Transformation design of 2-D recursive digital filters using spectral factorization.

Natarajan Nagamuthu

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/4416
NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L’AVONS RECUE
TRANSFORMATION, DESIGN OF 2-D RECURSIVE DIGITAL FILTERS USING SPECTRAL FACTORIZATION

by

Natarajan Nagamuthu

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

Windsor, Ontario, Canada
1982
To my Family Members
ABSTRACT

Group-delay specifications of Two Dimensional (2-D) recursive digital filter have not been considered in many of the design techniques. These are important requirements for many applications, especially image processing. Stability is the biggest problem in 2-D recursive filter design. Even though several methods have been presented in the literature, they can only be applied to special cases. Many of the design techniques address quite specific problems (e.g., separable, circularly-symmetric, low-pass) with simplifications (e.g., second-order, all pole, magnitude approximation). For a good design these procedures involve a high degree of complexity, and require more time and memory. A method has been described in this thesis which tries to reduce the above-mentioned deficiencies.

The method is basically a transformation design, but coupled with non-linear optimization techniques. It is an efficient method for the design of 2-D recursive digital filters with specified magnitude and group-delay specifications. It effectively uses the well developed 1-D filter theory and McClellan Transformation (MT). Methods are suggested for the design of MT Coefficients. Spectral factorization is used to incorporate stability. A stability error criteria is formed on the basis of complex cepstrum, and it guarantees the stability of the filter. The errors in the frequency response are corrected. Numerical implementation, limitations and examples are presented. The
method is simple, fast, requires less memory and gives a comparable result compared to other techniques. The implementation results of a direct design are used for the comparison. The theoretical foundation for both these methods is given briefly. Finally, possible extensions to the method are suggested.
ACKNOWLEDGEMENTS

I would like to express my heartfelt appreciation to my supervisor, Dr. M.A. Sid-Ahmed for his advice, help and guidance throughout the course of this research. The comments of the other committee members are gratefully acknowledged.

I wish to thank Dr. M. Shridhar, Dr. G.A. Jullien, Dr. M. Ahmadi and Dr. A.T. Chottera for their advice, valuable suggestions and constant encouragement. I also wish to thank Dr. S. Tang of the Department of Computer Science for his help. Additionally, I sincerely appreciate the graduate students and others for their help.

My sincere thanks are to my Family Members, without their help and love, though far away, this work would not have started.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>v</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>vii</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xiii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>xvii</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td>xix</td>
</tr>
</tbody>
</table>

## CHAPTER 1: INTRODUCTION

1.1 1-D and 2-D Digital Signal Processing 1
1.2 Analog Versus Digital 1
1.3 Non-Recursive Versus Recursive 2
1.4 Summary of Previous Work 3
1.4.1 Transformation Design 5
1.4.2 Direct Design 6
1.4.3 Comparison 7
1.4.4 Stability 8
1.4.5 Developed Design 8
1.4.6 Half Plane Design 9
1.4.7 Half Plane Design Versus Developed Design 11
1.5 Problem Statement 11
1.6 Thesis Organization 12

## CHAPTER 2: FUNDAMENTALS OF 2-D FILTER DESIGN

2.1 Introduction 13
2.2 Definitions 13
2.2.1-2.2.4 Definition 1-4 13
2.2.5 Definition 5 16
2.3 Filter Equations 16
2.3.1 ++ QPF 18
2.3.2 ⊕⊕ NHPF 19
2.3.3 ++ plus ⊕ SHPF 19
2.4 Other Equations 21
2.4.1-2.4.3 DFT, IDFT, MR 21
2.4.4-2.4.5 PR, GDR 22
2.5 Requirements for Design 22
2.6 Requirements for Stabilization 24
2.7 Stability Theorems
  2.7.1 Theorem 1
  2.7.2-2.7.4 Theorem 2-3, PLSI
  2.7.5-2.7.7 Conjecture 1, DPLSI,
          DPLSI Method
  2.7.8 Counter Examples
  2.7.9 Theorem 4
2.8 Stability Tests and Methods
  2.8.1 Modified Jury Table
  2.8.2 Double Bi-Linear Transform
  2.8.3 Discrete Hilbert Transform
  2.8.4 Spectral Factorization

CHAPTER 3 DIRECT DESIGN OF 2-D FILTERS
  3.1 Introduction
  3.2 Cascaded Approach
  3.3 Single Filter Approach
  3.4 Zero-Phase Response
  3.5 Direct Design of 2-D Digital Filters
  3.5.1 Design Approach and Theory
  3.5.2 Problem Statement
  3.5.3 Stability Requirements
  3.5.4 Problem Formulation
  3.5.5 Problem Analysis
  3.5.6 Design Algorithm
  3.5.7 Remarks
  3.5.8 Features of Implementation
  3.5.9 Example
  3.5.9.1 Specifications
  3.5.9.2 Chosen Parameters
  3.5.9.3 Results
  3.5.10 Comparison of Results
  3.5.11 Advantages
  3.5.12 Disadvantages
  3.5.13 Conclusion

CHAPTER 4 FUNDAMENTALS OF SPECTRAL FACTORIZATION
  4.1 Introduction
  4.2 1-D Spectral Factorization
  4.3 2-D Spectral Factorization
  4.4 Minimum Phase in 2-D
  4.4.1 Definition 6
  4.5 2-D Cepstrum
  4.6 Existence of 2-D Cepstrum
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.6.1-4.6.3</td>
<td>Theorem 5, Corollary, Theorem 6</td>
<td>73</td>
</tr>
<tr>
<td>4.6.4-4.6.5</td>
<td>Theorem 7-8</td>
<td>74</td>
</tr>
<tr>
<td>4.7</td>
<td>General Factorization</td>
<td>74</td>
</tr>
<tr>
<td>4.7.1-4.7.2</td>
<td>Definition 7-8</td>
<td>74</td>
</tr>
<tr>
<td>4.7.3-4.7.6</td>
<td>Definition 9-10, Lemma, Theorem 9</td>
<td>75</td>
</tr>
<tr>
<td>4.8</td>
<td>Canonical and Eight-Factor Factorizations</td>
<td>75</td>
</tr>
<tr>
<td>4.9</td>
<td>Four Factor Spectral Factorization</td>
<td>76</td>
</tr>
<tr>
<td>4.9.1</td>
<td>Definition 11</td>
<td>76</td>
</tr>
<tr>
<td>4.9.2</td>
<td>Numerical Implementation</td>
<td>77</td>
</tr>
<tr>
<td>4.9.3</td>
<td>Advantages and Disadvantages</td>
<td>79</td>
</tr>
<tr>
<td>4.10</td>
<td>Eight-Factor Spectral Factorization</td>
<td>79</td>
</tr>
<tr>
<td>4.10.1</td>
<td>Definition 12</td>
<td>79</td>
</tr>
<tr>
<td>4.10.2</td>
<td>Advantages and Disadvantages</td>
<td>79</td>
</tr>
<tr>
<td>4.11</td>
<td>Two-Factor Spectral Factorization</td>
<td>83</td>
</tr>
<tr>
<td>4.11.1</td>
<td>Definition 13</td>
<td>83</td>
</tr>
<tr>
<td>4.11.2-4.11.3</td>
<td>Definition 14, Theorem 10</td>
<td>84</td>
</tr>
<tr>
<td>4.11.4</td>
<td>Theorem 11</td>
<td>85</td>
</tr>
<tr>
<td>4.11.5</td>
<td>Advantages and Disadvantages</td>
<td>85</td>
</tr>
<tr>
<td>4.12</td>
<td>Approximation Errors</td>
<td>86</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>87</td>
</tr>
<tr>
<td>5.2</td>
<td>Design Approach and Theory</td>
<td>87</td>
</tr>
<tr>
<td>5.2.1</td>
<td>1-D Filter Theory</td>
<td>87</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Magnitude Approximation</td>
<td>89</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Stability</td>
<td>89</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Group Delay Approximation</td>
<td>90</td>
</tr>
<tr>
<td>5.3</td>
<td>Design of McClellan Transformation Coefficients</td>
<td>91</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Transformation</td>
<td>91</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Single Contour Approximation</td>
<td>92</td>
</tr>
<tr>
<td>5.3.2.1</td>
<td>Problem</td>
<td>92</td>
</tr>
<tr>
<td>5.3.2.2</td>
<td>i) Unconstrained</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>Non-Linear Approach</td>
<td>94</td>
</tr>
</tbody>
</table>

CHAPTER 5

TRANSFORMATION DESIGN OF 2-D FILTERS USING SPECTRAL FACTORIZATION 87

5.1 Introduction 87

5.2 Design Approach and Theory 87

5.2.1 1-D Filter Theory 87

5.2.2 Magnitude Approximation 89

5.2.3 Stability 89

5.2.4 Group Delay Approximation 90

5.3 Design of McClellan Transformation Coefficients 91

5.3.1 Transformation 91

5.3.2 Single Contour Approximation 92

5.3.2.1 Problem 92

5.3.2.2 i) Unconstrained Non-Linear Approach 94
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3.2.3</td>
<td>ii) Unconstrained Linear Approach</td>
<td>95</td>
</tr>
<tr>
<td>5.3.2.4</td>
<td>iii) Constrained Non-Linear Approach</td>
<td>96</td>
</tr>
<tr>
<td>5.3.2.5</td>
<td>iv) Constrained Linear Approach</td>
<td>96</td>
</tr>
<tr>
<td>5.3.2.6</td>
<td>Scaling</td>
<td>96</td>
</tr>
<tr>
<td>5.3.2.7</td>
<td>Test Results and Comparison</td>
<td>97</td>
</tr>
<tr>
<td>5.3.2.8</td>
<td>Advantages and Disadvantages</td>
<td>97</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Multi-Contour Approximation</td>
<td>100</td>
</tr>
<tr>
<td>5.4</td>
<td>Windows in Recursive Filter Design</td>
<td>101</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Window Functions</td>
<td>101</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Proof for the Stabilization Effect of the Exponential Window</td>
<td>102</td>
</tr>
<tr>
<td>5.5</td>
<td>Design of 2-D Filters Using Spectral Factorization</td>
<td>102</td>
</tr>
<tr>
<td>5.6</td>
<td>Numerical Implementation</td>
<td>105</td>
</tr>
<tr>
<td>5.6.1</td>
<td>Steps</td>
<td>105</td>
</tr>
<tr>
<td>5.6.2</td>
<td>Computational Time</td>
<td>106</td>
</tr>
<tr>
<td>5.6.3</td>
<td>Memory Requirement</td>
<td>106</td>
</tr>
<tr>
<td>5.6.4</td>
<td>Complex Cepstrum</td>
<td>107</td>
</tr>
<tr>
<td>5.6.4.1</td>
<td>Computation of Complex Cepstrum</td>
<td>107</td>
</tr>
<tr>
<td>5.6.4.2</td>
<td>Methods</td>
<td>107</td>
</tr>
<tr>
<td>5.6.4.3</td>
<td>Advantages</td>
<td>108</td>
</tr>
<tr>
<td>5.7</td>
<td>Remarks</td>
<td>108</td>
</tr>
<tr>
<td>5.8</td>
<td>Examples</td>
<td>112</td>
</tr>
<tr>
<td>5.8.1</td>
<td>Specifications</td>
<td>112</td>
</tr>
<tr>
<td>5.8.2</td>
<td>Chosen Parameters</td>
<td>112</td>
</tr>
<tr>
<td>5.8.3</td>
<td>Results</td>
<td>112</td>
</tr>
<tr>
<td>5.8.4</td>
<td>Comparison of Results</td>
<td>125</td>
</tr>
<tr>
<td>5.9</td>
<td>Conclusion</td>
<td>155</td>
</tr>
</tbody>
</table>

CHAPTER 6

DISCUSSIONS, EXTENSIONS AND CONCLUSIONS 156

6.1 Introduction 156

6.2 Discussion of the Design 156
6.2.1 Advantages 156
6.2.2 Disadvantages 157
6.2.3 Limitations 158
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3 Extensions</td>
<td>158</td>
</tr>
<tr>
<td>6.4 Conclusions</td>
<td>159</td>
</tr>
<tr>
<td>LIST OF REFERENCES</td>
<td>161</td>
</tr>
<tr>
<td>VITA AUCTORIS</td>
<td>166</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Overview of Filters and Design Methods..</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>2-D Frequency Domain</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>2-D Spatial Domain</td>
<td>14</td>
</tr>
<tr>
<td>2.3</td>
<td>Initial Conditions for ++ Quarter Plane Filter</td>
<td>15</td>
</tr>
<tr>
<td>2.4</td>
<td>Four Ways of Recursion for Quarter Plane Filters</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Support Region for ++ Non-Symmetric Half-Plane Filter</td>
<td>17</td>
</tr>
<tr>
<td>2.6</td>
<td>Support Region for ++ Plus -- Symmetric Half Plane Filter</td>
<td>17</td>
</tr>
<tr>
<td>2.7</td>
<td>Output Bi-Sequence for ++ Quarter Plane Filter</td>
<td>20</td>
</tr>
<tr>
<td>2.8</td>
<td>Output Bi-Sequence for ++ Non-Symmetric Half Plane Filter</td>
<td>20</td>
</tr>
<tr>
<td>2.9</td>
<td>Output Bi-Sequence for ++ Plus -- Symmetric Half Plane Filter</td>
<td>20</td>
</tr>
<tr>
<td>2.10</td>
<td>Mapping of $Z_1$ Unit Disc into $Z_2$ Plane for a Stable Filter</td>
<td>27</td>
</tr>
<tr>
<td>2.11</td>
<td>Mapping of $Z_1$ Unit Disc into $Z_2$ Plane for an Unstable Filter</td>
<td>27</td>
</tr>
<tr>
<td>2.12</td>
<td>Impulse Response of the Original Filter (Unstable)</td>
<td>27</td>
</tr>
<tr>
<td>2.13</td>
<td>Impulse Response of the Stabilized Filter</td>
<td>27</td>
</tr>
<tr>
<td>2.14</td>
<td>Effects of Double Bi-Linear Transform..</td>
<td>32</td>
</tr>
<tr>
<td>3.1</td>
<td>Cascaded Approach</td>
<td>35</td>
</tr>
<tr>
<td>3.2</td>
<td>Single Filter Approach</td>
<td>35</td>
</tr>
<tr>
<td>3.3</td>
<td>First Method of Achieving Zero-Phase Response</td>
<td>37</td>
</tr>
<tr>
<td>3.4</td>
<td>Second Method of Achieving Zero-Phase Response</td>
<td>37</td>
</tr>
<tr>
<td>3.5</td>
<td>Flow Chart for Filter Design</td>
<td>45</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>3.6</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>3.7</td>
<td>46</td>
<td></td>
</tr>
<tr>
<td>3.8</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>3.9</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>3.10</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>3.11</td>
<td>53</td>
<td></td>
</tr>
<tr>
<td>3.12</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td>3.13</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>3.14</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>3.15</td>
<td>57</td>
<td></td>
</tr>
<tr>
<td>3.16</td>
<td>58</td>
<td></td>
</tr>
<tr>
<td>3.17</td>
<td>59</td>
<td></td>
</tr>
<tr>
<td>3.18</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>3.19</td>
<td>61</td>
<td></td>
</tr>
<tr>
<td>3.20</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>3.21</td>
<td>63</td>
<td></td>
</tr>
<tr>
<td>3.22</td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>78</td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>4.6</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>4.7</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td>4.8</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>93</td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>93</td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>93</td>
<td></td>
</tr>
<tr>
<td>5.6</td>
<td>98</td>
<td></td>
</tr>
<tr>
<td>5.7</td>
<td>98</td>
<td></td>
</tr>
<tr>
<td>5.8</td>
<td>98</td>
<td></td>
</tr>
<tr>
<td>5.9</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>5.10</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>5.11</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>5.12</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>5.13</td>
<td>115</td>
<td></td>
</tr>
<tr>
<td>5.14</td>
<td>116</td>
<td></td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>5.15</td>
<td>Magnitude Response of 1-D Digital Low-Pass Filter</td>
<td>117</td>
</tr>
<tr>
<td>5.16</td>
<td>Phase Response of 1-D Digital Low-Pass Filter</td>
<td>118</td>
</tr>
<tr>
<td>5.17</td>
<td>Group-Delay Response of 1-D Digital Low-Pass Filter</td>
<td>119</td>
</tr>
<tr>
<td>5.18</td>
<td>Magnitude Response of 1-D Zero Phase Digital Low-Pass Filter</td>
<td>120</td>
</tr>
<tr>
<td>5.19</td>
<td>Phase Response of 1-D Zero Phase Digital Low-Pass Filter</td>
<td>121</td>
</tr>
<tr>
<td>5.20</td>
<td>Group-Delay Response of 1-D Zero Phase Digital Low-Pass Filter</td>
<td>122</td>
</tr>
<tr>
<td>5.21</td>
<td>Magnitude Response of 2-D Zero Phase Low-Pass Filter</td>
<td>123</td>
</tr>
<tr>
<td>5.22</td>
<td>Group-Delay-1 Response of 2-D Zero Phase Low-Pass Filter</td>
<td>124</td>
</tr>
<tr>
<td>5.23</td>
<td>Magnitude Response of 2-D Low-Pass Filter (After Factorization)</td>
<td>126</td>
</tr>
<tr>
<td>5.24</td>
<td>Group-Delay-1 Response of 2-D Low-Pass Filter (After Factorization)</td>
<td>127</td>
</tr>
<tr>
<td>5.25</td>
<td>Group-Delay-2 Response of 2-D Low-Pass Filter (After Factorization)</td>
<td>128</td>
</tr>
<tr>
<td>5.26</td>
<td>Magnitude Response of 2-D Low-Pass Filter (After Error Correction)</td>
<td>129</td>
</tr>
<tr>
<td>5.27</td>
<td>Group-Delay-1 Response of 2-D Low-Pass Filter (After Error Correction)</td>
<td>130</td>
</tr>
<tr>
<td>5.28</td>
<td>Group-Delay-2 Response of 2-D Low-Pass Filter (After Error Correction)</td>
<td>131</td>
</tr>
<tr>
<td>5.28a</td>
<td>Impulse Response of 2-D Low-Pass Filter (After Error Correction)</td>
<td>132</td>
</tr>
<tr>
<td>5.29</td>
<td>Value of Error Function Versus Number of Iterations</td>
<td>133</td>
</tr>
<tr>
<td>5.30</td>
<td>Number of Function Evaluations Versus Number of Iterations</td>
<td>134</td>
</tr>
<tr>
<td>5.31</td>
<td>Value of Alpha Versus Number of Iterations</td>
<td>135</td>
</tr>
<tr>
<td>5.32</td>
<td>Computational Time Versus Number of Iterations</td>
<td>136</td>
</tr>
</tbody>
</table>
Figure
5.33 Magnitude Response of 2-D Zero Phase Fan Filter #1 ........................... 137
5.34 Magnitude Response of 2-D Fan Filter #1 (After Factorization) .............. 138
5.35 Group-Delay-1 Response of 2-D Fan Filter #1 (After Factorization) .......... 139
5.36 Group-Delay-2 Response of 2-D Fan Filter #1 (After Factorization) .......... 140
5.37 Magnitude Response of 2-D Fan Filter #1 (After Error Correction) .......... 141
5.38 Group-Delay-1 Response of 2-D Fan Filter #1 (After Error Correction) ...... 142
5.39 Group-Delay-2 Response of 2-D Fan Filter #1 (After Error Correction) ...... 143
5.39a Impulse Response of 2-D Fan Filter #1 (After Error Correction) ........... 144
5.40 Magnitude Response of 2-D Zero Phase Fan Filter #2 ........................ 145
5.41 Magnitude Response of 2-D Fan Filter #2 (After Factorization) ............ 146
5.42 Group-Delay-1 Response of 2-D Fan Filter #2 (After Factorization) ......... 147
5.43 Group-Delay-2 Response of 2-D Fan Filter #2 (After Factorization) ......... 148
5.44 Magnitude Response of Fan Filter #2 (After Error Correction) ............. 149
5.45 Group-Delay-1 Response of Fan Filter #2 (After Error Correction) .......... 150
5.46 Group-Delay-2 Response of Fan Filter #2 (After Error Correction) .......... 151
5.46a Impulse Response of Fan Filter #2 (After Error Correction) ............... 152
<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>General Class and Sub-Class</td>
</tr>
<tr>
<td>3.1</td>
<td>Symmetry Operations</td>
</tr>
<tr>
<td>3.2</td>
<td>New Parameters for the Analog Polynomial</td>
</tr>
<tr>
<td>3.3</td>
<td>Filter Coefficients of 2-D Low-Pass Filter After 30 Iterations</td>
</tr>
<tr>
<td>3.4</td>
<td>Filter Coefficients of 2-D Low-Pass Filter After 100 Iterations</td>
</tr>
<tr>
<td>3.5</td>
<td>Comparison of Results for Direct Design</td>
</tr>
<tr>
<td>5.1</td>
<td>McClellan Transformation Coefficients</td>
</tr>
<tr>
<td>5.2</td>
<td>Filter Coefficients of 2-D Low-Pass Filter After 38 Iterations</td>
</tr>
<tr>
<td>5.3</td>
<td>Filter Coefficients of 2-D Fan Filter #1 After 31 Iterations</td>
</tr>
<tr>
<td>5.4</td>
<td>Filter Coefficients of 2-D Fan Filter #2 After 51 Iterations</td>
</tr>
<tr>
<td>5.5</td>
<td>Results of Transformation Design</td>
</tr>
</tbody>
</table>
# LIST OF ABBREVIATIONS

1-D  One Dimensional
2-D  Two Dimensional
#    Number
*    Conjugate Operation
.    Multiplication in the Frequency Domain
\(\Delta W\)  Triangular Window
AF   Analog Filter
APF  All Pass Filter
APFDA All Pass Filter Design Approach
BIBO Bounded Input Bounded Output
BT   Bilinear Transformation
CC   Complex Cepstrum
CL   Complex Logarithm
CT   Computational Time
D    Denominator
DBT  Double Bilinear Transform
DF   Digital Filter
DFT  Discrete Fourier Transform
DHT  Discrete Hilbert Transform
DPLSI Double Planar Least Square Inverse
DSP  Digital Signal Processing
EW   Exponential Window
FDA  Filter Design Approach
FF   Final Filter
FF1  Fan Filter #1
FF2  Fan Filter #2
FFT  Fast Fourier Transform
FMA    Filter with Magnitude Approximation
FR     Frequency Response
FT     Fourier Transform
GD1RMS Relative RMS Error for Group-Delay-1
GD2RMS Relative RMS Error for Group-Delay-2
GDR    Group Delay Response
GDS    Group Delay Specifications
GMI    Generalized McClellan Implementation
HP     Half Plane
HPF    Half Plane Filter
IDFT   Inverse Discrete Fourier Transform
iff    if and only if
IPT    Inverse Fourier Transform
IR     Impulse Response
IZT    Inverse Z Transform
LPF    Low-Pass Filter
LPR    Linear Phase and Removal
LSI    Linear Shift Invariant
M and GDS Magnitude and Group-Delay Specifications
MJT    Modified Jury Table
MMS    Minimum Mean Squared
MR     Magnitude Response
MRMS   Relative RMS Error for Magnitude
MS     Magnitude Specifications
MT     McClellan Transformation
N      Numerator
NF     Non-Recursive Filter
NFE    Number of Function Evaluations
NHP  Non-Symmetric Half Plane
NHPF Non-Symmetric Half Plane Filter
NI  Number of Iterations
NL  Natural Logarithm
NSSK Non-Essential Singularities of the Second Kind
OPT Optimization
PLSI Planar Least Square Inverse
PU  Phase Unwrapping
QP  Quarter Plane
QPF Quarter Plane Filter
RC  Real Cepstrum
RDF Recursive Digital Filter
RF  Recursive Filter
RMS Root Mean Square
S Stable
SF  Spectral Factorization
SHP Strict Hurwitz Polynomial
SHPF Symmetric Half Plane Filter
SSHPF Semi-Recursive Symmetric Half Plane Filter
ST  Stability Test
TF  Transfer Function
U  Unstable
VA Value of Alpha
VSHP Very Strict Hurwitz Polynomial
W  Window
ZP Zero-Phase
ZPF Zero-Phase Filter
ZPR Zero-Phase Response
ZT  Z Transform
CHAPTER 1

INTRODUCTION

1.1 1-D and 2-D Digital Signal Processing

In recent years digital technology has become more advanced and economical. Hence, there has been a considerable growth of interest in the processing of both One-Dimensional (1-D) and Two-Dimensional (2-D) digital signals. Such processing has broad spectrum of applications in various fields. For example speech signals (1-D) are processed to identify speakers. Picture signals and seismic signals (2-D) are processed to get specific features and informations. 2-D digital filters play an important role in 2-D Digital Signal Processing (DSP).

1.2 Analog Versus Digital

Analog filters are difficult to tune. On the other hand, digital filters offer greater flexibility in their realization using digital components. The signal to noise ratio of digital filters is high compared to analog filters. Digital systems are used to simulate analog systems or more importantly, to realize signal transformations which are impossible using analog systems. Digital systems have the disadvantage of limited accuracy and relatively high cost. In the future, it is expected that the accuracy will improve and the cost will decrease. In short, we can say that digital techniques are employed for sophisticated signal processing.
1.3 Non-Recursive Versus Recursive

The design problem of Non-recursive Filter (NF) is less difficult than that of the Recursive Filter (RF) for a number of distinct reasons. First, the NF problem is inherently linear, while the RF problem is non-linear. Second, NF's are inherently stable, while the RF's are in general not stable. To design RF's, stability constraints have to be incorporated in the algorithm, causing further complication in the design process. Finally the NF's can achieve constant group delay (linear phase) easily, while the RF's can only approximate constant group delay. This adds more complexity to the design process.

Now the question arises, "Why RF's?" RF's offer greater speed of filtering, smaller memory requirements, and easier implementations compared to NF's. For RF's, the poles can lie anywhere within the unit circle. For NF's, on the other hand, the poles are fixed at the origin. Because of the increased freedom of poles RF's (possessing rational functions) are capable of approximating a given Frequency Response (FR) with less filter order compared to NF's (possessing polynomial functions). For example, in order to meet the FR of a 90° recursive fan filter [41 coefficients; Numerator: 3 x 3 Quarter Plane (QP); Denominator: 3 x 3 Half Plane (HP)] using NF, it will require order \(\geq 26 \times 26\). To meet any FR, NF's require 5 to 10 times more order than the RF's. The implementation of the above RF is more efficient (based on number of multiplications and
additions) than the direct convolution, Fast Fourier Transform (FFT) and Generalized McClellan Implementations (GMI) [5.1]. When the available memory is significantly smaller than the image to be filtered, the input/output is quite costly for FFT and GMI's. Further, it is experimentally observed that spectra and desired filters derived from real data tend to be better approximated even by all pole filters than by all zero filters (NF's) [3.2].

Thus, even though the RF design is inherently difficult, still the significant improvement in FR is well worth the extra effort. Even though the lagging features in recursive filters have been tackled [3.1, 3.2, 3.3], yet the advancement is not satisfactory. Considering simplicity, design time and memory requirements these methods require improvements. Such an improved method is discussed in this thesis.

1.4 Summary of Previous Work*  

An overview of filters and design methods is given in Figure 1.1. Within the past decade vast amounts of papers have been published in the area of 'Filter Design.' Since the concern is only on 2-D designs, 1-D designs [refer 1.1, 1.2] will not be discussed here. In both the designs the developments started from simple Magnitude Response (MR) consideration. Later on, Group Delay Response (GDR) is also considered and included in the designs. Let

* Some of the terms which will appear in this section may/may not be familiar to the reader. These terms are explained in the coming Chapters.
FIG. 1.1 OVERVIEW OF FILTERS AND DESIGN METHODS
us use simply 'Filter' to denote '2-D Recursive Digital Filter', unless otherwise specified.

1.4.1 Transformation Design

Shanks et al [2.1] suggested 2-D analog transformations to rotate the analog axes by an angle $\beta$. Using one of these transformations, Costa et al [4.1] designed Circularly Symmetric Low-Pass Filters (CSLPF's) by rotating $(\frac{3\pi}{2} < \beta < 2\pi)$ 1-D analog filters and by using Double Bilinear Transform (DBT). A CSLPF consists of two elliptically shaped filters (each is a cascade of three second order Butterworth Filters). Since several rotated filters are used, it is difficult to control the cut-off contour and attenuation at different contours. Ali [4.2] obtained a similar filter using only two one-quadrant filters.

Digital transformations are suggested in [4.3] and [4.4]. The general transformation of [4.3] affects the peak value and the ripples of the MR; but it has tunable characteristics. The transformation of [4.4] results in separable filters unlike filters of [2.1] which are non-separable. When these filters are realized using interpolation functions, there are two inherent problems:

(i) High-frequency attenuation in the vicinity of the Nyquist frequencies, and

(ii) Contraction of the Nyquist region.

Bernabo et al [4.6] designed a Zero-Phase (ZP) CSLPF using McClellan Transformation (MT). The filter is a cascade of four—all pole filters and a NF. Obviously
the filter is inefficient from the point of view of the number of multiplications per sample.

1.4.2 Direct Design

In this let us limit our discussions to the optimization design. Maria et al developed $\ell_p$ design techniques to approximate MR [3.4] and GDR [3.5]. Ramamoorthy et al applied the theory of multivariable positive real functions and multivariable passive networks to the design of stable filters [3.6]. In [3.5] and [3.6] constant GDR in the pass-band are achieved by cascading an equalizer with the original filter. Since two filters are cascaded, let us call this approach as "Cascaded Approach" (see Figure 3.1 in Chapter 3). If the GDR of the original filter is highly non-linear, then the equilization is very difficult, and at times impractical. Further, even though the two sections of the final filter are optimal in their own right, the overall filter is not in general optimal in meeting the FR. In contrast to this is another approach. Let us call this approach as "Single Filter Approach" (see Figure 3.2 in Chapter 3). As per the name, this approach requires only one filter. It eliminates the design work for an equalizer and results in an efficient optimum filter, in comparably less time. Aly et al [3.1] and Chottera et al [3.3] followed this approach.

Linear design [e.g. 3.3] restricts the filter realization to the direct form, which has relatively large roundoff error for higher order filters. Because of the
increased number of parameters, the method takes more time. Moreover, the memory requirements are extremely large and general MR's and GDR's are difficult to achieve. The stability constraints used in [3.3] are sufficient conditions, hence the designed filters belong to a subclass of stable filters. The advantage of linear optimization technique is that it gives global optimum. The disadvantage is that it is slow and is restricted to problems of linear objective function and constraints. Most of the practical problems are non-linear in nature. Hence, non-linear optimization techniques are preferred. Particularly, the gradient methods are faster. The problem is that a global optimum is not guaranteed.

1.4.3 Comparison

Most of the previous 2-D designs address quite specific problems [e.g., separable: 3.7-3.9; circularly symmetric: 3.10, 4.1, 4.2, 4.6; low-pass: 3.8, 3.10, 4.1, 4.2, 4.5, 4.6] with simplifications [e.g., second-order: 3.1, 3.4, 3.5, 3.15, 3.19, 3.20, 4.2; all pole: 3.6, 3.11, 3.12, 4.6; magnitude approximation: 3.4, 3.7-3.19, 4.1-4.4]. Many of the designs result in subclass of filters [3.3, 3.6-3.9, 3.11, 3.12, 3.16, 3.19, 4.2-4.4, 4.6] and do not consider cascaded structure [2.1, 2.2, 3.2, 3.6, 3.8-3.13, 3.17, 3.18, 3.21, 4.1-4.4, 4.6]. Filters designed by methods [3.6, 3.9-3.12, 3.19, 3.20, 4.1-4.4, and 4.6] are sub-optimum. For a good design these procedures involve a high degree of complexity.
1.4.4 Stability

Many of the theories in 1-D cannot be easily extended to 2-D (e.g., stability). That is why 2-D DSP remains a challenging and interesting field of study. The main reason is the lack of fundamental theorem of algebra for polynomials in two or more variables. Even though there are many stability theorems [e.g., 2.1, 2.2], tests [2.3, 2.4, 3.11] and methods [2.5, 2.6], the practical feasibility is still not very good except for certain cases [e.g., 2.4]. Several theorems and tests are summarized in [2.7].

1.4.5 Developed Design

In this thesis an efficient general method is presented for the design of 2-D (QP) Recursive Digital Filters (RDF's) with specified magnitude and group delay specifications. Since, our eyes are sensitive to phase variations, the GDR should be given as much importance as the MR. As mentioned previously, many researchers omitted GDR to simplify the design problem. Constant GDR's are important for image processing applications, and are therefore considered in the design (using the single filter approach). As presented in several publications [3.4, 3.5, 3.13, 3.20] stability can be checked after each iteration. This procedure increases the computational time. If stability is checked at the end of all the iterations [3.21] to reduce time, then stability is not guaranteed. Guaranteed stabilization is the best approach, and is adopted in the design. 1-D filter theory, MT, Spectral Factorization (SF) [based on
Complex Cepstrum (CC) and non-linear optimization techniques are employed in the design. Since the method satisfies most of the requirements for a general design, and general class of filters, it is definitely an improved approach. A preliminary investigation of this method by the author is reported in [4.7] and [4.8].

It should be noted here, that at the time the research was started, the aim was to improve the existing QP designs by eliminating the problems. Hence, High Pass Filters (HPF's) are not designed. However, the theory of HPF's and suggestions for their design are well presented in the thesis. A summary of the work done in this area, advantages and disadvantages are given below.

1.4.6 Half Plane Design

The general class of HPF's have more capability in achieving general FR than the sub-class of QP Filters (QPF's). HPF is of two types:

(i) Non-symmetric HPF (NHPF), and

(ii) Symmetric HPF (SHPF).

Some of the work published in this area and the relative comparisons based on features are already given in Section 1.4.3.

Ramamoorthy et al's method [3.17] is based on the construction of two variable analog network with positive and negative reactive elements in one variable $s_1$, positive reactive elements in the other variable $s_2$ (resistors and gyrators). The denominator of the driving point function of this network is a mix-min phase polynomial in $s_1$ and $s_2$. 
Using DBT, SHPF's are obtained. By suitably constraining the network element values, NHPF's are designed. The disadvantage is that the number of constraints increases with increase in filter order. Moreover, it is assumed that the denominator polynomial is free from Non-essential Singularities of the Second Kind (NSSK) [2.8].

Chang et al's method [3.18] uses Inverse Fourier Transform (IFT) and Planar Least Square Inverse (PLSI) [2.1]. Since numerator and denominator are considered separately, there is no full interaction between these two. ZP NHPF's are designed by methods [3.12] and [3.21]. Only MR is considered in [3.10, 3.17, 3.18 and 3.19], while in [3.2] and [3.20], GDR is also considered.

In [3.20] an equalizer is used for this purpose. Moreover, after each iteration the stability of the filter is checked and if unstable, it is stabilized using Double PLSI (DPLSI) [2.1] method, followed by another stability test. These are tedious procedures and involve extremely more time. Because of the triangular support, general FR is impossible. Since, the optimization is carried out sequentially for each section, the overall filter is not in general optimal. If NF's are cascaded as presented in Example 2, the efficiency goes down.

The same problem is encountered in [3.19], where only SHPF's are designed. They are incapable of approximating general FR because they belong to sub-class of NHPF's. More capability can be obtained by cascading a 1-D filter with the SHPF. If the 1-D filter is recursive,
then the overall filter is NHPF; if it is non-recursive, then the overall filter is semi-recursive SHPF (SSHPF) (this does not mean that an arbitrary NHPF or SHPF can be factored in this way). Moreover, the approximation procedure used in [3.19] is slightly more complicated. It approximately doubles the filter coefficients and exercises very poor control over the pass-band gain. Since the approximating polynomial is squared after choosing the order, in general one cannot do well with half the number of coefficients.

1.4.7 Half Plane Design Versus Developed Design

From the above discussions, it will be clear that the developed QP design is better than many of the HP designs in several aspects. All the HP designs mentioned above are direct designs using non-linear optimization techniques. Some of them [3,2, 3.10, 3.11] use SF based on Real Cepstrum (RC). Direct designs are complicated (hence slow and require more memory), but do yield optimum filters. On the other hand, transformation designs are simple (hence fast and require less memory), but do not yield optimum filters. The developed design is a compromise; it is simple; fast, requires less memory and yields optimum (locally) filters. It is basically a transformation design, but coupled with non-linear optimization techniques.

1.5 Problem Statement

Given the arbitrary magnitude and group delay specifications (in the frequency domain), the problem is to find the filter coefficients so that the filter is stable and at the same time best satisfies the specifications.
For this purpose, nonlinear optimization techniques can be employed.

1.6 Thesis Organization

Chapter 2 explains the fundamentals of 2-D filter design. The basics of QPF and HPF are covered with definitions, notations, equations and theorems.

Chapter 3 describes a direct design method [3.1] to simultaneously approximate both the magnitude and group delay specifications. This method is implemented in software. Advantages and disadvantages are given along with results.

Chapter 4 explains the fundamentals of SF. Definitions, theorems and factorization procedures are presented.

Chapter 5 discusses the development of a transformation design to obtain the given frequency specifications (MR and GDR) using SF. Numerical implementation of the design algorithm is described and its design capabilities are demonstrated with several examples. Finally, advantages and disadvantages are given.

Chapter 6, the last chapter presents the conclusions from the research work, the possible extensions of the work, and the applications of the filters.
CHAPTER 2
FUNDAMENTALS OF 2-D FILTER DESIGN

2.1 Introduction
This chapter explains the fundamentals of 2-D filter design. Definitions, equations, requirements for design and stabilization, stability theorems, tests and methods are presented.

2.2 Definitions

2.2.1 Definition 1
Frequency domain is a domain in which frequency variables define the axes. A 2-D frequency domain is shown in Figure 2.1 with \( w_1 \) and \( w_2 \) frequency variables. The quadrants are marked as 1, ..., 4.

2.2.2 Definition 2
Spatial domain (also called Time domain) is a domain in which spatial variables define the axes. A 2-D Spatial domain is shown in Figure 2.2 with \( n_1 \) and \( n_2 \) spatial variables. The quadrants 1, ..., 4 are denoted as \( R_{++}, ..., R_{+-} \).

2.2.3 Definition 3
Recursive filter is a filter in which output at any time is a function of past outputs as well as present and past inputs. The required initial conditions for a ++ filter are shown in Figure 2.3.

2.2.4 Definition 4
A 2-D Recursive Digital Filter (RDF) is said to be a Quarter Plane Filter (QPF) (also called Causal Filter) if its Impulse Response (IR) is confined to the first quadrant.
FIG. 2.1 2-D FREQUENCY DOMAIN

FIG. 2.2 2-D SPATIAL DOMAIN
FIG. 2.3 INITIAL CONDITIONS FOR QUARTER PLANE FILTER.

FIG. 2.4 FOUR WAYS OF RECURSION FOR QUARTER PLANE FILTERS.
of the spatial domain.

In this thesis the phrase 'direction of recursion' is not used because it is somewhat inappropriate [3.11]. The direction in which data are processed is unimportant. The important thing is that the chosen 'sequence of computation' have the property that each output point makes use of allowable inputs as determined by the type of filter (denoted as ++ etc.). For example, a ++ QPF filter has support only in the region $R_{++}$ (see Figure 2.2). The IR $h(n_1, n_2)$ of the ++ filter is zero for $(n_1, n_2) \in R_{++}$. Similarly other filters are defined. Four ways of recursion (sequence of computations) for QPF's are shown in Figure 2.4.

2.2.5 Definition 5

A 2-D RDF is said to be a Half Plane Filter (HPF) (also called Semi Causal Filter) if its IR is confined to one half of the spatial domain.

For example, a $\bigoplus^+$ Non-symmetric HPF (NHPF) has support only in the region $R_{\bigoplus^+}$ (Figure 2.5). The IR $h((n_1, n_2))$ of the $\bigoplus^+$ filter is zero for $(n_1, n_2) \notin R_{\bigoplus^+}$. A ++ plus -- Symmetric HPF (SHPF) has support only in the region $R_{++}$ plus -- (Figure 2.6). The IR $h((n_1, n_2))$, of the ++ plus -- filter is zero for $(n_1, n_2) \notin R_{++}$ plus --.

2.3 Filter Equations

In this section, Transfer Function (TF), $Z$ transform and difference equations are presented for QPF, NHPF and SHPF.

*1 For definition see (4.12) on page 76.
*2 For definition see (4.30) on page 83.
FIG. 2.5 SUPPORT REGION FOR $\oplus^+$
NON-SYMMETRIC HALF-PLANE FILTER.

FIG. 2.6 SUPPORT REGION FOR $\oplus^+$ PLUS
$-\oplus$ SYMMETRIC HALF-PLANE FILTER.
2.3.1 ++ QPF

\[
H(z_1, z_2) = \frac{\sum_{i=0}^{P_n} \sum_{j=0}^{Q_n} a_{ij} z_{1}^{-i} z_{2}^{-j}}{\sum_{i=0}^{P_d} \sum_{j=0}^{Q_d} b_{ij} z_{1}^{-i} z_{2}^{-j}}
\]  \quad (2.1)

\[
= \frac{A(z_1, z_2)}{B(z_1, z_2)}
\]  \quad (2.2)

\[
= \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} h(n_1, n_2) z_{1}^{-n_1} z_{2}^{-n_2}
\]  \quad (2.3)

\[
b_{00} = 1; \quad z_i = e^{Si T_i}; \quad s_i = jw_i; \quad i = 1, 2
\]  \quad (2.4)

where \(a_{ij}\) and \(b_{ij}\) are the numerator (order: \(P_n x Q_n\)) and denominator (order: \(P_d x Q_d\)) filter coefficients; \(z_1\) and \(s_i\) are complex variables and \(T_i\) is the sampling period along \(n_i\) spatial axis.

\[
y(n_1, n_2) = \sum_{i=0}^{P_n} \sum_{j=0}^{Q_n} a_{ij} x(n_1-i, n_2-j) - \sum_{i=0}^{P_d} \sum_{j=0}^{Q_d} b_{ij} y(n_1-i, n_2-j); \quad (i, j) \neq (0, 0)
\]  \quad (2.5)

where \(y(n_1, n_2)\) and \(x(n_1, n_2)\) are the output and input sequences. The output bi-sequence is shown in Figure 2.7. The input bi-sequence (not shown) consists of the present and the past values which are known. Equation (2.4) is used in other equations in this chapter as well as other chapters.
2.3.2 \( \oplus + \text{NHPF} \)

\[
H(z_1, z_2) = \frac{\sum_{i=0}^{P_n} a_{i0} z_1^{-i} + \sum_{j=0}^{Q_n} a_{i j} z_1^{-i} z_2^{-j}}{\sum_{i=0}^{P_d} b_{i0} z_1^{-i} + \sum_{j=0}^{Q_d} b_{i j} z_1^{-i} z_2^{-j}} \tag{2.6}
\]

\[
= \sum_{(n_1, n_2) \in \mathbb{N}^+} h(n_1, n_2) z_1^{-n_1} z_2^{-n_2} \tag{2.7}
\]

\[
y(n_1, n_2) = \sum_{i=0}^{P_n} a_{i0} x(n_1-i, n_2) + \sum_{j=0}^{Q_n} a_{i j} x(n_1-i, n_2-j) - \sum_{i=0}^{P_d} b_{i0} y(n_1-i, n_2) - \sum_{j=0}^{Q_d} b_{i j} y(n_1-i, n_2-j) \quad (i, j) \neq (0, 0) \tag{2.8}
\]

The output bi-sequence is shown in Fig. 2.8.

2.3.3 \( \oplus + \text{SHPF} \)

\[
H(z_1, z_2) = \frac{\sum_{i=-P_n}^{P_n} a_{i0} z_1^{i} + \sum_{j=0}^{Q_n} a_{i j} z_1^{i} z_2^{-j}}{\sum_{i=-P_d}^{P_d} b_{i0} z_1^{i} + \sum_{j=0}^{Q_d} b_{i j} z_1^{i} z_2^{-j}} \tag{2.9}
\]

\[
= \sum_{(n_1, n_2) \in \mathbb{N}^+} h(n_1, n_2) z_1^{-n_1} z_2^{-n_2} \tag{2.10}
\]

\[
y(n_1, n_2) = \sum_{i=-P_n}^{P_n} a_{i j} x(n_1-i, n_2-j) - \sum_{i=-P_d}^{P_d} b_{i0} y(n_1-i, n_2) - \sum_{j=0}^{Q_d} b_{i j} y(n_1-i, n_2-j) \quad (i, j) \neq (0, 0) \tag{2.11}
\]

The output bi-sequence is shown in Fig. 2.9. Similar
FIG. 2.7 OUTPUT BI-SEQUENCE FOR
++ QUARTER PLANE FILTER.

FIG. 2.8 OUTPUT BI-SEQUENCE FOR
⊕+ NON-SYMMETRIC HALF PLANE FILTER.

FIG. 2.9 OUTPUT BI-SEQUENCE FOR
⊕+ PLUS ⊕ Symmetric HALF PLANE FILTER.
expressions can be written for other filters.

2.4 Other Equations

The Discrete Fourier Transform (DFT) and Inverse DFT (IDFT) equations are as follows.

2.4.1 DFT

\[ X_p(k_1, k_2) = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1, n_2) e^{-j(\frac{2\pi}{N_1} n_1 k_1 \frac{2\pi}{N_2} n_2 k_2)} \]  \hspace{1cm} (2.12)

2.4.2 IDFT

\[ X_p(n_1, n_2) = \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} X_p(k_1, k_2) e^{j(\frac{2\pi}{N_1} n_1 k_1 \frac{2\pi}{N_2} n_2 k_2)} \]  \hspace{1cm} (2.13)

The 2-D DFT coefficients of a finite duration sequence are the values of the 2-D Z transform of the same sequence at \( N_1 \) and \( N_2 \) evenly spaced points around the unit circles \( T^2 = \{(Z_1, Z_2): |Z_1| = 1, |Z_2| = 1\} \) (called unit bi-circle). One of the Fourier Transform (FT) properties is \( H(e^{jw_1}, e^{jw_2}) = H^*(e^{-jw_1}, e^{-jw_2}) \). Hence the responses in the four quadrants (see Figure 2.1) are anti-symmetric (also called centrally symmetric) \( 1 \equiv 3; \ 2 \equiv 4 \).

The Frequency Response (FR) consists of the Magnitude Response (MR) and the Phase Response (PR) or the Group Delay Response (GDR). They are defined as follows:

2.4.3 MR

\[ |H(Z_1, Z_2)| = \sqrt{\Re\{H(Z_1, Z_2)\} + \Im\{H(Z_1, Z_2)\}} \]  \hspace{1cm} (2.14)

* \( \Re \hat{=} \) Real part of; \( \Im \hat{=} \) Imaginary part of.
2.4.4 \( \text{PR} \)

\[
\beta(Z_1, Z_2) = \tan^{-1} \left[ \frac{\text{Im}(H(Z_1, Z_2))}{\text{Re}(H(Z_1, Z_2))} \right]
\]  

(2.15)

2.4.5 \( \text{GDR} \)

\[
\tau_i(Z_1, Z_2) = \frac{-\beta(Z_1, Z_2)}{w_i}
\]

(2.16)

\[
= -\text{Re} \left[ \frac{Z_i}{H(Z_1, Z_2)} \cdot \frac{\partial H(Z_1, Z_2)}{\partial Z_i} \right]_{i=1,2}
\]  

(2.17)

The GD of a filter is a measure of the average spatial or time delay of the filter as a function of frequency.

2.5 \textbf{Requirements for Design}

The preferable requirements for any (non-recursive and recursive) filter design technique are:

1) The design should achieve general class of filters for generic applications rather than sub-class of filters suitable for particular application. (General class and sub-class are shown in Table 2.1);

2) The design should yield best (optimum) filters satisfying specified MR and GDR;

3) The design should yield stable filters (this is the most important requirement);

4) The design should be simple and fast;

5) The design should use minimum memory;

It is important to point out that some of the above requirements are contradictory to each other. For example, optimum filter is possible only at the expense of complexity and increased computational time. Normally, speed is
<table>
<thead>
<tr>
<th>General Class</th>
<th>Sub-Class</th>
</tr>
</thead>
</table>
| **1. Filters:** | Quarter plane filters  
Half plane filters  
Non-symmetric filters  
Pole-zero filters  
Linear phase filters  
Non-separable filters  
N-D filters | Symmetric filters  
All pole filters, all pass filters  
Zero phase filters  
Separable filters  
1-D filters, 2-D filters |
| **2. Order:** | 1 x 1 order, 2 x 2 order |
| Any order | |
| **3. Response:** | Magnitude or group delay response  
Low-pass, high-pass, etc.  
Constant group delay response |
| Frequency response  
Specified magnitude response  
Specified group delay response | |
| **4. Approximation:** | Numerator or denominator approximation, numerator and then denominator approximation or vice versa |
| Numerator and denominator simultaneous approximation | |
| **5. Stability:** | Checking the stability after each iteration or at the end |
| Guaranteed stability | |

**TABLE 2.1 GENERAL CLASS AND SUB-CLASS**
achieved only at the expense of increased memory. Even in hardware, speed is possible only at the expense of increased hardware and power. In these cases, compromise is the best solution in order to satisfy most of the requirements.

As mentioned in Section 1.4.7, a compromised approach is followed in this research work. So far, there is no design technique either in the non-recursive case or in the recursive case which satisfies all the above requirements. Now one can be clear as to why researchers go for simplifications and the inherent problems in the design [Chapter 1].

2.6 **Requirements for Stabilization**

The preferable requirements for any stabilization method are:

1) The method should stabilize an unstable filter without perturbing the MR;

2) The method should be efficient.

So far, there is no stabilization method which satisfies all the above requirements. Most of the methods fail to satisfy requirements 1 or 2. These points will be clear from the following discussions and from the discussions in the next two Sections.

Stabilization in 1-D is easily accomplished by replacing the poles outside the unit circle, by the corresponding poles in the conjugate reciprocal locations with respect to the unit circle. This is equivalent to cascading an unstable filter with an all-pass filter, which will not
affect the shape of the MR. In the case of 2-D a similar algebraic approach fails due to the inability to factor bi-variate polynomials.

2.7 Stability Theorems

Consider a 2-D Linear Shift Invariant (LSI) ++ QPF having a rational TF,

\[ G(Z_1, Z_2) = \frac{P(Z_1, Z_2)}{Q(Z_1, Z_2)} \]  

(2.18)

where \( P(Z_1, Z_2) \) and \( Q(Z_1, Z_2) \) are two variable polynomials in \( Z_1 \) and \( Z_2 \). We will assume that \( Q(0,0) \neq 0 \) so that \( Q(Z_1, Z_2) \neq 0 \) in some neighborhood \( U_\varepsilon \triangleq \{ (Z_1, Z_2) : |Z_1| < \varepsilon, |Z_2| < \varepsilon \} \) of \( (0,0) \); hence in \( U_\varepsilon \) the function is analytic and has power.

Series expansion,

\[ G(Z_1, Z_2) = \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} g(n_1, n_2) Z_1^{n_1} Z_2^{n_2} \]  

(2.19)

where \( g(n_1, n_2) \) is the IR. A well known stability theorem can be stated as follows.

2.7.1 Theorem 1

A filter represented by (2.18 and 2.19) is Bounded Input-Bounded Output (BIBO) stable if and only if (iff) \( \{ g(n_1, n_2) \} \in \ell_1 \), i.e.,

\[ \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} |g(n_1, n_2)| < \infty \]  

(2.20)

We define \( U^2 \triangleq \{ (Z_1, Z_2) : |Z_1| < 1, |Z_2| < 1 \} \) to be the open unit bi-disc, \( \bar{U}^2 \triangleq \{ (Z_1, Z_2) : |Z_1| < 1, |Z_2| < 1 \} \) to be the closed unit bi-disc and \( T^2 \) as already defined. Two polynomials which have no irreducible factors in common
are said to be mutually prime. Let \( P(Z_1, Z_2) \) and \( Q(Z_1, Z_2) \) be mutually prime. A point \((Z_1, Z_2) \in Q(Z_1, Z_2) = P(Z_1, Z_2) = 0\) is called a non-essential singularity of the second kind (NSSK).

2.7.2 Theorem 2

A filter defined by (2.18) is stable if there are no values \( Z_1 \) and \( Z_2 \) such that \( Q(Z_1, Z_2) = 0 \) and \( \forall (Z_1, Z_2) \in \overline{U^2} \).

This is the first stability theorem in 2-D and is called Shanks' Theorem [2.1]. To test the stability using this theorem, the \( Z_1 \) unit disc is mapped into the \( Z_2 \) plane, setting \( Q(Z_1, Z_2) = 0 \). If the mapped region does not intersect the \( Z_2 \) unit disc, then the filter is stable (Figure 2.10); otherwise unstable (Figure 2.11). Alternatively, \( Z_2 \) unit disc can be mapped into the \( Z_1 \) plane for checking the stability. Although this theorem works well in theory, it is very difficult to apply in practice because all the points in one unit disc in one plane should be mapped into the other plane.

2.7.3 Theorem 3

A filter defined by (2.18) is minimum phase if \( Q(Z_1, Z_2) = 0 \) has no zeros \( \in \overline{U^2} \).

From theorems 2 and 3 we see that the filter defined by (2.18) is stable if \( Q(Z_1, Z_2) \) is minimum phase.

2.7.4 Planar Least Square Inverse

Let \( Q \) be a given array. Let \( S_p \) be another array such that \( Q \) convolved \( (\otimes) \) with \( S_p \) is equal to the unit impulse array \( \delta \).
FIG. 2.10 MAPPING OF Z₁ UNIT DISC INTO Z₂ PLANE FOR A STABLE FILTER.

FIG. 2.11 MAPPING OF Z₁ UNIT DISC INTO Z₂ PLANE FOR AN UNSTABLE FILTER.

FIG. 2.12 IMPULSE RESPONSE OF THE ORIGINAL FILTER (UNSTABLE).

FIG. 2.13 IMPULSE RESPONSE OF THE STABILIZED FILTER.
\[ Q \oplus S_p = \delta \]  

Dimension: \((i_1 \times j_1) \oplus (i_2 \times j_2) \oplus [(i_1 + i_2 - 1) \times (j_1 + j_2 - 1)]\)

Let \(C = Q \oplus S_p\). \(S_p\) can be found approximately minimizing \((C - \delta)\), in the Minimum Mean Squared (MMS) sense. Then \(S_p\) is called the Planar Least Square Inverse (PLSI) of \(Q\).

2.7.5 Conjecture 1

Given an arbitrary real finite array \(Q\), any PLSI of \(Q\) is minimum phase.

This is an important conjecture because it implies that the filter defined by (2.18) with \(Q(Z_1, Z_2)\) replaced by \(S_p(Z_1, Z_2)\) must be stable.

2.7.6 Double Planar Least Square Inverse

Let \(S_d\) be the PLSI of \(S_p\). Then \(S_d\) is the Double Planar Least Square Inverse (DPLSI) of \(Q\).

2.7.7 Double Planar Least Square Inverse Method

To stabilize an unstable filter without perturbing the MR, \(S_d(Z_1, Z_2)\) should be used instead of \(S_p(Z_1, Z_2)\). If the order of the intermediate array \(S_p >\) the order of the original array \(Q\), then the approximation is good. If the order of \(S_p\) is increased the approximation is further improved, and at some stage the improvement is negligible and does not justify the increase in order. For higher order filters this method takes lots of computer time.

This method of stabilization was tested [Example 1 of 2.1] for order 2 x 2 with an intermediate array of
order 7 x 7. An answer was obtained as in [2.1] using
Hook and Jeeve's non-gradient optimization technique. The
execution time is six hours*. The IR of the original filter
(unstable) and the stabilized filter are shown in Figure
2.12 and 2.13. For an intermediate array of order 3 x 4,
the approximation is poor compared to that of order 7 x 7.

2.7.8 Counter Examples

The above DPLSI method is based on Conjecture 1.
Since there is no proof for the Conjecture (2-D case), Shanks
et al [2.1] tested it for a large number of problems and
failed to find a counter example. Genin et al [2.9] showed
a counter example. They told that zeros may lie in D={(Z_1,Z_2):
\[|Z_1| > 1, \quad |Z_2| > 1\]} which contradicts the Conjecture. They
did not investigate the conditions under which it is valid,
which is crucial for the design of filters. Jury et al
[2.10] proved that it is valid for a class of polynomials
which are linear in one variable and quadratic in the other
and then extended to a class of polynomials of higher degrees
in the same variables. Shanks' Theorem is essentially correct
except that cases may arise where (2.18) has a NSSK. Such an
example was given by Goodman [2.11]. He also gave the
necessary conditions for the IR to be bounded and sufficient
conditions for it to be square summable and to approach
zero geometrically along any fixed column (or row). The
proof of Theorems 2 and 3 are given in [2.1].

* All the execution times mentioned in this thesis correspond
to the computer IBM 3031.
2.7.9. Theorem 4

A filter defined by (2.18) is stable if:

1) the map \( \mathcal{A}_c \equiv (Z_1:|Z_1|\leq 1) \) in the \( Z_2 \) plane, according to \( Q(Z_1,Z_2) = 0 \) lies outside \( d_2 \equiv (Z_2:|Z_2|\leq 1) \) and

2) no point in \( d_1 \equiv (Z_1:|Z_1|\leq 1) \) maps into the point \( Z_2 = 0 \) by the relation \( Q(Z_1,Z_2) = 0 \).

This is the second stability theorem in 2-D and is called Huang's Theorem [2.2]. To test the stability using this theorem, \( \mathcal{A}_c \) is mapped into the \( Z_2 \) plane according to \( Q(Z_1,Z_2) = 0 \) and to see whether the image lies outside \( d_2 \). Also \( Q(Z_1,0) = 0 \) is solved to see whether any root has magnitude \( \leq 1 \). If the former procedure gives a map outside \( d_2 \) and the latter procedure gives a root with magnitude \( > 1 \), then the filter is stable; otherwise unstable. This is simpler than Shanks' approach because only the rim of the unit disc \( Z_1 \) and a point \( Z_2 = 0 \) has to be mapped into the \( Z_2 \) and \( Z_1 \) planes respectively. However, both the methods are difficult to apply for higher order filters. He gave the proof and showed that his theorem reduces to Ansell's Theorem which is an analog equivalent. His results are equivalent to that of Shanks.

2.8 Stability Tests and Methods

Stability tests are simply used to test whether a filter is stable or not. On the other hand, stability methods are used to design stable filters. Stability methods are mostly stability tests suitably woven into the
direct design techniques.

2.8.1 Modified Jury Table

Maria et al. [2.3] proposed a Modified Jury Table (MJT) to test the first condition of Huang's Theorem. This method has the advantage that all determinants used in the computation are of dimension two, while Huang's method uses dimensions up to the order of the filter.

2.8.2 Double Bi-Linear Transform

Filters can be designed by applying Double Bilinear Transform (DBT) (Figure 2.14) on two variable analog TF's possessing Strict Hurwitz Polynomial (SHP) denominators. Goodman [2.12] showed that not all two variable analog TF's upon DBT yield stable 2-D digital TF's. Reddy et al [2.8] defined a new type of SHP known as Very SHP (VSHP), to overcome this problem. They studied the properties of this polynomial and derived the necessary and sufficient conditions on a 1-D to 2-D transformation so that NSSK are absent. A simplified testing procedure for VSHP's is given in [2.13]. Ahmadi et al [3.13] extended the method [3.6] to N-D filters by including the conditions for VSHP. The problems of NSSK are eliminated in the design techniques [3.1, 3.14, 3.15 and 3.22].

2.8.3 Discrete Hilbert Transform

Read et al [2.14] defined a 2-D Discrete Hilbert Transform (DHT) for the stabilization of filters. Their idea was to obtain the log-magnitude function of the denominator polynomial and use the DHT to calculate the minimum
\[ S_i = \frac{2(1-Z_i^{-1})}{T_i(1+Z_i^{-1})}, \ i = 1, 2 \]

**FIG. 2.14** EFFECTS OF DOUBLE BILINEAR TRANSFORM.
phase function associated with it. A new denominator polynomial is then constructed by complex exponentiation. The disadvantages are the nonexact computations of the DHT and the infinite degree minimum phase function. An example to this method of stabilization was given by Woods [2.15].

2.8.4 Spectral Factorization

A similar attempt at stabilization utilizing the notion of a 2-D cepstrum was performed by Pistor [2.5]. He provided a procedure for the factorization (also called decomposition) of unstable filters having non-zero, non-negative, non-imaginary Frequency Response (FR) into stable quadrant filters. He decomposed the magnitude-squared function into a cascade of four stable QPF's, each of whose IR is defined on a different quadrant. Ahmadi et al [2.16] extended Pistor's idea to N-D. A more general treatment is given in [3.11]. More details are given in Chapter 4.

There are basically two types of stability methods:

1) Mapping Method,
2) Algebraic Method.

Mapping methods are generally inefficient and stability is not guaranteed. Algebraic Methods are efficient and guaranteed stability is possible. But, in practice, problems arise due to finite precision arithmetic. Out of the very few methods available today, Spectral Factorization (SF) is one of the fastest methods.
CHAPTER 3

DIRECT DESIGN OF 2-D FILTERS

3.1 Introduction

This chapter describes a direct design method to simultaneously approximate both the magnitude and group delay specifications. Design algorithm, example, comparison of results, advantages and disadvantages are given. Two approaches are first presented to obtain the given Frequency Response (FR).

3.2 Cascaded Approach

The block diagram of this approach is shown in Figure 3.1. Magnitude Specifications [MS], Filter Design Approach [FDA] and Stability Test [ST] are coupled in an Optimization Technique [OPT] to get the Filter with Magnitude Approximation [FMA].

Its Group Delay Response (GDR), All-Pass Filter Design Approach [APFDA], Group Delay Specifications [GDS] and ST are coupled in another OPT to get the All-Pass Filter [APF]. These two filter Transfer Functions (TFS) are multiplied in the frequency domain [ ] to get the TF of the Final Filter [FF], i.e., the FF is a cascade of the FMA and the APF.

3.3 Single Filter Approach

The block diagram of this approach is shown in Figure 3.2. Magnitude and Group Delay Specifications [M and GDS], FDA and ST are coupled in a single OPT to get
FIG. 3.1 CADCADED APPROACH.

FIG. 3.2 SINGLE FILTER APPROACH
the FF with specified FR. The advantages and disadvantages of both the approaches are already mentioned in Sections 1.4.2 and 1.4.3.

3.4 Zero-Phase Response

Zero-Phase Response (ZPR) in 2-D is achieved using a Zero-Phase Filter (ZPF). It has the particular property that its Impulse Response (IR) is centrally symmetric. Since a Quarter Plane Filter (QPF) has IR only in one quadrant, it cannot have a ZPR. However, by proper combination of two or more QPF's or by properly possessing the input data, one can achieve ZPR or other useful symmetries. Two methods are shown in Figure 3.3 and 3.4. The symmetry operations with respect to \( n_1 \), \( n_2 \) and \( n_1 \) and \( n_2 \) spatial axes are given in Table 3.1. The problems with these approaches are:

(i) The computations per output are increased,

(ii) Memory requirements are extremely large.

So, a single filter with constant GDR is more advantageous than a ZPF.

3.5 Direct Design of 2-D Digital Filters

3.5.1 Design Approach and Theory

This approach [3.1] is an improved version of [3.6] by incorporating the stability requirements of [2.12]. Digital TF's are generated by applying Double Bilinear Transform (DBT) on two variable analog TF's possessing Strict Hurwitz Polynomial (SHP) denominators [2.13]. These TF's are again modified so that the Non-essential Singularities of
\[ H_{eq}(z_1, z_2) = H(z_1, z_2) + H(z_1^{-1}, z_2^{-1}) = 2 \Re\{H(z_1, z_2)\} \]  

(3.1)

**FIG. 3.3 FIRST METHOD OF ACHIEVING ZERO-PHASE RESPONSE.**

\[ H_{eq}(z_1, z_2) = H(z_1, z_2) + H(z_1^{-1}, z_2^{-1}) = 2 \Re\{H(z_1, z_2)\} \]  

(3.2)

**FIG. 3.4 SECOND METHOD OF ACHIEVING ZERO-PHASE RESPONSE.**

\[
\begin{align*}
H_{n_1}(z_1, z_2) &= H(z_1, z_2) \cdot H(z_1^{-1}, z_2^{-1}) \\
H'_{n_1}(z_1, z_2) &= H(z_1, z_2) + H(z_1^{-1}, z_2^{-1}) \\
H_{n_2}(z_1, z_2) &= H(z_1, z_2) \cdot H(z_1^{-1}, z_2^{-1}) \\
H'_{n_2}(z_1, z_2) &= H(z_1, z_2) + H(z_1^{-1}, z_2^{-1}) \\
H_{n_1n_2}(z_1, z_2) &= H(z_1, z_2) \cdot H(z_1^{-1}, z_2^{-1}) \cdot H(z_1^{-1}, z_2^{-1}) \cdot H(z_1, z_2^{-1}) \\
H'_{n_1n_2}(z_1, z_2) &= H(z_1, z_2) + H(z_1^{-1}, z_2^{-1}) + H(z_1^{-1}, z_2^{-1}) + H(z_1, z_2^{-1})
\end{align*}
\]  

(3.3)  

(3.4)  

(3.5)  

(3.6)  

(3.7)  

(3.8)

**TABLE 3.1 SYMMETRY OPERATIONS**
the Second Kind (NSSK) are absent. Using the modified
TF's, digital filters are designed. Fletcher and Powell's
(non-linear, gradient) optimization technique is employed
for this purpose.

3.5.2 Problem Statement

Consider a 2-D ++ QPF. Its TF in the cascade form

\[ H(\z_1, \z_2) = A \prod_{k=1}^{K} \frac{P_{nk} Q_{nk}^{\mathbf{k}}}{P_{dk}^{\mathbf{k}} Q_{dk}} \sum_{i=0}^{\mathbf{k}} \sum_{j=0}^{\mathbf{k}} a_{ij} \z_1^{-i} \z_2^{-j} \]

(3.9)

\[ a_{00}^k = b_{00}^k = 1; \mathbf{k} - \text{right super-script/sub-script} = \text{cascade } \# , \text{ where} \ A \text{ is the gain constant,} \ K \text{ is the total}
\text{number (\#) of sections cascaded,} \ a_{ij}^k \text{ and} \ b_{ij}^k \text{ are the}
umerator and denominator filter coefficients and} \ P_{nk}^{\mathbf{k}} Q_{nk}^{\mathbf{k}} \text{ and}
P_{dk}^{\mathbf{k}} Q_{dk}^{\mathbf{k}} \text{ are the numerator and denominator filter orders}
(1x1 \text{ or} 2x1 \text{ or} 1x2 \text{ or} 2x2). \text{ The problem statement is as in}
Section 1.5.

3.5.3 Stability Requirements

Digital TF (3.9) is obtained starting from an analog
TF. The denominator analog polynomial given by:

\[ S_a(s_1, s_2) = \sum_{i=0}^{\mathbf{k}} \sum_{j=0}^{\mathbf{k}} d_{ij} s_1^i s_2^j \]

(3.10)

where \( s_1 \) and \( s_2 \) are complex variables defined by (2.4).
Similarly, the numerator analog polynomial is defined using
the coefficients \( c_{ij} \)'s. It has been shown in [3.6] that:

\[ S_a(s_1, s_2) \neq 0 \text{ for } \bigcap_{i=1}^{2} \text{ Re}(s_i) > 0 \]

(3.11)
and it is a SHP if the $d_{ij}$'s can be expressed in terms of another set of real (non-zero) parameters [$q_k$'s] as in Table 3.2. Let $T_o$ and $T_n$ represent the total # of old and new parameters. They are given by:

$$T_o = (P_{dk} + 1)(Q_{dk} + 1)K; \quad T_n = (P_{dk} + Q_{dk}) (P_{dk} + Q_{dk} + 1)K/2 \quad (3.12)$$

If $q_k$'s are used for $d_{ij}$'s, then,

$$S_d(z_1, z_2) \Delta (S_a(s_1, s_2) | s_i = \frac{2(1-z_i^{-1})}{T_i (1+z_i^{-1})} \quad \text{(called DBT)}; \quad \text{let } T_i = 2; \quad i = 1, 2 \quad \neq 0 \text{ for } i = 1, 2 |z_i| > 1 \quad (3.13)$$

Similarly, the numerator digital polynomial is obtained by applying DBT on the numerator analog polynomial.

3.5.4 Problem Formulation

The resulting digital TF's numerator and denominator are multiplied by a polynomial [Shown in (3.14)] to get the modified TF,

$$H(z_1, z_2) = \prod_{k=1}^{K} \left[ (1+z_1^{-1})^P_{nk} (1+z_2^{-1})^Q_{nk} \right] \prod_{i=0}^{P_{nk}} \prod_{j=0}^{Q_{nk}} \left[ \frac{1-z_i^{-1}}{1+z_i^{-1}} \right]$$

where $d_{ij}$'s are expressed in terms of $q_k$'s. This TF is free from NSSK [2.1]. So, the filter given by (3.14) is always stable. Goodman [2.12] has shown that even if (3.11) is satisfied, $S_d(z_1, z_2)$ may still have zeros at $(z_1^{-1}, -1)$ with $|z_1| > 1$ and
<table>
<thead>
<tr>
<th>Case</th>
<th>$P_{dk}$</th>
<th>$Q_{dk}$</th>
<th>$k=1$</th>
<th>$d_{ij}$'s Expressed in Terms of $q_e$'s</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$T_0$</td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>II</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>III</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IV</td>
<td>2</td>
<td>2</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 3.2** NEW PARAMETERS FOR THE ANALOG POLYNOMIAL
and \((-1, z_2^{-1})\) with \(|z_2| \gg 1\). To avoid these zeros, the following conditions must be satisfied:

\[
\sum_{i=0}^{P_{dk}} d_{iP_{nk}} (1-z_1^{-1})^i (1+z_1^{-1})^P_{dk}^{-i} \neq 0 \forall |z_1| \gg 1 \quad (3.15)
\]

\[
\sum_{j=0}^{Q_{dk}} d_{P_{nk}}^j (1-z_2^{-1})^j (1+z_2^{-1})^P_{dk}^{-j} \neq 0 \forall |z_2| \gg 1 \quad (3.16)
\]

The proposed choice for \(d_{ij}^k\)'s always satisfy the above conditions.

By noting that \(s_i = (1-z_i^{-1})/(1+z_i^{-1}) = j \tan(w_i/2)\) and \((1+z_i^{-1}) = 2 \exp(-jw_i/2) \cos(w_i/2)\), (3.14) can be simplified as follows.

\[
H(w_1, w_2) = B \sum_{k=1}^{K} G_h(w_1, w_2)^k \prod_{k=1}^{K} H_k(w_1, w_2) \quad (3.17)
\]

where,

\[
G_h(w_1, w_2) = [\cos(w_1/2)]^{u_1} [\cos(w_2/2)]^{u_2} \left[\exp[-j(u_1 w_1 + u_2 w_2)/2]\right]
\]

where,

\[
u_1 = \sum_{k=1}^{K} (P_{nk} - P_{dk})
\]

\[
u_2 = \sum_{k=1}^{K} (Q_{nk} - Q_{dk})
\]

and,
\[ H_k(w_1, w_2) = \frac{N_k(w_1, w_2)}{D_k(w_1, w_2)} \]

\[ \sum_{i=0}^{P_{nk}} \sum_{j=0}^{Q_{nk}} c_{ij} [j \tan(w_1/2)]^i [j \tan(w_2/2)]^j \]

\[ \sum_{i=0}^{P_{dk}} \sum_{j=0}^{Q_{dk}} d_{ij} [j \tan(w_1/2)]^i [j \tan(w_2/2)]^j \]

The relationship between the old parameters \( a_{ij}^k, b_{ij}^k \) and \( A \) in (3.9) and the new parameters \( c_{ij}^k, d_{ij}^k \) \( \{F_n(d_k^k)\}^2 \) and \( B \) in (3.14) are straightforward.

The error functions \( E_m \) for magnitude (sub-script:m) and group delays (sub-script:i) are defined using the desired (sub-script:d) magnitude \( (H_d) \) and group delay \( (G_d) \) specifications.

\[ E_m(\vec{x}) = \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} F_m(w_1, w_2) (|H(w_1, w_2) - H_d(w_1, w_2)|)^p \]

\[ E_i(\vec{x}) = \sum_{n_1, n_2 \in R_p} F_i(w_1, w_2) (G_i(w_1, w_2) - G_d(w_1, w_2))^p \]

where \( F_m \) and \( F_i \) are the frequency weighting functions and \( G_i(w_1, w_2) \) are the designed group delays. \( p=2 \) for Least Mean Squared (LMS) design. \( R_p = \{ \text{discrete set of frequency prints in the pass-band} \} \). The final error function \( E \) is defined using the relative weights \( W_m \) and \( W_i \).

\*1 \( j \) (Near Tan, within []) = \( \sqrt{-1} \).

\*2 \( F_n \) \( \Delta \) Function of .
\[ E(\bar{x}) + \gamma^1 = \bar{w}_m E_m(\bar{x}) + \sum_{i=1}^{2} \bar{w}_i E_i(\bar{x}) \]  \tag{3.21}

\[ \bar{x} \triangleq (c_{ij}, q^k, B)^T \]
\[ \bar{w}_m = \alpha/R_m; \quad \bar{w}_i = [(1-\alpha)/2]/R_i \] \tag{3.22}

0 < \alpha < 1 where \( \bar{x} \) is the parameter vector, \( \alpha \) is the relative importance parameter and \( R_m \) and \( R_i \) are the reference functions defined as:

\[ R_m = \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} F_m(w_{1}, w_{2}) H_d^{p}(w_{1}, w_{2}) \] \tag{3.23}

\[ R_i = \sum_{n_1, n_2 \in R_p} F_i(w_{1}, w_{2}) G_d^{p}(w_{1}, w_{2}) \] \tag{3.24}

### 3.5.5 Problem Analysis

Let \( N \) be the dimension of \( \bar{x} \). The gradient vector is:

\[ \nabla(\bar{x}) = \frac{\partial E(\bar{x})}{\partial \bar{x}_1}, \ldots, \frac{\partial E(\bar{x})}{\partial \bar{x}_r}, \ldots, \frac{\partial E(\bar{x})}{\partial \bar{x}_N} \]

Using (3.21),

\[ \nabla(\bar{x}) = \bar{w}_m \nabla m(\bar{x}) + \sum_{i=1}^{2} \bar{w}_i \nabla i(\bar{x}) \] \tag{3.25}

The \( r \)th component of \( \bar{V}_m(\bar{x}) \) and \( \bar{V}_i(\bar{x}) \) are defined as given below:

\[ \nabla m_r(\bar{x}) = \frac{\partial E_m(\bar{x})}{\partial \bar{x}_r} = \prod_{i=1}^{N_1} \sum_{j=1}^{N_2} F_m(w_{1}, w_{2}) \]

\[ (|H(w_{1}, w_{2})| - H_d(w_{1}, w_{2}))^{p-1} \frac{\partial H(w_{1}, w_{2})}{\partial \bar{x}_r} \] \tag{3.26}

*1 \( \triangleq \) Minimization.

*2 \( \triangleq \) Matrix Transpose.
in which,

\[ \frac{\partial |H(w_1, w_2)|}{\partial X_r} = (|H(w_1, w_2)|)^{-1} \text{Re}(H^*(w_1, w_2) \frac{\partial H(w_1, w_2)}{\partial X_r}) \]  

(3.27)

\[ v_{i_r}(X) = \frac{\partial E_i(X)}{\partial X_r} = p \sum_{n_1, n_2 \in R_p} W_i(w_{1,2}) \]  

(3.28)

\[ \ln(H_k) = \ln(|H_k|) + j\beta_k; \ln(Z_i) = j\omega_i \]

Therefore,

\[ G_{ik} = -\frac{\partial \beta_k}{\partial \omega_i} = -\text{Re}(\frac{\partial \ln(H_k)}{\partial \ln(Z_i)}) \]

Using the above expressions, \( G_i \) is derived as:

\[ G_i = \frac{u_i}{2} + \text{cosec}(\omega_i) \sum_{k=1}^{K} \text{Im}(\frac{D_{ik}}{N_k}) - \frac{N_{ik}}{N_k} \]  

(3.29)

where \( N_{ik} \) and \( N_{2k} \) have similar numerator expressions as in (3.18); but \( c_{ij} \) are multiplied by \( i \) and \( j \) for the respective cases. Similarly, \( D_{1k} \) and \( D_{2k} \) are defined.

3.5.6 Design Algorithm

This is illustrated in Figure 3.5-3.7. The steps involved in the design are:

1) Choose \( K, P_{nk}, Q_{nk}, P_{dk} \) and \( Q_{dk} \) depending on the specifications. Choose \( N_{1}, N_{2}, F_m, F_i \), initial value of \( \alpha \), error norm \( p \), initial

\(+ * \Delta \text{ Complex Conjugate.}\)
FIG. 3.5 FLOW CHART FOR FILTER DESIGN.

Start

Read the input parameters

Store the frequency domain specifications

Calculate the necessary values* for the calculation of error function and gradients

* $F_0$ (Parameter Vector)

Call optimization subroutine (DFMFP)

Print the results

STOP

FIG. 3.6 FLOW CHART FOR OPTIMIZATION SUBROUTINE (DFMFP)

START

Pick a starting point

Determine search direction

Print the result after each iteration

Conduct 1-D search

Is convergence obtained?

No

Yes

Return
FIG. 3.7 FLOW CHART FOR SUBROUTINE 'FUNCT'.

Start

Pick the current value.

Calculate the necessary values* for the calculation of error function and gradients.

* = F^_n(Parameter Vector)

Calculate magnitude and group delay responses.

Calculate the final error function.

Calculate the gradient vector.

Return
guess for \( \bar{x} \) and optimization parameters
\((\text{EST, EPS, LIMIT})\).

2) Store the frequency domain specifications in
\( H_d \) and \( G_{id} \).

3) Calculate \( H, G_i, E_m, E_i \) and \( E \).

4) Calculate \( v_m, v_i \) and \( \bar{V} \).

5) Using the above values, minimize \( E \) until
satisfactory results (\( E \leq \text{EST} \) or \# of iterations
\( = \text{LIMIT} \)) are obtained. Then calculate \( A, a^k_{ij} \)
and \( b^k_{ij} \) from the final \( \bar{x} \).

3.5.7 Remarks

1) \( E_m \) and \( E_i \) given by (3.19) and (3.20) have
different orders of magnitude. Hence, combining these
functions as one function (\( E \)) may result in unintentional
emphasis of one function at the expense of the others.
To avoid this in (3.21), normalization is done through
(3.22)-(3.24).

2) To accelerate the convergence to the optimum,
especially when the initial parameters are selected at
random, \( \alpha \) may be varied as the search for the optimum
progress. Initially \( \alpha = 1 \), i.e., full weight to magni-
tude and zero weight to group delay error functions. It
is reduced gradually, till it attains the designer
selected value (usually, \( \alpha = 0.5 \), i.e., equal weight
to magnitude and group delay error functions) and then
it is kept constant till the end of the computation.

3) \( p \rightarrow \infty \) for Chebyshev (Mini-Max) approximation.
\( p \) is a positive even integer.
4) The technique can be extended to higher order sections.

5) 1-D filters can be designed as a special case of this design.

3.5.8 Features of Implementation

1) The program is written in WATFIV using double precision complex arithmetic. Double precision is not uncommon in filter designs. It is necessary because the results are sensitive to the accuracy of the calculations.

2) It has flexibility for arbitrary specifications, controlling weights and frequency weighting functions.

3) All the filter orders (1x1, 2x1, 1x2 and 2x2) up to 2x2 are implemented with a limitation of 1 section. From the examples of [3.1] it appears that they implemented only the filter order 2x2.

4) Efficient programming is done by minimizing memory (≈170K) and computational time (only 9.5 sec. for each function evaluation) as much as possible.

5) All the inputs and outputs are printed.

6) The FR is calculated in the upper half plane.

Exponent overflow and underflow problems (not mentioned in [3.1]) occur due to $\tan$ (3.18) and $\cosec$ (3.29) functions at the origin and at the edge frequencies of the axes. These problems are avoided by selecting the nearby frequencies.

3.5.9 Example

Consider Example 1 of [3.1] given below.
3.5.9.1 Specifications

Circularly Symmetric Low-Pass Filter (CSLPF);

\[ H_d(w_1, w_2) = 1.0, 0.0, 0.8, 0.44, 0.14, 0.03, 0.002, 0.001, 0.001, 0.001 \text{ for } w = \sqrt{w_1^2 + w_2^2/\pi} = 0.0, 0.1, 0.2, \ldots, 1.0 \]

respectively; \( G_{id}(w_1, w_2) = 2 \) in the pass-band (\( w < 0.4 \)).

The stop-band is \( w > 0.6 \).

3.5.9.2 Chosen Parameters

\[ K = 1; P_{nk} = Q_{nk} = P_{dk} = Q_{dk} = 2; N_1 = 21, N_2 = 11; \]

\[ P_m(w_1, w_2) = F_i(w_1, w_2) = 1; \alpha = 1(-0.1)0.5^*; p = 2; \]

Initial guess for \( X \) is unity for all the elements; \( EST = 0.1; EPS = 10^{-5}; LIMIT = 100 \).

3.5.9.3 Results

The desired FR is shown in Figure 3.8-3.10. The designed FR and IR are shown in Figure 3.11-3.14 for 30 iterations. The same are shown in Figures 3.15-3.18 for 100 iterations. The filter coefficients are given in Table 3.3 and 3.4 respectively. The FR after 100 iterations is better than that of 30 iterations. The value of Error Function [VEF], the Number of Function Evaluations [NFE], the value of Alpha [VA] and the Computational Time [CT] versus the Number of Iterations [NI] are shown in Figure 3.19-3.22 (NI=30).

3.5.10 Comparison of Results

The obtained relative Root Mean Square (RMS) errors for magnitude [MRMS], group delay-1 [GD1RMS] and group

\[
\text{for } x(\Delta x) y \triangleq x, x+\Delta x, x+2\Delta x, \ldots, y. \text{ In this case the step size is added for every five iterations.}
\]
FIG. 3.8 DESIRED MAGNITUDE RESPONSE OF 2-D LOW-PASS FILTER.
FIG. 3.9 DESIRED GROUP DELAY-1 RESPONSE OF 2-D LOW-PASS FILTER.
FIG. 3.10 DESIRED GROUP DELAY-2 RESPONSE OF 2-D LOW-PASS FILTER.
FIG. 3.11 MAGNITUDE RESPONSE OF 2-D LOW-PASS FILTER (AFTER 30 ITERATIONS).
FIG. 3.12 GROUP DELAY-1 RESPONSE OF 2-D LOW-PASS FILTER (AFTER 30 ITERATIONS).
FIG. 3.13 GROUP DELAY-2 RESPONSE OF 2-D LOW-PASS FILTER (AFTER 30 ITERATIONS).
FIG. 3.14 IMPULSE RESPONSE OF 2-D LOW-PASS FILTER (AFTER 30 ITERATIONS).
FIG. 3.15 MAGNITUDE RESPONSE OF 2-D LOW-PASS FILTER (AFTER 100 ITERATIONS).
FIG. 3.16  GROUP DELAY-1 RESPONSE OF 2-D LOW-PASS FILTER (AFTER 100 ITERATIONS).
FIG. 3.17 GROUP DELAY-2 RESPONSE OF 2-D LOW-PASS FILTER (AFTER 100 ITERATIONS).
FIG. 3.18 IMPULSE RESPONSE OF 2-D LOW-PASS FILTER (AFTER 100 ITERATIONS).
FIG. 3.19 VALUE OF ERROR FUNCTION VERSUS NUMBER OF ITERATIONS.

$X_{\text{MIN}} = 1.0000 \times 10^1$  \hspace{1cm} $Y_{\text{MIN}} = 4.2946 \times 10^0$

$X_{\text{MAX}} = 3.0000 \times 10^2$  \hspace{1cm} $Y_{\text{MAX}} = 9.9257 \times 10^0$
FIG. 3.20 NUMBER OF FUNCTION EVALUATIONS VERSUS NUMBER OF ITERATIONS.
FIG. 3.21 VALUE OF ALPHA VERSUS NUMBER OF ITERATIONS.
Fig. 3.22 Computational time versus number of iterations.

XMIN = 1.000E+01
XMAX = 3.000E+02
YMIN = 7.9365E-01
YMAX = 3.3968E+02
<table>
<thead>
<tr>
<th>Index i</th>
<th>Numerator Coefficients $a_{ij}$</th>
<th>Denominator Coefficients $b_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0.10000000E 01</td>
<td>0.10000000E 01</td>
</tr>
<tr>
<td>0 1</td>
<td>0.99522206E 00</td>
<td>0.23300390E 00</td>
</tr>
<tr>
<td>0 2</td>
<td>0.47413186E 00</td>
<td>-0.11040919E 00</td>
</tr>
<tr>
<td>1 0</td>
<td>0.94938208E 00</td>
<td>0.22365127E 00</td>
</tr>
<tr>
<td>1 1</td>
<td>0.15003642E 01</td>
<td>0.24653511E-01</td>
</tr>
<tr>
<td>1 2</td>
<td>0.85246736E 00</td>
<td>-0.19750524E 00</td>
</tr>
<tr>
<td>2 0</td>
<td>0.44251927E 00</td>
<td>-0.99015121E-01</td>
</tr>
<tr>
<td>2 1</td>
<td>0.85003148E 00</td>
<td>-0.20882038E 00</td>
</tr>
<tr>
<td>2 2</td>
<td>0.36991412E 00</td>
<td>-0.35450749E 00</td>
</tr>
</tbody>
</table>

$A \quad \text{Gain Constant} = 0.53006484E-01$

**TABLE 3.3 FILTER COEFFICIENTS OF 2-D LOW-PASS FILTER AFTER 3D-ITERATIONS**

<table>
<thead>
<tr>
<th>Index i</th>
<th>Numerator Coefficients $a_{ij}$</th>
<th>Denominator Coefficients $b_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0.10000000E 01</td>
<td>0.10000000E 01</td>
</tr>
<tr>
<td>0 1</td>
<td>0.97630518E 00</td>
<td>0.42312320E 00</td>
</tr>
<tr>
<td>0 2</td>
<td>-0.24447745E-01</td>
<td>-0.30982164E 00</td>
</tr>
<tr>
<td>1 0</td>
<td>0.77221681E 00</td>
<td>0.54747679E 00</td>
</tr>
<tr>
<td>1 1</td>
<td>-0.34089738E 01</td>
<td>-0.74930987E-01</td>
</tr>
<tr>
<td>1 2</td>
<td>-0.53698408E 01</td>
<td>-0.50763945E 00</td>
</tr>
<tr>
<td>2 0</td>
<td>-0.32832052E-01</td>
<td>0.21010567E-01</td>
</tr>
<tr>
<td>2 1</td>
<td>-0.36825861E 01</td>
<td>-0.30838047E 00</td>
</tr>
<tr>
<td>2 2</td>
<td>-0.41914083E 01</td>
<td>-0.28783609E 00</td>
</tr>
</tbody>
</table>

$A \quad \text{Gain Constant} = -0.17810962E-01$

**TABLE 3.4 FILTER COEFFICIENTS OF 2-D LOW-PASS FILTER AFTER 100 ITERATIONS**
delay-2 [GD2RMS] are compared in Table 3.5 with that of [3.1]. The NI, the NFE, the VEF and the CT are also given. In [3.1], the important values, i.e., the VEF and the NFE are not given. Hence, the obtained test results need not have to match with that of [3.1] and exact comparison is impossible.

3.5.11 Advantages

1) Approximation is done simultaneously for M and GDS.

2) Designed filters are guaranteed to be stable.

3) Considered cascaded structure has many advantages [3.4].

4) Obtained filters are optimum.

3.5.12 Disadvantages

1) The design procedure is complex.

2) The required memory is very high.

3) The computational time is more.

4) It is very difficult to achieve higher order in a section.

For example, consider a filter with denominator order 10x10. Using (3.12) the new parameters (210) are approximately two times the old parameters (121). Since, the new parameters are included in the parameter vector, the computational time increases enormously. Also the procedure becomes more complex and the memory requirement increases.
<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>NI</th>
<th>NPE</th>
<th>VEF</th>
<th>CT (min)</th>
<th>MRMS</th>
<th>GD1RMS</th>
<th>GD2RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Results: (CSLPF)</td>
<td>I</td>
<td>0</td>
<td>1</td>
<td>13.6877</td>
<td>0.1587</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>F</td>
<td>30</td>
<td>214</td>
<td>0.4928</td>
<td>33.9683</td>
<td>0.0950</td>
<td>1.3140</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>60</td>
<td>581</td>
<td>0.4022</td>
<td>92.2222</td>
<td>0.1149</td>
<td>1.0387</td>
</tr>
<tr>
<td>3</td>
<td>F</td>
<td>100</td>
<td>1221</td>
<td>0.3951</td>
<td>193.8095</td>
<td>0.1143</td>
<td>1.0856</td>
</tr>
</tbody>
</table>

**Results of [3.1]* (CSLPF)**

| F  | 30  | NK  | NK   | NK     | 0.0720 | 0.5773 | 0.5071 |

I - Initial (same for 1, 2 and 3); F - Final; * RMS error values are calculated from the desired responses.

**TABLE 3.5 COMPARISON OF RESULTS FOR DIRECT DESIGN**
3.5.13 Conclusion

A direct design method has been described. Test results of an example have been presented for the implemented design. Advantages and disadvantages have been discussed.
CHAPTER 4
FUNDAMENTALS OF SPECTRAL FACTORIZATION

4.1 Introduction

This Chapter explains the fundamentals of Spectral Factorization (SF). Definitions, theorems and factorization procedures are presented.

4.2 1-D Spectral Factorization

This was first introduced by Wiener [5.5] in the early 1940's; it involves factoring the spectral density \( F(z) \) into its minimum (min) phase factor \( F_+(z) \) and maximum (max) phase factor \( F_-(z) \). All the singularities and zeros of \( F_+(z) \) and \( F_-(z) \) are within the unit circle and outside the unit circle respectively. The inverse \( Z \) transform of \( F_+(z) \) and \( F_-(z) \) are stable, causal and anticausal respectively. Wiener made a passing reference to an alternate technique, based on the analytic properties \( F_+(z) \) and \( F_-(z) \). Doob [5.6] subsequently elaborated this technique, but it received little attention till the 1970's.

4.3 2-D Spectral Factorization

The extension of the mentioned Wiener-Doob technique to 2-D is called 2-D SF. Consider an equation similar to (2.7) with general \( n_1 \) and \( n_2 \) and \( (z_1, z_2) \in \mathbb{C}^2 \) where \( \mathbb{C}^2 \) is the space of 2-D complex variables. Corresponding to an autocorrelation or positive definite function \( f(n_1, n_2) \), let \( F(z_1, z_2) \) be the spectral function. 2-D SF is defined as the decomposition of \( F(z_1, z_2) \) into factors which are free from singularities and zeros in certain distinguished
4.4 Minimum Phase in 2-D

Unlike 1-D, in 2-D two unit circles are of concern in the region of analyticity of the Z transform. Let us define the following regions:

\[ D_{++} = \{(Z_1, Z_2): |Z_1| > 1, |Z_2| > 1\} \]
\[ D_{--} = \{(Z_1, Z_2): |Z_1| < 1, |Z_2| < 1\} \]
\[ D_{-+} = \{(Z_1, Z_2): |Z_1| < 1, |Z_2| > 1\} \]
\[ D_{+-} = \{(Z_1, Z_2): |Z_1| > 1, |Z_2| < 1\} \]

(4.1)

4.4.1 Definition 6

\( H(Z_1, Z_2) \) is defined to be min-min phase if none of its singularities or zeros lie on \( D_{++} \). Similarly max-min, max-max and min-max phases are defined if none of them lie on \( D_{--}, D_{--} \) and \( D_{+-} \) respectively.

4.5 2-D Cepstrum

The 2-D Cepstrum (also called Homomorphic Transform) is defined using the operator notation as:

\[ C = Z^{-1} \ln Z \]

(4.2)

where \( Z^{-1}, \ln \) and \( Z \) are the Inverse Z Transform (IZT), Natural Logarithm (NL) and Z Transform (ZT) operators respectively (Figure 4.1). IZT and ZT can be replaced by Inverse Fourier Transform (IFT) and Fourier Transform (FT). The Inverse Cepstrum is similarly defined as (Figure 4.2):

\[ C^{-1} = Z^{-1} \exp Z \]

(4.3)
FIG. 4.1 CEPSTRUM

FIG. 4.2 INVERSE CEPSTRUM

FIG. 4.3 EXAMPLES OF ADMISSIBLE REGIONS.
4.6 Existence of 2-D Cepstrum

The question, "When is cepstrum in \( l_1 \) [See (2.20)] (summable bi-sequence)?", was answered in 1-D in 1933 Wiener and Levy [5.7, 5.8]. Their result and method of proof are not changed for 2-D. Dudgeon showed in 1974 that 2-D cepstrum exists for (2.18) provided that:

i) \( P(Z_1, Z_2) \) and \( Q(Z_1, Z_2) \) are never zero at any point in \( T^2 \)

ii) any linear phase components are removed from the FT by an appropriate linear shift of the sequence \( g(n_1, n_2) \).

Let \( q(n_1, n_2) \) be the sequence corresponding to the Z transform \( Q(Z_1, Z_2) \). Then, this can be written as:

\[
q(n_1, n_2) \ast Q(Z_1, Z_2)
\]

Let,

\[
\hat{Q}(Z_1, Z_2) = \ln \{Q(Z_1, Z_2)\}
\]

The cepstrum of \( q \) is the cepstrum sequence \( \hat{q} \). Its Z transform is \( \hat{Q}(Z_1, Z_2) \). Therefore,

\[
\hat{q}(n_1, n_2) \ast \hat{Q}(Z_1, Z_2) = \ln \{Q(Z_1, Z_2)\}
\]

Let \( o(n_1, n_2) \) and \( i(n_1, n_2) \) be the output and input sequences. The input-output relation in the frequency domain is:

\[
o(Z_1, Z_2) = I(Z_1, Z_2)G(Z_1, Z_2)
\]

The cepstrum of (4.7) in the frequency domain is:

\[
\hat{o}(Z_1, Z_2) = \hat{i}(Z_1, Z_2) + \hat{G}(Z_1, Z_2) \circ \hat{i}(Z_1, Z_2)
\]
The input-output relation in the spatial domain is:

\[ o(n_1, n_2) = i(n_1, n_2) \odot g(n_1, n_2) \]  (4.9)

The cepstrum of (4.9) [i.e., cepstrum of (4.7) in the spatial domain or the spatial relation of (4.8)] leads to the additive relation in the cepstrum domain,

\[ \hat{o}(n_1, n_2) = \hat{i}(n_1, n_2) + \hat{g}(n_1, n_2) \]  (4.10)

4.6.1 Theorem 5

The sequence \( \hat{q}(n_1, n_2) \) is recursively stable if there exists a power series similar to (2.19) with \( G \) and \( g \) replaced by \( \hat{Q}_1 \) and \( \hat{q} \) that is absolutely convergent and equal to \( \ln [Q_1(z_1, z_2)] \) for \( \tilde{U}^2 \).

This theorem is also applicable to other quadrant sequences, if they are transformed to a first quadrant sequence by reflection (about \( n_1=0 \) or \( n_2=0 \) axis) and/or rotation (180° about the origin).

4.6.2 Corollary

The \( \ell \)th-quadrant function \( \hat{g}_\ell \), in which \( \ell = 2, 3, 4 \), is recursively stable if \( \ln \{Q_2(z_1^{-1}, z_2^1)\}, \ln \{Q_3(z_1, z_2^{-1})\}, \) or \( \ln \{Q_4(z_1^1, z_2^{-1})\} \), respectively, is equal to a power series of the form discussed in Theorem 5 that is absolutely convergent for \( \tilde{U}^2 \).

4.6.3 Theorem 6

Assume that a power series similar to (2.19) with \( G \) and \( g \) replaced by \( Q_1 \) and \( \hat{q} \) is absolutely convergent and nonzero in \( \tilde{U}^2 \). There then exists a pair of \( N_1 \) and \( N_2 \) such

* \( \odot \) Convolution Operation
that,
\[
|\hat{\varphi}(z_1, z_2)| = \sum_{n_1=0}^{n_1^*} \sum_{n_2=0}^{n_2^*} \frac{1}{q(n_1, n_2)} z_1^{n_1} z_2^{n_2} > 0
\]
(4.11)
for all \((n_1^*, n_2^*) \in \{(n_1, n_2): n_1 > N_1, n_2 > N_2\}\) and for \(\bar{U}\).

This theorem is also applicable to other quadrant sequences. The proof of the above theorems are given in [2.5].

4.6.4 Theorem 7 [3.11]

If a continuous function \(F(Z_1, Z_2) > 0\) has an absolutely convergent Fourier series \(f(n_1, n_2)\), then \(\hat{F}(z_1, z_2)\) has an absolutely convergent Fourier series.

This theorem is called "Wiener-Levy Theorem". It can be restated as:

4.6.5 Theorem 8 [3.11]

Let \(p \in \ell_1\) with continuous \(B(Z_1, Z_2) > 0\), then \(b\) exists and is in \(\ell_1\).

This theorem is more general than Theorem 6 where the array is of finite support.

4.7 General Factorization

4.7.1 Definition 7

A Non-Symmetric Half Plane (NHP) is a region of the form \(\{n_1 \geq 0, n_2 > 0\} \cup \{n_1 < 0, n_2 > 0\} \cup \{n_1 > 0, n_2 > 0\}\) or their rotations.

4.7.2 Definition 8

A sector is defined as the lattice points inside a continuous region.
4.7.3 Definition 9
An admissible region is the set theoretic product of sectors and a NHP (Figure 4.3).

4.7.4 Definition 10
A 2-D SF is determined by a decomposition of the range of C into non-overlapping (except at boundaries) admissible regions along with a corresponding set of projection operators whose sum is the identity. The factors are then the inverse transform of the projections onto these regions.

4.7.5 Lemma
Project \( b \xi \) onto a sector \(<180^\circ\) with projection operator \( P_s \), then \( b_s = C^{-1}(P_s(b)) \) has support on this sector.

4.7.6 Theorem 9
Let \( b \xi \) and let \( P \) be a projection operator which projects onto an admissible region. Then \( b = C^{-1}(P(b)) \) is recursively computable and stable.

The proof of Theorems 7-9 are given in [3.11].

4.8 Canonical and Eight-Factor Factorizations
Some useful sets of admissible regions are the Quarter Plane (QP) sectors \((\theta = 90^\circ)\) and the NHP sectors \((\theta < 180^\circ)\). The former leads to the four-factor factorization of min-min \((++)\) phase and latter leads to the two-factor factorization of mix-min \((\odot +)\) or min-mix \((+\odot)\) phase. These two factorizations are most important and they are called "Canonical Factorizations." Another factorization is the eight-factor factorization. All the factorizations and their

* Defined in page 83 and 84.
advantages and disadvantages are described in the following sections.

4.9 Four Factor Spectral Factorization

4.9.1 Definition 11

\[
R_{++} \triangleq \{n_1 > 0, n_2 > 0\}; \quad R_{+-} \triangleq \{n_1 < 0, n_2 > 0\}; \\
R_{--} \triangleq \{n_1 < 0, n_2 < 0\}; \quad R_{+-} = \{n_1 > 0, n_2 < 0\}.
\] (4.12)

These regions are shown in Figure 2.2. Let \( \varepsilon \{++, --, -+, +-\} \). Let \( \hat{b} \) be the cepstrum of \( f \varepsilon L_1 \). Let \( \hat{b} \) be the projection of \( \hat{b} \) onto \( R \ldots \).

On the overlapping boundaries \( \hat{b} \) is divided arbitrarily in consistent with \( b \ldots \). The four-factor SF is:

\[
\hat{b} = \hat{b}_{++} + \hat{b}_{+-} + \hat{b}_{--} + \hat{b}_{+-}
\] (4.13)

Let \( b \) be the inverse cepstrum of \( \hat{b} \). Then:

\[
b = b_{++} \odot b_{+-} \odot b_{--} \odot b_{+-}
\] (4.14)

The following three equations \([2.5]\) are used for this purpose.

\[
l_{b, 0, 0} = \mathcal{E}\{b_{0, 0}\}; \quad l_{b, p, q} = \sum_{n_1=0}^{P} \sum_{n_2=1}^{Q} (n_2/q) \hat{b}_{n_1, n_2} \quad l_{b, p-n_1, q-n_2}
\] (4.15)

\[
q \neq 0 \quad (4.16)
\]

\[
l_{b, p, q} = \sum_{n_1=1}^{P} \sum_{n_2=0}^{Q} (n_1/p) \hat{b}_{n_1, n_2} \quad l_{b, p-n_1, q-n_2}
\] (4.17)

\[
p \neq 0
\]

Using Theorem 5 and its Corollary or Theorem 9, the filters of (4.14) are found to be stable.
Consider the problem of factorizing an unstable filter into four stable filters.

\[
\frac{1}{C(Z_1, Z_2)} = \prod_{\ell = 1}^{4} \frac{1}{K_{\ell}(Z_1, Z_2)}
\]  
(4.18)

A solution can be obtained by transforming (4.18) to the cepstrum domain:

\[
\hat{c} = \sum_{\ell = 1}^{4} \hat{k}_{\ell}
\]  
(4.19)

4.9.2 Numerical Implementation

The factorization is shown in Figure 4.4. \(c\) and \(\hat{c}_a\) are centrally symmetric. In practice, FT and IFT are replaced by FFT and IFFT. Hence the obtained cepstrum \(\hat{c}_a\) is an aliased version of the actual cepstrum \(\hat{c}\). The degree of aliasing can be controlled by controlling the sampling frequency. Let \(k\) be the approximate of \(k\). The factorization has the advantage that only \(1^k\) and \(4^k\) have to be determined. The values of \(3^k\) and \(2^k\) are obtained by rotating the subscript planes through 180°.

The computation of an \((N_1+1) \times (N_2+1)\) array \(1^k\) from \(1^k\) should be performed in three steps. First \(1^k_{0,0}\) is obtained by (4.15). Second, the row vector \(1^k_{0,n_1} \mid 1 \leq n_1 \leq N_1\) is determined by (4.16). Third, the sub-matrix \(1^k_{n_1,n_2} \mid 1 \leq n_1 \leq N_1, 0 \leq n_2 < N_2\) is obtained by (4.17). The same procedure is applied to obtain \(4^k\) from \(4^k\). Figure 4.5 shows the factorization procedure.

The input-output relation,

\[
O(Z_1, Z_2) = I(Z_1, Z_2) / C(Z_1, Z_2)
\]

\[
= I(Z_1, Z_2) \prod_{\ell = 1}^{4} K_{\ell}(Z_1, Z_2)
\]  
(4.20)
FIG. 4.4 2-D SPECTRAL FACTORIZATION WITH QUARTER PLANE PROJECTION.
can be evaluated only by implementing the causal recursive algorithm. Figure 4.6 demonstrates how this can be done.

4.9.3 Advantages and Disadvantages

The four-factor SF is used for designing Quarter Plane Filters (QPFs) and circularly symmetric filters. A difficulty with this is "non-uniqueness" due to overlapping boundaries. This problem can be solved by defining non-overlapping regions. In two-factor SF, this problem is solved by including the projection on to the boundary as part of the definition.

4.10 Eight-Factor Spectral Factorization

4.10.1 Definition 12

\[ R_{+0} \triangleq \{ n_1 > 0, n_2 = 0 \}; \quad R_{0+} \triangleq \{ n_1 = 0, n_2 > 0 \}; \]

\[ R_{-0} \triangleq \{ n_1 < 0, n_2 = 0 \}; \quad R_{0-} = \{ n_1 = 0, n_2 < 0 \} \]

\[ R' \triangleq R_{+0} \cap \{ n_1 \neq 0 \text{ and } n_2 \neq 0 \} \]

These regions are shown in Figure 4.7. The eight-factor SF is:

\[ \hat{b} = \hat{b}_{++} + \hat{b}_{+-} + \hat{b}_{-+} + \hat{b}_{-0} + \hat{b}_{0+} + \hat{b}_{-0} + \hat{b}_{0-} \]

or

\[ b = b_{++} \bigcirc b_{+-} \bigcirc b_{-+} \bigcirc b_{-0} \bigcirc b_{0+} \bigcirc b_{-0} \bigcirc b_{0-} \]

\[ b_{++} = b_{++} \bigcirc b_{0+} \bigcirc b_{0+} = b_{++} \bigcirc [b_{0+}(n_1, 0) b_{0+}(0, n_2)] \]

where \( +0 \) and \( 0+ \) are factors in 1-D. If \( b \) is separable, i.e.,
RF - Recursive Filtering
by
RC - Reverse Columns
S1stQ - Shift into 1st
Quadrant
RO - Rotate by

FIG. 4.5 FACTORIZATION PROCEDURE.

DAC - Determine
Approximate Cepstrum
FBQ - Factorize by Quadrant
R1Q - Reflect into 1st
Quadrant
UCR - Use Convolutional
Relations
R4Q - Reflect into 4th
Quadrant
RO - Rotate by

FIG. 4.6 IMPLEMENTATION OF FACTORIZED
RECURSIVE FILTER.
FIG. 4.7 SUPPORT REGION FOR EIGHT FACTOR SPECTRAL FACTORIZATION.

FIG. 4.8 SUPPORT REGION FOR TWO FACTOR SPECTRAL FACTORIZATION.
\[ b(n_1, n_2) = \_b(n_1) \_b(n_2) \quad (4.24) \]

then,
\[ b_+^0(n_1, 0) = \_b_+^0(n_1) \]

and,
\[ b_0^+(0, n_2) = \_b_+^0(n_2) \quad (4.25) \]

where \(_b_+^0\) and \(_b_+^0\) are the minimum phase factors of \(_b^0\) and \(_b^0\) respectively.

In (4.23), the factor \(b^{++}_+\) represents the nonseparable component. The mentioned results are equivalent to the fact that:
\[ b(n_1, n_2) = 0 \quad (n_1, n_2) \notin \{(n_1, n_2): n_1=0 \text{ or } n_2=0\} \quad (4.26) \]

for separable \(b^0\). Hence, for separable functions, the four and eight factor SF's are identical. The resulting factors are Cartesian product of the factors arising from the 1-D factorization.

\(b^{++}_+\) is of the form:
\[ b^{++}_+ = 1 \text{ for } n_1=n_2=0; \quad 0 \text{ for } (n_1=0, n_2 \neq 0) \text{ or } (n_1 \neq 0, n_2=0); \]
\[ \text{(4.27)} \]

\(b^{++}_+\) otherwise

The factors \(b^{++}_+\) are analytic \((\min-\min \text{ phase etc.})\) in the regions \((\text{including } T^2)\) with their separable component removed. Huang [2.2] has shown that for a factor \(b^0\) with support on \(R^{++}_+\), the \(\min-\min\) phase property is equivalent to:
\[ B(Z_1, \infty) \neq 0 \text{ for } \{|Z_1| > 1\} \quad (4.28) \]
\[ B(e^{j2\pi f_1}, Z_2) \neq 0 \text{ on } \{|Z_2| > 1\} \text{ for } f_1 \in (-0.5, 0.5) \quad (4.29) \]
Rewriting (4.23) as \( b_{++} = (b_{++}^* b_{0+}) \oplus b_{+0} \), it is seen that (4.28) is the requirement that \( b_{+0} \) be 1-D minimum phase and (4.29) is the condition that \( (b_{++}^* b_{0+}) \) be min-min phase. It has support on the region \( \{n_1 \geq 0, n_2 > 0\} \). Similarly, analogous factorizations into separable and non-separable components exist for the other factors \( b_{-+}, b_{--} \) and \( b_{+-} \).

### 4.10.2 Advantages and Disadvantages

The eight-factor SF is important only from a theoretical point of view. Since the regions overlap only at the origin, the projection operators are unique up to a scalar multiplier. It provides factorizations for the factors \( b_{++} \).

### 4.11 Two-Factor Spectral Factorization

#### 4.11.1 Definition 13

\[
\begin{align*}
R_{++} & \triangleq R_{++} U R_{+-} \oplus R_{-+} U R_{++} ; \\
R_{--} & \triangleq R_{--} U R_{+ -} \oplus R_{+-} U R_{--} \\
R_{+-} & \triangleq R_{+-} U R_{--} ; \\
R_{--} & \triangleq R_{--} U R_{+-} 
\end{align*}
\] (4.30)

In the above definitions, the two signs determine the \( R \) part and the "\( \ominus \)" determines \( R' \) part by a change of the corresponding sign. Similarly \( R_{+-}, R_{-+} \) and \( R_{+-} \) are defined. These regions are shown in Figure 4.8. There are eight NHP's.

The two-factor SF is:

\[
\hat{b} = \hat{b}_{++} + \hat{b}_{--}
\]

or,

\[
b = b_{++} \ominus b_{--}
\] (4.31)
From (4.30), \( b_{++} = b_{++} \otimes b'_{+} \quad (4.32) \)

where \( b_{++} \) is min-min phase and \( b'_{+} \) is max-min phase. Therefore, \( b_{++} \otimes + \) is mix-min phase. Let us define the following regions of analyticity:

\[
\begin{align*}
D_{++} &= \{(Z_1,Z_2): |Z_1|=1, \ |Z_2|\leq1 \} \\
D_{+-} &= \{(Z_1,Z_2): |Z_1|=1, \ |Z_2|\leq1 \} \\
D_{+\otimes} &= \{(Z_1,Z_2): |Z_1|>1, \ |Z_2|=1 \} \\
D_{-\otimes} &= \{(Z_1,Z_2): |Z_1|\leq1, \ |Z_2|=1 \}
\end{align*}
\quad (4.33)
\]

4.11.2 Definition 14

\( H(Z_1,Z_2) \) is defined to be mix-min phase if none of its singularities or zeros lie on \( D_{++} \). Similarly, mix-max, min-mix and max-mix phases are defined if none of them lie on \( D_{+-}, D_{+\otimes} \) and \( D_{-\otimes} \) respectively.

It is to be noted that even though there are eight factorizations, there are only four combinations of mix-min etc. They don't fully describe the two-factor SF. For example, the additional analyticity of \( b_{++} \) is that \( b_{\otimes+}(n_1,0) \) is minimum phase in 1-D. Two theorems are given below \([2.4, 3.12]\).

4.11.3 Theorem 10

A \( + + \) Half Plane Filter (HPF) \( H(Z_1,Z_2) \) is stable iff:

1) \( B(Z_1,Z_2) \neq 0 \) on \( D'_{++} \), and

2) \( B(Z,\infty) \neq 0 \) on \( (Z_1: |Z_1|\geq1) \).
4.11.4 Theorem 11

A \( \bigoplus^+ \) HPF \( H(z_1, z_2) \) is stable iff its cepstrum \( \hat{h}(n_1, n_2) \) has support on \( \mathbb{R}^+ \).

It is relatively straightforward to generalize Theorem 11 to other HPF's. Because of the first condition of Theorem 10, \( \hat{h}(n_1, n_2) \) takes support on the entire \( \mathbb{R}^+ \), plus \( \mathbb{R}^- \).

The second condition ensures that \( \hat{h}(n_1, n_2) = 0 \) for \( n_1 < 0 \).

If we define the symmetric HP's (SHP's), e.g., \( \{ n_2 \geq 0 \} \), instead of NHP's, the defined mixed phase notation itself is sufficient to characterize the factors. Then they will have stable inverses. However, since those factors have support on SHP's not NHP's, they are not recursively computable. Thus the NHP support constraint leads to the \( 1-\)D minimum phase constraint on \( b \bigoplus^+ \) and vice-versa. This can be considered as a theorem.

4.11.5 Advantages and Disadvantages

The two-factor SF is used for designing NHPF's. The regions overlap at the origin. It is advantageous over four-factor SF. For example, in four-factor SF a Zero Phase Filter (ZPF) will require the ++, +-, -, and -+ factors. In two-factor SF any two mutually opposite NHP factors are enough. A magnitude only approximation will require one of (++,--) and one of (-+,+-) factors. This is contrary to the views expressed in [2.1]. It is clear that the QPF's cannot be used to obtain a general Magnitude Response (MR) and only a restrictive class of MR is possible.
4.12 Approximation Errors

The approximation errors stem from the aliasing exhibited in \( \hat{c}_a \) and also from the truncation applied to \( \hat{\alpha}_k, \ell = 1-4 \). Aliasing error can be reduced by increasing the size of FFT. For almost all factorizations of practical interest, a 1024 x 1024 point FFT will provide accuracies well within the numerical error of the FFT itself. Normally a 32 x 32 or 64 x 64 point FFT is found to be sufficient and error values \( \leq 10^{-6} \) are reported in [2.4, 3.11, and 3.12].

A 1-D polynomial of order 2N can be factorized into minimum and maximum phase polynomials of order N. In 2-D the obtained factors [Eqn. (4.16) and (4.17)] are generally infinite order. Hence, to get a practical realization, truncation of the factors becomes mandatory. This means not only that the factorization (4.18) becomes an approximate one but also that the recursive stability of the factors \( \hat{\alpha}_k \) may be affected. Theorem 6 guarantees that there exist truncations for recursively stable factors \( \hat{\alpha}_k \) such that the resulting factors \( \hat{\alpha}_k \) remain recursively stable. \( N_1 = P_d \) and \( N_2 = Q_d \) may give satisfactory results [2.5].

It is in utilizing this case that the SP becomes a useful tool in the Recursive Filter (RF) design. The next Chapter discusses a design method.
CHAPTER 5
TRANSFORMATION DESIGN OF 2-D FILTERS USING
SPECTRAL FACTORIZATION

5.1 Introduction
This chapter discusses the development of a transformation design to obtain the given frequency specifications using Spectral Factorization (SF). 2-D SF is relatively a new area. The computation of 2-D cepstra [5.2] appeared at the end of 1977. Since then, as mentioned in Section 1.4.7, only very few direct design methods based on Real Cepstrum (RC) have been developed. The developed transformation design is based on Complex Cepstrum (CC). It has several advantages over many of the designs available in the literature (see Section 1.4). Numerical implementation, examples, advantages and disadvantages are given.

5.2 Design Approach and Theory
The problem statement is as in Section 1.5.

5.2.1 1-D Filter Theory
The block diagram of the design is shown in Figure 5.1. The method starts from a stable 1-D Analog Filter [1-D AF(S)]. This is converted into a stable 1-D Digital Filter [1-D DF(S)] by applying Bilinear Transformation [BT]. The Conjugate [*] of the Digital Filter's Transfer Function (TF) [1-D DF*(U)] and the original TF are multiplied in the frequency domain [·] to get the 1-D Zero Phase Digital Filter's [1-D ZPDF(U)] TF. This is nothing but the magnitude squared TF of the 1-D DF(S). These
FIG. 5.1 TRANSFORMATION DESIGN OF 2-D FILTERS USING SPECTRAL FACTORIZATION.

FIG. 5.2 CONTOUR LINES FOR AN APPROXIMATELY CIRCULARLY SYMMETRIC FILTER.
operations are necessary because McClellan Transformation [MT] can be applied only on a 1-D ZP DF (U)'s TF. After applying MT, the result is a 2-D Zero Phase Digital Filter [2-D ZPDF(U)]. Obviously the conjugated filter and the Zero Phase Filters (ZPF's) are unstable (U).

5.2.2 **Magnitude Approximation**

The above-mentioned approach is mainly employed because, the direct design in the $Z_i$, $i=1,2$ planes takes a great deal of computational time. In fact, MT was developed to transform 1-D Zero Phase Non-recursive Filters (ZPNF's) into 2-D ZPNF's [4,5]. But it can also be utilized in the design of Recursive Filters (RF's). It is important to point out that it can be applied to all forms (direct, cascade and parallel) of 1-D ZPDF(U) of any order. The MT Coefficients (MTC's) should be suitably selected using an optimization technique [OPT 1] so that the desired Magnitude Specification [MS] is obtained. The design of MTC's is discussed in the next section.

5.2.3 **Stability**

It is necessary to stabilize the resulting magnitude approximated 2-D ZP DF(U). Even though there are several stabilization methods, SF is chosen because of its advantages. Since FFT is used, SF is faster. Moreover, in stabilizing an unstable filter, SF approach is slightly better than the Discrete Hilbert Transform (DHT) method and significantly better than the Double Planar Least Square Inverse (DPLSI) Method [2,6].
The basic theories presented in the previous chapter are followed in this design. Since the exact decomposition of the bi-variate polynomial gives doubly infinite series, in order to get a practical realization a Truncation Window (TW) \([W]\) is employed along with an Exponential Window (EW) \([3.11]\). The latter has the effect of smoothing any oscillations that may occur due to the application of the former. Moreover, it has the ability to drive out the zeros that may occur due to the TW in the zero free regions. Thus, it leads to additional stability. The EW gives sharper cut-off than the Triangular Window (\(\Delta W\)) \([3.11]\). More details will be given in Section 5.4.

For most of the filters it is observed that the factors have their energy concentrated in a region near the origin. Hence, stable filters are obtained. In this method, a stability error criteria is formed on the basis of CC and it guarantees the stability of the filter. Magnitude error and stability error are optimized using an optimization technique \([OPT 2]\).

5.2.4 **Group Delay Approximation**

Only the Denominator \([D]\) of the 2-D ZP DF(U) is spectrally factorized. The Magnitude Response (MR) of the factorized filters slightly differ from that of the unfactorized filter. But the Group Delay Response (GDR) is very much affected and turns out to be highly non-linear. These effects are mainly due to the approximate factorization procedure.
The errors in the Frequency Response (FR) are corrected using a non-linear optimization technique [OPT 3]. Only the Numerator [N] filter coefficients of the selected quadrant are used as parameters of optimization. The MS and the Group Delay (1 and 2) Specification [GDS] are included in the optimization process. Minimization is carried out simultaneously for both using suitable weighting factors and normalization. The procedure is exactly the same as the design in Chapter 3*, except for the change in the parameter vector and the gradients. Finally, a stable 2-D Digital Filter [2-D DF(S)] with specified MR and GDR is obtained.

5.3 Design of McClellan Transformation Coefficients

5.3.1 Transformation

The generalized MT of order \( P_m \times Q_m \) is defined as [4.5]:

\[
\cos(w) = \sum_{i=0}^{P_m} \sum_{j=0}^{Q_m} t_{ij} \cos(iw_1) \cos(jw_2) \quad (5.1)
\]

\[
= P_m(w_1, w_2)
\]

where \( w \) and \( w_1 \) and \( w_2 \) are the 1-D and 2-D frequency variables and \( t_{ij} \)'s are the MTC's. The MTC's should satisfy the relation:

\[
\left| \sum_{i=0}^{P_m} \sum_{j=0}^{Q_m} t_{ij} \cos(iw_1) \cos(jw_2) \right| \leq 1 \quad (5.2)
\]

\[
0 \leq w_i \leq \pi, \ i = 1, 2
\]

* Same notations and definitions are used for the derivations in this chapter.
Consider the MT of order 1 x 1. The MTC's are given below for three different filters. Their contours are shown in Figure 5.2-5.4.

1) Approximately Circularly Symmetric Filter (CSR):

\[ t_{00} = t_{01} = t_{10} = t_{11} = 0.5 \]  \hspace{1cm} (5.3)

2) Fan Filter #1 (FF1) having pass-band along \( w_1 \) axis:

\[ t_{00} = t_{11} = 0; \ t_{01} = t_{10} = 0.5 \] \hspace{1cm} (5.4)

3) Fan Filter #2 (FF2) having pass-band along \( w_2 \) axis:

\[ t_{00} = t_{11} = 0; \ t_{01} = t_{10} = 0.5 \] \hspace{1cm} (5.5)

The choice of the method for the design of MTC's depends on the shape of the desired contours. For filters like circular low-pass and high-pass where a constant value is desired in the pass-band and stop-band, the only important contour to be approximated is the band edge; the other contours are unimportant. In this case, a "Single Contour Approximation" method is enough. For filters with desired MR not piecewise constant, several contours must be matched simultaneously. Hence, a "Multi-Contour Approximation" method is necessary. For the piecewise constant case, it is advantageous to simultaneously optimize the transition band.

5.3.2 Single Contour Approximation

5.3.2.1 Problem

Design the MTC's to obtain a 2-D Low Pass Filter (LPF) with pass-band having ellipse shape. \( w_p \) and \( w_q \) are the
FIG. 5.3 CONTOUR LINES FOR FAN FILTER #1.

FIG. 5.4 CONTOUR LINES FOR FAN FILTER #2.

FIG. 5.5 DESIRED MAGNITUDE RESPONSE OF 2-D LOW-PASS FILTER.
cutoff frequencies along \( w_1 \) and \( w_2 \) axis. Consider only the cross-hatched region in Figure 5.5. \( w_{cp} \) is the cutoff frequency of the 1-D proto-type LPF.

This problem can be solved by using any one of the following four approaches. Each approach is described below.

5.3.2.2 i) Unconstrained Non-Linear Approach

The equation of the desired 2-D contour is:

\[
\left( \frac{w_1}{w_{ap}} \right)^2 + \left( \frac{w_2}{w_{bp}} \right)^2 = 1 \quad (5.6)
\]

For the given problem, the 1-D origin should be mapped into the 2-D origin, i.e.,

\[
w = 0 \rightarrow w_i = 0, \, i = 1, 2 \quad (5.7)
\]

In (5.1) let \( P_m = Q_m = 1 \). Substituting (5.7) in (5.1), we get:

\[
t_{00} + t_{01} + t_{10} + t_{11} = 1 \quad (5.8)
\]

\( t_{11} \) is expressed as a function of other three coefficients. Another condition is that the 1-D cutoff frequency should be mapped into the contour given by (5.6), i.e.,

\[
w = w_{cp} \rightarrow \text{contour of (5.6)} \quad (5.9)
\]

Substituting (5.9) and (5.8) into (5.1) and rearranging, we get:

\[
w_2 = \cos^{-1} \left[ \frac{\cos w_{cp} - t_{00} - t_{10} \cos w_1}{t_{01} + \{1 - (t_{00} + t_{01} + t_{10})\} \cos w_1} \right] \quad (5.10)
\]

\[
= G_p(w_1, w_{cp}, t_{00}, t_{01}, t_{10})
\]
From (5.6),

$$w_2 = w_p \sqrt{1 - \left(\frac{w_1}{w_p}\right)^2} \text{ and } w_1 \leq w_p$$  \hspace{1cm} (5.11)

$$= G_{pd}(w_1, w_p, w_p)$$

Using (5.10) and (5.11) a non-linear error function is defined:

$$E_p(\bar{X}_1) = \sum_{n_1 \in \mathbb{R}^p} F_p(w_1)(G_p - G_{pd})^p$$  \hspace{1cm} (5.12)

$$\bar{X}_1 \triangleq (t_{00}, t_{01}, t_{11})^T$$  \hspace{1cm} (5.13)

An unconstrained non-linear optimization technique is used to minimize $E_p$.

5.3.2.3 ii) **Unconstrained Linear Approach**

Assume that the 1-D cutoff frequency exactly maps into the desired 2-D contour, i.e., (5.9) is exact. Substituting (5.8) and (5.11) in (5.1) we get:

$$\cos \frac{w_c}{w_p} = t_{00} + t_{01} \cos \left[ w_p \sqrt{1 - \left(\frac{w_1}{w_p}\right)^2} \right] + t_{10} \cos \frac{w_1}{w_p} +$$

$$\left(1-(t_{00} + t_{01} + t_{10})\right) \cos \frac{w_1}{w_p} \cos \left[ w_p \sqrt{1 - \left(\frac{w_1}{w_p}\right)^2} \right]$$

$$= G_{\ell}(w_1, w_p, w_p, t_{00}, t_{01}, t_{10})$$  \hspace{1cm} (5.14)

A linear error function is written from (5.14):

$$E_2(\bar{X}_1) = \sum_{n_1=1}^{N_1} F_2(w_1)(\cos \frac{w_c}{w_p} - G_{\ell})^p$$  \hspace{1cm} (5.15)
An unconstrained linear optimization technique is used to minimize $E_L$.

5.3.2.4 iii) Constrained Non-Linear Approach

The problem is stated as minimize (5.12) subject to (5.2). To solve this problem, a constrained non-linear optimization technique such as Box's Algorithm or Constrained Fletcher and Powell's Algorithm can be used.

5.3.2.5 iv) Constrained Linear Approach

The problem is stated as minimize (5.15) subject to (5.2). To solve this problem, a constrained linear optimization technique such as Simplex Method can be used.

5.3.2.6 Scaling

The coefficients obtained by i) and iii) may not satisfy (5.2). So, scaling is necessary. For a constant value of $w$, the shape of the contours of (5.1) are not changed by using:

$$F_S(w_1, w_2) = c_1 F_m(w_1, w_2) - c_2 \text{ and } c_1 \neq 0 \quad (5.16)$$

c_2 can be any value. By choosing,

$$c_1 = \frac{2}{\text{max}(F_m) - \text{min}(F_m)}; \quad c_2 = c_1 \text{max}(F_m) - 1 \quad (5.17)$$

$F_S(w_1, w_2)$ always satisfies (5.2). The proof is relatively straightforward. The 1-D cutoff frequency changes from $w_c$ to:

$$w_s = \cos^{-1}(c_1 \cos w_c - c_2) \quad (5.18)$$

This implies that the 1-D prototype must be redesigned after the transformation. Typical contours before and after
scaling are shown in Figure 5.6 and 5.7.

5.3.2.7 Test Results and Comparison

The problem is solved using a frequency resolution of 11 points in each axis. For i), the Fletcher and Powell's Algorithm is slightly modified so that if the search is in the nonfeasible region and if (5.2) is not satisfied, the search is forced back into the feasible region. The results are shown in Table 5.1. The minimum value of error function is 0.01.

In [4.5] the same problem is solved by ii). The minimum value of error function is 0.77. To make a comparison, the test results of i) are substituted in the error function of ii) and a value of 0.76 is obtained. Similarly the results of [4.5] by ii) are substituted in the error function of i) and a value of 1.15 is obtained. From these results, it is clearly seen that i) is better than ii).

5.3.2.8 Advantages and Disadvantages

The difference in values will definitely become large for MT of order >1x1 because the exact-mapping assumption ii) is not in general valid. It may be approximately true only for MT of order 1x1. In i) there is no exact mapping assumption and is valid for even MT of order >1x1. Moreover, i) gives optimum (locally) results while ii) gives only sub-optimum results. The computational time of i) is much less than that of ii). Approaches iii) and iv) give similar results like the scaled values. They do not change the cutoff frequency of the 1-D prototype and there is no necessity to redesign. The computational time
FIG. 5.6 TYPICAL CONTOURS BEFORE SCALING.

FIG. 5.7 TYPICAL CONTOURS AFTER SCALING.

FIG. 5.8 COMPLEX CEPSTRUM.
### Table 5.1 McClellan Transformation Coefficients

<table>
<thead>
<tr>
<th>Index</th>
<th>Unscaled $t_{ij}$</th>
<th>Scaled $t_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>-0.23999770E 01</td>
<td>0.17469520E-02</td>
</tr>
<tr>
<td>0 1</td>
<td>-0.58295090E-02</td>
<td>-0.17473480E-02</td>
</tr>
<tr>
<td>1 0</td>
<td>0.23710050E 01</td>
<td>0.71068970E 00</td>
</tr>
<tr>
<td>1 1</td>
<td>0.96519849E 00</td>
<td>0.28931050E 00</td>
</tr>
</tbody>
</table>

**SP**

\[ C_1 = 0.29774190E 00; \quad C_2 = -0.21120701E 00 \]

**S**

\[ w_{cp} = 0.5; \quad w_{ap} = 0.25; \quad b_p = 0.5 \]

**SP** - Scaling Parameters; **S** - Specifications

### Table 5.2 Filter Coefficients of 2-D Low-Pass Filter After 38 Iterations

<table>
<thead>
<tr>
<th>Index</th>
<th>Numerator Coefficients $p_{ij}$</th>
<th>Denominator Coefficients $q_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0.29093697E 00</td>
<td>0.10024860E 01</td>
</tr>
<tr>
<td>0 1</td>
<td>0.78712471E 00</td>
<td>0.48711520E-02</td>
</tr>
<tr>
<td>0 2</td>
<td>0.756402b4E 00</td>
<td>0.45430020E-02</td>
</tr>
<tr>
<td>1 0</td>
<td>0.83302937E 00</td>
<td>0.2-561470E-02</td>
</tr>
<tr>
<td>1 1</td>
<td>0.14657083E 01</td>
<td>0.34522960E-02</td>
</tr>
<tr>
<td>1 2</td>
<td>0.12863470E 01</td>
<td>0.28120710E-02</td>
</tr>
<tr>
<td>2 0</td>
<td>0.73706586E 00</td>
<td>0.64503890E-03</td>
</tr>
<tr>
<td>2 1</td>
<td>0.12702447E 01</td>
<td>0.56223120E-03</td>
</tr>
<tr>
<td>2 2</td>
<td>0.11708756E 01</td>
<td>0.26747070E-03</td>
</tr>
</tbody>
</table>

\[ A_f = \text{Gain Constant} = -0.84285550E-01 \]
for all the approaches are in the order i<ii<iii<iv. Based on computational considerations i) followed by scaling is the best approach for designing MTC's. This statement is contradictory to the view in [4,5] that ii) is preferable and superior. i) is also suited for higher order MT for designing higher order filters. MT is highly capable and filters of different shapes are possible like square, diamond, ring and fan.

5.3.3 Multi-Contour Approximation

For some of the problems, single contour approximation method may not serve the purpose. Test results indicate that for the same problem with \( w_c = 0.24\pi \), ii) failed to control the stop-band contours along the \( w_2 \) axis. The reason is that there is no control over the stop-band contours in the previous approach. So a method is suggested to control both the pass-band and stop-band contours.

Let \( w_{c_s} \) and \( w_{a_s} \) and \( w_{b_s} \) be the 1-D and 2-D stop-band cutoff frequencies (Figure 5.5). An error function is defined similar to (5.12).

\[
E_s(\tilde{X}_1) + \sum\limits_{t \in R_{p+t}} P_l(w_1)(G_s - G_1)P_s = 0
\]

(5.19)

where \( R_{p+t} \) is the line of frequency points in the pass and transition bands up to and including \( w_{c_s} \). The final error function is:

\[
E_1(\tilde{X}_1) = W'_p E'_p(\tilde{X}_1) + W'_s E'_s(\tilde{X}_1)
\]

(5.20)

where \( 0 < W_p + W_s < 1 \) and ' denotes the normalized values.
5.4 Windows in Recursive Filter Design

The use of windows in the Non-Recursive Filter (NF) design is well known. But in the Recursive Filter (RF) design, an area where they can be considerably useful, windows are not used. This is due to the lack of a theorem guaranteeing that the windowed filter would be recursively stable. The windows considered here are actually smoothing sequences except for the TW and they are not optimum. Some of the window functions and a proof for the stabilization effect of the EW are given below.

5.4.1 Window Functions

1) EW:

\[ w_e(n_1, n_2) = e^{-\left(\alpha |n_1| + \beta |n_2|\right)} ; \alpha, \beta \neq 0 \]  \hspace{1cm} (5.21)

2) W:

\[ w_\Delta(n_1, n_2) = \max\left\{\frac{|n_1|}{N_1} \left(1 - \frac{|n_2|}{N_2}\right), 0\right\} \]  \hspace{1cm} (5.22)

where \( N_1 \times N_2 \) is the dimension of the truncated sequence. The Gaussian Window (GW) and the Kaiser Window (KW) functions are defined as follows.

3) GW:

\[ w_g(n_1, n_2) = e^{-\frac{(n_1^2 + n_2^2)}{2s^2}} / (s\sqrt{2\pi}) \]  \hspace{1cm} (5.23)

4) KW:

\[ w_k(n_1, n_2) = \alpha J_1\left(2\pi \alpha \sqrt{n_1^2 + n_2^2}\right) / \sqrt{n_1^2 + n_2^2} \]  \hspace{1cm} (5.24)

where \( s \) is the standard deviation, \( \alpha \) is ratio between cutoff frequency and sampling frequency and \( J_1 \) is the Bessel Function of the first kind.
5.4.2 Proof for the Stabilization Effect of the Exponential Window

\[ b(n_1, n_2) \ast B(z_1, z_2), b_w(n_1, n_2) \ast B_w(z_1, z_2) \quad (5.25) \]

where \( b_w \) is the windowed sequence of \( b \).

\[
B_w(z_1, z_2) = \sum_{n_1=0}^{P} \sum_{n_2=0}^{Q} b(n_1, n_2) (z_1 e^{\alpha})^{-n_1}(z_2 e^{\beta})^{-n_2}
\]

\[= B(z_1 e^{\alpha}, z_2 e^{\beta}) \quad (5.26) \]

This accomplishes a mapping of the zero loci according to:

\[(z_1, z_2) \rightarrow (z_1 e^{-\alpha}, z_2 e^{-\beta}) \quad (5.27)\]

thus moving the zeros out of the region \( D_{++} \) which is required to be zero free for the stability of the ++ Quarter Plane Filter (QPF).

5.5 Design of 2-D Filters Using Spectral Factorization

In this section, the design is formulated by equations using the design approach and theory presented in Section 5.2.

The 1-D AF(S)'s TF in the cascade form is:

\[
H_a(s) = A_a \Pi_{k=1}^{P} \frac{\sum_{i=0}^{k} a_{ik} s^{-i}}{\sum_{i=0}^{d} b_{ik} s^{-i}} \quad (5.28)
\]

Applying BT, i.e., \( s \rightarrow 2(1-z^{-1})/T(1+z^{-1}) \), (Let \( T=1 \)) on (5.28), we get the 1-D DF(S)'s TF:

\[
H_d(z) = A_d \Pi_{k=1}^{P} \frac{\sum_{i=0}^{k} c_{ik} z^{-i}}{\sum_{i=0}^{d} d_{ik} z^{-i}} \quad (5.29)
\]
The 1-D ZP DF(U)'s TF is:

\[ H_z(Z) = H_d(Z)H_d(Z^{-1}) = |H_d(Z)|^2 \]  \hspace{1cm} (5.30)

'\(Z\)' form is converted into 'cos' form by using:

\[ z^i + z^{-i} = 2\cos(iw) \]  \hspace{1cm} (5.31)

Therefore, (5.30) becomes:

\[
H_z(w) = A_z \prod_{k=1}^{P} \frac{\sum_{i=0}^{K} e^{i \cos(iw)}}{\sum_{i=0}^{P} f^{k \cos(iw)}}
\]  \hspace{1cm} (5.32)

'Sum' form of cos can be expressed in the 'product' form. For example:

\[ \cos(2w) = 2\cos^2w - 1 \]  \hspace{1cm} (5.33)

Applying MT on (5.32), we get the 2-D ZP DF(U)'s TF,

\[
H_b(w_1, w_2) = A_b \prod_{k=1}^{P} \frac{\sum_{i=-P}^{P} \sum_{j=-P}^{P} g^{k \cos(iw_1) \cos(jw_2)}}{\sum_{i=-P}^{P} \sum_{j=-P}^{P} h^{k \cos(iw_1) \cos(jw_2)}}
\]  \hspace{1cm} (5.34)

which is before SF. Using (5.31), (5.34) can be written as:

\[
H_b(z_1, z_2) = B_b \prod_{k=1}^{P} \frac{\sum_{i=-P}^{P} \sum_{j=-P}^{P} i^{k \cos(iw_1) \cos(jw_2)}}{\sum_{i=-P}^{P} \sum_{j=-P}^{P} m^{k \cos(iw_1) \cos(jw_2)}}
\]  \hspace{1cm} (5.35)
Factorizing (5.35) using four-factor SF, we get:

\[ H_{D}(z_1, z_2) = B_{D} \prod_{k=1}^{K} \left( H_{b_{++}}^k H_{b_{+-}}^k H_{b_{-+}}^k H_{b_{--}}^k \right) \] (5.36)

where \( H_{b_{++}}^k \ldots \) are stable filters in ++ \ldots quadrants. The final ++ QPF's TF is:

\[ H_f(z_1, z_2) = A_f \prod_{k=1}^{K} \left( \sum_{i=0}^{P} \sum_{j=0}^{P} p_{ij} z_1^{-i} z_2^{-j} \right) \] (5.37)

\[ H_f(z_1, z_2) = \frac{P_f(z_1, z_2)}{Q_f(z_1, z_2)} \]

Theorems 5-9 can easily be implemented to incorporate stability. If \( m^k(n_1, n_2) \approx \hat{m}_a^k(n_1, n_2) \), then in practice \( m^k(n_1, n_2) \approx \hat{m}_a^k(n_1, n_2) \), an aliased version. If \( \hat{m}_k(n_1, n_2) = 0 \) for \( n_1, n_2 \not\in R_{++} \) then the filter is stable; otherwise unstable [2,4]. Using these theories, the stability error criteria is written as:

\[ E_s^k(\bar{x}_2) = \max_{n_1, n_2 \not\in R_{++}} \left| m^k_{a_{i=1}}(n_1, n_2) - \hat{m}(n_1, n_2) \right| \] (5.38)

\[ i \triangleq \text{Iteration } #; \bar{x}_2 = [q^k(n_1, n_2)]^T \]

Let \( q^k_{a_{i=1}}(n_1, n_2) \) be the denominator coefficients corresponding to \( \hat{m}_a^k(n_1, n_2) \). \( n_1, n_2 \in R_{++} \). \( R_{++} \) can be taken as a finite order \((N_1 \times N_2)\) array. In this case, rapid decay of the cepstrum sequence is taken into account. Hence, the stability...
of the filter is guaranteed. The error function for
magnitude is:

\[ E_m(\bar{x}_2) = \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} F_m(w_1, w_2)(|Q_{f_1}(w_1, w_2)| - S_d(w_1, w_2))^p \]  

(5.39)

where \( S_d \) is the inverse spectrum of the desired MR. The
final error function is:

\[ E_2(\bar{x}_2) = \left[ \sum_{k=1}^{K} \beta^k E_s^k(\bar{x}_2) \right] + \gamma E_m^r(\bar{x}_2) \]  

(5.40)

\[ E_2(\bar{x}_2) = [ \sum_{k=1}^{K} \beta^k E_s^k(\bar{x}_2) ] + \gamma E_m^r(\bar{x}_2) \]

\( \gamma = 1 \)

\[ \sum_{k=1}^{K} \beta^k = 1 \]

where \( \beta^k \) is a parameter which determines the relative impor-
tance between stability and magnitude. To stabilize an
unstable 2-D filter, its ZP TF should be used in the calcu-
luation.

The errors in the MR and GDR are corrected as dis-
cussed in Section 5.2.4, to get the final filter.

5.6 Numerical Implementation

So far the design has been described in detail. The
important steps in the design are summarized as follows.

5.6.1 Steps

1. Choose a stable 1-D filter. Derive the relation-
ships between different coefficients. Get the 1-D prototype
filter [1-D ZP DF(U)].

2. Design the MTC's by using any one of the described
approaches so as to satisfy the MS. Apply the MT to get the 2-D ZP DF(U).

3. Spectrally factorize this filter using CC. If the filter is stable (i.e., $\varepsilon_s \leq \varepsilon; \varepsilon = 10^{-8}$) go to the next step; otherwise follow the procedures of the previous section.

4. Correct the errors in the FR by using the MS and the GDS.

5.6.2 **Computational Time**

Even though the block diagram (Figure 5.1) appears with several blocks, the design is simple and relatively fast. To get a 2-D ZP DF(U), starting from a 1-D AF(S), the computer takes only 0.25 second for known MTC's. Design of MTC's using approach i) takes only 5 seconds. To get a stable factorized filter, it takes only 10 seconds using a 1024 point radix-4 FFT which is very fast. So far there is no design technique which can give a 2-D filter with approximate MR within 15.25 seconds. Error correction takes much time (several minutes) depending on the optimization technique. Even then, the total design time for a filter with desired MS and GDS is less than 1/3 of the time for the same filter using a direct design.

5.6.3 **Memory Requirement**

The computer memory required at various stages of this design is much less ($<50K$) than the direct design. The direct design requires more than three times the memory required for the transformation design for the same frequency
resolution.

5.6.4 Complex Cepstrum

The 2-D CC (Figure 5.8) is defined as the Inverse Fourier Transform (IFT) of the Complex Logarithm (CL) of the Fourier Transform (FT) of a sequence.

5.6.4.1 Computation of Complex Cepstrum

Consider the CL of the array \( Q(z_1, z_2) \). Let \( \beta(z_1, z_2) \) be its phase angle as defined in (2.15). Then:

\[
Q(z_1, z_2) = |Q(z_1, z_2)| \text{Exp}(j\beta(z_1, z_2))
\]

(5.41)

Therefore,

\[
\hat{Q}(z_1, z_2) = \text{Ln}(|Q(z_1, z_2)|) + j\beta(z_1, z_2)
\]

(5.42)

In order to ensure the analyticity of \( \hat{Q} \), the phase angle \( \beta \) must be continuous and periodic. After taking the FFT, the computer gives the principal value of the phase between \(-\pi\) and \(\pi\), introducing phase jumps of \(2\pi\). To make it continuous, appropriate multiples of \(2\pi\) are added to the principal value. This process is called "Phase Unwrapping [PU]". Finally, the phase is made periodic by subtracting the observed amount of linear phase from the unwrapped phase. This process is called "Linear Phase Removal [LPR]".

5.6.4.2 Methods

Several methods [2.4, 2.7, 5.2] are available for the computation of the CC. Tribolet's PU algorithm [5.3] is very efficient and accurate and well suited for higher order sequences. It combines the information contained in both the phase derivative and the principal value of the phase into an adaptive numerical integration scheme. It is very
reliable and gives better results than the numerical integration and the Schafer's algorithm [1.3]. The algorithm can be modified to suit the need. When the phase derivative is quite irregular and shows large peaks, more time is spent in one particular loop during the execution. In order to reduce the time, the improvements suggested by Bonzanigo [5.4] and a radix-4 FFT are incorporated.

5.6.4.3 Advantages

Evaluation of the CC involves the FFT and the IPFT operations and the computations associated with the CL, the PU and the LPR. The Modified Jury Table (MJT) method [2.3] requires computations of $N^2 2^{N-1}$ for an $N \times N$ filter. The autocorrelation method [3.11, 3.12] based on RC requires only $N^2 \log N$ computations. The CC method is better than these two methods and requires 50% less calculations than the autocorrelation method [2.4]. The MJT method and many of the methods in sections 2.7 and 2.8 are extremely difficult to implement for higher order filters and are limited to QPF's. But the CC method is easy to implement, also suited for higher order filters and even applicable to Half Plane Filters (HPF's).

5.7 Remarks

1. The method can be started from a 1-D DF(S) or 1-D ZP DF(U) or directly from a 2-D ZP DF(U) instead of starting from a 1-D AF(S).

2. Analog and/or digital transformation (1-D and 2-D) can be used with or without conversion transformation
to design various filters (e.g., low-pass, high-pass, band-pass, band-stop, fan, etc.).

3. Suppose we want to stabilize an unstable filter. Suppose the same MR and arbitrary GDR are desired. With the existing methods [2.5, 2.6] the filter can be stabilized. But the MR is affected and the GDR can be any response. In this method, the unstable filter can be stabilized and since error correction is employed, the same MR and arbitrary GDR are possible to achieve.

4. Because of the MT characteristics, the method is highly flexible for any order and any structure at the same time versatile.

5. Unlike other methods (DHT, DPLSI, DBT) in SF no counter example was shown because, Pistor [2.5] showed that the stability depends on the rate of convergence of the factors and the order of approximation.

6. If the energy of the factors is not concentrated in a region near the origin which is very rare, then the size of the FFT and/or the order of the filter can be increased to get a stable filter. Alternatively, the filter can be realized as a cascade of all pole filters and a Non-recursive Filter (NF) to compensate for the loss of energy [4.6]. But this filter has the disadvantage of more arithmetic operations per output.

7. The stability error $E_s$ is due to two sources of errors. The first source is the numerical error associated with computing the FFT. It is a function of the machine wordlength and the size of the FFT, and generally quite
small. The second source is the aliasing error associated with computing the cepstrum. It is a function of the size of the FFT and the spatial spread of the cepstrum sequence, and is much higher than the first error. Both the errors are controllable to certain extent by varying the size of the FFT.

The size of the FFT should be \( \geq \) the dimension of the truncated factors for smaller errors [3.11]. For almost all factorizations of practical interest, a FFT of size 1024 x 1024 would provide accuracies well within the numerical error of the FFT itself. For a modest-size FFT (e.g., 32x32, 64x64), the errors are small.

8. The choice of \( \varepsilon \) depends on the desired FR and the size of the FFT.

9. In this design, prior knowledge of the factors is unknown unless the sequence is factorized. Hence, the stability error formulation is different from the direct design [3.12] where an initial guess is available for the factors at the beginning.

10. The method is also suited for higher order filters because of the partitioning of the numerator and the denominator. For full interaction between these two, optimization stages 2 and 3 can be combined. In this case, the algorithm will take more time because of the increased number of parameters. Moreover, once the denominator is factorized, only very little improvement can be done in the FR using the denominator coefficients without affecting the
stability. These are the reasons why we are going to step 4 after checking the stability in step 3. Since the full interaction results in very little improvement in the FR and since it can be obtained only at the expense of increased computational time, it is not followed in this design.

11. This design is a 'pole-zero' design and is better than the 'all pole' [3.6,3.11,3.12,4.6] and 'all-zero' designs to obtain a general FR.

12. Zero-Phase Filters (ZPF's) are obtained by simply cascading the factorized filters.

13. Half-Plane Filters (HPF's) are obtained by simply taking the support region as half-planes and by using the two-factor SF.

14. The design is extendable to N-D filters and the design of 1-D filters is a sub-class of this design.

15. It is easy to use other error norms and optimum Chebyshev Filters are possible.

16. Since non-linear optimization techniques are employed, the usual problems (e.g., sensitivity to initial guess, slow convergence, local minimum) may be encountered. However, the successive approximation procedure allows quite a direct control over the form and magnitude of the approximation error. This is accomplished through choice of error norm, frequency weighting function, weighting factor, frequency resolution and of course the order of the approximation.
5.8 Examples

5.8.1 Specifications

1) Design an approximately Circularly Symmetric Low-Pass Filter (CS LPF) starting from a second order Butterworth Filter. \( G_{id}(w_1, w_2) = 1.15 \) in the pass-band.

2) Using the same starting filter design a Fan Filter (FF1) having pass-band along \( w_1 \) axis. \( G_{id}(w_1, w_2) = 1.19 \) in the pass-band.

3) Similarly design another Fan Filter (FF2) having pass-band along \( w_2 \) axis. \( G_{id}(w_1, w_2) = 1.16 \) in the pass-band.

For all the cases, take the desired MS as that of the 2-D ZP DF(U). In the pass-band \( |H_b(w_1, w_2)| > 0.707 \) and in the stop-band \( |H_b(w_1, w_2)| < 0.1 \). The desired specifications are shown in Figure 5.9-5.11.

5.8.2 Chosen Parameters

\( \alpha = \beta = 0.13 \) for the EW. The initial guess for the parameter vector \( \bar{X} \), for the error correction is taken to be the corresponding coefficient value of the factorized filter. The other parameter values are the same as in Section 3.5.9. The value of LIMIT is set at a desired value.

5.8.3 Results

The FR for the 1-D AF(S), the 1-D DF(S) and the 1-D ZP DF(U) are shown in Figures 5.12-5.14, 5.15-5.17 and 5.18-5.20.

The MR and the GDR for the 2-D ZP DF (CS LPF) are shown in Figures 5.21 and 5.22. The same GDR is obtained for the other 2-D ZP DF's (FF1 and FF2). The FR of the factorized and the error corrected 2-D DF(S) (CS LPF) are
FIG. 5.9 DESIRED MAGNITUDE SPECIFICATIONS FOR LOW-PASS FILTER.

FIG. 5.10 DESIRED MAGNITUDE SPECIFICATIONS FOR FAN FILTER #1.

FIG. 5.11 DESIRED MAGNITUDE SPECIFICATIONS FOR FAN FILTER #2.
FIG. 5.12 MAGNITUDE RESPONSE OF 1-D ANALOG LOW-PASS FILTER.
FIG. 5.13 PHASE RESPONSE OF 1-D ANALOG LOW-PASS FILTER.
FIG. 5.14 GROUP DELAY RESPONSE OF 1-D ANALOG LOW-PASS FILTER.
FIG. 5.15 MAGNITUDE RESPONSE OF 1-D DIGITAL LOW-PASS FILTER.
FIG. 5.16 PHASE RESPONSE OF 1-D DIGITAL LOW-PASS FILTER.
FIG. 5.17 GROUP DELAY RESPONSE OF 1-D DIGITAL LOW-PASS FILTER.
FIG. 5.18 MAGNITUDE RESPONSE OF 1-D ZERO PHASE DIGITAL LOW-PASS FILTER.
FIG. 5.19 PHASE RESPONSE OF 1-D ZERO PHASE DIGITAL LOW-PASS FILTER.
FIG. 5.20 GROUP DELAY RESPONSE OF 1-D ZERO PHASE DIGITAL LOW-PASS FILTER.
FIG. 5.21 MAGNITUDE RESPONSE OF 2-D ZERO PHASE LOW-PASS FILTER.
FIG. 5.22 GROUP DELAY-1 RESPONSE OF 2-D ZERO PHASE LOW-PASS FILTER (GROUP DELAY-2 RESPONSE IS EXACTLY THE SAME).

[SAME GROUP DELAY RESPONSES ARE OBTAINED FOR FAN FILTER #1 AND #2].
shown in Figures 5.23-5.25 and 5.26-5.28. The filter coefficients for the final filter are given in Table 5.2. The plots related to the optimization parameters are shown in Figures 5.29-5.32.

The MR of the 2-D ZP DF (FF1) is shown in Figure 5.33. The FR of the factorized and the error corrected 2-D DF(S) (FF1) are shown in Figures 5.34-5.36 and 5.37-5.39. The filter coefficients for the final filter are given in Table 5.3.

The results for the FF2 are given in the same order in Figures 5.40, 5.41-5.43, 5.44-5.46 and in Table 5.4.

5.8.4 Comparison of Results

The obtained relative Root Mean Square (RMS) errors for the FR and the other important values are given in Table 5.5 for all the examples. To get an idea of the performance of this method, one can approximately compare this Table with Table 3.5 which is for the direct design. In this method, for most of the filters the relative RMS errors for the MR and the GDR are ≈0.08 and ≈0.02. The same values for Example 2 of [3.1] are ≈0.15 and ≈0.17. The error values in [3.1,3.2,3.4 and 3.5] are higher than the values of this method. Moreover, those values are obtained with more iteration and/or increased filter order.
FIG. 5.23 MAGNITUDE RESPONSE OF 2-D LOW-PASS FILTER (AFTER FACTORIZATION).
FIG. 5.24 GROUP-DELAY-1 RESPONSE OF 2-D LOW-PASS FILTER (AFTER FACTORIZATION).
FIG. 5.25 GROUP-DELAY-2 RESPONSE OF 2-D LOW-PASS FILTER (AFTER FACTORIZATION).
FIG. 5:26 MAGNITUDE RESPONSE OF 20D LOW-PASS FILTER (AFTER ERROR CORRECTION).
FIG. 5.27 GROUP-DELAY-1 RESPONSE OF 2-D LOW PASS FILTER (AFTER ERROR CORRECTION)
FIG. 5.28 GROUP-DELAY-2 RESPONSE OF 2-D LOW-PASS FILTER (AFTER ERROR CORRECTION).
FIG. 5.28a  IMPULSE RESPONSE OF 2-D LOW-PASS FILTER (AFTER ERROR CORRECTION).
FIG. 5.29 VALUE OF ERROR FUNCTION VERSUS NUMBER OF ITERATIONS.
FIG. 5.30 NUMBER OF FUNCTION EVALUATIONS VERSUS NUMBER OF ITERATIONS.

XMIN = .100000E 01
XMAX = .380000E 02
YMIN = .600000E 01
YMAX = .333000E 03
 FIG. 5.31 VALUE OF ALPHA VERSUS NUMBER OF ITERATIONS.
FIG. 5.32 COMPUTATIONAL TIME VERSUS NUMBER OF ITERATIONS.

XMIN = .10000E 01
XMAX = .38000E 02
YMIN = .49995E 00
YMAX = .27747E 02
FIG. 5.33 MAGNITUDE RESPONSE OF 2-D ZERO PHASE FAN FILTER II (FOR GROUP DELAY 1 AND 2 RESPONSES REFER FIG. 5.22).
FIG. 5.34 MAGNITUDE RESPONSE OF 2-D FAN FILTER #1 (AFTER FACTORIZATION).
FIG. 5.35 GROUP-DELAY-1 RESPONSE OF 2-D FAN FILTER 
#1 (AFTER FACTORIZATION).
FIG. 5.36 GROUP-DELAY-2 RESPONSE OF 2-D FAN FILTER #1 (AFTER FACTORIZATION).
FIG. 5.37 MAGNITUDE RESPONSE OF 2-D FAN FILTER #1 (AFTER ERROR CORRECTION).
FIG. 5.38 GROUP-DELAY-1 RESPONSE OF 2-D FAN FILTER 
#1 (AFTER ERROR CORRECTION).
FIG. 5.39 GROUP-DELAY-2 RESPONSE OF 2-D FAN FILTER #1 (AFTER ERROR CORRECTION).
FIG. 5.39a IMPULSE RESPONSE OF 2-D FAN FILTER #1 (AFTER ERROR CORRECTION)
FIG. 5.40  MAGNITUDE RESPONSE OF 2-D ZERO PHASE
FAN FILTER #2 (FOR GROUP DELAY 1 AND
2 RESPONSES REFER FIG. 5.22).
FIG. 5.41 MAGNITUDE RESPONSE OF 2-D FAN FILTER #2 (AFTER FACTORIZATION).
FIG. 5.42  GROUP-DELAY-1 RESPONSE OF 2-D FAN FILTER #2 (AFTER FACTORIZATION).
FIG. 5.43 GROUP-DELAY-2 RESPONSE OF 2-D FAN FILTER #2 (AFTER FACTORIZATION).
FIG. 5.44 MAGNITUDE RESPONSE OF 2-D FAN FILTER #2
(AFTER ERROR CORRECTION).
FIG. 5.45 GROUP-DELAY-1 RESPONSE OF 2-D FAN FILTER #2 (AFTER ERROR CORRECTION).
FIG. 5.46 GROUP-DELAY-2 RESPONSE OF 2-D PAN FILTER #2 (AFTER ERROR CORRECTION).
FIG. 5.46a IMPULSE RESPONSE OF 2-D FAN FILTER
#2 (AFTER ERROR CORRECTION).
<table>
<thead>
<tr>
<th>Index i</th>
<th>Numerator Coefficients $p_{ij}$</th>
<th>Denominator Coefficients $q_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0.46810814E-01</td>
<td>0.10032320E 01</td>
</tr>
<tr>
<td>0 1</td>
<td>0.82073250E 00</td>
<td>0.52616160E-05</td>
</tr>
<tr>
<td>0 2</td>
<td>0.76466253E 00</td>
<td>-0.35404830E-02</td>
</tr>
<tr>
<td>1 0</td>
<td>-0.73908315E 00</td>
<td>-0.21489600E-02</td>
</tr>
<tr>
<td>1 1</td>
<td>-0.16814818E 01</td>
<td>-0.26726040E-03</td>
</tr>
<tr>
<td>1 2</td>
<td>-0.15794920E 01</td>
<td>0.16356650E-02</td>
</tr>
<tr>
<td>2 0</td>
<td>0.74659214E 00</td>
<td>0.31076250E-03</td>
</tr>
<tr>
<td>2 1</td>
<td>0.16043604E 01</td>
<td>-0.23567350E-04</td>
</tr>
<tr>
<td>2 2</td>
<td>0.14379816E 01</td>
<td>-0.18971710E-03</td>
</tr>
</tbody>
</table>

$A_f$ - Gain Constant $= 0.11492024E 00$

**TABLE 5.3 FILTER COEFFICIENTS OF 2-D PAN FILTER #1 AFTER 31 ITERATIONS**

<table>
<thead>
<tr>
<th>Index i</th>
<th>Numerator Coefficients $p_{ij}$</th>
<th>Denominator Coefficients $q_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0.28423542E 00</td>
<td>0.10032310E 01</td>
</tr>
<tr>
<td>0 1</td>
<td>-0.85248748E 00</td>
<td>-0.52763780E-05</td>
</tr>
<tr>
<td>0 2</td>
<td>0.73660024E 00</td>
<td>-0.35406900E-02</td>
</tr>
<tr>
<td>1 0</td>
<td>0.87439326E 00</td>
<td>0.21482690E-02</td>
</tr>
<tr>
<td>1 1</td>
<td>-0.16731863E 01</td>
<td>-0.26735770E-03</td>
</tr>
<tr>
<td>1 2</td>
<td>0.13283249E 01</td>
<td>-0.16351230E-02</td>
</tr>
<tr>
<td>2 0</td>
<td>0.76255851E 00</td>
<td>0.31066820E-03</td>
</tr>
<tr>
<td>2 1</td>
<td>-0.13412837E 01</td>
<td>0.23576660E-04</td>
</tr>
<tr>
<td>2 2</td>
<td>0.11451028E 01</td>
<td>-0.18965140E-03</td>
</tr>
</tbody>
</table>

$A_f$ - Gain Constant $= 0.12065690E 00$

**TABLE 5.4 FILTER COEFFICIENTS OF 2-D PAN FILTER #2 AFTER 51 ITERATIONS**
<table>
<thead>
<tr>
<th>Description</th>
<th>NI</th>
<th>NFE</th>
<th>VEF</th>
<th>CT (min)</th>
<th>MRMS</th>
<th>GD1RMS</th>
<th>GD2RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSLPF</td>
<td>I</td>
<td>0</td>
<td>1</td>
<td>217.0</td>
<td>0.0833</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>38</td>
<td>333</td>
<td>0.1762</td>
<td>27.7472</td>
<td>0.1073</td>
<td>0.0116</td>
</tr>
<tr>
<td>FF1</td>
<td>I</td>
<td>0</td>
<td>1</td>
<td>241.0</td>
<td>0.0833</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>31</td>
<td>207</td>
<td>0.1218</td>
<td>17.2482</td>
<td>0.0765</td>
<td>0.0269</td>
</tr>
<tr>
<td>FF2</td>
<td>I</td>
<td>0</td>
<td>1</td>
<td>264.0</td>
<td>0.0833</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>51</td>
<td>333</td>
<td>0.1266</td>
<td>27.7472</td>
<td>0.0799</td>
<td>0.0229</td>
</tr>
</tbody>
</table>

I - Initial; F - Final

**TABLE 5.5 RESULTS OF TRANSFORMATION DESIGN**
5.9 Conclusion

A method has been described for the transformation design of 2-D filters using SF. The advantages and disadvantages have been presented in the course of discussion. (They are summarized in the next chapter). Design examples and results have been given to demonstrate the capabilities of the method.
CHAPTER 6

DISCUSSIONS, EXTENSIONS AND CONCLUSIONS

6.1 Introduction

This last chapter presents a discussion of the design method, the possible extensions and the derived conclusions. The advantages, the disadvantages and the limitations of the design are also presented.

6.2 Discussion of the Design

From the discussion in Section 1.3, one can observe that the features in the advantages of the Nonrecursive Filter (NF) are not met by the Recursive Filter (RF). Similarly, the features in the advantages of the RF are not met by the NF. So an attempt to eliminate the disadvantages in either of the filters is very much preferred and will lead to a very good design.

Only very few attempts [3.1, 3.2, 3.3] have been made in the RF case and they suffer from some other disadvantages (see Section 3.5.12). In this design, these disadvantages are also taken into consideration and the problems are eliminated or reduced as much as possible.

6.2.1 Advantages

1) The method is simple, fast and requires less memory.

2) It is a general method. It satisfies most of the preferable requirements for a design (Section 2.5).

3) General class of 'pole-zero' filters are designed. Almost all types of practical filters (low-
4) Filters with any order and any structure can be designed.

5) The method is used to obtain both the Magnitude Specification (MS) and the Group Delay Specification (GDS).

6) The Spectral Factorization (SF) method of achieving stability is better than the other methods (Section 5.2.3).

7) The Exponential Window (EW) gives sharper cut-off, smoothing and additional stability.

8) The designed filters are guaranteed to be stable.

9) Non-linear optimization techniques, particularly the gradient methods which are faster are used to obtain optimum (locally) filters.

10) The method can be used to stabilize unstable filters with specified MS and GDS.

11) The design is extendable to N-D filters and other error norms are possible. The design of 1-D filters is a sub-class of this design. Many of the advantages presented here are not present in many of the existing designs (Refer to the discussions in the previous chapter and in the Section 1.4).

6.2.2 Disadvantages

1) Due to the partitioning of the numerator and the denominator there is no full interaction between these two. However, partitioning has the advantages given in
Remark 10 of the previous chapter.

2) The usual problems of the non-linear optimization may be encountered (Remark 16).

The first disadvantage can be eliminated at the expense of increased computational time. The second disadvantage is there in all the methods which use the non-linear optimization techniques. These two disadvantages are minor in nature, compared to the major disadvantages (see Section 1.4) present in many of the methods available in the literature.

6.2.3 Limitations

The design is limited only by the designs available for finding the McClellan Transformation Coefficients. Since, almost all practical filters can be designed, this is not a serious limitation.

6.3 Extensions

1) This Quarter Plane Filter\(^3\) (QPF) design can be extended to the general class of Half Plane Filter (HPF) design.

2) More work can be done to obtain the MTC's for arbitrary specifications.

3) Optimum windows can be designed for the RF case.

4) Under what suppositions is it possible to factorize exactly an unstable filter into a stable filter that have a minimum finite number of coefficients; and what is the number of these coefficients? If such a
factorization exists, how can it be found? These two questions remain unanswered.

In spite of these unanswered questions, the results presented in this thesis are of practical value.

6.4 Conclusions

A new method has been developed for the design of 2-D recursive digital filters with arbitrary MS and GDS (Chapter 5). This is the first transformation design using SF to design 'pole-zero' filters. Also, this is the first transformation design to guarantee the stability of the filter using Complex Cepstrum (CC). The method uses non-linear optimization techniques and yields optimum (locally) filters. It eliminates the stability, higher order and structure problems and very much reduces the complexity, time and memory problems which are present in the existing designs (Chapter 1). The design capabilities of the method have been proven to be better than the direct designs. The implementation results of a direct design (Chapter 3) have been used for the comparison. The theoretical foundation for both these methods has been given briefly in Chapter 4 and Chapter 2. Advantages, disadvantages, limitations and extensions of the design have been discussed in this Chapter 6.

Methods for designing MTC's, stabilization of unstable filters, stability error-criteria based on CC, guaranteed stability, GDS, cascaded structure and other advantages mentioned are the special features of this design and are improvements overs several designs in
various aspects.
LIST OF REFERENCES

1. General


2. Stability


3. Direct Design.


4. Transformation Design


5. Other References


VITA AUCTORIS

April 22, 1957

Born in Modakurichi, Coimbatore District (Dt). Tamil Nadu (TN) State, India.

Dates Attended

June 1967-Mar. 1973
Passed the Secondary School Leaving Certificate (SSLC) Examination conducted by the Board of Secondary Education, Government of Tamil Nadu, through the Government Boys' High School, SULUR-641402, Coimbatore (Dt), TN, India.

June 1973-May 1974
Passed the Pre-University Course (PUC) conducted by the University of Madras, MADRAS-600002, through the P.S.G. Arts College, COIMBATORE-641014, TN, India.

July 1974-May 1979
Passed the Bachelor of Engineering (Honours) (Electronics and Communication Engineering) [BE Hons. (ECE)] degree course conducted by the University of Madras, MADRAS-600002, through the P.S.G. College of Technology, COIMBATORE-641004, TN, India.

July 1979-Oct. 1979
Studied in the Master of Engineering (Computer Science) [ME (Comp. Sc.)] course offered by the same University through the same College.

January 1980
Joined in the Master of Applied Science (M.A.Sc.) course offered by the University of Windsor, WINDSOR, N9B 3P4, Ontario, Canada.