i \leqslant \pi \qquad i = 1,2,\ldots,n \tag{4.42}$$

Therefore, a linear scaling of the design parameters must be applied so that equation (4.42) above holds. Using scaling factors defined by:

$$F'(u_1,u_2,\ldots,u_n) = C_1F(u_1,u_2,\ldots,u_n) - C_2 \tag{4.43}$$

for a non-zero C_1 and any C_2, the shape of the contours $F(u_1,\ldots,u_n) = \text{constant}$ will remain unchanged. Hence, if F_{max} denotes the maximum value of $F(u_1,\ldots,u_n)$ and F_{min} denotes its minimum value, then by choosing:

$$\text{and} \qquad \begin{aligned} C_1 &= 2/(F_{max}-F_{min}) \\ C_2 &= C_1F_{max}-1 \end{aligned} \tag{4.44}$$

the condition

$$\left|F'(u_1,\ldots,u_n)\right| = \left|\sum_{p_1=0}^{1}\ldots\sum_{p_n=0}^{1} t'(p_1,\ldots,p_n)\cos p_1u_1 \ldots \cos p_nu_n\right| \leqslant 1 \tag{4.45}$$

is always satisfied. The shape of the contours remains unchanged, as desired, and all that changes is the (1-D) frequency that is associated with each contour. In particular, if $F(u_1,\ldots,u_n)$ were originally associated with the value u, then $F'(u_1,\ldots,u_n)$ defined by equation (4.42) will be associated with the value u' where:

$$u' = \cos^{-1}(C_1\cos u - C_2) \tag{4.45}$$

This implies that, in most cases, the (1-D) prototype must be designed after the transformation has been found.

4.7 DESIGN TECHNIQUES FOR (N-D) ZERO-PHASE RECURSIVE FILTERS

The proposed design technique which will now be described is an extension to the work of Bernabo et al. [31]. It is based on the transformation of the magnitude square of the frequency response of a (1-D) digital filter by using an extension of McClellan's transformation to (N-D) described above, followed by decomposition technique of Ahmadi and King [50,63] to obtain 2^N single quadrant filters.

4.7.1 Transformation of (1-D) Filter to (N-D)

An N-dimensional first quadrant recursive filter is defined by its z-transfer function:

$$H(z_1,\ldots,z_n) = \frac{\sum_{m_1=0}^{M_1}\cdots\sum_{m_n=0}^{M_n} a(m_1,\ldots,m_n) z_1^{m_1}\ldots z_n^{m_n}}{\sum_{m_1=0}^{M_1}\cdots\sum_{m_n=0}^{M_n} b(m_1,\ldots,m_n) z_1^{m_1}\ldots z_n^{m_n}} \tag{4.46}$$

A zero-phase filter can be written [50]:

$$H(e^{ju_1},\ldots,e^{ju_n}) = \frac{\sum_{m_1=1}^{M_1}\cdots\sum_{m_n=0}^{M_1} p(m_1,\ldots,m_n)\cos m_1u_1 \ldots \cos m_nu_n}{\sum_{\ell_1=0}^{L_1}\cdots\sum_{\ell_n=0}^{L_n} q(\ell_1,\ldots,\ell_n)\cos \ell_1u_1\ldots\cos \ell_nu_n} \tag{4.47}$$

The transformation to be described here is based on transforming the squared magnitude frequency response of a (1-D) recursive digital filter

$$H(e^{ju}) = \frac{\sum_{n=0}^{N} a(n)\cos nu}{\sum_{m=0}^{M} b(m)\cos mu} \tag{4.48}$$

which can alternatively be written as:

$$H(e^{ju}) = \frac{\sum_{n=0}^{N} a'(n)\left[\cos u\right]^{n}}{\sum_{m=0}^{M} b'(m)\left[\cos u\right]^{m}} \tag{4.49}$$

into a (N-D) function by the change of variable of the form:

$$\cos u = \sum_{p_1=0}^{P_1} \cdots \sum_{p_n=0}^{P_n} t(p_1,\ldots,p_n)\cos p_1u_1 \,..\cos p_nu_n \qquad (4.50)$$

Substituting equation (4.50) into equation (4.49) will result in:

$$H(e^{ju_1},\ldots,e^{ju_2}) = \frac{\sum_{n=0}^{N} a'(n)\left[\sum_{p_1=0}^{P_1}\cdots\sum_{p_n=0}^{P_n} t(p_1,\ldots,p_n)\cos p_1u_1 \,..\cos p_nu_n\right]^n}{\sum_{m=0}^{M} b'(m)\left[\sum_{p_1=0}^{P_1}\cdots\sum_{p_n=0}^{P_n} t(p_1,\ldots,p_n)\cos p_1u_1 \,..\cos p_nu_n\right]^m} \qquad (4.51)$$

Furthermore, by exploiting the recurrence formula for Chebyshev polynomials, this can be written in the form of equation (4.47). The (N-D) filter resulting from the transformation is of size $(2NP_1+1)$ x $(2NP_2+1)$ x ... x $(2NP_n+1)$ in the numerator and $(2MP_1+1)$ x $(2MP_2+1)$ x .. .. x $(2MP_n+1)$ in the denominator, but only the case when $\bigcap_{i=1}^{n} P_i = 1$ is to be considered here.

The design procedure is divided into three parts. First, design of the (1-D) filter prototype; second, calculating the parameter t such that the contour u = constant is an approximation to the desired cut-off boundary in a least square sense. Finally, because the zero-phase property of the design filter implies instability [50,63] , a decomposition technique should be applied to the obtained filter coefficients array p and q to find the quadrants filter coefficients a and b, equation (1.6).

The decomposition technique of Ahmadi and King [50,63] can be used to decompose the obtained unstable filter into 2^N stable-single quadrant filters which each recurse in a different direction. This can be done by evaluating the cepstrum of the filter array and by dividing it into the sum of 2^N arrays.

CHAPTER FIVE

SPATIAL DESIGN TECHNIQUES

In this chapter, several spatial design techniques for (2-D) recursive digital filters are reviewed, and their extensions to the (N-D) case are derived wherever possible.

5.1 THE SHANKS' METHOD

This is an extension of the time domain design technique in one dimension [62,72]. A filter is designed by this technique so as to give an impulse response that approximates some given desired response.

5.1.1 The (2-D) Case [20]

Given the desired impulse response d_{ℓ_1,ℓ_2} if a (2-D) recursive filter for $\ell_1 = 1,2,\ldots,L_1$ and $\ell_2 = 1,2,\ldots,L_2$, its z-transform is then:

$$D(z_1,z_2) = \sum_{\ell_1=1}^{L_1+1} \sum_{\ell_2=1}^{L_2+1} d_{\ell_1,\ell_2} z_1^{\ell_1-1} z_2^{\ell_2-1} \tag{5.1}$$

while the approximating (2-D) recursive filter will always be of the form of equation (1.5):

$$F(z_1,z_2) = \frac{A(z_1,z_2)}{B(z_1,z_2)} = \frac{\displaystyle\sum_{j_1=1}^{M_1+1} \sum_{j_2=1}^{M_2+1} a_{j_1,j_2} z_1^{j_1-1} z_2^{j_2-1}}{\displaystyle\sum_{j_1=1}^{N_1+1} \sum_{j_2=1}^{N_2+1} b_{j_1,j_2} z_1^{j_1-1} z_2^{j_2-1}} \tag{5.2}$$

where M_1, M_2, N_1 and N_2 are arbitrary (but fixed) parameters. Equation (5.2) above can be re-written as:

$$F(z_1,z_2)B(z_1,z_2) = A(z_1,z_2) \tag{5.3}$$

and since multiplication of the z-polynomials is equivalent to convolution of the arrays in the space domain, it follows that:

$$a_{j_1,j_2} = \sum_{m_1=1}^{N_1+1} \sum_{m_2=1}^{N_2+1} b_{m_1,m_2} \, f_{j_1-m_1+1,\, j_2-m_2+1} \tag{5.4}$$

The coefficients a_{j_1,j_2} are defined by equation (5.4) over the integers $j_1 = 1,2,\ldots,M_1$, $j_2 = 1,2,\ldots,M_2$. For all other values of j_1 and j_2, the a_{j_1,j_2} are zero. Therefore, if set of integer pairs S_a is defined as

$$S_a = \left\{(j_1,j_2) : 1 \leqslant j_1 \leqslant M_1, \quad 1 \leqslant j_2 \leqslant M_2\right\} \tag{5.5}$$

and another set $\hat{S}_a$ defined as the set of all other values of (j_1,j_2) greater than zero:

$$\hat{S}_a = \left\{(j_1,j_2) : j_1 > 0, \quad j_2 > 0, \quad (j_1,j_2) \notin S_a\right\} \tag{5.6}$$

then equation (5.4) can be re-written as:

$$f_{j_1,j_2} = -\sum_{m_1=1}^{N_1+1} \sum_{m_2=1}^{N_2+1} b_{m_1,m_2} \, f_{j_1-m_1+1,\, j_2-m_2+1} \tag{5.7}$$

$$m_1 \cdot m_2 \neq 0 \qquad \text{for } (j_1,j_2) \in \hat{S}_a$$

and when the b_{m_1,m_2} are judiciously chosen in equation (5.7) above, the f_{j_1,j_2} approximates the desired impulse response d_{ℓ_1,ℓ_2}. Thus:

$$d_{j_1,j_2} \cong -\sum_{m_1=1}^{N_1+1} \sum_{m_2=1}^{N_2+1} b_{m_1,m_2} \, d_{j_1-m_1+1,\, j_2-m_2+1} \tag{5.8}$$

$$m_1 \cdot m_2 \neq 1 \text{ for } (j_1,j_2) \in \left\{\hat{S}_a \cap S_d\right\}$$

where S_d is the set of integer pairs over which the desired filter is defined:

$$S_d = \{(\ell_1, \ell_2) : 1 \leqslant \ell_1 \leqslant L_1, \quad 1 \leqslant \ell_2 \leqslant L_2\} \tag{5.9}$$

An error e_{j_1,j_2} can be added to the right-hand side of equation (5.8) to produce the equality:

$$d_{j_1,j_2} = e_{j_1,j_2} - \sum_{m_1=1}^{N_1+1} \sum_{m_2=1}^{N_2+1} b_{m_1,m_2} . d_{j_1-m_1+1, j_2-m_2+1} \qquad (j_1,j_2) \in \{\hat{S}_a \cap S_d\} \tag{5.10}$$

which, with d_{j_1,j_2} moved to the summation, can be written as:

$$e_{j_1,j_2} = \sum_{m_1=1}^{N_1+1} \sum_{m_2=1}^{N_2+1} b_{m_1,m_2}\, d_{j_1-m_1+1, j_2-m_2+1} \qquad \text{for } (j_1,j_2) \in \{\hat{S}_a \cap S_d\} \tag{5.11}$$

The obvious choice of the b_{m_1,m_2} is that based on the minimisation of the mean-square-error:

$$e^2 = \sum_{j_1} \sum_{j_2} \left[\sum_{m_1} \sum_{m_2} b_{m_1,m_2}\, d_{j_1-m_1+1, j_2-m_2+1} \right]^2 \tag{5.12}$$

$$(j_1,j_2) \in \{\hat{S}_a \cap S_d\} \text{ and } (m_1,m_2) \in S_b$$

so that by differentiating equation (5.12) with respect to the b_{m_1,m_2} and setting the resultant equations equal to zero, the b_{m_1,m_2} that minimise e^2 above can be found. This involves solving $(N_1.N_2-1)$ equations of the type:

$$\sum_{m_1=1}^{N_1} \sum_{m_2=1}^{N_2} b_{m_1,m_2}\, \varphi_{k_1 k_2 m_1 m_2} = \varphi_{k_1 k_2} \qquad \text{for } k_1 = 1,2,\ldots,N_1; \; k_2 = 1,2,\ldots,N_2; \text{ but } k_1.k_2 \neq 1 \tag{5.13}$$

where:

$$\varphi_{k_1 k_2 m_1 m_2} = \sum_{j_1} \sum_{j_2} d_{j_1-m_1+1,\, j_2-m_2+1}\, d_{j_1-k_1+1,\, j_2-k_2+1} \qquad (5.14)$$

and

$$\varphi_{k_1,k_2} = -\sum_{j_1} \sum_{j_2} d_{j_1,j_2}\, d_{j_1-k_1+1,\, d_2-m_2+1} \quad (j_1,j_2) \in \{\hat{S}_a \cap S_d\} \qquad (5.15)$$

With the denominator $B(z_1,z_2)$ of the equation (5.2) computed, the numerator $A(z_1,z_2)$ must next be determined.

One way of doing this is to compute those coefficients of $A(z_1,z_2)$ that minimise the mean square difference between the coefficients of $F(z_1,z_2) = \frac{A(z_1,z_2)}{B(z_1,z_2)}$ and the coefficients of the desired response $D(z_1,z_2)$. This is a Wiener filtering problem in two dimensions. It consists of finding the optimum filter $A(z_1,z_2)$, given an input $\frac{1}{B(z_1,z_2)}$ and a desired output $D(z_1,z_2)$.

In a simpler but less accurate method the array B can be convolved with D giving the A array. Since $\frac{A(z_1,z_2)}{B(z_1,z_2)} = D(z_1,z_2)$ the coefficients a_{j_1,j_2} are computed from $A(z_1,z_2) = B(z_1,z_2).D(z_1,z_2)$ for $(j_1,j_2) \subset S_a$. This is easy to apply but has the disadvantage that the stability of the resulting filter cannot be guaranteed.

5.1.2 Extension to the (N-D) Case [73]

This is when the impulse response of an (N-D) filter $d_{\ell_1,\ldots,\ell_n}$ for $\ell_1 = 1,2,\ldots,L_1$, $\ell_2 = 1,2,\ldots,L_2,\ldots,$ and $\ell_n = 1,2,\ldots,L_n$ is given. That is:

$$D(z_1,\ldots,z_n) = \sum_{\ell_1=1}^{L_1+1}\cdots\sum_{\ell_n=1}^{L_n+1} d_{\ell_1,\ldots,\ell_n}\, z_1^{\ell_1-1}\cdots z_n^{\ell_n-1} \tag{5.16}$$

The approximating (N-D) recursive filter can then be assumed to be of the form (equation (1.6)):

$$F(z_1,\ldots,z_n) = \frac{A(z_1,\ldots,z_n)}{B(z_1,\ldots,z_n)} = \frac{\sum_{j_1=1}^{M_1+1}\cdots\sum_{j_n=1}^{M_n+1} a_{j_1,\ldots,j_n}\, z_1^{j_1-1}\cdots z_n^{j_n-1}}{\sum_{j_1=1}^{N_1+1}\cdots\sum_{j_n=1}^{N_n+1} b_{j_1,\ldots,j_n}\, z_1^{j_1-1}\cdots z_n^{j_n-1}}$$

$$= \sum_{j_1=1}^{\infty}\cdots\sum_{j_n=1}^{\infty} f_{j_1,\ldots,j_n}\, z_1^{j_1-1}\cdots z_n^{j_n-1} \tag{5.17}$$

where $M_1,M_2,\ldots,M_n$, $N_1,\ldots,N_n$ are arbitrary (but fixed) parameters. The numerator and denominator coefficients a and b should be calculated such that the $f_{j_1,\ldots,j_n}$ approximate the desired filter's impulse response $d_{\ell_1,\ldots,\ell_n}$ in a least-square sense.

Equation (5.17) can be re-written as:

$$F(z_1,\ldots,z_n).B(z_1,\ldots,z_n) = A(z_1,\ldots,z_n) \tag{5.18}$$

and since multiplication of the z-polynomial is equivalent to convolution of the arrays in the space domain:

$$a_{j_1,\ldots,j_n} = \sum_{m_1=1}^{N_1+1}\cdots\sum_{m_n=1}^{N_n+1} b_{m_1,\ldots,m_n}\, f_{j_1-m_1+1,\ldots,\, j_n-m_n+1} \quad \text{for } (j_1,j_2,\ldots,j_n) \in S_M \tag{5.19}$$

$$= 0 \qquad \text{for } (j_1,j_2,\ldots,j_n) \in \hat{S}_M$$

where S_M and $\hat{S}_M$ are the sets of integers defined by:

$$S_M = \left\{(j_i,\ i=1,n) : 0 \leqslant j_1 \leqslant M_1,\ \ 0 \leqslant j_2 \leqslant M_2,\ldots,0 \leqslant j_n \leqslant M_n\right\} \tag{5.20}$$

and:

$$\hat{S}_M = \left\{(j_i,\ i=1,n) : j_1 > 0,\ j_2 > 0, \ldots, j_n > 0,\ (j_i,\ i=1,n) \notin S_M\right\} \tag{5.21}$$

Therefore, assuming that $b(1,\ldots,1) = 1$ and using equation (5.19) gives:

$$f_{j_1,\ldots,j_n} = -\sum_{m_1=1}^{N_1+1} \cdots \sum_{m_n=1}^{N_n+1} b_{m_1,\ldots,m_n} f_{j_1-m_1+1,\ldots,j_n-m_n+1} \tag{5.22}$$

$$\prod_{i=1}^{n} m_i \neq 1 \quad \text{for } (j_i,\ i=1,n) \in \hat{S}_M$$

and when the $b_{m_1,\ldots,m_n}$ are judiciously chosen, the $f_{j_1,\ldots,j_n}$ will closely approximate the desired impulse response $d_{\ell_1,\ldots,\ell_n}$. Thus:

$$d_{j_1,\ldots,j_n} \cong -\sum_{m_1=1}^{N_1+1} \cdots \sum_{m_n=1}^{N_n+1} b_{m_1,\ldots,m_n} \cdot d_{j_1-m_1+1,\ldots,j_n-m_n+1} \tag{5.23}$$

$$\prod_{i=1}^{n} m_i \neq 1 \quad \text{for } (j_i,\ i=1,n) \in \left\{\hat{S}_M \cap S_L\right\}$$

where S_L is the set of integers defined as

$$S_L = \left\{(\ell_i,\ i=1,n) : 0 \leqslant \ell_1 \leqslant L_1,\ 0 \leqslant \ell_2 \leqslant L_2, \ldots, 0 \leqslant \ell_n \leqslant L_n\right\} \tag{5.24}$$

An error function $e_{j_1,\ldots,j_n}$ is defined as that which can be added to the right-hand-side of equation (5.23) yielding the equality:

$$d_{j_1,\ldots,j_n} = e_{j_1,\ldots,j_n} - \sum_{m_1=1}^{N_1+1} \cdots \sum_{m_n=1}^{N_n+1} b_{m_1,\ldots,m_n} \cdot d_{j_1-m_1+1,\ldots,\ j_n-m_n+1}$$

$$\prod_{i=1}^{n} m_i \neq 1 \text{ for } (j_i,\ i=1,n) \in \left\{\hat{S}_M \cap S_L\right\} \tag{5.25}$$

and therefore by moving the $d_{j_1,\ldots,j_n}$ to the summation:

$$e_{j_1,\ldots,j_n} = \sum_{m_1=1}^{N_1+1} \cdots \sum_{m_n=1}^{N_n+1} b_{m_1,\ldots,m_n} \, d_{j_1-m_1+1,\ldots,j_n-m_n+1}$$

$$\text{for } (j_i,\ i=1,n) \in \{\hat{S}_M \cap S_L\} \qquad (5.26)$$

The $b_{m_1,\ldots,m_n}$ are chosen such that the mean-square-error

$$e^2 = \sum_{j_1} \cdots \sum_{j_n} \left[\sum_{m_1=1} \cdots \sum_{m_n=1} b_{m_1,\ldots,m_n} \, d_{j_1-m_1+1,\ldots,j_n-m_n+1} \right]^2$$

$$(j_i,\ i=1,n) \in \{\hat{S}_M \cap S_L\} \qquad (5.27)$$

is minimised. This is so when the partial derivatives with respect to each $b(m_1,\ldots,m_n)$ are equal to zero. Differentiating e^2 with respect to $b(k_1,\ldots,k_n)$, where $k_1 = 1,2,\ldots,N_1$, $k_2 = 1,2,\ldots,N_2,\ldots$, $k_n = 1,2,\ldots,N_n$ and $\prod_{i=1}^{n} k_i \neq 1$, and setting each partial derivative to zero, gives:

$$\frac{\partial e^2}{\partial b(k_1,\ldots,k_n)} = \sum_{j_1} \cdots \sum_{j_2} 2 \left[\sum_{m_1=1}^{N_1+1} \cdots \sum_{m_n=1}^{N_n+1} (b_{m_1,\ldots,m_n} \cdot d_{j_1-m_1+1,\ldots,j_n-m_n+1}) \cdot (d_{j_1-k_1+1,\ldots,j_n-k_n+1}) \right] = 0$$

$$\text{for } (j_i,\ i=1,n) \in \{\hat{S}_M \cap S_L\} \qquad (5.28)$$

which can be written as

$$\sum_{m_1=1}^{N_1} \cdots \sum_{m_n=1}^{N_n} b_{m_1,\ldots,m_n} \, \varphi_{k_1,\ldots,k_n,\ m_1,\ldots,m_n} = \varphi_{k_1,\ldots,k_n}$$

$$\text{for } (k_i,\ i=1,n) = (N_i,\ i=1,n)$$

$$\text{but } \bigcap_{i=1}^{n} k_i \neq 1 \qquad (5.29)$$

where:

$$\varphi_{k_1,\ldots,k_n,m_1,\ldots,m_n} = \sum_{j_1} \cdots \sum_{j_n} d_{j_1-m_1+1,\ldots,j_n-m_n+1} \, d_{j_1-k_1+1,\ldots,j_n-k_n+1} \qquad (5.30)$$

and:

$$\varphi_{k_1,\ldots,k_n} = -\sum_{j_1}\cdots\sum_{j_n} d_{j_1,\ldots,j_n}\, d_{j_1-k_1+1,\ldots,j_n-k_n+1}$$

$$(j_i,\ i=1,n) \in \{\hat{S}_M \cap S_L\} \qquad (5.31)$$

Thus, equation (5.29) describes a set of $((N_1+1)(N_2+1)\ldots(N_n+1)-1)$ simultaneous linear equations with the same number of unknowns. The solution gives the $b(m_1,\ldots,m_n)$ coefficients that minimise the mean-square-error e^2.

After computing $B(z_1,z_2,\ldots,z_n)$ coefficients of the denominator in equation (5.23), the numerator coefficients $A(z_1,z_2,\ldots,z_n)$ should be calculated. In a similar way to the (2-D) case, the mean square difference between the coefficients of $F(z_1,\ldots,z_n) = \dfrac{A(z_1,\ldots,z_n)}{B(z_1,\ldots,z_n)}$ and the coefficients of the desired response $D(z_1,\ldots,z_n)$ should be minimised. This is a Wiener filtering problem in N-dimensions. It consists of finding the optimum filter $A(z_1,\ldots,z_n)$ given an input $\dfrac{1}{B(z_1,\ldots,z_n)}$ and a desired output $D(z_1,\ldots,z_n)$.

$A(z_1,\ldots,z_n)$ can be calculated less accurately by convolving the array D with the array B, since

$$D(z_1,\ldots,z_n) = \frac{A(z_1,\ldots,z_n)}{B(z_1,\ldots,z_n)}$$

directly yields the coefficients of the array A.

5.2 THE BORDNER TECHNIQUE [32]

This is a modified form of the Shanks' Method [20]. Stability of the designed filter is now guaranteed by means of augmenting the desired impulse response with an infinite sequence.

5.2.1 The (2-D) Case

The finite (2-D) sequence of equation (5.1) representing the desired impulse response can be augmented with an infinite sequence. It is intuitively apparent that the best such sequence is that which would represent the most natural extension of the original impulse response sequence $d(\ell_1,\ell_2)$ for $\ell_1 = 1,2,\ldots,L_1$ and $\ell_2 = 1,2,\ldots,L_2$, and which is square summable over the region $(0, \infty)$. In other words, given the impulse response of the filter $d(\ell_1,\ell_2)$, a square summable tail g_{ℓ_1,ℓ_2} should be added to it, yielding a new (2-D) sequence $\hat{d}(\ell_1,\ell_2)$, where:

$$\hat{d}(\ell_1,\ell_2) = \begin{cases} d(\ell_1,\ell_2) & \text{for} \quad 1 \leqslant \ell_1 \leqslant L_1 \text{ and } 1 \leqslant \ell_2 \leqslant L_2 \\ g(\ell_1,\ell_2) & \text{for} \quad L_1+1 \leqslant \ell_1 \leqslant \infty \text{ and } L_2+1 \leqslant \ell_2 \leqslant \infty \end{cases} \tag{5.32}$$

The square summability constraint on the augmented sequence implies stability since:

$$\sum_{\ell_1}\sum_{\ell_2} \left|\hat{d}(\ell_1,\ell_2)\right|^2 < \infty \tag{5.33}$$

The transfer function of the filter is restricted to the bilinear (2-D) form:

$$H(z_1,z_2) = A_o + \sum_{i=1}^{N} \frac{A_i}{1 + B_i z_1 + C_i z_2 + D_i z_1 z_2} \tag{5.34}$$

The parameter set $(A_i,\ B_i,\ C_i,\ D_i)$ must be calculated so as to minimise the error function e in a least square sense:

$$\underset{\substack{\text{all "stable"}\\ g_{\ell_1,\ell_2}}}{\text{Min}} \left[\underset{\text{all } A_i,B_i,C_i,D_i}{\text{Min}} \left\{ e = \sum_{\ell_1=0}^{L_1}\sum_{\ell_2=0}^{L_2} (d_{\ell_1,\ell_2} - h_{\ell_1,\ell_2})^2 + \right.\right.$$

$$\left.\left. + \left(\sum_{\ell_1=0}^{\infty}\sum_{\ell_2=0}^{\infty} (g_{\ell_1,\ell_2} - h_{\ell_1,\ell_2})^2 - \sum_{\ell_1=0}^{L_1}\sum_{\ell_2=0}^{L_2} (g_{\ell_1,\ell_2} - h_{\ell_1,\ell_2})^2 \right) \right\} \right] \tag{5.35}$$

The only restriction in this method is that the infinite array g_{ℓ_1,ℓ_2} must be square summable. This forces the inner minimisation, denoted by braces $\{\ \}$ in equation (5.35) above, to give a square summable result, hence ensuring the stability of the impulse response.

5.2.2 Extension to the (N-D) Case [73]

Given a finite-length impulse response $d_{\ell_1,\ldots,\ell_n}$ which is defined for $(\ell_i,\ i=1,n) \in S_L$, an augmenting (N-D) sequence $g_{\ell_1,\ldots,\ell_n}$ can be added to it to form a new (N-D) sequence $\hat{d}_{\ell_1,\ldots,\ell_n}$ where:

$$\hat{d}_{\ell_1,\ldots,\ell_n} = \begin{cases} d_{\ell_1,\ldots,\ell_n} & \text{for } (\ell_i,\ i=1,n) \in S_L \\ g_{\ell_1,\ldots,\ell_n} & \text{for } L_1+1 \leqslant \ell_1 \leqslant \infty\ ,\ L_2+1 \leqslant \ell_2 \leqslant \infty\ ,\ldots, L_n+1 \leqslant \ell_n \leqslant \infty \end{cases} \tag{5.36}$$

In the above, $g_{\ell_1,\ldots,\ell_n}$ is chosen as the most natural extension of the original impulse response sequence $d_{\ell_1,\ldots,\ell_n}$ such that the new completed sequence is square-summable over the region S_∞ which is defined as follows:

$$S_\infty = \{(\ell_i,\ i=1,n) : 0 \leqslant \ell_1 \leqslant \infty\ ,\ 0 \leqslant \ell_2 \leqslant \infty\ ,\ldots,\ 0 \leqslant \ell_n \leqslant \infty\} \tag{5.37}$$

The mean-square error is then defined as:

$$\begin{aligned} e = \Bigg\{ & \sum_{\ell_1=0}^{L_1} \cdots \sum_{\ell_n=0}^{L_n} (d_{\ell_1,\ldots,\ell_n} - h_{\ell_1,\ldots,\ell_n})^2 + \\ & \sum_{\ell_1=0}^{\infty} \cdots \sum_{\ell_n=0}^{\infty} (g_{\ell_1,\ldots,\ell_n} - h_{\ell_1,\ldots,\ell_n})^2 - \\ & \sum_{\ell_1=0}^{L_1} \cdots \sum_{\ell_n=0}^{L_n} (g_{\ell_1,\ldots,\ell_n} - h_{\ell_1,\ldots,\ell_n})^2 \Bigg\} \end{aligned} \tag{5.38}$$

and the coefficients of the (N-D) recursive digital filter are determined by minimising the error function of equation (5.38) above.

5.3 THE LAL SPATIAL DESIGN TECHNIQUE [33]

This is again a modified version of the Shanks' Method for designing (2-D) recursive digital filters in the space domain. The resulting filter is always stable.

The technique is based on sectioning the desired (2-D) impulse response of equation (5.1) into N groups as follows:

$$\left\{ d_{\ell_1,\ell_2} \right\}_{\ell_1,\ell_2=1}^{L_1,L_2} = \left\{ d_{\ell_1,\ell_2} \right\}_{\ell_1,\ell_2=1}^{k_1 k_2} , \left\{ d_{\ell_1,\ell_2} \right\}_{\substack{\ell_1=k_1+1 \\ \ell_2=k_2+1}}^{2k_1,2k_2} ,$$

$$\ldots \left\{ d_{\ell_1,\ell_2} \right\}_{\substack{\ell_1=L_1-k_1+1 \\ \ell_2=L_2-k_2+1}}^{L_1,L_2} \tag{5.39}$$

Each group can be approximated with a second order (2-D) recursive z-transfer function of the form:

$$H(z_1,z_2) = \frac{a_{11} + a_{12}z_1 + a_{21}z_2 + a_{22}z_1z_2}{b_{11} + b_{12}z_1 + b_{21}z_2 + b_{22}z_1z_2} \tag{5.40}$$

An optimisation procedure is employed to calculate the coefficients of the above filter transfer function with some constraints on the coefficients of the denominator as stated in [21]. The filter realisation is that of the form shown in Fig. (5.1).

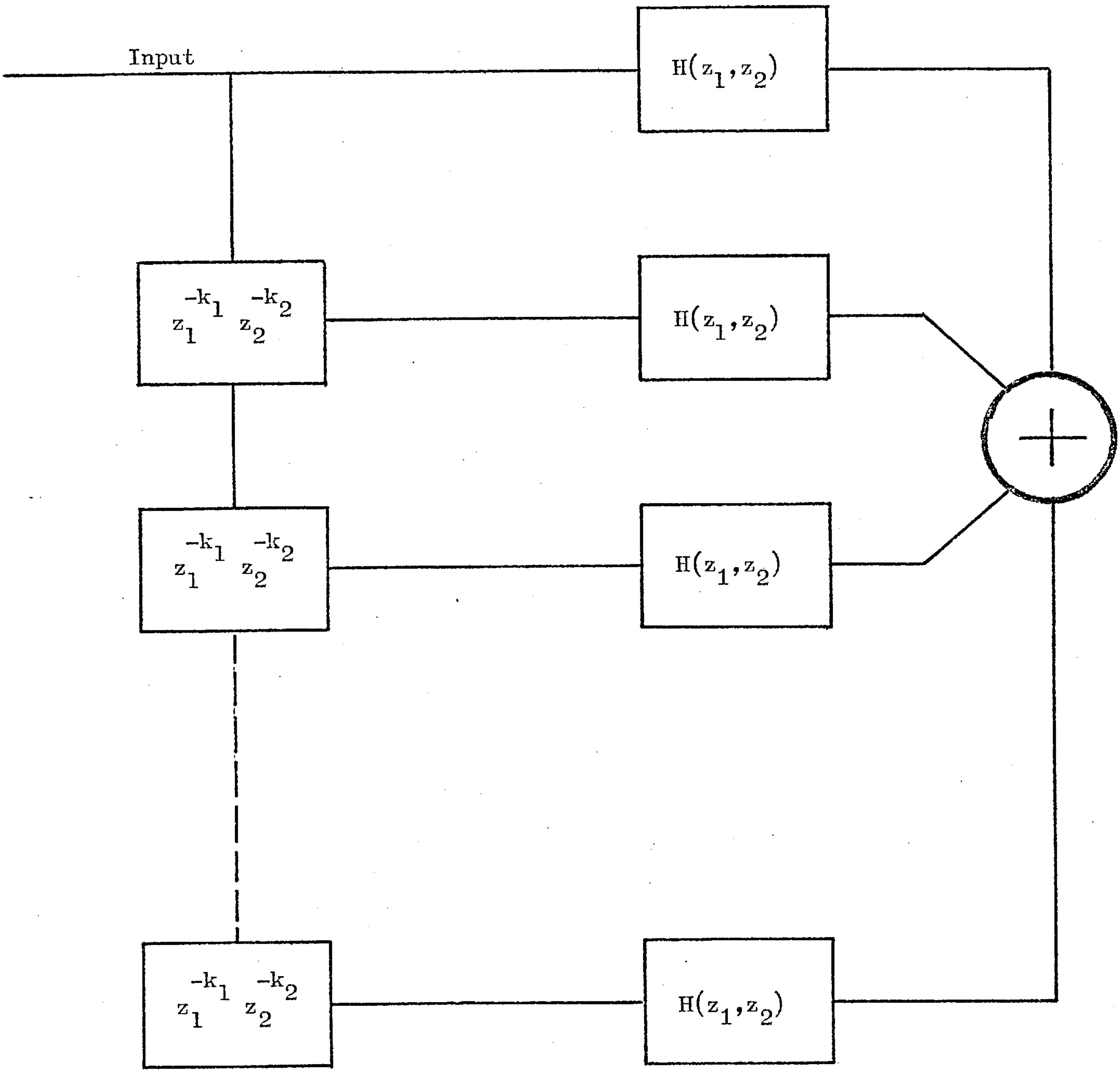

Fig. (5.1): Realisation structure of a digital filter designed by Lal's Method.

CHAPTER SIX

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

This chapter consists of a summary of the contributions of this thesis and suggestions for further research.

In Chapter Two the concept of stability for (2-D) and (N-D) recursive digital filters is defined and conditions for stability of such filters are discussed. Several stability tests are reviewed.

The stability test of Shanks [20] for (2-D) case requires the construction of a theoretically infinite number of mappings; this is not practical, especially for filters of higher orders.

The procedure of Huang [21] , although requiring a finite number of mappings, incorporates the mandatory application of two bilinear transformations in order to enable the use of the method of Ansell [43] . Basically, Ansell's contribution is to couple the use of a Hermite test to check stability, with a series of Sturm tests to check positiveness which is still tedious to apply. The procedure of Anderson and Jury [22] is also finite and does not require bilinear transformation; the Hermite test of Huang [21] is replaced by a Schur-Cohn matrix test [45,46] followed by a series of Sturm tests, or, equivalently, by a series of tests for establishing the root distribution of a polynomial. Finally the last algorithm for checking the stability of (2-D) recursive filters is given by Maria and Fahmy [24] ; it provides a method for checking the condition (2.13) of Huang's method [21] by means of a modified Jury's table. For (N-D) case the stability is checked by means of the repeated applications of the extended Jury's theorem [22] to (N-D) [39].

All the techniques mentioned above suffer from the high cost in terms of computing and complexity in terms of handling and so we need a design technique which does not involve stability checks.

Chapter Three is concerned with the techniques for stabilising an unstable (2-D) recursive digital filter and the presentation of a new stabilisation method for (N-D) zero-phase recursive digital filters. The stabilisation method of Shanks [20] is based on the property of a planar least square inverse of a matrix. This method has been successful for many examples given by the same author, although a counter-example by [53] has placed a question mark on the validity of the procedure. The stabilisation technique of Read and Treitel [54] has the same kind of problems as Shanks'. Failure of the technique for some examples is due to the approximation made to an integral by a finite summation associated with the truncation of the array.

Pistor [26] introduced the stability criteria in terms of a transformation of the complex frequency response. It provides not only a test for stability but also a method of decomposition of an unstable filter into stable filters recursing in different directions; this technique is applicable to zero-phase filters. The method still has unanswered questions concerning the approximation due to transformation which must be made for a practical algorithm. A general treatment of Pistor's technique is presented by Ekstrom and

Woods [27] . In their method the application of two Window sequences is recommended for removal into the stable domain of the possible poles in unstable domain caused by the approximation and truncation used in Pistor's method.

The new decomposition technique presented in Chapter Four for (N-D) zero-phase recursive filters is an extension to (N-D) of the work of Pistor. This method also makes use of the two (N-D) weighting sequences to overcome the possible instability caused by the truncation and approximation used in this technique. These sequences are extensions to (N-D) of the ones proposed by Ekstrom and Wood [27] .

In Chapter Four, different frequency domain design techniques are studied. A new (2-D) design technique for (2-D) filters is presented. A new design technique for (N-D) FIR filters is also proposed and followed by another design technique for (N-D) zero-phase recursive digital filters. The separable product technique of Hall [28.29] gives a simple way of designing (2-D) recursive filters using the well-established (1-D) technique. Therefore these filters have no stability problem, while the choice of the cut-off boundary for these filters is restricted to the rectangular type. This is the only technique for designing linear phase filters. The rotated filters of Shanks [20] share the simplicity of the design technique of Hall, while the stability of the designed filter cannot be guaranteed. The design technique as it stands cannot design a filter with specified cut-off boundary or phase, which is a disadvantage of this technique. This method has been improved by Costa et al. [30]. The new procedure no longer suffers from the stability problem and design of circularly symmetric filters which are of zero-phase with circular cut-off boundary is possible. Bernabo [31] proposed a method for designing (2-D) zero-phase recursive filters with circular cut-off boundary. It is based on the McClellan transformation [15] applied to a squared magnitude of a (1-D) filter followed by the decomposition technique of Pistor [26]. The technique gives the filter designer control over the desired shape of the cut-off boundary while in return it suffers from complexity and high cost of the computation. The technique also relies on the decomposition technique of Pistor which makes the stability of the designed filter to be in doubt.

The new design technique for (2-D) zero-phase filters presented in this thesis is based on the use of a 2-variable reactance function as a transformation applied to a (1-D) recursive filter. The technique is simple to use and the stability of the designed filter is guaranteed, while the shape of the cut-off boundary is restricted.

A new design technique for (N-D) FIR filter is also presented. This is based on the extension to (N-D) of McClellan's method [15]. The chapter ends with a new design technique presented for the (N-D) zero-phase recursive digital filters. The technique makes use of the extension to the (N-D) of McClellan's transformation [71] to a magnitude square function of a (1-D) filter followed by a decomposition technique [50,63]. This technique is also an extension to (N-D) of Bernabo's [31], which has the same property as Bernabo's method.

In Chapter Five, spatial domain design techniques are considered.

The first (2-D) spatial domain design technique is due to Shanks [20]. The technique consists of solving a system of linear equations with the same number of unknowns. The solution to this can easily be obtained by using the well-known Gauss elimination technique. The computational labour involved is rather simple and a guaranteed solution is also provided. This has been extended to (N-D) [73]. In spite of the simplicity of this method for both (2-D) and (N-D) case, it has the disadvantage of not using the minimisation routine over the true mean square error and the stability of the designed filter cannot be guaranteed. A modified version of the above method has been derived by Bordner [32] for (2-D) case and also an extension of this is presented for (N-D) case [73].

In this method the finite impulse response sequence is extended by a square summable infinite tail and an approximation method is used to minimise the mean square error function. A guaranteed solution is provided and the designed filter is stable for both (2-D) and (N-D) case; nevertheless the solution does not necessarily converge to a global minimum. Furthermore, the problem of selecting an (N-D) square summable sequence which is the most natural extension of the desired impulse response is rather difficult. The technique also suffers from the high computation cost involved. The method introduced by Lal [33], which is also derived from Shanks' method, guarantees the stability of the designed filter by using a constraint optimisation routine to approximate the filter's impulse response with a group of second order (2-D) filters. This method lacks the controllability of the sharpness of the transition region.

6.1 SUGGESTIONS FOR FUTURE WORK

As with the development of any new technique, more questions are developed than are answered. The various possibilities which come to mind and deserve consideration are as follows:

(1) The high computational cost and labour involved in checking the stability of (2-D) and (N-D) recursive digital filters with the existing methods makes these methods uneconomical to use for filters of higher orders. The time has therefore come for the development of an efficient algorithm for checking the stability of high order multi-dimensional filters.

(2) Most of the existing stabilisation algorithms lack the reliability needed for ensuring the stability of the designed filter. This is due to the approximation and truncation needed for practical implementation of each technique. Ekstrom [27] in his method suggests the use of a weighting sequence to improve the reliability of the algorithm. The question may arise as to whether a similar procedure can be used in the stabilisation technique of Read and Treitel [54].

(3) The stabilisation techniques mentioned in Chapter Three have the disadvantage of distorting the impulse response of the filter The question may again arise as to whether one can develop an algorithm which stabilises an unstable filter without distorting the impulse response.

(4) In the frequency domain an algorithm is desired which enables us to design a stable filter which has a linear

phase characteristic and arbitrary shape cut-off boundary. Unfortunately none of the present design techniques can offer all these requirements at the same time. Therefore another possible field of work is the development of a technique which offers these properties for the designed filters.

REFERENCES

1. A.S. Samulon, "Separation of Man-Made and Natural Patterns in High Altitude Imagery of Agricultural Areas", IEEE Trans. on Circuits and Systems, April 1975.

2. R. Elefsen et al., "Urban Land-Use Mapping by Machine Processing of ERTS-1 Multispectral Data: A San Francisco Bay Area Example", LARS Inf. Note 101573, Purdue University 1973.

3. G.W. Wecksung and K. Campbell., "Digital Image Processing at EG8G", Computer, Vol. 7, No. 5, pp. 63-71, May 1974.

4. F.C. Billingsby, "Applications of Digital Image Processing", Applied Optics, Vol. 9, February 1970.

5. R.W. Selzer, "Improving Biomedical Image Quality with Computers", NASA Technical Report 32-1336, 1968.

6. C.S. Clark and A.F. Metherell, "Digital Image Processing of Acoustical Holograms", Acoustical Holography, Vol. 5, pp. 471-480, Plenum, 1970.

7. B.R. Hunt and D.H. Janney, "Digital Image Processing at Los Alamos Scientific Laboratory", Computer, Vol. 7, No. 5, pp. 57-62, May 1974.

8. M.P. Ekstrom, "Digital Image Processing at Lawrence Livermore Laboratory, Part I: Diagnostic Radiography Applications", Computer, Vol. 7, No. 5, pp. 72-78, May 1974.

9. S. Treitel, J.L. Shanks and C.W. Frasier, "Some Aspects of Fan Filtering", Geophysics, Vol. 32, pp. 789-800, October 1967.

10. M.B. Dolrin, Introduction to Geophysical Prospecting, McGraw-Hill Book Co., Inc., New York, 1960.

11. J.L. Holloway, Jr., "Smoothing and Filtering of Time Series and Space Fields", Advances in Geophysics, Vol. 4, pp. 381-389, Academic Press Inc., New York, 1958.

12. J.W. Cooley and J.W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", Mathematics of Computation, Vol. 19, pp. 297-301, 1965.

13. D.K. Darby and E.B. Davies, "The Analysis and Design of Two-Dimensional Filters for Two-Dimensional Data", Geophysical Prospecting, Vol. 15, pp. 383-406, 1967.

14. T.S. Huang, "Two-Dimensional Windows", IEEE Trans. Audio and Electroacoustics, Vol. AU-20, pp. 88-89, March 1972.

15. J.H. McClellan, On the Design of One-Dimensional and Two-Dimensional FIR Digital Filters, Ph.D. Dissertation, Dept. of Electrical Engineering, Rice University, 1973.

16. J.H. McClellan and T.W. Parks. "Equiripple Approximation of Fan Filters", Geophysics, Vol. 37, pp. 573-583, August 1972.

17. J.V. Hu and L.R. Rabiner, "Design Techniques for Two-Dimensional Digital Filters", IEEE Trans. Audio and Electroacoustics, Vol. AU-20, pp. 249-257, October 1972.

18. J.V. Hu, Frequency Sampling Design of Two-Dimensional Finite Impulse Response Digital Filters, M.Sc. Thesis, Dept. of Electrical Engineering, MIT, 1972.

19. R.M. Mersereau and D.E. Dugeon", The Representation of Two-Dimensional Sequences as One-Dimensional Sequences", IEEE Trans, Acoustics, Speech and Sig. Proc., Vol. ASSP-22, No. 5, pp. 320-325, October 1974.

20. J.L. Shanks et al., "Stability and Synthesis of Two-Dimensional Recursive Filters", IEEE Trans. on Audio and Electroacoustics, Vol. AU-20, pp. 155-128, June 1972.

21. T.S. Huang, "Stability of Two-Dimensional Recursive Filters", IEEE Trans. on Audio and Electroacoustics, Vol. AU-20, pp. 158-163, June 1972.

22. B.D.O. Anderson and E.I. Jury, "Stability Test for Two-Dimensional Recursive Filters", IEEE Trans. on Audio and Electroacoustics, Vol. AU-21, pp. 366-372, August 1973.

23. E.I. Jury, Inners and Stability of Dynamic Systems, John Wiley and Sons, New York, Chapter 6, 1974.

24. G.A. Maria and M.M. Fahmy, "On the Stability of Two-Dimensional Digital Filters", IEEE Trans. Audio and Electroacoustics, Vol. AU-21, pp. 470-472, October 1973.

25. M.P. Ekstrom and J.W. Woods, "An Algorithm for Testing the Stability of Two-Dimensional Recursive Filters using a Homomorphic Transform", University of California, Lawrence Livermore Laboratory, UCRL-75581, May 1974.

26. P. Pistor, "Stability Criterion for Recursive Filters", IBM Journal of Research and Development, Vol. 18, pp. 59-71, January 1974.

27. M.P. Ekstrom and J.W. Woods, "Two-Dimensional Spectral Factorization with Applications in Recursive Digital Filtering", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-24, No. 2, April 1976.

28. E.L. Hall, Digital Filtering of Images, Ph.D. Thesis, University of Missouri-Columbia, 1971.

29. E.L. Hall et al., "Spatial Digital Filter Design Techniques", Proceedings of the UMR-Marvin J. Kelly Communication Conference, Rolla, Missouri, October 1970.

30. J.M. Costa et al., "Design of Circularly Symmetric Two-Dimensional Recursive Filters", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-22, pp. 432, 443, December 1974.

31. M. Bernabo et al., "A Method for Designing (2-D) Recursive Digital Filters having a Circular Symmetry", Proceedings of Florence Conference on Digital Signal Processing, Florence, Italy, September 1975.

32. G.W. Bordner, "Time Domain Design of Stable Recursive Digital Filters, Ph.D. Dissertation, State University of New York at Buffalo, May 1974.

33. M. Lal, "Space Domain Design of Two-Dimensional Recursive Digital Filters", Proceedings of 1975 IEEE International Symposium on Circuits and Systems, Boston, U.S.A.

34. G.A. Maria and M.M. Fahmy, "An ℓ_p Design Technique for Two-Dimensional Digital Recursive Filters", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-22, pp. 15-21, February 1974.

35. G.A. Maria and M.M. Fahmy, "ℓ_p Approximation of the Group Delay Response of One- and Two-Dimensional Filters", IEEE Trans. Circuits and Systems, Vol. CAS-21, No. 3, pp. 431-436, May 1974.

36. A.G. Deczky, "Synthesis of Recursive Digital Filters using the Minimum p-Error Criterion", IEEE Trans. Audio and Electroacoustics, Vol. AU-20, No. 4, October 1972.

37. N.A. Pendergrass, "Comments on 'An ℓp Design Technique for Two-Dimensional Recursive Filters' ", IEEE Trans. on Acoustics, Speech and Signal Processing, pp. 389-340, August 1975.

38. J.H. Justics and J.L. Shanks, "Stability Criterion for n-Dimensional Digital Filters", IEEE Trans. Automation Control, Vol. AC-18, pp. 284-286, June 1973.

39. B.D.O. Anderson and E.I. Jury, "Stability of Multi-Dimensional Recursive Filters", IEEE Trans. Circuits Syst., Vol. CAS-21, pp. 300-307, March 1974.

40. N.K. Bose and E.I. Jury, "Positivity and Stability Tests for Multi-Dimensional Filters (Discrete-Continuous)", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-22, pp. 174-180.

41. N.K. Bose and P.S. Kamat, "Algorithm for Stability Test of Multi-Dimensional Filters", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-22, No. 5, October 1974.

42. R.E. Bogner and A.G. Constantinides, Introduction to Digital Filtering, John Wiley and Sons, New York.

43. H.G. Ansell, "On Certain Two-Variable Generalisations of Circuit Theory, with Applications to Networks of Transmission Lines and Lumped Reactances", IEEE Trans. on Circuit Theory, pp. 214-223, June 1964.

44. E.A. Guillemin, Synthesis of Passive Networks, John Wiley and Sons, New York, 1957.

45. A. Cohn, "Uber die Anzahl der Wurzeln einer algebraischen Gleichung in einer Kreise", Math. J., pp. 110-148, 1922.

46. E.I. Jury, Theory and Application of the z-Transform Method, John Wiley and Sons, New York, 1964.

47. E.I. Jury, "A Note on the Reciprocal Zeros of a Real Polynomial with respect to the Unit Circle", IEEE Trans. Circuit Theory, pp. 292-294, June 1964.

48. M. Vidyasagar and N.K. Bose, "Input-Output Stability of Linear Systems defined over Measure Space", Proc. of 1975 Mid-West Symposium on Circuit and Systems.

49. C. Farmer and J.B. Bednar, "Stability of Spatial Digital Filters", Mathematical Biosciences 14, pp. 113-119, 1972.

50. M. Ahmadi and R.A. King, "A Stability Criterion for N-Dimensional Zero-Phase Recursive Digital Filters", Proceedings of 19th Mid-West Symposium on Circuits and Systems, University of Wisconsin, Milwaukee, U.S.A., 16-17 August 1976.

51. E.A. Robinson, Statistical Communication and Detection, Mafner, New York, 1967.

52. R.A. Wiggins, "On Factoring the Correlations of Discrete Multi-Variable Stochastic Processes", Mass. Institute of Technology, Cambridge, Sci. Report 9 of Contract AF 19(604)7378, pp. 127-152, 1965.

53. Y. Genin and Y. Kamp, "Comments on 'On the Stability of the Least Mean-Square Inverse Process in Two-Dimensional Digital Filters '", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-25, No. 1, February 1977.

54. R.R. Read and S. Treitel, "The Stabilisation of Two-Dimensional Recursive Filter via the Discrete Hilbert Transform", IEEE Trans. on Geoscience Electronics, Vol. GE-11, No. 3, pp. 153-160, July 1973.

55. A. Papoulis, The Fourier Integral and its Applications, McGraw-Hill Book Co., Inc., 1962.

56. B. Gold and C.M. Rader, Digital Processing of Signals, McGraw-Hill Book Co., Inc., New York, 1969.

57. J.L. Shanks and R.R. Read, "Techniques for Stabilising Two-Dimensional Finite Difference Operator", Proceedings of Colloque International Point Transformations and their Applications, Toulouse, France, September 1973.

58. V. Cizek, "Discrete Hilbert Transform", IEEE Trans. on Audio and Electroacoustics, Vol. AU-18, 1970.

59. G.P. Tolstov, Fourier Series, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1962.

60. W.T. Cochran et al., "What is the Fast Fourier Transform?", Proc. of IEEE 55, 1664 (1967).

61. A.V. Oppenheim et al., "The Non-Linear Filtering of Multiplied and Convolved Signals", Proc. IEEE, Vol. 56, August 1968.

62. J.L. Shanks, "Recursion Filters for Digital Processing", Geophysics 32, 33 (1967).

63. M. Ahmadi and R.A. King, "A Stability Criterion for N-Dimensional Zero-Phase Recursive Digital Filters". To appear in the Proceedings of IEEE, June 1977.

64. M. Ahmadi, A.G. Constantinides and R.A. King, "Design Technique for a Class of Stable Two-Dimensional Recursive Digital Filters", Proc. 1976 IEEE International Conference on Acoustics, Speech and Signal Processing, Philadelphia, Pa., April 1976.

65. Tosino Koga, "Synthesis of Finite Passive n-port with prescribed Two-Variable Reactance Matrices", IEEE Trans. on Circuit Theory, Vol. CT-13, No. 1, March (1966).

66. Lars Hormander, "An Introduction to Complex Analysis in Several Variables", North Holland Publishing Co., Amsterdam (1973).

67. E. Dubois and M.L. Bolstein, "A Circuit Analysis Method for the Design of Recursive Two-Dimensional Digital Filters", Proc. 1975 IEEE International Symposium on Circuits and Systems, Boston, U.S.A. (1975).

68. H. Ozabi and T. Kasami, "Positive Real Functions of Several Variables and their Application to Variable Network", IRE Trans. on Circuit Theory, Vol. CT-7, pp. 251-260, September (1960).

69. R.M. Mersereau et al., "McClellan Transformation for Two-Dimensional Digital Filtering", IEEE Trans. on Circuit and Systems, Vol. CAS-23, No. 7, July (1976).

70. Lawrence R. Rabiner and Berbard Gold, "Theory and Application of Digital Signal Processing", Prentice-Hall, Englewood Cliffs, N.J., U.S.A. (1975).

71. M. Ahmadi and R.A. King, "Transformation Technique for N-Dimensional FIR Filter". To appear in the Proceedings of 20th Midwest Symposium on Circuits and Systems, August 15 and 16, 1977, Hilton Inn, Lubbock, Texas (1977).

72. C.S. Burrus and T.W. Parks, "Time Domain Design of Recursive Digital Filters", IEEE Trans. on Audio Electroacoustics, June (1970).

73. B. Nowrouzian, M. Ahmadi and R.A. King, "On the Space Domain Design Techniques of N-Dimensional Recursive Digital Filters", Proc. of the 1977 IEEE International Conference on Acoustics, Speech and Signal Processing, Connecticut, U.S.A., May (1977).