NOTICE

The quality of this microform 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 an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.
Echo Tail Canceller based on AIFIR Filtering

by

Anis Abousaada, B.A.Sc.

A thesis submitted to the School of Graduate Studies and Research in partial fulfilment of the requirements for the degree Master of Applied Science

Ottawa-Carleton Institute for Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
University of Ottawa

© Anis Ahmed Abousaada, Ottawa, Canada, 1990
The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

I hereby declare that I am the sole author of this thesis.

I authorize The University of Ottawa to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Anis Abousaada

I further authorize The University of Ottawa to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Anis Abousaada
To my family
Abstract

Echoes are due to impedance mismatches in the hybrid transformer of the subscriber loop. A solution to the echo problem is to use an adaptive FIR (AFIR) echo canceller to simulate the echo path impulse response. Due to the long tail of the echo path impulse response, high order AFIR echo canceller is needed. In this thesis, the problem of computationally efficient design for echo cancellers is studied. This thesis contributes a computationally efficient echo tail canceller. The proposed structure is an adaptive version of the Interpolated FIR (IFIR) filter. The results obtained with this new approach are compared to those obtained with an AFIR direct form echo canceller. The results show that significant savings in the required number of arithmetic operations can be attained. The proposed system requires approximately 31% of the hardware of the AFIR echo canceller.

A two-stage echo canceller utilized in the U-interface at Basic Rate Access-ISDN is investigated. The first stage (Main Echo Canceller) is comprised of an AFIR filter which cancels the main echo pulse amid the first few symbol intervals. Cancellation of the echo tail is performed by the AIFIR tail canceller. In this thesis, 60 dB echo cancellation accuracy was selected as the performance requirement to be met by the proposed two stage canceller. To evaluate the performance of this architecture, echo impulse responses were produced for the Bellcore loop structures. Adequate echo cancellation performance (60 dB) was achieved for various Bellcore loops. Simulations were carried out to investigate the performance of the proposed two-stage echo canceller. These investigations were on the effects of parameters such as the pass-band edge and the stopband attenuation of the interpolator as well as length of the first stage echo canceller. The performance of a compromise interpolator was also investigated.
Acknowledgements

I would like to express my utmost gratitude and appreciation to my thesis supervisor Dr T. Aboulnasr and co-supervisor Dr W. Steenaart. I am deeply indebted to them for their teaching, their advice, their encouragement, and their patience during times of stress and difficulty. This thesis would not have been accomplished without their combined auspices.

*My utmost thanks go to my family for their patience, sacrifice, and support during all those years of my studies.*

I am grateful for the prolific stimulating discussions with Mr G. Davidson and Dr P. Mohanraj. I am also thankful to Dr P. Galko for negotiating my DTFW scholarship. Furthermore, his help and expertise in IATpX are acknowledged. I would also like to thank Dr S. Aly and Dr D. Falconer for supplying some loop responses and the loop generating program. A special thanks to, M. Abdulrahman, for reading parts of this thesis. Finally, I would like to thank all my graduate/undergraduate teachers at Ottawa University (especially Dr A. Yongaçoğlu) and my fellow graduate students in the Digital Communications and Signal Processing Group (especially Dr D. Makrakis & Mr S. Spenler).

I wish to thank the examiners Dr R. Gouban, Dr D. Falconer, and Dr W. McGee for constructive comments and suggestions.

I would like to acknowledge the Ottawa University, NSERC, TRIO, and Civil Aviation (*LIBYA*) funding which made this research possible.
# Contents

1 **Introduction**

1.1 Problem Statement .............................................. 20  
1.2 Thesis Organization ............................................. 25  
1.3 Conclusion .................................................................. 27  

2 **Review of Transmission Issues for ISDN Basic Rate Access**  

2.1 Impairment Constraints of the DSL Transmission Medium ........ 30  
2.2 ISDN Basic Access Transmission Techniques ......................... 31  
2.2.1 Comparison of Line Codes for DSL ................................. 33  
2.2.2 Time Compression Multiplexing vs Echo Canceller Hybrid .... 33  
2.3 Echo Canceller ....................................................... 41  
2.3.1 Linear Echo Canceller ............................................. 42  
2.3.2 Nonlinear Echo Canceller ......................................... 48  
2.3.3 Recent Techniques in Echo Cancellation .......................... 54  
2.4 Echo Tail Cancellers .................................................. 59  
2.5 Conclusion .................................................................. 65  

3 **Theory of Interpolated FIR Filters**  

3.1 Introduction .......................................................... 66  
3.2 Mathematical Formulation of IFIR Filters in the Time Domain .... 69  
3.3 Design Procedure ..................................................... 71  
3.3.1 Selection of $l$ ..................................................... 72  

3.3.2 Design Steps ............................................ 72
3.4 Computational Requirements ........................................ 74
  3.4.1 Model Filter ............................................ 74
  3.4.2 Interpolator Filter ........................................ 75
3.5 Finite Wordlength Properties of IFIR filters ......................... 80
3.6 Design Examples ............................................... 82
3.7 Conclusion ................................................... 87

4 AIFIR Echo Tail Canceller ........................................... 88
  4.1 System Description of Proposed Two-Stage EC ................. 89
  4.2 Analysis of Adaptive IFIR Structures ...................... 91
    4.2.1 Complexity Comparison: AIFIR vs AFIR ............ 94
    4.2.2 Two Stage Echo Cancellation Utilizing AIFIR Filter .... 96
  4.3 Conclusion ................................................. 101

5 Simulations and Results ............................................ 103
  5.1 Simulation Details ........................................... 104
    5.1.1 Echo Canceller Length Requirement ................... 105
  5.2 AIFIR Echo Tail Canceller: Design ......................... 110
    5.2.1 Echo Tail Frequency Response ....................... 110
    5.2.2 AIFIR Design Steps ................................... 111
  5.3 Simulation Set-Up ........................................... 121
    5.3.1 Investigation#1: Effects of Main Canceller Length ...... 122
    5.3.2 Investigation#2: Effects of Interpolator Passband $f_p$ .... 133
    5.3.3 Investigation#3: Performance of a Compromise Interpolator .... 152
    5.3.4 Investigation#4: Effects of Interpolator Stopband Attenuation $\delta$ 156
  5.4 Comparison With Orthonormal-Functions-Based EC ................ 167
  5.5 Conclusion ................................................... 170
6 Conclusions and Recommendations

6.1 Concluding Remarks ........................................... 172
6.2 Suggestions for Future Work ................................. 173

A Loop Information .................................................. 175
A.1 Bellcore Loop Configurations ................................. 175
A.2 Analysis of Hybrid Systems ................................. 179
A.3 Cable Characteristics ........................................... 181

B Simulations Information ......................................... 186
B.1 Interpolator Specifications ................................. 186
B.2 Learning Curves ........................................... 188

C Proofs .......................................................... 190
C.1 Is $R$ Diagonal Matrix? ................................. 190

Bibliography ......................................................... 191
List of Figures

1.1 Block Diagram of ISDN Basic Rate Access. .......................... 21
1.2 Block diagram of echo cancellation technique. ..................... 22
1.3 Block Diagram of DSP Transceiver Architecture. ................... 23
1.4 Simplified diagram of a hybrid transformer. ....................... 26
1.5 Block diagram of the proposed two-stage echo canceller. .......... 28

2.1 NEXT and FEXT Crosstalk on the DSL. ............................. 32
2.2 Low frequency cut-off and echo tail of a hybrid transformer. ....... 39
2.3 Time compression multiplexing method of full-duplex data transmission. 41
2.4 Classic configuration of echo canceller. .......................... 41
2.5 Block diagram of a digital echo canceller with analog domain echo cancellation. .................................................. 43
2.6 Block diagram of a digital echo canceller with digital domain echo cancellation. .................................................. 43
2.7 Block diagram of linear echo canceller with 2B1Q input reference. .. 44
2.8 Block diagram of memory-based echo canceller. ..................... 49
2.9 Block diagram of multi-memory echo canceller. ..................... 53
2.10 Data echo canceller with symbol reference proposed by Mueller. ..... 55
2.11 Implementation of Adaptive Reference Echo Cancellation. ........ 56
2.12 Kanemasa proposed echo canceller block diagram. ................. 57
2.13 Configuration of three-stage echo cancellation incorporating DA as tail canceller. .................................................. 60
2.14 Block diagram of two-stage echo cancellation. ........................................... 61
2.15 Configuration of echo tail canceller using IIR filter with variable gain. .......... 63
2.16 Block diagram of parallel IIR filter to suppress echo tail. .............................. 64
2.17 Block diagram of the \((1 - D)\) filter. .............................................................. 65

3.1 IFIR filter structure with \(L=4\). ................................................................. 69
3.2 Block diagram of Interpolated FIR (IFIR) filter. ........................................... 70
3.3 An example to illustrate frequency response of model, sparse, interpolater, and IFIR filter. \(L = 15\). ................................................................. 73
3.4 A view of the response of the sparse filter and its first spectral image central at \(2\pi/L\). ................................................................. 74
3.5 Direct-form implementation of the RRS. ......................................................... 76
3.6 A plot of the number of taps versus \(L\) for the sparse filter \(F(Z^L)\) and the interpolator \(G(Z)\). ................................................................. 77
3.7 A view of the response of the interpolator filter and its first stopband centered at \(1/L\). ................................................................. 79
3.8 Magnitude responses of Example 1. (a) Conventional. (b) Model. (c) Interpolator. (d) IFIR. ................................................................. 84
3.9 Magnitude responses of Example 2. (a) Conventional. (b) Model. (c) Interpolator. (d) IFIR. ................................................................. 86

4.1 Block Diagram of Adaptive IFIR Filter. ......................................................... 90
4.2 Block diagram of two-stage echo cancellation incorporating AIFIR filter as a tail canceller. ................................................................. 97

5.1 Echo path impulse response of Loop\#2. ....................................................... 106
5.2 Echo path impulse response of Loop\#6. ....................................................... 107
5.3 Echo path impulse response of Loop\#9. ....................................................... 108
5.4 An exploded view of Loop\#2 tail frequency response. ................................... 112
5.5 An exploded view of Loop#5 tail frequency response. 113
5.6 An exploded view of Loop#6 tail frequency response. 114
5.7 An exploded view of Loop#7 tail frequency response. 115
5.8 An exploded view of Loop#9 tail frequency response. 116
5.9 An exploded view of Loop#11 tail frequency response. 117
5.10 An exploded view of Loop#12 tail frequency response. 118
5.11 An exploded view of Loop#13 tail frequency response. 119
5.12 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#2. 124
5.13 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#5. 125
5.14 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#6. 126
5.15 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#7. 127
5.16 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#9. 128
5.17 A graph to illustrate the effects of Main Echo Canceller length \(N_M\) on performance. The same \(G(Z)\) is used for each value of \(N_M\). As \(N_M\) increases echo cancellation performance improves: Loop#11. 129
5.18 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#12. .... 130

5.19 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#13. .... 131

5.20 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#2. For each passband edge, a specific $G(Z)$ is designed. ................. 135

5.21 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#5. For each passband edge, a specific $G(Z)$ is designed. ................. 136

5.22 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#6. For each passband edge, a specific $G(Z)$ is designed. ................. 137

5.23 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#7. For each passband edge, a specific $G(Z)$ is designed. ................. 138

5.24 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#9. For each passband edge, a specific $G(Z)$ is designed. ................. 139

5.25 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#11. For each passband edge, a specific $G(Z)$ is designed. ................. 140

5.26 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#12. For each passband edge, a specific $G(Z)$ is designed. ................. 141

9
5.27 A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#13. For each passband edge, a specific $G(Z)$ is designed. ........................................... 142

5.28 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#2. ........................................... 144

5.29 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#5. ........................................... 145

5.30 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#6. ........................................... 146

5.31 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#7. ........................................... 147

5.32 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#9. ........................................... 148

5.33 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#11. ......................................... 149

5.34 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#12. ......................................... 150

5.35 A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#13. ......................................... 151

5.36 An exploded view of Loop#6 simulated tail frequency response. ... ........................................... 155

5.37 A view of the wanted passband of the sparse filter $F(Z^L)$ (Loop#6) and its spectral images (only showing seven of these images). All spectral images must be attenuated by the interpolator $G(Z)$. ........................................... 155

5.38 An exploded view of the compromise interpolator. ........................................... 156

5.39 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#2. For each $\delta_s$, a specific $G(Z)$ is designed. ........................................... 158
5.40 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#5. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 159

5.41 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband edge $\delta_s$: Loop#6. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 160

5.42 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#7. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 161

5.43 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#9. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 162

5.44 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#11. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 163

5.45 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#12. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 164

5.46 A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#13. For each $\delta_s$, a specific $G(Z)$ is designed. .............................................. 165

A.1 Bellcore loops: Loop#1 to Loop#5. ................................. 176

A.2 Bellcore loops: Loop#6 to Loop#10. ................................. 177

A.3 Bellcore loops: Loop#11 to Loop#15. ................................. 178

A.4 Model of a hybrid connected with subscriber line. .................. 180

B.1 Compromise interpolator characteristics and magnitude response. 187
B.2 Convergence curves of the two-stage echo canceller: MSE vs Baud periods. ......................................................... 189
List of Tables

2.1 Codes Considered for BRA-DSL. .......................... 36
2.2 Measured NEXT Limited Range in km of 0.5 mm copper twisted pair. 36
2.3 2B1Q Conversion Table. .................. 36

3.1 FIR vs IFIR, a complexity comparison. .................. 80
3.2 Hardware details for the filters designed in Example 1. ..... 83
3.3 Hardware details for the filters designed in Example 2. ..... 85

4.1 AFIR vs AIFIR, a complexity comparision. .................. 96

5.1 The first selection of passband frequencies $f_p$. ........ 111
5.2 The passband and stopband widths of Loop#2 interpolator, $L = 15$, sampling frequency normalized to 1 Hz. ........ 121
5.3 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#2. The main canceller multiplier size is $3 \times r$. The $\textit{COMPLEXITY}_AIFIR$ of the tail echo canceller is $45 \times r \times r$ multipliers. ........ 121
5.4 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#5. The main canceller multiplier size is $3 \times r$. The $\textit{COMPLEXITY}_AIFIR$ of the tail echo canceller is $45 \times r \times r$ multipliers. ........ 125
5.5 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#6. The main canceller multiplier size is $3 \times r$. The $\textit{COMPLEXITY}_AIFIR$ of the tail echo canceller is $45 \times r \times r$ multipliers. ........ 126
5.6 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#7.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $45 \ r \times r$ multipliers. .......... 127

5.7 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#9.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $45 \ r \times r$ multipliers. .......... 128

5.8 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#11.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $45 \ r \times r$ multipliers. .......... 129

5.9 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#12.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $45 \ r \times r$ multipliers. .......... 130

5.10 Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#13.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $45 \ r \times r$ multipliers. .......... 131

5.11 Echo Cancellation Performance (ECP) vs $f_p$: Loop#2. .......... 135

5.12 Echo Cancellation Performance (ECP) vs $f_p$: Loop#5. .......... 136

5.13 Echo Cancellation Performance (ECP) vs $f_p$: Loop#6. .......... 137

5.14 Echo Cancellation Performance (ECP) vs $f_p$: Loop#7. .......... 138

5.15 Echo Cancellation Performance (ECP) vs $f_p$: Loop#9. .......... 139

5.16 Echo Cancellation Performance (ECP) vs $f_p$: Loop#11. .......... 140

5.17 Echo Cancellation Performance (ECP) vs $f_p$: Loop#12. .......... 141

5.18 Echo Cancellation Performance (ECP) vs $f_p$: Loop#13. .......... 142

5.19 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#2.
The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$
of the tail echo canceller is $35 \ r \times r$ multipliers. .......... 144
5.20 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#5.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 145

5.21 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#6.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 146

5.22 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#7.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 147

5.23 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#9.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $34\,r \times r$ multipliers. .............................................. 148

5.24 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#11.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 149

5.25 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#12.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 150

5.26 Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#13.

The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35\,r \times r$ multipliers. .............................................. 151

5.27 Echo The effects of using a compromise interpolator in the tail canceller on the echo cancellation performance (ECP): various Bellcore lines are considered .................................................. 153

5.28 Important specifications of the proposed two-stage echo canceller. .............................................. 154

5.29 Echo Cancellation Performance (ECP) vs $\delta_5$: Loop#2. .............................................. 158

5.30 Echo Cancellation Performance (ECP) vs $\delta_5$: Loop#5. .............................................. 159
5.31 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#6. .......... 160
5.32 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#7. .......... 161
5.33 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#9. .......... 162
5.34 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#11. ....... 163
5.35 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#12. ....... 164
5.36 Echo Cancellation Performance (ECP) vs $\delta_2$: Loop#13. ....... 165

A.1 Cable characteristics for 19 AWG PIC line @ 70° F. .......... 182
A.2 Cable characteristics for 22 AWG PIC line @ 70° F. .......... 183
A.3 Cable characteristics for 24 AWG PIC line @ 70° F. .......... 184
A.4 Cable characteristics for 26 AWG PIC line @ 70° F. .......... 185

B.1 The passband and stopband widths of compromise interpolator. ... 186
## Acronyms and Definitions

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISDN</td>
<td>Integrated services Digital Network</td>
</tr>
<tr>
<td>BRA</td>
<td>Basic Rate Access</td>
</tr>
<tr>
<td>DSL</td>
<td>Digital Subscriber Loop</td>
</tr>
<tr>
<td>NT</td>
<td>Network Termination</td>
</tr>
<tr>
<td>LT</td>
<td>Line Termination</td>
</tr>
<tr>
<td>PAM</td>
<td>Pulse Amplitude Modulation</td>
</tr>
<tr>
<td>2B1Q</td>
<td>2 Binary 1 Quaternary</td>
</tr>
<tr>
<td>DFE</td>
<td>Decision Feedback Equalizer</td>
</tr>
<tr>
<td>EC</td>
<td>Echo Canceller</td>
</tr>
<tr>
<td>CCITT</td>
<td>Comité Consultatif International Télégraphique et Téléphonique</td>
</tr>
<tr>
<td>ISI</td>
<td>Intersymbol Interference</td>
</tr>
<tr>
<td>AFIR</td>
<td>Adaptive Finite Impulse Response</td>
</tr>
<tr>
<td>AIFIR</td>
<td>Adaptive Interpolated Finite Impulse Response</td>
</tr>
<tr>
<td>NEXT</td>
<td>Near End Cross Talk</td>
</tr>
<tr>
<td>FEXT</td>
<td>Far End Cross Talk</td>
</tr>
<tr>
<td>TCM</td>
<td>Time Compression Multiplexing</td>
</tr>
<tr>
<td>ECH</td>
<td>Echo Canceller Hybrid</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>DPLL</td>
<td>Digital Phase Locked Loop</td>
</tr>
<tr>
<td>RDS</td>
<td>Running Digital Sum</td>
</tr>
<tr>
<td>DA</td>
<td>Digital Accumulator</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Squared Error</td>
</tr>
<tr>
<td>SSE</td>
<td>Steady State Error</td>
</tr>
<tr>
<td>LMS</td>
<td>Least Mean Squares</td>
</tr>
<tr>
<td>AREC</td>
<td>Adaptive Reference Echo Canceller</td>
</tr>
<tr>
<td>HPRF</td>
<td>High Pass Receiving Filter</td>
</tr>
<tr>
<td>IIR</td>
<td>Infinite Impulse Response</td>
</tr>
<tr>
<td>RRS</td>
<td>Recursive Running Sum</td>
</tr>
<tr>
<td>ECSA</td>
<td>Exchange Carrier Standard Association</td>
</tr>
</tbody>
</table>
Nomenclature

\( x_n \)  Uncoded data symbols
\( \mathbf{X} \)  Near-end data vector
\( \mathbf{B} \)  Echo path vector, main pulse
\( \mathbf{T} \)  Echo path vector, tail pulse
\( \mathbf{C} \)  Main EC tap vector
\( \mathbf{H} \)  Tail EC taps vector
\( r_n \)  Received signal
\( d_n \)  Desired response
\( s_n \)  Far-end signal plus noise
\( \ell_n \)  Residual uncancelable echo
\( e_n \)  Error signal
\( \mathbf{U} \)  Tap error vector, main canceller
\( \mathbf{V} \)  Tap error vector, tail canceller
\( E \)  Expectation operator
\( \nabla_n \)  Gradient operator
\( \widehat{\nabla}_n \)  Gradient estimate
\( \sigma^2_x \)  Variance of \( x \)
\( \| \mathbf{X} \| \)  Euclidean norm of vector \( \mathbf{X} \)
\( \mu \)  Adaptation step-size
\[ n \quad \text{Time index} \]
\[ N \quad \text{Filter order} \]
\[ N_M \quad \text{Order of main EC} \]
\[ N_F \quad \text{Order of sparse filter} \]
\[ N_G \quad \text{Order of interpolator filter} \]
\[ G(Z) \quad \text{Interpolator filter transfer function} \]
\[ F(Z^L) \quad \text{Sparse filter transfer function} \]
\[ L \quad \text{Interpolation factor} \]
\[ \beta \quad \text{Number of non-zero taps in the } F(Z^L) \]
\[ j \quad \text{Number of taps in } G(Z) \]
\[ F \quad \text{A vector of non-zero taps of } F(Z^L) \]
\[ G \quad \text{A vector of } G(Z) \text{ taps} \]
\[ H(Z) \quad \text{Interpolator filter transfer function} \]
\[ \tilde{X} \quad \text{Input data matrix} \]
\[ M \quad \text{The inner product vector} \]
\[ m_i \quad \text{An element of an inner product vector} \]
Chapter 1

Introduction

With the recent technological advances in digital transmission and switching in the telecommunications field, the evolution towards the Integrated Services Digital Network (ISDN) is well under way. The ISDN will provide end-to-end digital connectivity, along with increased capability, flexibility and reliability of services to the telecommunications user.

In order for Basic Rate Access (BRA), the initial phase of the ISDN, to be successful, the efficient and economical implementation of the digital subscriber loop¹ (DSL) on existing metallic, two-wire transmission media is of basic importance. Further to this, the successful deployment of the BRA-ISDN DSL will depend on the cost-effective implementation of its components, the principal one being the ISDN U-interface transceiver VLSI chip.

There are several interfaces that provide the ISDN basic rate access at the physical layer. The physical interfaces for the basic rate access are shown in Figure 1.1, [1]. Of these, the two most important are the S-interface and the U-interface. The S-interface is for transmission within the customer premises (residential, business, or from a private branch exchange to the telephone instrument).

The S-interface uses separate wire pairs for the two directions of transmission between the network termination (NT) and terminal equipment (TE). The S-interface

¹The DSL is basically the transmission line between the subscriber and the central office, including the U-interface transceivers at each end.
line rate is 192 kb/s [2] (i.e. 144 kb/s plus 48 kb/s for framing). The U-interface is used for the connection of central office to the customer premises. The U-interface describes the full duplex data signal on the two-wire subscriber loop between central office line termination (LT) and customer premises network termination (NT).

ISDN basic access consists of a 2B+D format, with a total channel capacity of 144 kb/s. Each of the B channels has a capacity of 64 kb/s and will be used for voice/data (including graphics and video) transmission, while the 16 kb/s D channel is planned for signaling purposes. The U-interface line bit rate of the system is 160 kb/s, including a framing signal of 12 kb/s and a maintenance channel of 4 kb/s. The method chosen to obtain full duplex transmission at the U-interface is the echo cancellation technique, since it has greater range than Time Compression Multiplexing (TCM) due to narrow bandwidth, less cable attenuation, and reduced crosstalk coupling. Figure 1.2 shows the echo cancellation method. The proposed U-interface line code is 2B1Q (2 binary, 1 quaternary). This is a 4-level pulse amplitude modulation (PAM) code with zero redundancy [3]. Thus the proposed symbol rate at the U-interface is 80 kbaud. A block diagram of a typical echo cancellation transceiver\textsuperscript{2}

\textsuperscript{2}e.g., MC145472, a single chip implementation of an ISDN U-transceiver [4].
is shown in Figure 1.3. The transmitted data is scrambled to ensure sufficient timing information and also to ensure that data signals in the two directions are uncorrelated (correlation would bias the echo canceller adaptation). The 2B1Q symbols are formed in the encoder block and passed as symbols to the D/A converter. The transmit filter converts two bit 2B1Q symbols into filtered analog voltage levels to drive the DSL via a coupling circuit. In the receive direction, the receive filter prevents noise aliasing in the subsequent sampling. An echo canceller is used to cancel the echo of the local transmitter. A Decision Feedback Equalizer (DFE) [5] is used to cancel the far-end data intersymbol interference. A timing recovery circuit recovers the proper phase/frequency of the receive sampling in relation to the received data symbols. The most critical issues in the design of the transceiver are Equalization, Timing Recovery, and Echo Cancellation.

Figure 1.2: Block diagram of echo cancellation technique.

3The scrambler will break up long sequences of zeros which have no timing information.
Figure 1.3: Block Diagram of DSP Transceiver Architecture.

SC = Scrambler
EN = 2B1Q Encoder
TXF = Transmit Filter
AAF = Anti-Aliasing Filter
S&C = Sampling & A/D
LEQ = Linear Equalizer
SLR = Slicer
DSC = Descrambler
• Equalization Techniques:

Dispersion caused by the $\sqrt{f}$ attenuation and bridged taps of the DSL makes adaptive equalization necessary. One of the drawbacks of linear equalization is the noise enhancement when there are spectral nulls caused by the bridged taps. Adaptive equalization with reduced complexity and with less noise enhancement can be achieved by using a DFE. Based on the past decisions on the transmitted data, a DFE generates the estimated ISI and subtracts it from the received signal. Since the estimation of the ISI is based on past decisions, the DFE is effective only in eliminating post-cursor ISI and cannot reduce the precursor ISI. Unless special care is taken [6], [7] a linear equalizer is necessary in conjunction with the decision feedback equalizer to remove both the precursor and post-cursor ISI.

• Timing Recovery Issues:

Timing recovery involves recovering the frequency of the data signals transmitted from the other end and determining the sampling phase. The sampling phase is important because most equalizers (linear equalizer and DFE, for example) are sensitive to the sampling phase. The timing jitter, random fluctuation in the recovered clock, must be minimized since it affects the accuracy of the echo cancellation [8], [9]. Only discrete-time timing recovery techniques are considered suitable for the DSL application due to the requirement of economic integrated circuit implementation. Since timing recovery is performed after echo cancellation, and the hardware complexity of the echo canceller increases linearly with sampling rate, the sampling rate must be as low as possible in order to limit the complexity of the echo canceller.

• Echo Canceller Design Problems:

1. a high degree of accuracy is required.
2. A long echo pulse tail necessitates many tap coefficients.

3. Timing jitter produces a time-varying echo response and imperfect cancellation.


5. Adaptation must be continuous, in the presence of far-end data.

1.1 Problem Statement

The echo canceller is one of the most important blocks in the transceiver of Figure 1.3. Echoes result from imperfections in the hybrid transformer\(^4\) of the subscriber loop: impedance mismatches, changes in wire gauges at junctions which of which cause reflections, and bridged taps in multiple subscriber loop configurations which can produce reflections. Figure 1.4 illustrates a simplified model of a hybrid transformer. If \(Z_{bat}\) is matched to the channel impedance \(Z_{line}\), the receiver will be isolated from the transmitter and no echo is generated (i.e., the hybrid will provide infinite loss; Transhybrid Loss = \(\log \frac{(Z_{bat} + Z_s)(Z_{line} + Z_{tran})}{Z_{bat}Z_s - Z_{line}Z_{tran}}\)). Since this complete isolation of the receiver from the transmitter requires that \(Z_{bat}\) to be matched to the channel impedance \(Z_{line}\) at all frequencies, it is obviously not achievable and some echo will always exist. Since, the hybrid transformer does not pass d.c., we end up with a main echo impulse response, large in magnitude for short duration followed by a tail echo response of smaller magnitude and longer duration. The echo impulse response is also affected by the loop length, line gauges, and bridged taps.\(^5\) The overall response can be classified into three different regions:

- A first region in which the echo response has large magnitude and shape widely varying from loop to loop.

- A second region in which the echo response has a pseudo exponential nature.

---

\(^4\)A detailed guide of transformer circuits and parameters can be found in [111].

\(^5\)Appendix A contains Bellcore loop configurations.
• A third region in which the echo response can be approximated as an exponential function (tail section).

In this thesis, an adaptive transversal FIR filter representation for the first and second regions is used. The tail region is then modelled as a narrow-band low-pass response. In order to achieve adequate echo cancellation performance, the echo tail response must be dealt with efficiently. The objective of this thesis is to propose an efficient echo tail canceller. In Appendix A, we derive the model for the hybrid circuit with the subscriber line.

![Diagram of a hybrid transformer](image)

Figure 1.4: Simplified diagram of a hybrid transformer.

There has been a number of published designs that deal with the cancellation of the echo tail response (see section 2.4 for details). In current U-transceivers, a popular method is to simulate the echo tail of the echo path response with a fixed pole first-order IIR (infinite impulse response) filter or filters, with adaptive gain [33], [80]-[83],
[108]. The value of the fixed pole is calculated from the d.c. cutoff characteristics of the hybrid. Due to the fact that the echo tail response decreases very slowly, an IIR filter with large time constant (pole near the unit circle) is required. Thus, this filter requires high precision implementation. In [33], [80]-[83], high-pass receiving filter was used to shorten the echo tail. This filter does allow more of the effects of noise and crosstalk. Consequently, the performance of the forward equalizer is compromised. In [82], the use of (1 – D) filter caused at least 3 dB degradation in the overall SNR of the transceiver. In [109], an IIR filter with adaptive pole(s) has been used to approximate the echo tail response. However, complicated adaptation algorithms are required to maintain stability during adaptation. In all of the above methods, a compromise balancing network is needed to provide at least 15 dB of echo power reduction.

With our approach, the need for a precise compromise network and high-pass receiving filter is eliminated. The proposed tail canceller is an adaptive version of the interpolated FIR filter (IFIR) introduced in [10]. It is based on providing a long impulse response using fewer coefficients and making use of the special properties of narrow-band frequency response filters. The computational efficiency and performance of the adaptive Interpolated FIR (AIFIR) filter [10] will be shown. Trade-offs between complexity and performance are emphasized. A simple block diagram of the proposed system is illustrated in Figure 1.5. A detailed system block diagram is given in chapter 4.

1.2 Thesis Organization

This thesis is organized in 6 chapters. Chapter 2 reviews transmission issues for ISDN Basic Rate Access, echo canceller, and echo tail suppression techniques. Chapter 3 presents the theory of the Interpolated FIR (IFIR) filters. Chapter 4 proposes the adaptive Interpolated FIR (AIFIR) echo tail canceller. Chapter 5 presents the simu-
Figure 1.5: Block diagram of the proposed two-stage echo canceller.

...lation results of the proposed system. Finally, the various conclusions and extensions for further work are presented in chapter 6.

1.3 Conclusion

In this chapter, we reviewed the ISDN Basic Rate Access. A great advantage of ISDN is that a variety of services (voice, data, or image) can be digitized or processed locally and transmitted on the same digital loop, the DSL. The concept and overview of DSL transmission were introduced. The S-interface is for transmission within the subscriber premises. The S-interface line bit rate is 192 kb/s. The U-interface connects the serving center line termination (LT) to the customer premises network termination (NT). The U-interface line bit rate is 160 kb/s and symbol rate is 80 kbaud. The transmission scheme that has been chosen to achieve full-duplex at the U-interface is the echo cancellation method. Moreover, the chosen U-interface line code is 2B1Q.
Issues and challenges affect the design of the U-transceiver were briefly outlined. Of these, 2B1Q signal is rich in low frequency content. That leads to long echo tail. Therefore, without any special solution for the echo tail pulse, a large echo canceller is required. In this thesis, we propose a computationally efficient tail echo canceller employing AIFIR filter to reduce the complexity of the overall echo canceller. The performance of the proposed canceller is investigated in chapter 5. Satisfactory echo cancellation performance was achieved by this new structure.
Chapter 2

Review of Transmission Issues for ISDN Basic Rate Access

It is the intent of this chapter to summarize the issues concerning the Basic Rate Access-ISDN (BRA-ISDN) and echo cancellation. The proposed U-interface bit rate is 160 kb/s. The U-interface transceiver design is complicated due to crosstalk, ISl, echo, and impulse noise. In this chapter, we begin by presenting a brief overview of the various DSL impairments that are present on the loop plant. The second issue to be discussed is the ISDN basic rate access transmission techniques (i.e., the selection of the transmission method and the line code). The technical reasons that led to the selection of the Echo Canceller Hybrid (ECH) transmission method and the 2B1Q line code are outlined in section 2.2. A comparison of various line codes proposed for BRA-ISDN is given in subsection 2.2.1. Special emphasis is placed on 2B1Q line code characteristics and the implementation design challenges it imposes on the transceiver system design. In section 2.3, a review of the principles of echo cancellation is given. This review includes a mathematical treatment of linear and nonlinear echo canceller, and recent schemes in echo cancellation are summarized. The analysis of the linear echo canceller is utilized later in deriving the convergence behavior of the suggested AIFIR structure. In section 2.4, an overview of many techniques that deal with the cancellation of the echo tail is presented. Concluding remarks on this chapter are given in section 2.5.
2.1 Impairment Constraints of the DSL Transmission Medium

Because the overall transmission performance of the DSL is greatly affected by the environmental impairments of the physical loop plant, it would offer insight to the relevant DSL design issues by studying what the DSL impairments are, and what specific effects they have on the DSL performance. We will now describe the various environmental impairments that are present in the DSL.

Crosstalk

Crosstalk is mainly due to the capacitive coupling between DSL wire pairs (i.e., transmission lines). There are two types of crosstalk mechanisms: near-end crosstalk (NEXT) and far-end crosstalk (FEXT) (see Figure 2.1). Of all the loop plant impairments, NEXT is the problem which represents the most significant constraint on a DSL [11], [12]. NEXT is basically crosstalk between local transmitter to receiver, with a frequency-dependent transfer function of:

\[ |H_{NEXT}(f)| = K_{NEXT}|f|^{3/2} \]  

(2.1)

The constant \( K_{NEXT} \) has been empirically determined to be \( 0.88 \times 10^{-13} \) which represents 99% worst case NEXT loss of 57 dB at 80 kHz for 49 disturbing pairs [13], [14]. On the other hand, FEXT is regarded as crosstalk between a local transmitter and a distant receiver at the other end of the DSL. FEXT has a frequency-dependent transfer function of:

\[ |H_{FEXT}(f)| = K_{FEXT}|C(f)|^2|f|^2 \]  

(2.2)

Where \( C(f) \) is the transfer function of the subscriber loop. The constant \( K_{FEXT} \) has been empirically determined to be \( 2.60 \times 10^{-16} \) which represents 99% worst case FEXT loss of 33 dB at 3.15 MHz for 49 disturbing pairs [13], [14].
Since both forms of crosstalk increase with frequency, it is desirable to minimize the transmission bandwidth of the data signal. This can be achieved by appropriate low-pass filtering or by the choice of a line code with small bandwidth [15], [16].

![Diagram showing NEXT and FEXT Crosstalk on the DSL](image)

**Figure 2.1: NEXT and FEXT Crosstalk on the DSL.**

**Impulse Noise**

This impairment, consisting of infrequent, high energy noise bursts, is due mainly to switching transients from relays of electromechanical switches in the central office and/or by lighting. Impulse noise varies substantially in frequency of occurrence, intensity, and characteristics from location to location [17]. The energy in many impulses is concentrated in a band below 50 kHz, but this is not always the case. Impulse noise peaks in the range 10 mV to 40 mV were found to occur as often as every 6 seconds on some central office loop terminations and two to three times a minute at the subscriber premises. Mean durations of impulses ranged from about 30 $\mu$s to about 100 $\mu$s [18].

**Intersymbol Interference (ISI)**

This impairment is due to transmitted data pulse dispersion over the band-limited DSL channel. From the Nyquist Criterion for zero ISI, the required minimum band-
width on the data signal is equal to half the baud rate. However, it would be desirable to have excess bandwidth, since this yields faster decaying signal impulse response with reduced ISI, and a wider horizontal ‘eye’ opening. This corresponds to an increased immunity to timing jitter and, thus, easier timing recovery at the receiver [19], [20]. On the other hand, a larger bandwidth does allow more NEXT contamination of the digital signal. Recall that NEXT, one of the significant impairments, has a power spectrum increasing rapidly as frequency increases. It is anticipated that post-cursor ISI will be essentially canceled out by a DFE.

2.2 ISDN Basic Access Transmission Techniques

At present, much current research is showing that the key to the success of BRA-ISDN will be the economical deployment of the DSL on the two-wire loop plant. Since it will be a few years before the metallic two-wire medium is fully replaced with fiber optic cables [21], the present VLSI technology and digital transmission techniques (e.g., line codes) will have to accommodate BRA-ISDN data rates of up to 160 kb/s on the existing two-wire loop environment. At such high BRA-ISDN data rates, the two-wire medium (originally intended only for voice transmission) imposes severe constraints on the design of the DSL (e.g., selection of transmission technique and line code) and its components (e.g., U-interface transceiver chip). Next, we will give a comparison of the various line codes that were proposed for DSL transmission.

2.2.1 Comparison of Line Codes for DSL

The choice of a line code for the DSL is of paramount importance because the properties of the line code affect the complexity and performance of the other DSL components. The requirement of a small concentration of low-frequency power in the line code is because such long echo tails require more complex echo cancellation. On the other hand, the requirement to avoid high-frequency content in a line code is
due to the fact that a signal is attenuated to a greater extent at higher frequencies, which, consequently, would require more complex equalization. In addition, a small line code bandwidth is desirable in minimizing the signal susceptibility to crosstalk noise. In the formulation of the U-interface standards by the ECSA/T1D1.3\(^1\), many line codes and variations of line codes were studied\(^2\). This subsection is a review of the attributes\(^3\) of line code characteristics that led to the selection of the 2B1Q line code as the North American Standard. 2B1Q line code was determined to have the best performance/complexity tradeoff of the proposed line codes. NEXT is the crucial factor used by the ESCA in selecting the 2B1Q line code as the North American Standard [34]. The most important line codes considered were (see Table 2.1)

- **AMI**
  
  Alternate Mark Inversion (AMI), transmits the data zero with no pulse (i.e., no encoding) and a data one with alternating positive and negative pluses. Because of this pulse alternation, the AMI code is d.c. balanced (i.e., it contains a zero at d.c.). Its baud rate is equal to its information rate. Simulation studies of the NEXT performance of the AMI code showed that AMI code falls about 2 dB below MDB [34].

- **MDB**
  
  In Modified Doubinary (MDB),\(^4\) every information bit is represented by two pulses of opposite polarity with a zero pulse in the baud interval between them. Simulation studies of the NEXT performance of the MDB code indicated that its NEXT performance was 2 to 3 dB poorer than 3B2T or 4B3T [34].

---

\(^1\) Exchange Carrier Standards Association/Committee T1D1.3 is the organization responsible for setting the US standard for the U-interface.

\(^2\) The codes were compared using computer simulation studies to measure their performance [34].

\(^3\) The baud rate is the fundamental attribute of a line code, since it determines its crosstalk and ISI performance.

\(^4\) Also known as Class IV Partial Response.
• 3B2T

This line code has lower baud rate compression than 2B1Q line code since it utilizes three-level signals for transmission instead of four-level. Another factor that lower the baud rate compression slightly is that the code maps the eight possible combinations of three bits of information into the nine possible combinations of two ternary symbols. Simulation studies indicated that 3B2T had NEXT performance that was close to that of 4B3T [34].

• 4B3T

This code converts blocks of 4 bits (16 signal levels) into words of 3 ternary digits (27 signal levels). This ternary line code uses three code books to produce desirable transmission properties in the transmitted signal, such as no d.c. component in the transmitted signal spectrum and a reduction in the baud rate. This reduction in baud rate is desirable for increasing crosstalk noise immunity. One of the books contains code words that are biased toward positive polarity, one contains words that are negatively biased, and a third is neutral. Furthermore, the Running Digital Sum (RDS), determines which code book is used. Excessively positive RDS selects the negatively biased code book, negative RDS decides to use the positively biased code book, and neutral RDS selects the neutral code book [73].

• 2B1Q

2B1Q line code offers the greatest baud rate reduction of any of the codes considered. It gave NEXT performance that was 2 to 3 dB better than 4B3T [34]. Consequently, the 2B1Q line code performs better than any of the other

---

5 A code book is a table that converts blocks of input bits to be encoded into blocks of transmitted symbol values.

6 Reducing the baud rate is equivalent to increasing the number of transmitted signal levels.

7 RDS is the algebraic sum of a sequence of k transmitted symbols.
line codes that were considered for application in BRA-DSL (see Table 2.2)\textsuperscript{8}.

<table>
<thead>
<tr>
<th>Code</th>
<th>Input</th>
<th>Output</th>
<th>Compression</th>
<th>Redundancy</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>2B1Q</td>
<td>2 bits</td>
<td>1 Quaternary</td>
<td>50%</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>3B2T</td>
<td>3 bits</td>
<td>2 Ternary</td>
<td>33%</td>
<td>1/9</td>
<td>00 unused</td>
</tr>
<tr>
<td>4B3T</td>
<td>4 bits</td>
<td>3 Ternary</td>
<td>25%</td>
<td>11/27</td>
<td>3 Code Books</td>
</tr>
<tr>
<td>AMI</td>
<td>1 bit</td>
<td>1 Ternary</td>
<td>0</td>
<td>1/3</td>
<td>Dicode</td>
</tr>
<tr>
<td>MDB</td>
<td>1 bit</td>
<td>1 Ternary</td>
<td>0</td>
<td>1/3</td>
<td>Class IV PR</td>
</tr>
</tbody>
</table>

Table 2.1: Codes Considered for BRA-DSL.

<table>
<thead>
<tr>
<th>Line Code</th>
<th>NEXT Limited Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMI</td>
<td>4.4 km</td>
</tr>
<tr>
<td>MDB</td>
<td>5.3 km</td>
</tr>
<tr>
<td>3B2T</td>
<td>6.4 km</td>
</tr>
<tr>
<td>2B1Q</td>
<td>7.2 km</td>
</tr>
</tbody>
</table>

Table 2.2: Measured NEXT Limited Range in km of 0.5 mm copper twisted pair.

2B1Q’s high performance\textsuperscript{9} (i.e., longer range, see Table 2.2) and relatively low complexity makes it a suitable line code for DSL applications in ISDN. It is a 4-level PAM code with zero redundancy. This code converts blocks of 2 bits\textsuperscript{10} into one quaternary symbol as given in Table 2.3. As a result, the line baud rate is half of the bit rate.

<table>
<thead>
<tr>
<th>Transmitted Bits</th>
<th>Quaternary Symbols</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>+3</td>
</tr>
<tr>
<td>11</td>
<td>+1</td>
</tr>
<tr>
<td>01</td>
<td>-1</td>
</tr>
<tr>
<td>00</td>
<td>-3</td>
</tr>
</tbody>
</table>

Table 2.3: 2B1Q Conversion Table.

\textsuperscript{8}Next Limited Range is defined as the longest twisted pair length that can be used for a given error rate [34].

\textsuperscript{9}Recent work has indicated that in a NEXT-limited environment five levels is optimum, and four levels are almost as good [32].

\textsuperscript{10}The first bit is called the "sign bit". The second bit is called the "amplitude bit".
Although the efficient transmission capability of the 2B1Q transmission system has been proven (e.g., 2B1Q provides good performance in the presence of crosstalk noise [32]), there are many implementation problems to be solved. The high low-frequency content of the 2B1Q line code presents unique problems in the design of a transceiver. The most significant issues are discussed next.

- **Echo Tail**

  The echo tail length has a direct effect on the complexity of the echo canceller. One drawback of low-frequency transmission system (i.e., transformer) is that the pulses on the input tend to develop long echo tails. Several consecutive pulses will tend to have effect on their neighbors, resulting in echo response with long tails. Mathematically, a simple transformer can be represented by a high pass transfer function \( T(z) = \frac{1-z^{-1}}{1-\lambda z^{-1}} \), where \( \lambda \approx 1 \). The response of a simple transformer to a rectangular pulse or unit step (see Figure 2.2) explains this echo tail phenomenon. The transformer can’t sustain a d.c. voltage, therefore the positive pulse is followed by a negative decaying pulse. The magnitude of the decaying pulse varies inversely with its duration. If the transformer’s time constant \( \tau \) is much longer than the pulse duration \( T \), then the magnitude of the decaying pulse equals

  \[
  1 - e^{-T/\tau} \approx T/\tau \tag{2.3}
  \]

  Thus, if there is a long period of similar symbols\(^{11}\) (i.e., single d.c. level), their echo tail contributions add to produce a substantial total effect. Minimizing the complexity of the echo canceller is a priority. Echo tail cancellation techniques are discussed in section 2.3. In chapter 4, we propose an efficient echo tail

\(^{11}\)A long period of 2B1Q similar symbols can be considered as a unit step, which yields a decaying exponential \( e^{-t/\tau} \)
canceller to reduce the complexity of the two-stage echo canceller. The performance of the two-stage canceller incorporating an AIFIR filter is investigated in chapter 5.

- **DC Restoration**

  The 2B1Q signal has no redundancy and can have extended periods of a single d.c. level. Currently, subscriber loop coupling is done using transformers which are low-frequency transmission system. Thus, DC restoration in the receiver is a necessity. A prolonged time-length\textsuperscript{12} DFE is used also for DC restoration.

- **Transmit Level Accuracy**

  In practice, the levels of the 2B1Q symbols will deviate from the ideal values (that causes nonlinearities). This is due to the limited component precision of the 2B1Q generation circuit. In [33], it is suggested that if highly accurate level control is not possible, the DSP based transceiver must use a high precision D/A converter in the transmitter.

- **Nonlinearities**

  Nonlinear effects in the echo path can severely degrade the performance of the echo canceller. As an example of the kinds of nonlinearity appearing in the system, consider the DSP transceiver shown in Figure 1.3. The following sources of nonlinear distortion can be identified.

  1. *Saturation in the Transformers*

     This leads to a nonlinearity which can be controlled by the choice of a bulkier transformer with gapped core and lower inductance. The low-frequency content of the 2B1Q signal imposes limitations on the design of the loop coupling transformer. It appears that it has been accepted

\textsuperscript{12}This is obtained by using a tail suppression scheme, e.g., [85].
Response to rectangular pulse

Response to unit step

Figure 2.2: Low frequency cut-off and echo tail of a hybrid transformer.
by the industry, that with 2B1Q the echo canceller must have nonlinear compensation capability. Nonlinear capability can be added in a variety of ways (e.g., nonlinear compensation over a limited number of taps). A fully nonlinear echo canceller will be beyond the limits of semiconductor technology for some time to come [32]. Thus there is still a need for a carefully designed analog part.

2. Line Driver.

Transmitted pulse asymmetry caused by the line driver and the amplifiers in the line driver are a source of nonlinearity (i.e., saturation and clipping of the analog devices).

3. Nonlinearity of Data Converters.

The D/A and A/D converters introduce the most important source of nonlinearity in the DSL transceiver.

2.2.2 Time Compression Multiplexing vs Echo Canceller Hybrid

A fundamental choice in DSL technology is the transmission method. There are two prevailing techniques: Time Compression Multiplexing (TCM) [22]-[26]. and Echo Canceller Hybrid (ECH)\textsuperscript{14} [24], [1], [27]-[31]. TCM allows only one-way data flow at a time (Figure 2.3); thus, its line baud rate is at least twice its source baud rate. However, an advantage of TCM is that it does not require any echo cancellation, since data flow in only one direction at a time. This, of course, would reduce the complexity of the DSL transceiver design. On the other hand, ECH allows full-duplex transmission (i.e., 2-way, simultaneous data flow), and, consequently, requires only half the transmission bandwidth of TCM. However, ECH does require echo cancellation and the associated increase in complexity. Next, we will discuss the

\textsuperscript{13}Nonlinearities are reduced by careful design of pulse generation circuit and transformers.

\textsuperscript{14}May, 1985, ECH was chosen as the transmission method for North America.
fundamentals of echo cancellation.

Figure 2.3: Time compression multiplexing method of full-duplex data transmission.

2.3 Echo Canceller

Echo cancellers have received a great deal of attention in the last few years as a viable technique in ISDN applications [35]-[42]. The echo canceller (EC), shown in Figure 2.4, ideally emulates the echo path, thus allowing a replica of the interfering echo to be subtracted from (cancel) the returned echo.

Figure 2.4: Classic configuration of echo canceller.
A received signal to echo noise ratio (SNR) at 20dB or more [43] is required. In practice less than 10 dB transhybrid loss can be expected through the hybrid with a single compromise balance network\(^{15}\). The subscriber loop may attenuate the far-end signal by as much as 40-50 dB. Therefore, a near-end echo signal may be 30-10 dB higher than a far-end signal. The echo canceller may be required to suppress the near-end echo signal by as much as 50-60 dB. The cancellation can be done in the analog domain (Figure 2.5) by converting the echo canceller output to analog, or in the digital domain (Figure 2.6) by converting the hybrid output to digital prior to cancellation. Several alternative realizations of echo cancellers are discussed in [44]:

1. an analog implementation of an echo canceller.

2. a fully digital echo canceller with digital cancellation and A/D conversion of the input signal.

3. a digital echo canceller with analog cancellation (sum in analog).

4. analog-digital echo canceller with analog cancellation.

Two different techniques of implementing the echo canceller that have gained wide acceptance are the AFIR filter structure (linear echo canceller) [45] and the memory-based or RAM structure (nonlinear echo canceller) [54]. The following subsection will provide a discussion of those techniques. The advantages and disadvantages of each method will be given. Derivations necessary for clarification are included.

### 2.3.1 Linear Echo Canceller

A linear echo canceller can be implemented with an adaptive FIR filter. AFIR filter-based echo cancellers have the advantage of faster coefficient adaptation and only a linear increase in complexity with increased filter length. Next, we will give an analytical treatment of the linear echo canceller.

\(^{15}\)Balance networks are generally resistor capacitor combinations designed to provide some balance over the transmission bandwidth.
Figure 2.5: Block diagram of a digital echo canceller with analog domain echo cancellation.

Figure 2.6: Block diagram of a digital echo canceller with digital domain echo cancellation.
Mathematical Formulation of the Linear Echo Canceller

An echo canceller intended for 2B1Q data transmission is depicted in Figure 2.7.

Figure 2.7: Block diagram of linear echo canceller with 2B1Q input reference.

In this subsection, we use the independence theory [46]-[48], [73] and assume uncorrelated data symbols to derive the mean squared error (MSE) of the echo canceller. The behavior of the mean and mean squared coefficient error vectors is derived. For facilitating the analysis of the echo canceller we introduce the following vectors:

1. the near-end data vector \( \mathbf{X} = [x_n \, x_{n-1} \, \ldots \, x_{n-N+1}]^T \)

2. the echo path vector \( \mathbf{B} = [b_0 \, b_1 \, \ldots \, b_{N-1} \, b_N \, \ldots]^T \)

3. the echo canceller coefficient vector \( \mathbf{C} = [c_{0,n} \, c_{1,n} \, \ldots \, c_{N-1,n}]^T \)

where \( c_{i,n} \) is the \( i^{th} \) coefficient at instant \( n \). Now, we can represent the received signal as

\[ e_n \]
\[ r_n = d_n + s_n + \ell_n \]  \hspace{1cm} (2.1)

where \( \ell_n = \sum_{k=N}^{\infty} b_k x_{n-k} \) is the residual uncancelable echo corresponding to echo delays that exceed the number of coefficients in the echo canceller. The far-end signal plus noise is \( s_n \). The \( d_n \) is the echo part that is emulated by the echo canceller and expressed as

\[ d_n = \sum_{k=0}^{N-1} b_k x_{n-k} = B^T X \]  \hspace{1cm} (2.5)

The output of the echo canceller is

\[ y_n = \sum_{k=0}^{N-1} c_k x_{n-k} = C^T X \]  \hspace{1cm} (2.6)

Thus, the error signal \( e_n \) used for updating the canceller coefficient vector \( C \) is expressed through

\[ e_n = r_n - y_n \]  \hspace{1cm} (2.7)

\[ e_n = (B - C)^T X + s_n + \ell_n \]  \hspace{1cm} (2.8)

\[ e_n = U^T X + s_n + \ell_n \]  \hspace{1cm} (2.9)

where \( U = B - C \) is the coefficient error vector. At the beginning of the analysis, we state the following standard assumptions ([46]-[48], [73]) which will be used in the derivations here and chapter 4.

- Each input vector \( X \) is statistically independent of all previous input vectors.
  Also, \( X \) and \( s_n \) are zero-mean wide-sense stationary, mutually independent discrete stochastic processes.
• Each input vector $X$ is stochastically independent of all previous samples of the desired response $d_n$. Moreover, $X$ and $U$ are statistically independent.

The MSE is found by taking the expectation of $e_n^2$

$$E[e_n^2] = \sigma_x^2 E[\|U_n\|^2] + \sigma_s^2 + \sigma_t^2$$  \hfill (2.10)

The update equation, for the echo canceller's coefficients, and the coefficient error vector,

$$C_{n+1} = C_n + \mu e_n X$$  \hfill (2.11)

$$U_{n+1} = U_n - \mu e_n X$$  \hfill (2.12)

$$U_{n+1} = (I - \mu XX^T) U_n - \mu (s_n + \ell_n) X$$  \hfill (2.13)

The mean of the coefficient error vector can be derived by taking the expected value of Eq.(2.13)

$$E[U_{n+1}] = (1 - \mu \sigma_x^2) E[U_n]$$  \hfill (2.14)

$$E[U_n] = (1 - \mu \sigma_x^2)^n E[U_0]$$  \hfill (2.15)

Therefore, the mean coefficient error vector converges to zero as $n$ approaches infinity, provided that the following condition is satisfied [48]

$$0 < \mu < \frac{2}{\sigma_x^2}$$  \hfill (2.16)
We next consider the convergence behavior of the mean squared coefficient error vector. Taking the expected value of the squared norm values of both sides of Eq.(2.13), we get

\[ E[\| U_{n+1} \|^2] = (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)E[\| U_n \|^2] + \mu^2\sigma_x^2N(\sigma_x^2 + \sigma_i^2) \]  

(2.17)

\[ E[\| U_n \|^2] = (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)^nE[\| U_0 \|^2] + \mu^2\sigma_x^2N(\sigma_x^2 + \sigma_i^2) \]  

(2.18)

Eq.(2.18) fully describes the dynamic behavior of the echo canceller based upon the LMS algorithm. Eq.(2.18) can be solved as

\[ E[\| U_n \|^2] = (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)^nE[\| U_0 \|^2] + \mu^2\sigma_x^2N(\sigma_x^2 + \sigma_i^2) \sum_{i=0}^{n-1} (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)^i \]  

(2.19)

\[ E[\| U_n \|^2] = (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)^nE[\| U_0 \|^2] + \mu^2\sigma_x^2N(\sigma_x^2 + \sigma_i^2) \frac{1 - (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)^n}{1 - (1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4N)} \]  

(2.20)

The transient term will go to zero as \( n \) approaches infinity and the steady state error (SSE) value is

\[ SSE = \lim_{n \to \infty} E[\| U_n \|^2] = \frac{\mu_N}{2 - \mu\sigma_x^2N}(\sigma_x^2 + \sigma_i^2) \]  

(2.21)

The mean squared coefficient error vector converges to a steady state if and only if the step-size parameter \( \mu \) satisfies the condition

\[ 0 < \mu < \frac{2}{\sigma_x^2N} \]  

(2.22)

\(^{17}\)The squared norm, \( \| U_n \|^2 = \sum_k u_k^2 \).
Then, total MSE is

\[ MSE = \left( \frac{\mu \sigma_s^2 N}{2 - \mu \sigma_s^2 N} + 1 \right) (\sigma_s^2 + \sigma_i^2) \]  \hspace{1cm} (2.23)

The condition of Eq.(2.22) is the necessary and sufficient condition for the overall stability of the echo canceller. The steady state error and the speed of convergence decrease as the step-size parameter \( \mu \) decreases. The fastest convergence is obtained by selecting \( \mu = \frac{1}{\sigma_s^2 N} \). As a result, an MSE at \( 2(\sigma_s^2 + \sigma_i^2) \) is given. Lower MSE can be obtained at the cost of slower convergence by selecting smaller value of \( \mu \).

Eq.(2.19) shows the importance of the parameter \( \mu \), its value determines both the rate of convergence and the minimum residual echo that can be obtained.

### 2.3.2 Nonlinear Echo Canceller

For completeness, the principles of nonlinear echo cancellers are given here although they will not be used in this thesis. Transmitted pulse asymmetry\(^\text{18}\) and nonlinearity in the line driver\(^\text{19}\) will occasionally cause the transformer and data converters to exceed the tolerable echo path nonlinearity and compromise the echo canceller performance. In such cases, nonlinear echo cancellers have to be used. Two methods have been proposed [49]-[54] to cancel the nonlinear echo response. They differ in the way that the nonlinear function is expanded. The first method uses the volterra expansion and is reported in [51]. The second method [54] that cancels echoes of general nonlinear echo path is the memory-based (table look-up) echo cancellation. The replica of such an echo canceller can be expressed as

\[ e_n^*_0 = f(x_n, x_{n-1}, \ldots, x_{n-N+1}) \]  \hspace{1cm} (2.21)

where \( f(.) \) is a general nonlinear function.

\(^{18}\)A linear echo canceller would require that the uncompensated transmitted pulse asymmetry be kept below -60 dB [51].

\(^{19}\)It is a power amplifier which drives the transmit port of the hybrid.
Mathematical Formulation of Lookup Table Echo Canceller

To implement such a general nonlinear echo canceller, a memory of size $2^N$ words is used, where each memory location stores the echo replica of a particular data pattern. This is illustrated in Figure 2.8 where the last $N$ transmitted data symbols form the address to select the correct echo estimate. If any address is read at time $n$, its content is updated with a correction term, while the contents of the other addresses stay unchanged.

![Diagram](image)

Figure 2.8: Block diagram of memory-based echo canceller.

In this subsection, we will show that lookup table echo canceller has the same kind of convergence characteristics as the linear canceller of subsection 2.3.1 [28]. For facilitating the analysis, we consider Figure 2.7 employing a lookup table echo canceller shown in Figure 2.8. We assume that the echo path is nonlinear and represented by nonlinear function $f_i$ (Eq.(2.24)). The near-end binary data vector $\mathbf{X}$ is used as an address for the RAM at time $n$. The set $A$ of the $M = 2^N$ distinct address vectors
consists of all possible binary $N$-tuples $X_i$:

\[ A = \{X_0, X_1, \ldots, X_{M-1}\} \]  \hspace{1cm} (2.25)

Now, we denote the contents of address $X_i$ at time $n$ by $z_{i,n}$, $0 \leq i \leq M - 1$. The total contents of the RAM at time $n$ is represented by the vector $Z$:

\[ Z = [z_{0,n} \ z_{1,n} \ \ldots \ z_{M-1,n}]^T \]  \hspace{1cm} (2.26)

At time $n$, $X$ is equal to one of the elements of $A$. The corresponding index will be denoted by $q_n$, thus

\[ X = X_{q_n} \]  \hspace{1cm} (2.27)

and the output $\epsilon_n$ of the RAM echo canceller is

\[ \epsilon_n = z_{q_n,n} = RAM_{q_n,n} \]  \hspace{1cm} (2.28)

At time $n$ the content of the RAM at address $X_{q_n}$ is updated by adding a correction term $\mu \epsilon_n$ (other addresses are unchanged). Thus the update equation for the RAM echo canceller is

\[ z_{i,n+1} = \begin{cases} z_{i,n} & i = 0, 1, \ldots, M - 1, i \neq q_n \\ z_{i,n} + \mu \epsilon_n & i = q_n \end{cases} \]  \hspace{1cm} (2.29)

The error signal $\epsilon_n$ is defined as $(s_n + f_i - \epsilon_n)$. We invoke the independence assumptions between $s_n$ and $X$, and introduce the residual error signal as

\[ \psi_{i,n} = f_i - \epsilon_n \hspace{1cm} 0 \leq i \leq M - 1 \]  \hspace{1cm} (2.30)

where for each $i$, $f_i$ is an arbitrary number (i.e., the convolution expansion of nonlinear echo path and near-end data). With Eq.(2.30) we introduce the vector
\[ \Psi_{i,n} = [\psi_{0,n}, \psi_{1,n}, \ldots, \psi_{M-1,n}]^T \] (2.31)

This represents the contents of a RAM that consists of all residual error values which are updated as follows

\[ \psi_{i,n+1} = \begin{cases} \hat{\psi}_{i,n} & i = 0, 1, \ldots, M-1, \ i \neq q_n \\ \psi_{i,n}(1-\mu) - \mu s_n & i = q_n \end{cases} \] (2.32)

From Eq.(2.32), the MSE is found by taking \( E[\psi_{i,n}^2] \)

\[ E[\psi_{i,n+1}^2] = p(q_n \neq i)E[\psi_{i,n}^2] + p(q_n = i)\{E[\psi_{i,n}^2](1-\mu)^2 - 2\mu(1-\mu)E[\psi_{i,n}s_n] + \mu^2 E[s_n^2]\} \] (2.33)

\[ E[\psi_{i,n+1}^2] = (1 - p(q_n = i))E[\psi_{i,n}^2] + p(q_n = i)E[\psi_{i,n}^2](1 - \mu)^2 + \mu^2 \sigma_s^2 \] (2.34)

where \( p(q_n = i) \) denotes the probability that \( q_n = i \) (i.e., the probability that \( X = X_i \)). Then if all addresses \( X_i \) \((0 \leq i \leq M-1)\) are equiprobable we have

\[ p(q_n = i) = 2^{-N} \] (2.35)

then, Eq.(2.34) becomes

\[ E[\psi_{i,n+1}^2] = E[\psi_{i,n}^2]\{1 - \mu 2^{-N}(2 - \mu)\} + \mu^2 2^{-N} \sigma_s^2 \] (2.36)

Eq.(2.36) fully describes the dynamic behavior of the look up table echo canceller based upon the LMS algorithm, and it can be solved as

\[ E[\psi_{i,n}^2] = E[\psi_{i,0}^2]|\{1 - \mu 2^{-N}(2 - \mu)\}^n + \mu^2 2^{-N} \frac{1 - (1 - \mu 2^{-N}(2 - \mu))}{1 - (1 - \mu 2^{-N}(2 - \mu))} \sigma_s^2 \] (2.37)

\(^{20}\)Scrambling of transmitted data is mandatory in memory-based echo cancellers, as in AFIR echo cancellers, to ensure that all addresses are generated with equal probability.
In Eq.(2.37) the quantity $E[\psi_{i,o}^2]$ dictates the initial condition of the convergence of the RAM echo canceller. Also, we see that the MSE converges to a steady-state residual echo power\(^{21}\) if and only if the step-size parameter $\mu$ satisfies the condition

$$0 < \mu < 2$$ \hspace{1cm} (2.38)

Some of the fundamental differences between a nonlinear echo canceller and a linear transversal-type canceller are:

1. The nonlinear echo canceller mitigates the effects of nonlinearities in the echo path by storing the echo replicas of all the input data pattern in the look up table.

2. The nonlinear echo canceller does not require any calculation to get the echo replica, as opposed to the linear echo canceller where a convolution has to be performed.

3. The nonlinear echo canceller requires much larger memory size than the linear canceller. The size grows exponentially as $N$ increases.

4. The rate of convergence of a nonlinear echo canceller is slower than a linear-type echo canceller.

A multi-memory echo canceller was proposed by Maitre et al., [55], [49]. This technique preserves all the advantages of nonlinear cancellation without the drawback of a large memory and a slow adaptation by dividing the input data vector $\mathbf{X}$ into blocks\(^{22}\) and performing nonlinear echo cancellation on each of the blocks (nonlinear cancellation between blocks is not possible). This is illustrated in Figure 2.9.

\(^{21}\)It is equal to $\mu/(2 - \mu)$.

\(^{22}\)Each block can store $2^{N/M}$ words.
SR = SHIFT REGISTER

Figure 2.9: Block diagram of multi-memory echo canceller.
2.3.3 Recent Techniques in Echo Cancellation

Several adaptive echo cancellation techniques have been proposed for two-wire full-duplex digital transmission system. This section will give an overview of different techniques used in echo cancellations. The emphasis will be on methods that have been applied to the U-interface. Some of these methods have utilized line codes such as AMI, 4B3T, etc. The advantages and disadvantages of each technique will be stated, along with any novel ideas or concepts used.

An approach for a data echo canceller for a two-wire full-duplex system has been first described by Koll and Weinstein [56]. Such an arrangement is shown in Figure 2.4. The transmitter output (i.e., data symbols) is applied simultaneously to the hybrid circuit and the AFIR filter. In the AFIR filter, each input data symbol is multiplied by a tap and then summed. Thus, the use of the transmitter output as a reference to the AFIR filter requires full precision multipliers. Since multipliers are expensive in terms of hardware implementation cost, it is undesirable to use this arrangement.

Another approach was considered by Mueller [57] where the input to the echo canceller is taken before the transmitter (i.e., scrambled binary data), (Figure 2.10). The advantage of this approach is that tap signals are binary, and multiplications are performed by addition and subtractions. Moreover, this is preferable for the adaptation process in the AFIR filter because of the nature of the uncorrelated data. This arrangement also simplifies the hardware of the echo canceller. From the viewpoint of the canceller, the transmitter is now part of the echo path.

Adaptive Reference Echo Canceller (AREC) is considered by D.D. Falconer [58]. In data echo cancellation, a formidable factor that slows the adaptation of the echo canceller is the presence of the far-end signal\footnote{Far-end signal is considered as a source of disturbance to the echo canceller adaptation algorithm.}. In AREC, the far-end signal is adaptively removed by a reference former\footnote{The reference former is an adaptive transversal filter whose input is the sequence of receiver} from the cancellation error in a decision-
HPRF = High Pass Receiving Filter

Figure 2.10: Data echo canceller with symbol reference proposed by Mueller.

directed scheme. As a result, the speed of adaptation increases. This is illustrated in Figure 2.11, where the adaptive reference former is attached to the output of the receiver. Reference former adaptively forms a linear combination of the current and past receiver decisions to form an estimate of the far-end signal appearing in the cancellation error signal. This estimate is subtracted from the local error signal to yield a new error signal, which is employed in the echo-canceller adaptation algorithm. A joint adaptation of the echo canceller and the reference former is used. Since the far-end signal has been extracted from the error signal, MMSE is made smaller and the adaptation step-size $\mu$ can be selected to be larger for a given excess mean-square cancellation error. This in turn speeds adaptation. The AREC offers much faster convergence and shorter coefficient wordlengths than the conventional echo canceller. However, there is a problem getting started, when the cancellation may not be enough decisions (see Figure 2.11) and output is an estimate of the far-end signal.
to support a mediocre receiver error rate. This problem is solved either by starting out with a known training sequence, or by disabling the adaptive reference former and using a large adaptation step-size initially until the echo is canceled adequately for an acceptable error rate.

\[ \text{HPRF} = \text{High Pass Receiving Filter} \]

Figure 2.11: Implementation of Adaptive Reference Echo Cancellation.

A method is introduced (by Kanemasa et al.) in [39] that adaptively changes the canceller's step-size accordingly to the residual echo level by using the correlation between the echo replica and the error signal (as illustrated in Figure 2.12). This structure permits the step-size parameter \( \mu \) to vary according to the residual echo level since both the echo replica and the residual echo are correlated during the canceller convergence process, whereas they are uncorrelated in the steady state after convergence. Therefore, at the start of the adaptation, the correlation level is highest (implies largest step-size \( \mu \)). Consequently, the speed of adaptation is maximum. As
the echo canceller convergence evolves, the correlation between the echo replica and the residual echo decreases and so does the value of the step-size parameter $\mu$. The idea of the correlator is quite interesting where a leaky integrator is used to gradually leak/decrease the step-size $\mu$. A theoretical analysis for the proposed scheme was given for AMI line code. The sign form of the LMS algorithm is used. Due to the step size adaptation, this method increased the echo canceller convergence speed by an order of magnitude over the conventional sign algorithm.

![Diagram](image)

**Figure 2.12:** Kanemasa proposed echo canceller block diagram.

Because of the large dynamic ranges of the echo and far-end signal (e.g., worst case -10 dB). An AFIR echo canceller requires a large number of bits for the tap representation. Park and Macchi introduced the controlled gain echo canceller in [60] to reduce the required number of bits. This technique uses an adaptive gain control (AGC) at the output of the echo canceller. A feedback loop between the echo canceller and the AGC enables approximate regulation of the echo canceller output power at a nominal level, independent of the echo and far-end signal levels. Thus, the precision required for the AFIR taps is reduced to a minimum value. This new technique offers a 4 bit reduction over the classical echo canceller. Moreover, the convergence speed
is improved over the conventional echo canceller.

The application of Recursive Least Squares (RLS) algorithm to Data Driven Echo C canceller (DDEC) has been reported in [61], [62]. The proposed method yields very short convergence times for the DDEC. Presently, the complexity of this technique renders it impractical for production.

A method is given in [63] that combines linear and nonlinear echo cancellation. Each echo canceller tap comprises an adaptive nonlinear compensator, and jitter compensator.

Due to the presence of both linear and nonlinear distortion of the echo path, a parallel combination of linear and nonlinear echo cancellation is used [64]. It was shown that it resulted in improved speed of convergence. The configuration involves a sequentially adaptive transversal echo canceller and memory echo canceller. The transversal echo canceller is allowed to initially adapt while the memory echo canceller is inoperative. Thus, the transversal echo canceller is adapted to the linear part of the echo path impulse response and its coefficients are frozen without the intercession of the memory echo canceller. Then, the memory echo canceller is adapted to cancel the echo response consisting of the nonlinear echo components. An analytical treatment of the case where the linear/nonlinear echo cancellers have joint adaptation is not given in [64]. A recent thesis by K. Yamazaki [65] investigated the convergence behavior of a jointly adaptive linear/nonlinear echo canceller. Intriguingly, it was also revealed that the nonlinear canceller was cancelling a portion of the linear echo component. Lastly, a comparison of the jointly adaptive linear/nonlinear canceller and the sequentially adaptive linear/nonlinear canceller showed that the later had faster convergence properties.

A technique proposed by Mogavero et al [66] utilizes three distinct methods to cancel different parts of the echo (see Figure 2.13). The echo impulse response is divided into three different regions.
• A main pulse in which the echo is high and irregular.

• A second region in which the echo is low and irregular.

• A tail pulse.

The first part of the echo is canceled with an adaptive nonlinear echo canceller (table look-up). The second section is an adaptive transversal filter that evaluates the second part of the echo response. The use of table look-up EC on the second region is unnecessary. That is due to the fact that the power of the nonlinear harmonics decrease with the decrease of the echo level. The final section is recursive to cancel the echo tail (to be discussed shortly).

A two-stage echo canceller (i.e., parallel configuration) has been presented in [69]. The required resolution of the A/D converters is reduced utilizing the two-stage (coarse/fine) canceller concept shown in Figure 2.14. The first stage echo canceller performs a coarse cancellation to reduce the dynamic range requirement of an A/D converter. Also, it will prevent saturation in the $\sqrt{f}$ equalizer caused by the high-amplitude echo in the analog domain. Saturation of analog devices introduces nonlinearity. The second stage echo canceller compensates for the residual echo due to the coarse cancellation of the first stage echo canceller. A $\sqrt{f}$ equalizer is used to compensate for subscriber loop loss and to shorten the echo tail [67], [68]. The A/D requirement was reduced from 14 to 10 bits. A novel vector quantization technique is used to suppress the long echo tail as will be explained in the next section.

2.4 Echo Tail Cancellers

There has been a number of published designs that deal with the suppression of the echo tail response. In this section, a review of those techniques will be given. It will be seen that the complexity of the tail canceller is depended on the nature of the line
Figure 2.13: Configuration of three-stage echo cancellation incorporating DA as tail canceller.
code to be used. AMI and 4B3T line codes have high pass spectrum.\textsuperscript{25} As a result, they cause short echo tail. Codes such as 2B1Q or 3B2T will cause very long echo tails.

A method proposed in [66] uses a recursive tail echo canceller based on a digital accumulator, with transfer function $\frac{1}{1-z^{-T}}$. To cancel the echo tail, the echo tail response is generated by the digital accumulator through the evaluation of the running digital sum (RDS) of the AMI line code\textsuperscript{26}. Since AMI line code has a finite RDS [70]-[72], the digital accumulator is always stable. Figure 2.13 shows this three-stage echo canceller. The hardware requirement of the digital accumulator is minimum. However, it can not be used as a tail echo canceller for 2B1Q line code, since this code has a large RDS [70].

Echo tail suppression using vector quantization was proposed by Takatori et al.,
[69]. In this realization a 4B3T line code at 120 kbaud is used. The vector quantization is performed by an infinite echo estimator with digital accumulator and gain controller. This digital accumulator with transfer function \( \frac{2z^{-1}}{1-2z^{-1}} \) calculates the RDS of the 4B3T input sequence. Infinite echo estimator is always stable, since, the RDS of 4B3T code is finite [71], [73]. The vector quantization technique, combined with high-pass receiving filter, reduced the number of taps to a minimum.

An analog tail echo canceller based on an automatically controlled balancing network circuit is reported in [74]. The purpose of this circuit is to suppress the echo by matching the line impedance and balancing network impedance as closely as possible. A selectable balancing network (eight possibilities), which considered eight impedance characteristics each, was used to reduce the transhybrid loss. An improvement of more than 18 dB over a simple resistor termination was obtained.

High pass receiving filter is used to reduce echo tail length (e.g. in [75]-[79]). This is very efficient method to suppress echo tails. However, excessive high-pass filtering does allow more NEXT noise. That will compromise the performance of the equalizer. A \((1 - D)\) version\(^{27}\) of this method is widely used.

An IIR tail echo canceller is reported in [80], [81]. Figure 2.15 shows the recursive-type filter connected in tandem after the second-stage echo canceller (covers the first 9 baud intervals). High pass receiving filter and balance network are used to shorten the echo tail. The value of the fixed pole of the IIR filter is calculated from the d.c. cutoff characteristics of the hybrid.\(^{28}\) The gain-parameter \(3\) is being adapted by LMS algorithm to track the echo tail response.

In [82], an IIR tail echo canceller and a \((1 - D)\) filter are compared. Figure 2.16 illustrates a parallel IIR filter used as tail canceller. The coefficients of the the IIR filter are designed from the d.c. cutoff characteristics of the hybrid. Since the echo tail decreases very slowly, an IIR filter with large time constant is required. Therefore,

\(^{27}\) \(D\) is one delay element.

\(^{28}\) The echo tail response is modeled by a first-order IIR filter.
Figure 2.15: Configuration of echo tail canceller using IIR filter with variable gain.

\[ \alpha = \text{Fixed Pole Value} \]
\[ \beta = \text{DC Gain} \]
this filter's transfer function has poles near unit circle, and a high degree of accuracy is required. A transversal-type AFIR filter of order 64 taps was used to cancel the main echo response. The \((1 - D)\) filter shown in Figure 2.17, reduces the echo tail, since the echo tail decreases monotonically \([82]\). The advantage of using \((1 - D)\) filter is that shorter echo canceller length can be achieved. However, the use of \((1 - D)\) filter decreases the SNR since it enhances the noise. Another disadvantage, a 4-level received signal is transformed into a 7-level signal by the \((1 - D)\) filter. Assume there is no channel distortion. Extra hardware is required to retransform the signal.

![Figure 2.16: Block diagram of parallel IIR filter to suppress echo tail.](image)

A combination of non-adaptive IIR filter and \(1 - e^{-\frac{RL}{T}}D\) receiving filter is used to reduce the number of echo canceller taps \([83]\). The IIR coefficient is set to \(e^{-\frac{RL}{T}}\), where \(R\), \(T\), and \(L\) and the termination resistance, the band interval, the inductance of the transformer, respectively. Since the gain of the IIR filter is non-adaptive, any parameter value changes of analog components could degrade the degree of cancella-
Echo tail cancellation based on analog continuous orthonormal functions (IIR filters) was investigated in [99].

A similar problem exists for the DFE due to the long echo tail of the impulse response of transmission path. Therefore, the complexity of the DFE can be reduced by using a similar tail suppression schemes. An adaptive quantized feedback is used in [S2] to reduce the complexity of the DFE. A technique is proposed by P. Mohanraj et al., [85] which uses an adaptive piecewise linear fit filter to fit the tail of the post-cursor ISI.

2.5 Conclusion

In this chapter, we discussed the transmission impairments of the loop media that affect the system performance and design. Two transmission schemes have been proposed to achieve full-duplex two-wire transmission. These two schemes are TCM and ECH. Since ECH was selected by the standard committee, we concentrated on this technique. Implementation design challenges for the 2B1Q transceiver were summarized. 2B1Q line code was determined to have the best performance/complexity tradeoff of the proposed line codes. The general theory of the echo canceller was presented. Recent techniques in echo cancellation were briefly reviewed. A comparison of various echo tail cancellers was presented.

---

29 Quantized feedback was first used by F. D. Waldhauser to compensate for the baseline wander ISI [84].
Chapter 3

Theory of Interpolated FIR Filters

It is the objective of chapter 3 to overview the general theory of the narrow-band interpolated FIR filters. This overview is a summary of the three major papers [88]-[90] that have been published on the IFIR filters. For completeness, it is essential to have a good understanding of the fundamental properties and limitations of these narrow-band filters. To this end, the chapter begins with the introduction of the IFIR filters in section 3.1. An analysis of IFIR filters in the time domain is given in section 3.2. This concept will be utilized in the derivation of the MSE behavior of the adaptive IFIR structure in chapter 4. The design steps of IFIR filters are outlined in section 3.3. It is important to note that the IFIR filter design process is based on digital interpolation formulated in the frequency domain. In section 3.4, a detailed description of the IFIR filter cascades (i.e., sparse filter and interpolator) along with the approximate design formulas is given. Section 3.5 briefly describes the finite wordlength attributes of the IFIR filters. The computational efficiency of the IFIR filters is demonstrated by several illustrative examples in section 3.6. Concluding remarks on the chapter are given in section 3.7.

3.1 Introduction

Finite impulse response (FIR) filters are known to possess desirable properties such as guaranteed stability, absence of limit cycles, and linear phase, if needed. Their
major disadvantage is the large number of arithmetic operations needed for the implementation, as higher order filters are often required. The number of multipliers needed for implementation is equal to the order of the filter in the case of a general phase. The minimum length of the low-pass FIR filter to meet the given frequency domain specifications is approximately [86]

\[ N = \frac{-20 \log_{10} \sqrt{\delta_p \delta_s} - 13}{14.6 \Delta F} + 1 \]  

(3.1)

where \( \delta_p \) and \( \delta_s \) are the passband and stopband ripples, and \( \Delta F \) (\( = f_p - f_s \)) is the transition bandwidth. The above estimate is more precise when the passband and stopband ripples are small. If the ripples are large, a better estimate is given by a more complicated expression in [87]. In direct form implementation, each multiplier determines the value of one impulse response sample independently of the other samples. However, in narrow-band FIR filters there is a relatively strong correlation between adjacent impulse response sample values. By developing filter structures that exploit this correlation, the number of multipliers required in the implementation can be significantly reduced.

One of the computationally efficient realizations for narrow-band FIR filters is the Interpolated FIR filter, or simply, the IFIR filter [88]-[90], which is a cascade realization of the form

\[ H(Z) = F(Z^L)G(Z) \]  

(3.2)

where \( L \) is the interpolation factor and

\[ F(Z^L) = \sum_{n=0}^{L} f(n)Z^{-nL} \]  

(3.3)

\[ G(Z) = \sum_{n=0}^{J} g(n)Z^{-n} \]  

(3.4)
In this realization, the first FIR structure, \( F(Z^L) \), called the sparse filter, has a very sparse impulse response (resulting in a greatly reduced number of arithmetic operations) with only every \( L \)th sample being non-zero. Moreover, \( F(Z^L) \) has a periodic frequency response with period \( 2\pi/L \). Because of the periodicity of the response of the \( F(Z^L) \), it has multiple undesired spectral images centered at \( 2\pi K/L \) with \( K = 1, 2, 3, \ldots, L - 1 \). The second FIR structure, \( G(Z) \), a very simple interpolator of short length, fills in the missing samples by interpolation and attenuates the undesired spectral images of the response of the sparse filter below the prescribed stopband level. In [SS], it was shown that the IFIR structure reduces the number of multipliers and adders by almost a factor of \( L \). In addition, roundoff noise and coefficient sensitivities improve.

In the design of IFIR filters, one important parameter to be selected is the interpolation factor \( L \). As \( L \) increases, the complexity of the sparse filter decreases but the complexity of the interpolator increases. The IFIR filter design method is performed in frequency domain. In order to generate the interpolated impulse response \( H(Z) \), we cascade the sparse filter \( F(Z^L) \) with the interpolator filter \( G(Z) \). For a chosen \( L \), the IFIR filter design is started with the design of the model filter \( \hat{F}(Z^L) \). The model filter is defined as a filter that has a passband and transition bandwidth that is \( L \) wider than the desired frequency response. The sparse filter \( F(Z^L) \) is obtained from the implementation of the model filter \( \hat{F}(Z^L) \) by replacing each delay with \( L \) delays. The sparse filter band edge frequencies (in the first period of the response) are equal to those of the desired response. After designing the model/sparse filter, the interpolator is designed to attenuate the spectral replicas of the response of the sparse filter. Thus, the passband of the interpolator is chosen to have a width equal to the stopband edge \( f_s \) of the overall IFIR design.

\(^1\)Mathematically, a \( k \)-th order interpolation is to fit \( k + 1 \) points by a \( k \)-th order polynomial. For example, first order interpolation is to use a straight line to fit two given points and second order interpolation is to fit three given points by a parabola.
Consider the model filter \( \hat{F}(Z) \) with the impulse response \( \hat{f}(n) \). \( \hat{f}(n) \) is filled in with \( L - 1 \) zero-valued samples between each pair of samples, giving the signal,

\[
f(n) = \begin{cases} 
\hat{f}(n/L) & n = iL, \ i = 0, \pm 1, \pm 2, \ldots \ \\
0 & \text{otherwise}
\end{cases} \quad (3.5)
\]

That is the response of \( F(Z^L) \). An example of IFIR implementation for \( L = 4 \) is shown in Figure 3.1.

![Diagram of IFIR filter structure with L=4](image)

Figure 3.1: IFIR filter structure with \( L=4 \).

### 3.2 Mathematical Formulation of IFIR Filters in the Time Domain

The IFIR filter is shown in Figure 3.2. The overall time response \( h_n \) is given by

\[
h_n = \sum_{k=0}^{\beta} f_{Lk}g_{n-Lk}
\]

(3.6)

\[
h_n = F^T G
\]

(3.7)
Figure 3.2: Block diagram of Interpolated FIR (IFIR) filter.

Then, we can represent a vector that contains the interpolated impulse response samples as

$$\mathbf{H} = [h_0, h_1, \ldots, h_i, \ldots]^T$$  \hspace{1cm} (3.8)

where

$$\mathbf{F} = [f_0, f_L, \ldots, f_{\beta L}]^T$$  \hspace{1cm} (3.9)

$\mathbf{F}$ is the vector of non-zero coefficients of $F(Z^L)$. The vector of the interpolator coefficients is defined as

$$\mathbf{G} = [g_0, g_1, \ldots, g_j]^T$$  \hspace{1cm} (3.10)

where $\beta$ is the number of the non-zero coefficients in $\mathbf{F}$ and $j$ is the number of taps in $\mathbf{G}$. The output $y_n$ can be expressed as

$$y_n = \sum_{i=0}^{j} \sum_{k=0}^{\beta} g_i f_{i,k} x_{n-i-L,k}$$  \hspace{1cm} (3.11)

Also, the output $y_n$ can be expressed in vector notation as\(^2\)

$$y_n = \mathbf{F}^T \hat{\mathbf{X}} \mathbf{G} = \mathbf{F}^T \mathbf{M}$$  \hspace{1cm} (3.12)

\(^2\)Here we have dropped the subscript $n$ from the matrix $\hat{\mathbf{X}}$ and the vector $\mathbf{M}$ for convenience.
where \( \bar{X} \) is the matrix for the input data, represented as

\[
\bar{X} = \begin{bmatrix}
    x_n & x_{n-1} & \cdots & x_{n-j} \\
    x_{n-L} & x_{n-1-L} & \cdots & x_{n-j-L} \\
    \vdots & \vdots & \ddots & \vdots \\
    x_{n-\beta L} & x_{n-1-\beta L} & \cdots & x_{n-j-\beta L}
\end{bmatrix}
\]

(3.13)

The vector \( \mathbf{M} \) consists of inner products of the input data and the interpolator coefficients,

\[
\mathbf{M} = \bar{X} \mathbf{G} = [m_0 \ m_L \ \ldots \ m_{\beta L}]^T
\]

(3.14)

or

\[
\mathbf{M} = \bar{X} \mathbf{G} = [\sum_i g_i x_{n-i} \ \sum_i g_i x_{n-1-i-L} \ \ldots \ \sum_i g_i x_{n-i-\beta L}]^T
\]

(3.15)

### 3.3 Design Procedure

The IFIR filter design process is based on frequency domain properties of digital interpolation. The design of the interpolator \( G(Z) \) can be formulated as the design of a multistage band FIR filter having passband matching the wanted response of the sparse filter \( F(Z^L) \) and stopbands overlapping the unwanted replicas of the sparse filter response. Y. Neuvo et al., [88], used the Parks-McClellan [91] algorithm to design simple interpolators. In a recent paper by T. Saramäki et al., [93], it is shown how the overall IFIR filter including the sparse filter \( F(Z^L) \), interpolator \( G(Z) \), and \( L \) can be optimized to arrive at the smallest number of required arithmetic operations. Both single-stage and multistage implementation of \( G(Z) \) were discussed. This recent method was based on iteratively designing the sparse filter and the interpolator using the Remez multiple exchange algorithm. Nevertheless, we will only outline the design
steps for the earlier (single-stage) method, since it will be utilized in the design of the
tail canceller in chapter 5.

3.3.1 Selection of \(L\)

The largest value of \(L\), \(L_{\text{max}}\), that can be used depends on the specifications of the
IFIR filter. If the stopband edge frequency of the low-pass IFIR filter is denoted by
\(w_s\) (radians), then the maximum value of \(L\) is

\[
L_{\text{max}} = \left\lceil \pi/w_s \right\rceil \tag{3.16}
\]

where \(\lceil x \rceil\) stands for integer part of \(x\). In practice it is prudent that a somewhat
smaller value than \(L_{\text{max}}\) be selected as the requirements for the interpolator become
otherwise more stringent if it has to select one of two very close passbands of \(F(Z^L)\).

3.3.2 Design Steps

The design of the IFIR filters can be summarized as follows.

1. From the given equivalent FIR filter stopband edge frequencies, calculate \(L_{\text{max}}\)
   (Eq.(3.16)) and select a suitable \(L < L_{\text{max}}\). After \(L\) has been selected, the
   positions of the unwanted repetitions of the response of the sparse filter are
   known.

2. Design the model filter \(\hat{F}(Z^L)\). The band edge frequencies of the model filter are
   obtained by multiplying the edge frequencies of the sparse filter in the interval
   \([0, \pi/L]\) by \(L\). Then, the sparse filter \(F(Z^L)\) is obtained from the implementation
   of the model filter \(\hat{F}(Z)\) by replacing each delay with \(L\) delays.

3. Design the interpolator \(G(Z)\) to attenuate these repetitions of the response of
   the sparse filter to or below the stopband level. \(G(Z)\) is designed as a multi-
   stop band FIR filter having passband matching the wanted band of \(F(Z^L)\) and
stopbands overlapping the unwanted replicas of this band. The interpolated impulse response $H(Z)$ is obtained by cascading $F(Z^L)$ with $G(Z)$.

To generate the interpolated impulse response, $F(Z^L)$ is cascaded with $G(Z)$ as shown in Figure 3.1. The design of the model filter and interpolator can be done using any FIR filter design programs. We used the program DFDP\(^3\) [105] (see examples in section 3.6). Frequency domain behavior of the different signals in the various stages of the implementation are illustrated in Figure 3.3. Note that the stopbands of the interpolator are equally spaced.

---

**Model**

---

**Sparse**

---

**Interpolator**

---

**IFIR**

---

$S\,B = \text{Stopband}$

---

**Figure 3.3:** An example to illustrate frequency response of model, sparse, interpolator, and IFIR filter. $L = 15$.

---

\(^3\text{Digital Filter Design Package}\)
3.4 Computational Requirements

In this section, the approximate design formulas for the IFIR filters are provided.

Length Estimation

For the length of the sparse filter $F(Z^L)$, a good estimate is

$$N_F = \lceil N/L \rceil$$  \hfill (3.17)

where $N$ is the minimum order of an optimum direct-form FIR design to meet the given overall criteria. The magnitude response of the sparse filter $F(Z^L)$ can be approximated as shown in Figure 3.4.

![Sparse Filter Response](image)

Figure 3.4: A view of the response of the sparse filter and its first spectral image central at $2\pi/L$.

3.4.1 Model Filter

In Figure 3.3, it is important to note that in the interpolated impulse response filter, the passband and transition bandwidths are $1/L$th of the corresponding widths of the model filter $\hat{F}(Z^L)$. Thus, the length of the model filter is approximately $1/L$th of that of the conventional FIR filter required to meet the given specifications, reducing
the multiplier requirement to $1/L$th of the original requirement (not including the interpolator). The required length of the model filter is given, approximately, by [88]

$$N_M \approx \frac{-20 \log_{10} \sqrt{\delta_p \delta_s} - 10}{14.6 \Delta F_M} + 1$$  \hspace{1cm} (3.18)

where $\Delta F_M (= L \Delta F)$ is the transition bandwidth of the model filter. The length of the model filter $N_M$ can be calculated in terms of the ratio of the transition bandwidth of the conventional FIR filter and the transition bandwidth of the model filter as

$$N_M \approx N \frac{\Delta F}{\Delta F_M} + 1$$  \hspace{1cm} (3.19)

### 3.4.2 Interpolator Filter

This subsection gives a detailed discussion on digital interpolators. Of these, two techniques of the design of the interpolator filter are considered: Recursive Running Sum interpolators and Optimal interpolators. Formulas characterizing the number of multipliers and the minimal value of $L$ minimizing the complexity of the IFIR will be provided.

**Recursive Running Sum (RRS) interpolator**

The RRS is multiplier-free FIR filter with unity coefficients [93]. Direct-form implementation of the RRS interpolator is shown in Figure 3.5 and mathematically is represented as

$$G(Z) = \sum_{n=0}^{N_G-1} g(n)Z^{-n}$$  \hspace{1cm} (3.20)

with $N_G = L$ and $g(n) = 1$ for $n = 0, 1, 2, ..., N_G - 1$. The frequency response of the RRS interpolator of length $L$ is given by

$$G(w) = \frac{\sin(Lw/2)}{L \sin(w/2)}$$  \hspace{1cm} (3.21)
The RRS interpolator has zeros at frequencies \(2\pi K/L\), \(K = 1, 2, 3, \ldots, L - 1\) which are the centers of the undesired images of the desired spectrum of the sparse filter. The first zero is located at the frequency \(2\pi K/L\) and the first sidelobe is the highest sidelobe with an approximate attenuation of 12.5 dB.

![Diagram](image)

**Figure 3.5:** Direct-form implementation of the RRS.

**Selection of Minimal \(L\)**

The total number of taps\(^4\) needed for the IFIR filter realization using a single stage the RRS interpolator is given by

\[
N_r = N_F + N_G
\]  
\[(3.22)\]

\[
N_r = N/L + L
\]  
\[(3.23)\]

where \(N_F\) and \(N_G\) are the number of taps in the sparse filter and the RRS interpolator, respectively. Differentiating \(N_r\) with respect to \(L\) and equating it to zero and solving the resulting equation, we get

\[
L_{\text{min}} = \sqrt{N}
\]  
\[(3.24)\]

\(L_{\text{min}}\) has to satisfy the condition that

---

\(^4\)The RRS interpolator has no multipliers.
\[ L_{\text{min}} \leq L_{\text{max}} = \lfloor \pi / w_s \rfloor \]  

(3.25)

The minimal number of taps is given by

\[ N_{r,\text{min}} = N / \sqrt{N} + \sqrt{N} \]  

(3.26)

To illustrate the dependence of the IFIR computational complexity on the interpolating factor \( L \), Eq.(3.23) is plotted in Figure 3.6. It can be observed, if \( L \) increases, the complexity of the sparse filter decreases while the complexity of the interpolator increases. For a given value of \( L \), the length of the interpolated impulse response \( N_{ir} \) using Eq.(3.8) and Eq.(3.17) is

\[ N_{ir} = L(N_F - 1) + N_G \]  

(3.27)

![Figure 3.6: A plot of the number of taps versus \( L \) for the sparse filter \( F(Z^L) \) and the interpolator \( G(Z) \).](image)
Optimal Interpolator

The multistop band interpolator is designed using the Parks-McClellan program to optimize the number and locations of the zeros. The interpolator’s frequency specifications are obtained from the overall IFIR filter. The stopband regions of the interpolator must totally contain the unwanted replicas of the response of the sparse factor \( F(Z^L) \). The passband of the interpolator \( f_{pi} \) is chosen to have a width equal to the stopband edge \( f_s \) of the overall IFIR design (see Figure 3.7). The interpolator specifications used in the following analysis are

\[
\text{Passband} \in [0, f_{pi}] \tag{3.28}
\]

\[
\text{Stopbands} \in [K/L - f_s, K/L + f_s], \quad K = 1, 2, \ldots, L - 1 \tag{3.29}
\]

\[
\text{Passband ripple} = \delta_p \tag{3.30}
\]

\[
\text{Stopband ripple} = \delta_s \tag{3.31}
\]

\( f_{pi} \) is chosen to be

\[
f_{pi} \doteq f_s \tag{3.32}
\]

The frequency \( f_t \) is defined as

\[
f_t \doteq (f_s + f_{pi}) \tag{3.33}
\]

The areas in between the prescribed stopbands can be considered \textit{don’t care bands} where the amplitude response of \( G(Z) \) is tolerated to take on any value less than one [88]. These areas coincide with the stopband regions in the sparse filter which already possess the desired attenuation. An approximate value of the length of \( G(Z) \) is [89]
Figure 3.7: A view of the response of the interpolator filter and its first stopband centered at $1/L$.

$$N_G \approx 1 + \alpha N \Delta F / \Delta F_G$$  \hspace{1cm} (3.34)

$\alpha$ is small number less than 1 [89]. The interpolator's transition bandwidth can be expressed as

$$\Delta F_G = \left(1/L\right) - f_s - f_{pi}$$  \hspace{1cm} (3.35)

**Choice of Minimal $L$**

The total number of multipliers $M_o$ required in the overall implementation (if an optimal single stage $G(Z)$ is used) is given by

$$M_o = M_F + M_G$$  \hspace{1cm} (3.36)

It can be rederived by substituting for $N_F$ and $N_G$ from Eq.(3.19), Eq.(3.34), and Eq.(3.35) respectively

$$M_o = N \Delta F \{ (1/L \Delta F) + \alpha L/(1 - L\phi) \} + 2$$  \hspace{1cm} (3.37)
$L_{\text{min}}$ is found by differentiating Eq. (3.37) with respect to $L$ and equating it to zero and solving the quadratic equation [89]

$$L_{\text{min}} = 1/[f_i + \sqrt{\alpha \Delta F}]$$  \hspace{1cm} (3.38)

The above equation reveals that, having $f_p$ and $f_s$ of the equivalent low-pass FIR filter, the minimal value of the interpolation factor $L$ can be determined and used in the sparse filter $F(Z^L)$ design. In Eq. (3.37), substitute for $L_{\text{min}}$, then the number of multipliers in the optimal case is

$$M_{o,\text{min}} = N\Delta F \{ (1/L_{\text{min}}\Delta F) + \alpha L_{\text{min}}/(1 - L_{\text{min}} f_i) \} + 2$$  \hspace{1cm} (3.39)

To summarize, Table 3.1 compares the number of computations required by the IFIR filter with that of an equivalent conventional FIR filter meeting the same frequency domain specifications.

<table>
<thead>
<tr>
<th>Method</th>
<th># of multipliers</th>
<th># of adders</th>
<th>Type of $G(Z)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIR</td>
<td>$N$</td>
<td>$N - 1$</td>
<td>-</td>
</tr>
<tr>
<td>IFIR</td>
<td>$N/L + L$</td>
<td>$N/L + L - 2$</td>
<td>RRS</td>
</tr>
<tr>
<td>IFIR</td>
<td>$N/L + N_G$</td>
<td>$N/L + N_G - 2$</td>
<td>Optimal</td>
</tr>
</tbody>
</table>

Table 3.1: FIR vs IFIR, a complexity comparison.

### 3.5 Finite Wordlength Properties of IFIR filters

In this section, coefficient sensitivities and output roundoff noise properties of the IFIR filter are analyzed and shown to be better than those of the direct form FIR filter. First, we consider the coefficient quantization noise in direct form FIR filters. An upper bound for the standard deviation of the coefficient quantization error is derived by L. Rabiner and B. Gold [93]:

$$\sigma(w) \leq \frac{q}{2} \sqrt{\frac{(2N - 1)}{3}}$$  \hspace{1cm} (3.40)
where

\[ q = \text{Quantization step size} \quad (3.11) \]

\[ N = \text{Filter length} \quad (3.42) \]

In the implementation of the IFIR filter, Y. Neuvo et al. [88] make the assumption that \( G(Z) \) does not affect the coefficient sensitivity. Therefore, it is adequate to analyze merely the effect of quantizing the coefficients of the sparse filter \( F(Z^L) \). This is a justifiable assumption as in the vicinity of the passband, where the gain of the interpolator is approximately one\(^5\), the distance to the zeros of the interpolator is large; i.e., the transition bandwidth of the interpolator filter is fairly relaxed.\(^6\) Closer to the zeros of the interpolator, the required stopband attenuation is exceeded so much that the coefficient sensitivity is no problem. On the unwanted replicas of the sparse filter \( F(Z^L) \) response, the attenuation caused by the interpolator is easy to check using quantized interpolator coefficients, if necessary. According to these arguments it is sufficient to analyze only the effect of quantizing the coefficient of \( F(Z^L) \). The standard deviation \( \sigma_{FIR} \) of the coefficient quantization error of a conventional filter of length \( LN \) can be compared with that of an IFIR of length \( N \), \( \sigma_{IFIR} \) derived from \( F(Z^L) \) of length \( N \). For large \( N \), it can be shown using Eq.\,(3.40), that IFIR and FIR are related as

\[ \sigma_{FIR} \approx L \sigma_{IFIR} \quad (3.13) \]

To determine the output roundoff noise variance \( \gamma_{FIR} \) of a direct form FIR filter with length \( LN \), we assume that all multiplication products are rounded before they are summed (implies that each tap has a roundoff noise source). Thus, the total

\(^5\)In all of our IFIR filter designs, an amplitude of one is used.

\(^6\)J. F. Kaiser showed that direct form FIR filters have insignificant coefficient sensitivities if the transition bandwidth is not steep [110].
output roundoff noise is the sum of $LN$ uncorrelated noise sources, each uniformly distributed between $-q/2$ and $q/2$, with variance $q^2/12$. Therefore, the output roundoff noise variance ($\tau_{FIR}$) at the output of a conventional filter of length $LN$ is

$$\tau_{FIR} \approx \frac{LNq^2}{12} \quad (3.14)$$

An IFIR filter derived from sparse filter of length $N$ has the output roundoff noise variance

$$\tau_{IFIR} = [N \sum_i g_i^2 + \Theta] \frac{q^2}{12} \quad (3.45)$$

where $g_i$ are the interpolator taps and $\Theta$ is resulted from the noise generated in the interpolator section. Y. Neuvo et al. [88] showed that if $N$ is large, then the noise contributed by the interpolator can be neglected. Thus, we can relate the variance of the IFIR to that of the FIR as

$$\tau_{FIR} \approx LN \tau_{IFIR} \quad (3.46)$$

### 3.6 Design Examples

We have designed numerous IFIR filters with various specifications. In this section, we introduce some illustrative examples and contrast the hardware complexity requirements with those of the direct form FIR implementations. All the filters are designed using the Parks-McClellen program available in the DFDP.

**Example 1**

In this example, the desired passband and stopband ripples are of equal magnitude. The conventional low-pass FIR filter has its passband edge at 20 mHz and its stopband edge at 33 mHz. The passband ripple is to be at most $\pm 0.03$ while the stopband attenuation is to be at least 30 dB. The sampling rate is 1 Hz. The conventional linear
phase FIR filter designed using the Parks-McClellan filter program in the DFDP, requires a filter length 120. The frequency response for the conventional filter is shown in Figure 3.8. The minimal value of $L$ calculated from Eq.(3.38) is 6. $L_{min}$ occurs well below the upper bound $L_{max} = [\pi/w_x]$ (= 15). Selecting $L$ to be 6 and following the design steps outlined earlier, the length of the model filter (shown in Figure 3.8) is 20, and the interpolator is of order 16 (also shown in Figure 3.8). The hardware requirement for the conventional FIR and the IFIR are summarized in Table 3.2. To illustrate the dependence of the computational efficiency of the IFIR structure on the value of $L$, another IFIR filter is designed with the above specifications and $L = 10$.

As it appears in Table 3.2, the use of non-minimal interpolation factor $L$ can increase the complexity of the IFIR filters.

<table>
<thead>
<tr>
<th>Method</th>
<th>$L$</th>
<th>order</th>
<th>$N_F$</th>
<th>$N_G$</th>
<th>multipliers</th>
<th>adders</th>
<th>savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIR</td>
<td>1</td>
<td>120</td>
<td>-</td>
<td>-</td>
<td>60</td>
<td>119</td>
<td>-</td>
</tr>
<tr>
<td>IFIR</td>
<td>6</td>
<td>36</td>
<td>20</td>
<td>16</td>
<td>18</td>
<td>34</td>
<td>70%</td>
</tr>
<tr>
<td>IFIR</td>
<td>10</td>
<td>58</td>
<td>13</td>
<td>45</td>
<td>29</td>
<td>56</td>
<td>51%</td>
</tr>
</tbody>
</table>

Table 3.2: Hardware details for the filters designed in Example 1.
Figure 3.8: Magnitude responses of Example 1. (a) Conventional. (b) Model. (c) Interpolator. (d) IFIR.
Example 2

This is an example where the passband is very narrow. The sampling rate is 1 Hz. The passband frequency is at $f_p = 1$ mHz and the stopband frequency is at $f_s = 13$ mHz. The desired passband and stopband ripples have equal weights in the design. The passband ripple is to be at most ±0.018 while the stopband attenuation is to be at least 35 dB. The conventional linear phase filter requires 130 taps. Using Eq. (3.38), $L_{\text{min}}$ is calculated to be 10.12. Rounding, we use 10 as the interpolating factor. For this value of $L$, the interpolator has an order 21 while the corresponding model filter has an order 16. The frequency response for the conventional and the IFIR realizations are shown in Figure 3.9 and the respective hardware details are summarized in Table 3.3

<table>
<thead>
<tr>
<th>Method</th>
<th>$L$</th>
<th>order</th>
<th>$N_F$</th>
<th>$N_G$</th>
<th>multipliers</th>
<th>adders</th>
<th>savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIR</td>
<td>1</td>
<td>130</td>
<td>-</td>
<td>-</td>
<td>65</td>
<td>129</td>
<td>-</td>
</tr>
<tr>
<td>IFIR</td>
<td>10</td>
<td>37</td>
<td>16</td>
<td>21</td>
<td>19</td>
<td>35</td>
<td>77%</td>
</tr>
</tbody>
</table>

Table 3.3: Hardware details for the filters designed in Example 2.

From these tables, it is observed that the IFIR implementation provides a substantial reduction in the number of arithmetic operations over the conventional direct-form structures.

---

\[7\text{The price paid for the reduction is a slight increase in the number of delay elements.}\]
Figure 3.9: Magnitude responses of Example 2. (a) Conventional. (b) Model. (c) Interpolator. (d) IFIR.
3.7 Conclusion

The general theory of the IFIR filter design has been presented. IFIR filters are based on interpolation of the impulse response. The IFIR filter is a cascade of a prefilter with sparse impulse response and a frequency response periodic with $2\pi/L$, followed by an interpolator filter filling the missing impulse response samples and attenuating the undesired spectral images of the sparse filter response. A simple design procedure based on frequency domain has been reviewed. Two approaches to the design of the interpolator filter of the IFIR structure have been explained: the RRS interpolator and the optimal interpolator. Formulas describing the total number of arithmetic operations and the minimal value of the interpolation factor $L$ minimizing the computational complexity have been given. It has been shown that IFIR filters take $1/L$th of the adders and multipliers and, in addition, possess $1/f_L$th of the output roundoff noise level and $1/\sqrt{L}$th of the coefficient sensitivity of an equivalent FIR filter. The IFIR filters are shown to be very efficient in synthesizing narrow-band response. This was demonstrated through some examples. Additionally, this computational efficiency will be fully exploited in approximating the tail narrow-band response (in the following chapters).
Chapter 4
AIFIR Echo Tail Canceller

Due to the low-frequency components of the 2B1Q signal, a long echo tail is generated from the hybrid transformer. The presence of such a long tail in the echo response can dramatically increase the complexity of the echo canceller. The use of AFIR to cancel the echo tail is not effective due to the very high order required. In this thesis, we propose the use of a two-stage canceller for echo cancellation. The first stage is an AFIR filter that covers the first few symbol intervals of the echo path impulse response. The second stage is an adaptive interpolated finite impulse response (AIFIR) filter which approximates the echo tail. The AIFIR filter is used as an echo tail canceller since it has been shown to be capable of efficiently implementing very narrow band impulse responses. Significant complexity reduction was attained by using this new structure. The AIFIR structure retains all the advantages of the IFIR filter [88] without affecting the simpler calculations requirement of the 2B1Q line code (3 bit/symbol; in binary 2's complement representation of the 4-level symbol). The adaptive IFIR filter is described in section 4.1. A comprehensive mathematical treatment of the AIFIR structures is given in section 4.2. Moreover, the AIFIR filter is compared with an AFIR filter in terms of the computational requirement in subsection 4.2.1. The analysis of the proposed two-stage echo cancellation employing an AIFIR filter is addressed in subsection 4.2.2. Finally, the chapter ends with some concluding remarks in section 4.3.
4.1 System Description of Proposed Two-Stage EC

The rationale behind the two-stage cancellation scheme is the shape of the echo path impulse response. The first section of the impulse response is coarse,\textsuperscript{1} whereas the tail part of the response is a slowly decaying exponential (with a large time constant). Thus, there is a strong correlation between neighboring impulse response values. The reason for the AIFIR approach is the nature of the echo tail response. In frequency domain, the echo tail frequency response is a narrow-band response.\textsuperscript{2} IFIR filters are very effective in simulating narrow-band responses (as shown in chapter 3). Therefore, IFIR filters are well suited for approximating the tail section. The IFIR filter has been compared with other computationally efficient FIR filters structure and proved to be the most efficient filter [90].

There has been a number of published designs that deal with the cancellation of the echo tail response. A popular method is to simulate the long echo tail of the echo path response with a fixed pole first-order IIR (infinite impulse response) filter or filters, with adaptive gain [80]-[83], [108]. The value of the fixed pole is calculated from the d.c. cutoff characteristics of the hybrid. Due to the fact that the echo tail response decreases very slowly, an IIR filter with large time constant (pole near the unit circle) is required. Thus, this filter requires high precision implementation. An IIR filter with adaptive pole(s) has been used to approximate the echo tail response [109]. However, complicated adaptation algorithms are required to maintain stability during adaptation. Another technique to suppress the echo tail is to use $(1 - D)$ high-pass receiving filter (e.g., in [82]). Nevertheless, this filter does allow more of the effects of noise and crosstalk. Another disadvantage, a 4-level received signal is transformed into a 7-level signal by the $(1 - D)$ filter. Extra hardware is required to

\textsuperscript{1}The first-stage AFIR canceller is used to represent this section.

\textsuperscript{2}This tail can be modeled by a first-order IIR filter with a pole close to the unit circle.
retransform the signal.

The proposed tail canceller is an adaptive version of the interpolated FIR filter (IFIR) introduced in [10]. It is based on providing a long impulse response using fewer coefficients and making use of the special properties of narrow-band frequency response filters. Moreover, the AIFIR technique is more general and can be utilized to reduce the complexity of DFEs. Next, we will describe the structure of the adaptive IFIR filter.

The proposed architecture of the tail canceller is based on adapting the coefficients of the sparse filter $F(Z^L)$ while keeping the interpolator taps fixed [10] when simulating the echo tail narrow-band response, as shown in Figure 4.1. The interpolator taps are held fixed to simplify and reduce the complexity of the proposed approach.

![Figure 4.1: Block Diagram of Adaptive IFIR Filter.](image)

Since the AIFIR tail canceller maintains the computational efficiency (i.e., the number of arithmetic operations) of the IFIR filter, a lower order echo canceller can be achieved. The received 2B1Q signal is coded in 3-bits/symbol. It is thus advantageous to preserve the precision needed at this low level. As shown in Figure 4.1, the first section of the AIFIR will receive this 3-bit data requiring 3-bit delays and 3×r bit
multipliers, \( r \) being the number of bits/coefficient. The coefficients of this section are updated according to the LMS algorithm [95], [96]. The output is then fed into the interpolative low-order section which will be working on a higher precision level (i.e., higher than the first cascade) with fixed coefficients. The following section analyzes the convergence properties of the AIFIR filter and compares them with those of a conventional AFIR.

### 4.2 Analysis of Adaptive IFIR Structures

Many forms of adaptive algorithms exist which vary greatly in computational complexity. In this thesis, the LMS algorithm is used to iteratively update the taps of the sparse filter \( F(Z^L) \) so that the overall AIFIR response matches the impulse response of the echo tail. We now proceed to derive the MSE for AIFIR filter based on the system depicted in Figure 4.1. the error is

\[
e_n = d_n - y_n \tag{4.1}
\]

The output of the AIFIR can be expressed as

\[
y_n = \sum_{i=0}^{j} \sum_{k=0}^{M} g_i f_{Lk}(n) x_{n-i-Lk} \tag{4.2}
\]

or expressed in vector notation as

\[
y_n = F^T M \tag{4.3}
\]

\( F \) and \( M \) have been defined in section 3.2. Subsequently, the error signal for the AIFIR filter can be written as

\[
e_n = d_n - F^T M \tag{4.1}
\]

The instantaneous squared error is
\[ e_n^2 = d_n^2 - 2d_nF^TM + F^TMM^TF \] (4.5)

If we take the expected value of Eq. (4.5) and invoke the statistical assumptions regarding input signal, echo path, and error signal defined in subsection 2.3.1, we have the MSE


Let \( R \) be defined as

\[ R = E[MM^T] \] (4.7)

\[
R = E \left[ \begin{array}{cccc}
  m_0^2 & m_0m_L & \cdots & m_0m_{BL} \\
  m_Lm_0 & m_L^2 & \cdots & m_Lm_{BL} \\
  : & : & \cdots & : \\
  m_{BL}m_0 & m_{BL}m_L & \cdots & m_{BL}^2 \\
\end{array} \right] \] (4.8)

\( R \) is designated as 'modified correlation matrix'. \( R \) is a modified version of the conventional correlation matrix in the sense that its entries are inner products of input data and interpolator coefficients (as defined in chapter 3). In fact \( R \) is a diagonal matrix with the uncorrelated input data assumption.\(^3\) The variance of any diagonal element is equal to \( \sigma_n^2 \|G\|^2 \), where \( \|G\|^2 = \sum_k g_k^2 \).

Let \( P \) be defined as

\[ P = E[d_nM^T] = E[d_nm_0, d_nm_L, \ldots, d_nm_{BL}]^T \] (4.9)

Then, we can reexpress the MSE (Eq. (4.6)) in terms of Eq. (4.7) and Eq. (4.8) as

\[ MSE = E[e_n^2] = E[d_n^2] - 2P^TF + F^TRF \] (4.10)

\(^3\)High value of \( L \) will make sure that \( m_0 \) and \( m_L \) are uncorrelated (see Appendix C for pseudo proof).
The gradient is

$$\nabla_n = 2RF - 2P \quad (4.11)$$

The minimum mean squared error (MMSE) is obtained by setting the gradient equal to zero and substituting for $R$ in Eq.(4.10)

$$MMSE = E[d_n^2] - P^TF_{opt} \quad (4.12)$$

where $F_{opt}$ is the optimal coefficient vector.

Now, the update equation of the LMS algorithm for AIFIR filters can be derived. From Eq.(4.4), we can express the gradient estimate as

$$\hat{\nabla}_n = -2e_nM \quad (4.13)$$

Then, the weight update equation is

$$F_{n+1} = F_n + \alpha e_nM \quad (4.14)$$

where $\alpha$ is the gain parameter.

The above expressions are similar to the expressions for the conventional adaptive FIR (AFIR) [97]. Thus, we are still dealing with the standard form of equations. However, there are some differences. Firstly, the input data is passed through two filters which results in a matrix $\tilde{X}$ for the input signal (see Eq.(3.13)) instead of the input data vector $X$ applied to the conventional AFIR (see Eq.(4.15)). Secondly, the weight update equation is different from the one for the standard AFIR. It differs in the tap weight update increment, Eq.(4.14) shows $M$ instead of $X$. The vector $M$ consists of inner products of input data and interpolator taps. However, the nature of the computation required in the AIFIR filter is not different from that used for the standard AFIR.
4.2.1 Complexity Comparison: AIFIR vs AFIR

We now compare the number of computations required for the AIFIR filter with that required for an equivalent conventional AFIR filter estimating the same response. Starting with AFIR filters, the LMS algorithm interactively adapts the tap weights by

\[ \mathbf{W}_{n+1} = \mathbf{W}_n + \alpha e_n \mathbf{X} \]  

(4.15)

Direct implementation of Eq.(4.15) requires two multipliers per filter tap, one\(^4\) to produce the tap weight update, \( \alpha e_n x_{n-t} \) and a second to calculate the signal sample-tap weight product, \( w_i x_{n-t} \). Thus, \( N \) multipliers are required to compute the sample output and \( N \) multipliers are required for an update of all weights. Also \( (N - 1) \) adders are required. For an AIFIR structure, the number of non-zero coefficients for the sparse filter \( F(Z^L) \) can be estimated as,

\[ N_F = \lfloor N/L \rfloor \]  

(4.16)

where \( \lfloor x \rfloor \) denotes the largest integer \( \leq x \), and \( \lceil x \rceil \) denotes the smallest integer \( \geq x \). Therefore, the number of multipliers required for the implementation of AIFIR is

\[ N_i = \lfloor N/L \rfloor (3 \times r) \text{ multipliers} + N_G (r \times r) \text{ multipliers} \]  

(4.17)

where \( r \) is the number of bits/coefficient and input data is assumed to be 2B1Q symbols, \( N_G \) is the order of \( G(Z) \). From Eq.(4.17), we can see the savings in the required number of multiplications/output computation as compared with an \( N \)th order AFIR approximating the same response requiring \( N \) multipliers of size \( (3 \times r) \).

Now consider the number of multiplications required for the update equation Eq.(4.14). The individual components of \( M \) are an inner product of data inputs and

\(^4\alpha \) is chosen to be a power of two.
interpolator tap values. Because of the inherent properties of AIFIR, \( m_0 \) will become \( m_L \) after \( L \) sample times. Thus, during the operation of the AIFIR, all we need to compute is \( m_0 \) and feed \( m_0 \) to a serial shift register. \( N_G \) multiplications of \((3 \times r)\) bits each are needed to compute one inner product. This computation is required in addition to the \([N/L]\) multiplications of \((r \times r)\) bit each needed to update the coefficients of the sparse filter as in Eq.(4.14). The number of multipliers needed for the AIFIR update equation

\[
n_{ip} = [N/L] \ (r \times r) \ multipliers + N_G \ (3 \times r) \ multipliers \tag{4.18}
\]

It follows from Eq.(4.18) and Eq.(4.17) that the AIFIR filter would require a total of \((N_i + n_{ip})\) to compute the output and update the coefficients compared to the transversal AFIR filter estimating the same response. For large \( L \)'s significant savings are obtained. The complexity of the two-stage canceller of Figure 4.2 can be determined. In order to implement the AIFIR echo tail canceller (assuming linear phase design for the interpolator), then \(2(N_F + [N_G/2])\) multipliers are required. The main echo canceller requires \(2N_M\) multipliers of size \((3 \times r)\). With respect to the complexity of the AIFIR tail canceller, to perform filtering, \(N_F\) multipliers of size \((3 \times r)\) and \([N_G/2]\) multipliers of size \((r \times r)\) are required. To perform adaptation, \(N_F\) multipliers of size \((r \times r)\) and \([N_G/2]\) multipliers of size \((r \times r)\). Because of the simple binary nature of the received 2B1Q data, the multipliers of size \((3 \times r)\) can be replaced by shift/add operation (that is only assumed for the AIFIR filter). Consequently, the complexity of the main echo canceller and the tail echo canceller, respectively

\[
COMPLEXITY_{MEC} = 2N_M \ (3 \times r) \ multipliers \tag{4.19}
\]

\[
COMPLEXITY_{AIFIR} = N_F + [N_G/2] \ (r \times r) \ multipliers \tag{4.20}
\]
Table 4.1 compares the number of computations required by the AIFIR filter with that of an equivalent conventional AFIR filter estimating the same response.

<table>
<thead>
<tr>
<th>Method</th>
<th># of multipliers</th>
<th># of adders</th>
<th>Type of G(Z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFIR</td>
<td>2N</td>
<td>2(N - 1)</td>
<td>-</td>
</tr>
<tr>
<td>AIFIR</td>
<td>2(N/L + L)</td>
<td>2(N/L + L - 2)</td>
<td>RRS</td>
</tr>
<tr>
<td>AIFIR</td>
<td>2(N/L + N_G)</td>
<td>2(N/L + N_G - 2)</td>
<td>Optimal</td>
</tr>
</tbody>
</table>

Table 4.1: AFIR vs AIFIR, a complexity comparison.

### 4.2.2 Two Stage Echo Cancellation Utilizing AIFIR Filter

Figure 4.2 shows the structure of the proposed two-stage echo canceller. The first stage is an AFIR echo canceller that spans the first few baud intervals of the impulse response, and the second stage estimates the echo tail by employing an AIFIR filter as a Tail Echo Canceller. In chapter 5, we will present the simulation results of the system in Figure 4.2.

This subsection examines the interaction between the Main Echo Canceller (MEC) and the Tail Echo Canceller (TEC). The echo impulse response is separated into a main echo pulse covering the first few baud intervals and an echo tail. Thus, we can model the echo path response as

\[ d_n = \sum_{k=0}^{N_M-1} b_k x_{n-k} + \sum_{k=N_M}^{\infty} t_k x_{n-k} \]  

(4.21)

where the first \( N_M \) terms are implemented by the Main Echo Canceller. To find the optimum set of coefficients and the MSE, it is assumed the data\(^5\) \( x_n \) is uncorrelated with the noise and far-end data, \( s_n \) (see assumptions in section 2.3). The following vectors will be used in the analysis:

1. the main echo pulse vector \( \mathbf{B} = [b_0 \ b_1 \ \ldots \ b_{N_M-1}]^T \)

2. the tail echo pulse vector \( \mathbf{T} = [t_0 \ t_1 \ \ldots \ t_{N_t-2} \ t_{N_t-1} \ \ldots]^T \)

\(^5\) Also assumed to be uncorrelated among themselves
Figure 4.2: Block diagram of two-stage echo cancellation incorporating AIFIR filter as a tail canceller.
3. the Main Echo Canceller coefficient vector \( \mathbf{C} = [c_0, c_1, \ldots, c_{N_M-1}, n] \)

4. the Tail Echo Canceller coefficient vector \( \mathbf{H} = [h_0, h_1, \ldots, h_{N_t-1}, n] \)

Using the above definitions, the output of the two stage echo canceller is

\[
y_n = \sum_{k=0}^{N_M-1} c_k x_{n-k} + \sum_{k=N_M}^{N_t-1} h_k x_{n-k}
\]  \hspace{1cm} (4.22)

The received signal can be expressed as

\[
r_n = \mathbf{B}^T \mathbf{X} + \mathbf{T}^T \mathbf{X} + s_n + \ell_n
\]  \hspace{1cm} (4.23)

where \( (\ell_n = \sum_{k=0}^{N_t} h_k x_{n-k}) \) is the residual uncancelable echo corresponding to echo delays that exceed the number of coefficients in the echo canceller. The \( s_n \) is far-end signal plus noise.

Then, the error after subtracting the echo canceller output is

\[
\epsilon_n = (\mathbf{B} - \mathbf{C})^T \mathbf{X} + (\mathbf{T} - \mathbf{H})^T \mathbf{X} + s_n + \ell_n
\]  \hspace{1cm} (4.24)

\[
\epsilon_n = \mathbf{U}^T \mathbf{X} + \mathbf{V}^T \mathbf{X} + s_n + \ell_n
\]  \hspace{1cm} (4.25)

\( \mathbf{U} \) and \( \mathbf{V} \) are the coefficient error vectors. The expression for the MSE is

\[
E[\epsilon_n^2] = \sigma_e^2 E[\| \mathbf{U} \|^2] + \sigma_r^2 E[\| \mathbf{V} \|^2] + \sigma_s^2 + \sigma_r^2
\]  \hspace{1cm} (4.26)

The coefficients of the Main Echo Canceller and the Tail Echo Canceller are updated every symbol interval according to

\[
\mathbf{C}_{n+1} = \mathbf{C}_n + \mu \epsilon_n \mathbf{X}
\]  \hspace{1cm} (4.27)

\[
\mathbf{H}_{n+1} = \mathbf{H}_n + \alpha \epsilon_n \mathbf{M}
\]  \hspace{1cm} (4.28)
From Eq.(4.26) it can be seen that investigating the behavior of the MSE involves finding expression for $E[\|U_n\|^2]$ and $E[\|V_n\|^2]$. Analogous to the derivation in the previous analysis, the mean squared coefficient error vector for the Main echo Canceller and the Tail Echo Canceller are

$$E[\|U_{n+1}\|^2] = (1-2\mu\sigma_x^2 + \mu^2\sigma_x^4 N_M)E[\|U_n\|^2] + \mu^2\sigma_x^4 N_M E[\|V_n\|^2] + \mu^2\sigma_x^4 N_M(\sigma_x^2 + \sigma_i^2)$$

(4.29)

$$E[\|V_{n+1}\|^2] = (1-2\alpha\zeta\sigma_x^2 + \alpha^2\|G\|^2\sigma_x^4)E[\|V_n\|^2] + \alpha^2\|G\|^2\sigma_x^4 E[\|U_n\|^2] + \alpha^2\|G\|^2\sigma_x^2(\sigma_x^2 + \sigma_i^2)$$

(4.30)

where $\zeta$ is a sum of interpolator taps. Now, the problem becomes the same as that solved in [58]. To express Eq.(4.29) and Eq.(4.30) in matrix form, let

$$D_n = \begin{bmatrix} E[\|U_n\|^2] \\ E[\|V_n\|^2] \end{bmatrix}$$

(4.31)

$$K = \begin{bmatrix} \mu^2\sigma_x^2 N_M(\sigma_x^2 + \sigma_i^2) \\ \alpha^2\|G\|^2\sigma_x^4(\sigma_x^2 + \sigma_i^2) \end{bmatrix}$$

(4.32)

$$H = \begin{bmatrix} 1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4 N_M & \mu^2\sigma_x^4 N_M \\ \alpha^2\|G\|^2\sigma_x^4 & 1 - 2\alpha\zeta\sigma_x^2 + \alpha^2\|G\|^2\sigma_x^4 \end{bmatrix}$$

(4.33)

Then, Eq.(4.29) and Eq.(4.30) can be expressed as

$$D_{n+1} = HD_n + K$$

(4.34)

Eq.(4.34) can be iterated to yield

$$D_n = H^n D_0 + \sum_{i=0}^{n-1} H^i K$$

(4.35)
The first term of Eq. (4.35) is transient\(^6\) and will converge to zero if all eigenvalues of \(\mathbf{H}\) are within the unit circle [98]. Its rate of convergence is governed by the largest eigenvalue \(\lambda_{\text{max}}\). The second term is the steady state error of both cancellers and can be evaluated as

\[
\lim_{n \to \infty} D_n = \sum_{i=0}^{\infty} \mathbf{H}^i \mathbf{K} = (\mathbf{I} - \mathbf{H})^{-1} \mathbf{K}
\]  

(4.36)

Therefore, the total steady state error (SSE) is

\[
\text{SSE} = \frac{\mu N_M + \alpha \rho}{2 - \sigma_e^2(\mu N_M + \alpha \rho)} (\sigma_e^2 + \sigma_i^2)
\]  

(4.37)

where \(\rho \doteq \|G\|^2 / \zeta\). Then the steady state MSE is

\[
\text{MSE} = (1 + \frac{\sigma_e^2(\mu N_M + \alpha \rho)}{2 - \sigma_e^2(\mu N_M + \alpha \rho)}) (\sigma_e^2 + \sigma_i^2)
\]  

(4.38)

The relative values of the step-size parameters \(\mu\) and \(\alpha\) should be chosen to make the maximum eigenvalue \(\lambda_{\text{max}}\) of \(\mathbf{H}\) as small as possible. Consequently, convergence rate is maximized. The fastest convergence is obtained by choosing \(\mu\) and \(\alpha\) as [58]

\[
\mu = \alpha = \frac{1}{\sigma_e^2 (N_M + \rho)}
\]  

(4.39)

That causes the highest MSE. Lower MSE can be achieved at the price of slower convergence by selecting smaller values of \(\mu\) and \(\alpha\).

We now consider the number of bits required for the tap representation of the AFIR echo canceller. In [58], it has been shown that the tap wordlength can be found as

\[
2^{W_e - 1} \geq \frac{\| U_0 \|}{\mu \sigma_a}
\]  

(4.40)

\(^{6}\)It is evaluated as a series of terms involving the \(n\)th power of the eigenvalues of \(\mathbf{H}\) [58].
\[ W_c = \text{tap wordlength in bits} \]

\[ \| U_0 \| = \text{rms value near-end signal} \]

\[ \sigma_s = \text{rms value far-end signal} \]

\[ \mu = \text{step-size} \]

In a typical worst case situation, where the ratio of near-end to far-end signal power \( \frac{\| U_0 \|}{\sigma_s} = 40dB \), the tap wordlength is calculated to be 16 bits (\( \mu = 5.6 \times 10^{-3} \)). It has been shown in [90] that the taps of IFIR filter required 1-3 bits less than the equivalent FIR filter. Therefore, we anticipate that the AIFIR filter will have a tap wordlength close to that of the AFIR filter.

### 4.3 Conclusion

The basic intent of this research is to develop adaptive thinned FIR filter structures to reduce the complexity of the echo canceller. In this chapter, we proposed a new architecture for the echo tail canceller. The proposed tail canceller is an adaptive version of the interpolated FIR filter introduced in [88]. The general theory of the AIFIR structure has been derived. The analysis of the proposed two-stage echo canceller was developed. It was shown that the AIFIR structure is very efficient in estimating the echo tail narrow-band response requiring significantly fewer multipliers and adders compared to the direct form adaptive FIR filter. The suggested AIFIR filter also maintains simpler calculations required by the direct form AFIR and requires only the basic FIR computations. In chapter 5, extensive simulations will be performed to
show that 60 dB echo cancellation performance is attained by the proposed two-stage echo canceller at highly reduced computational complexity.
Chapter 5
Simulations and Results

In this chapter, the proposed two-stage echo canceller is simulated. The first stage is simulated as an AFIR filter to cancel the main echo part. The second stage is simulated as a cascade of an adaptive sparse filter and a non-adaptive interpolator to cancel the tail section. All simulation programs were written in C and compiled using the VM/CMS Waterloo C compiler on the mainframe computer. The DFDP package [105] was used to design the interpolator filters. The essential details and definitions pertaining to the system simulations are explained in section 5.1. In order to have a wide sample of nonloaded\(^1\) subscriber loop lines (see Appendix A for details on these loops), echo path impulse responses were numerically generated\(^2\) for subscriber loops with various configurations. The AIFIR echo tail canceller design method is outlined in section 5.2 where special emphasis is placed on the loop echo tail frequency response. In section 5.3, extensive simulations were carried out to investigate the performance of the overall echo canceller. These investigations were primarily on the effects of parameters such as the passband edge\(^3\) frequency \(f_p\) and stopband ripple \(\delta_s\) of the interpolator, as well as the length of the first stage echo canceller \(N_M\) on the echo canceller performance. The first two parameters affect the design of the interpolator/tail canceller. It will be seen later that some selections of these parameters

---
\(^1\)Those lines not containing inductive coils make up 77% of the loop population.
\(^2\)For example, loops such as \#2, \#5, \#6, \#7, \#11, \#12, and \#13. Loop\#9 was supplied by Mr. G. Davidson. Loop\#10, and \#9 were supplied by Dr. S. Aly.
\(^3\)It is the actual stopband edge frequency of the tail frequency response.
resulted in more complex interpolators than necessary. Investigation #1 was carried out to study the favorable/adverse effects of the main canceller length \( N_M \) on the system performance. The objective of investigation #2 is to understand the effect of the interpolator passband frequency \( f_p \) on the system performance/complexity. Since the first two studies were based on the utilization of an interpolator specifically designed for the loop used, generalization of the interpolator section was essential. Thus, the effects of employing a compromise interpolator for all loops was considered in investigation #3. Finally, the effects of the stopband ripple \( \delta_s \) on the overall system performance were determined in investigation #4. The proposed AIFIR filter approach was compared with the echo tail canceller of Davidson and Falconer [100] in section 5.4. The results show that the suggested tail canceller is quite efficient. This proposed tail canceller approximately requires 27% of the hardware of the AIFIR canceller. Concluding remarks on the chapter are given in section 5.5.

5.1 Simulation Details

The simulation results for the two-stage echo canceller (shown in Figure 4.2) are reported here. A number of assumptions and details ought to be pointed out before presentation of the results. In order to ensure that adequate echo cancellation is attained from the proposed echo canceller under different line arrangements, our investigations cover the effects of loops of different lengths, number of bridged taps, and line gauges. We used a program that was developed by Dr. D. D. Falconer (Carleton University) to generate the impulse response for echo paths of any desired loop configuration including the effects of bridged taps, the transformer, and the hybrid. It generates the near end echo in two stages [99]:

1. The program finds the transfer function of the echo path at various points in frequency, up to half the sampling frequency\(^4\). This transfer function is

\(^4\)The used sampling frequency is 640 kHz (8 times the baud rate).
multiplied with the transmitted pulse transfer function (100% raised cosine transmit filter was used).

2. The echo path impulse response is obtained using an inverse FFT. This impulse response contains eight samples per baud. Our simulations use one sample per baud (i.e., 80 kbaud sampling rate). Thus, the echo path impulse response is decimated to 80 kbaud sampling rate.

These echo responses are the input signals to be cancelled by the suggested echo canceller. Some echo path impulse responses are shown in Figure 5.1 to Figure 5.3. From these figures, it can be observed that the echo tail response, eventually decays monotonically in an exponential manner, and lasts\(^5\) several hundred samples (see section 4.1 and [69]). Moreover, the echo tail responses possess the same general shape for most loops considered. From these figures, it appears that the echo path impulse response has three different regions:

- A first region in which the echo response changes rapidly (wide-band response).
- A second region in which the echo response has a *pseudo exponential* nature.
- A third region in which the echo response can be approximated as an exponential function(s).

In order to obtain adequate performance, an FIR representation of the coarse region main echo is a necessity.

### 5.1.1 Echo Canceller Length Requirement

The purpose of this subsection is to determine the minimum length of an echo canceller that is required to meet a specified Echo Cancellation Accuracy (ECA). To find the length requirement\(^6\) of an echo canceller for a given Echo Cancellation Accuracy

---

\(^5\)Echo tail lasts over a thousand sample for 320 kbaud sampling rate [100].

\(^6\)i.e., how long must the echo canceller spans to achieve the required cancellation accuracy?
Figure 5.1: Echo path impulse response of Loop #2.
Figure 5.2: Echo path impulse response of Loop#6.
Figure 5.3: Echo path impulse response of Loop #9.
(ECA), we determine, the ratio of the residual uncancelable echo\(^7\) to total echo on
an energy basis. This ratio\(^8\) is represented as

\[
\frac{\text{Residual Echo Energy}}{\text{Total Echo Energy}} \leq \text{ECA}
\] (5.1)

*In this thesis 60 dB echo cancellation accuracy is used unless otherwise stated.*

To attain the 60 dB echo cancellation accuracy, the residual echo energy must
be reduced below 60 dB compared to the total echo energy. Thus, we can find the
number of echo canceller taps needed for a given echo cancellation accuracy. It is
interesting to report that for all the loops considered, a careful look at the impulse
response showed that these responses possess numerical oscillations\(^9\) after about the
420\(^{th}\) sample. In practice the response of the echo path will not have these numerical
oscillations which are partly caused by the FFT accuracy\(^{10}\).

**Performance Measures**

Since the far end data is attenuated by the channel, a reduction in echo power of
as much as 60 dB is required for an SNR of 20 dB. With this criterion in mind, the
measure of echo cancellation performance\(^{11}\) (ECP) is defined as

\[
\text{ECP} = 10 \log_{10} \frac{\text{Actual Echo Energy Before Cancellation}}{\text{Residual Echo Energy After Cancellation}} \text{ dB}
\] (5.2)

For the first stage echo canceller, it can be defined as [103], [103], [104]

---

\(^7\)It is the echo part that exceeds the echo canceller tap length.

\(^8\)If this ratio is not satisfied, then the energy of uncancel\(\text{ed}\) echo will compromise the performance of any echo canceller.

\(^9\)Except Loop\#9.

\(^{10}\)A Radix Two FFT was used in the loop generating program to go from the frequency domain
to the time domain. The FFT assumes that a periodic function is being transformed and that the
function is bandlimited to 320 kHz. The truncation of the echo transfer function will cause numerical
oscillations in the echo path impulse response.

\(^{11}\)It means the degree of cancellation achieved by the echo canceller (i.e., how much of the near-end
echo power is canceled?).
\[ ECP_1 = 10 \log_{10} \frac{\sum_{i=0}^{N_i-1} b_i^2}{\sum_{i=0}^{N_i-1} (b_i - c_i)^2} \text{ dB} \] (5.3)

where \( \sum_{i=0}^{N_i-1} b_i^2 \) is the actual echo energy of the echo path impulse response before cancellation. For the second stage (tail canceller), if we assume perfect cancellation by the first stage canceller, the echo cancellation performance (ECP) can be defined as

\[ ECP_2 = 10 \log_{10} \frac{\sum_{i=0}^{N_i-1} b_i^2}{\sum_{i=0}^{N_i-1} (t_i - h_i)^2} \text{ dB} \] (5.4)

The above equations will be used in our investigations as performance merit for the echo canceller.

### 5.2 AIFIR Echo Tail C canceller: Design

The design of the AIFIR Echo Tail C canceller is based on the frequency response of the echo tail to be estimated. Due to the fact that the design of any IIR filter is performed in frequency domain (see chapter 3), a study of the tail frequency response for all considered loops is essential. Next, the frequency response of the echo tail for Loop#2, Loop#5, Loop#6, Loop#7, Loop#9, Loop#11, Loop#12, and Loop#13 is produced via an FFT. An interpolator is then designed for optimum performance on a given loop. Later, we will show how to design a compromise interpolator for 60 dB ECP performance on all loops considered.

#### 5.2.1 Echo Tail Frequency Response

The echo path impulse response was separated into main echo pulse and echo tail. An FFT of the tail echo pulse was performed to obtain the frequency response of the tail section of Loop#2, Loop#5, Loop#6, Loop#7, Loop#9, Loop#11, Loop#12, and Loop#13. Those frequency responses are illustrated in Figure 5.4 to Figure 5.11.
A parameter such as the passband frequency\textsuperscript{12} $f_p$ of the interpolator $G(Z)$ is selected from the frequency responses to be used in the design of AIFIR filter. This selection is based on a criterion of 22 dB ratio for the highest point in the amplitude response and the point corresponding to $f_p$. Table 5.1 lists the passband frequencies $f_p$ (in mHz, normalized) as a first trial. The selection of these passband frequencies is based on the tail spectrum of the above loops. It will be seen later that this selection resulted in more complex (i.e., higher order) interpolators than necessary.

<table>
<thead>
<tr>
<th>Loop#</th>
<th>$f_p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>15.62 mHz</td>
</tr>
<tr>
<td>5</td>
<td>15.00 mHz</td>
</tr>
<tr>
<td>6</td>
<td>13.67 mHz</td>
</tr>
<tr>
<td>7</td>
<td>14.65 mHz</td>
</tr>
<tr>
<td>9</td>
<td>11.72 mHz</td>
</tr>
<tr>
<td>11</td>
<td>16.60 mHz</td>
</tr>
<tr>
<td>12</td>
<td>15.43 mHz</td>
</tr>
<tr>
<td>13</td>
<td>15.23 mHz</td>
</tr>
</tbody>
</table>

Table 5.1: The first selection of passband frequencies $f_p$.

\textbf{5.2.2 AIFIR Design Steps}

The AIFIR filter is composed of the sparse filter $F(Z^L)$ and the interpolator filter $G(Z)$. The sparse filter $F(Z^L)$ is adapted so that the overall frequency response tracks the echo tail frequency response. The interpolation factor $L$ determines the number of non-zero taps of the sparse filter. The interpolator is then designed as a multi-stopband FIR filter. Its passband is selected on the basis of the actual stopband edge $f_s$ of the tail frequency response (Table 5.1). The design process of the AIFIR filter is summarized next:

1. From the given tail stopband edge frequency $f_s$ (i.e., the selected interpolator passband edge), we determine $L_{\text{max}} = \lfloor 1/2f_s \rfloor$. In this thesis, $L$ is selected\textsuperscript{13}

\textsuperscript{12}The passband width of the interpolator is chosen to equal the stopband frequency $f_s$ of the tail spectrum.

\textsuperscript{13}This selection satisfies the condition $L < L_{\text{max}}(= 30)$ for no spectral aliasing.
Figure 5.4: An exploded view of Loop#2 tail frequency response.
Figure 5.5: An exploded view of Loop#5 tail frequency response.
Figure 5.6: An exploded view of Loop#6 tail frequency response.
Figure 5.7: An exploded view of Loop#7 tail frequency response.
Figure 5.8: An exploded view of Loop#9 tail frequency response.
Figure 5.9: An exploded view of Loop#11 tail frequency response.
Figure 5.10: An exploded view of Loop#12 tail frequency response.
Figure 5.11: An exploded view of Loop#13 tail frequency response.
2. The sparse filter \( F(Z^L) \) is implemented as an adaptive FIR filter with \( L \) delays between the coefficients. These coefficients are initialized to zero value at the start of the adaptation. The length \( N_F \) of the sparse filter (i.e., the number of non-zero coefficients) is estimated using Eq.(4.16).

3. For the specified \( L \), the positions of the undesired spectral images of the sparse filter spectrum are determined from Eq.(5.6). We design the interpolator filter \( G(Z) \) to suppress these undesired spectral images to or below the specified stop-band attenuation\(^{14}\). The interpolator \( G(Z) \) is designed as a multistop band FIR filter having passband matching the wanted response of \( F(Z^L) \) and stopbands overlapping the unwanted replicas of the sparse filter spectrum. Thus, the overall cascade of the sparse filter \( F(Z^L) \) and the interpolator \( G(Z) \) approximates the echo tail spectrum.

The interpolator \( G(Z) \) specifications used in our simulations are

\[
\text{Passband} \in [0, f_p] \tag{5.5}
\]

\[
\text{Stopbands} \in [K/L - f_p, K/L + f_p], \ K = 1, 2, \ldots, 14. \tag{5.6}
\]

\[
\text{Amplitude gain} = 1 \tag{5.7}
\]

\[
\text{Interpolator order} = N_F; \tag{5.8}
\]

\[
\text{Passband ripple} = \delta_p \tag{5.9}
\]

\(^{14}\)It will be seen later that the stopband level affected the echo cancellation performance.
\[ \text{Stopband ripple} = \delta_s \]  

(5.10)

We start the design of the AIFIR filter such that each loop has a particular \( G(Z) \) tailored for it. The following is the design of \( G(Z) \) based on loop\#2 tail frequency response. The chosen interpolation factor \( L \) is selected to be equal to 15. As an example, the design method of the interpolator filter based on the frequency response of loop\#2 is given here. The passband edge frequency \( f_p \) (from Table 5.1) is equal to 15.62 mHz. Thus, the specifications of this particular \( G(Z) \) are listed in Table 5.2. The interpolator is designed using Parks-McClellan program in the DFDP. It was found through extensive simulations (to be reported later) that the degree of attenuation of the stopbands should be at least greater than 32 dB to attain the required cancellation accuracy. With this criterion in mind, the interpolator order found by the DFDP was 46 taps (i.e., 23 actual multipliers).

<table>
<thead>
<tr>
<th>Passband edge</th>
<th>( \rightarrow ) ( f_p ) mHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stopband#1</td>
<td>51.05 ( \rightarrow ) 82.29 mHz</td>
</tr>
<tr>
<td>Stopband#2</td>
<td>117.71 ( \rightarrow ) 148.95 mHz</td>
</tr>
<tr>
<td>Stopband#3</td>
<td>184.38 ( \rightarrow ) 215.62 mHz</td>
</tr>
<tr>
<td>Stopband#4</td>
<td>251.05 ( \rightarrow ) 282.29 mHz</td>
</tr>
<tr>
<td>Stopband#5</td>
<td>317.71 ( \rightarrow ) 348.95 mHz</td>
</tr>
<tr>
<td>Stopband#6</td>
<td>384.38 ( \rightarrow ) 415.62 mHz</td>
</tr>
<tr>
<td>Stopband#7</td>
<td>451.05 ( \rightarrow ) 482.29 mHz</td>
</tr>
</tbody>
</table>

Table 5.2: The passband and stopband widths of Loop\#2 interpolator, \( L = 15 \), sampling frequency normalized to 1 Hz.

### 5.3 Simulation Set-Up

Simulations were carried out to investigate the performance of the suggested two-stage echo canceller. Effects of parameters such as the passband edge frequency \( f_p \), stopband ripple \( \delta_s \), and \( N_M \) on performance will be studied and reported. We will also investigate whether or not adequate echo cancellation performance can be attained
if a *compromise* interpolator is used for all loops. The system depicted in Figure 4.2 was simulated as follows:

1. An adaptive FIR filter with variable length $N_M$ was simulated as the Main Echo Canceller. This canceller is used to cancel the main echo pulse of the echo path impulse response.

2. An adaptive IFIR filter based on the method of section 5.2 was simulated as the Tail Echo Canceller. The first cascade (sparse filter $F(Z^L)$) is being adapted so that the overall IFIR response tracks the tail response and the second cascade (interpolator) is kept fixed (i.e., non-adaptive). The adaptation of the echo canceller is being done in presence of the far-end data and noise. A noise source with variance $\sigma_n^2 = 10^{-11}$ was added to the output of the hybrid. The hybrid was simulated as a 420 sample FIR filter with taps equal to the baud rate sampled echo path impulse response of any loop considered. Pseudo random noise generated sequence was employed as an input to the two-stage echo canceller and the simulated echo path. A common error signal is used for joint adaptation of the Main and Tail Echo Canceller (see Figure 4.2).

### 5.3.1 Investigation#1: Effects of Main Canceller Length

The purpose of this simulation is to study the effects of the Main Echo Canceller (MEC) length $N_M$ on the echo cancellation performance. As stated earlier, 60 dB of echo cancellation is required from the two-stage canceller. The echo path impulse response is divided into main echo pulse of length $N_M$ and echo tail pulse of length $N_T$. The selection of the time instant ($N_M$) at which the echo path impulse response is separated into main echo pulse and tail echo pulse should be made on the basis of

---

15Initially, all the taps are set equal to zero.

16The difference between the far-end and near-end noise source is the use of different seeds in the noise generators (i.e., to obtain uncorrelated processes).
minimizing the complexity of the overall system and will be found through simulations. Since \((N_M + N_t)\) is fixed at the length of the loop response, then increasing \(N_M\) automatically decreases the potential for savings in the tail canceler. At the same time, decreasing \(N_M\) below some limit will result in attempting to cancel the main echo using the AIFIR section. This obviously will result in significant degradation of performance. In this section we will study the effects of \(N_M\) on the overall echo canceller performance. To isolate this effect only, we designed the interpolator to give best performance on a given loop (i.e., specific interpolator/loop). For stopband attenuation \(\delta_s \geq 32 \, dB\), the DFDP program resulted in \(N_G\) equal to 46 taps for all interpolators. For \(L\) equals to 15 for all the loops, the sparse filter order is 22 non-zero coefficients (from Eq.(4.16)). The locations of the spectral images of the response of the sparse filter are determined and attenuated by \(G(Z)\) (e.g., see Table 5.2). Thus, an interpolated response of 361 samples (from Eq.(3.27)) is generated by the AIFIR filter to cancel the tail pulse. In order to implement the AIFIR tail canceller, 68 taps (from Eq.(4.17)) are required. Results for some of the Bellcore lines using the two-stage echo cancellation (with various lengths of \(N_M\)) are given in Table 5.3 to Table 5.10. Also, included in these tables are attributes such as the two-stage echo canceller Total Length and the COMPLEXITY MEC of the main echo canceller (from Eq.(4.19)). The COMPLEXITY AIFIR of the tail echo canceller (from Eq.(4.20)) is \(45 \, r \times r\) multipliers. For the loops considered here, the echo cancellation performance (ECP) as a function of \(N_M\) is depicted in Figure 5.12 to Figure 5.19 clarifying the effects of increasing/decreasing the length of the first stage canceller \(N_M\) (i.e., X-axis) for unchanged interpolator.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_c$ in dB</th>
<th>$\left\lceil N_G/2 \right\rceil$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY_MEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>56.29</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>59.17</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>60.01</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>61.13</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>62.73</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>64.20</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>65.63</td>
<td>35.05</td>
<td>23</td>
<td>15.62</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.3: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#2. The main canceller multiplier size is $3 \times r$. The COMPLEXITY_MEC of the tail echo canceller is $45 \times r$ multipliers.

![Graph](image)

Figure 5.12: A graph to illustrate the effects of Main Echo Cancellation length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#2.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>N_M</th>
<th>N_r</th>
<th>ECP in dB</th>
<th>δ_0 in dB</th>
<th>[N_G/2]</th>
<th>f_p in mHz</th>
<th>COMPLEXITY_MECH</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>57.16</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>60.86</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>62.46</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>62.99</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>64.05</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>65.68</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>66.17</td>
<td>36.12</td>
<td>23</td>
<td>15.00</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.4: Echo Cancellation Performance (ECP) vs N_M (specific G(Z)): Loop#5. The main canceller multiplier size is 3 × r. The COMPLEXITY_MECH of the tail echo canceller is 45 r × r multipliers.

Figure 5.13: A graph to illustrate the effects of Main Echo C canceller length N_M on performance. The same G(Z) is used for each value of N_M. As N_M increases echo cancellation performance improves: Loop#5.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_i$</th>
<th>ECP $\text{in dB}$</th>
<th>$\delta_s$ $\text{in dB}$</th>
<th>$\frac{N_G}{2}$</th>
<th>$f_p$ $\text{in mHz}$</th>
<th>COMPLEXITY$_{MEC}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>56.48</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>58.89</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>60.13</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>60.25</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>62.39</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>64.95</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>66.53</td>
<td>38.72</td>
<td>23</td>
<td>13.67</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.5: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#6. The main canceller multiplier size is $3 \times r$. The COMPLEXITY$_{AIFIR}$ of the tail echo canceller is $45 \times r$ multipliers.

![Graph](image)

Figure 5.14: A graph to illustrate the effects of Main Echo Cancellation length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#6.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_\gamma$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY$_{MEC}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>57.79</td>
<td>36.85</td>
<td>29</td>
<td>14.65</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>58.45</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>61.13</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>61.67</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>62.89</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>64.03</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>65.07</td>
<td>36.85</td>
<td>22</td>
<td>14.65</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.6: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#7. The main canceller multiplier size is $3 \times r$. The COMPLEXITY$_{MEC}$ of the tail echo canceller is $45 \times r$ multipliers.

Figure 5.15: A graph to illustrate the effects of Main Echo Cancellation length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#7.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_{ir}$</th>
<th>ECP in dB</th>
<th>$\delta_s$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY$_{MEC}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>58.09</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>61.74</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>62.23</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>62.36</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>64.86</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>67.85</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>68.01</td>
<td>42.79</td>
<td>23</td>
<td>11.72</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.7: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#9. The main canceller multiplier size is $3 \times r$. The COMPLEXITY$_{AIFIR}$ of the tail echo canceller is $45\ r \times r$ multipliers.

![Graph](image)

Figure 5.16: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#9.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_s$ in dB</th>
<th>$[N_g/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY&lt;sub&gt;MIC&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>57.24</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>59.31</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>60.56</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>61.28</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>63.76</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>64</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>64.98</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>66.09</td>
<td>39.35</td>
<td>23</td>
<td>16.60</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.8: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#11. The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{ALFIR}$ of the tail echo canceller is $45 \times r$ multipliers.

Figure 5.17: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves: Loop#11.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$f_p$ in mHz</th>
<th>$N_g/2$</th>
<th>COMPLEXITY</th>
<th>MEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>56.90</td>
<td>15.43</td>
<td>23</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>60.04</td>
<td>15.43</td>
<td>23</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>61.95</td>
<td>15.43</td>
<td>23</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>62.91</td>
<td>15.43</td>
<td>23</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>65.08</td>
<td>15.43</td>
<td>23</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>66.59</td>
<td>15.43</td>
<td>23</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>67.24</td>
<td>15.43</td>
<td>23</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.9: Echo Cancellation Performance (ECP) vs $N_M$ (specific $G(Z)$): Loop#12. The main canceller multiplier size is $3 \times r$. The COMPLEXITY$_{MEC}$ of the tail echo canceller is $45 \times r$ multipliers.

Figure 5.18: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance. The same $G(Z)$ is used for each value of $N_M$. As $N_M$ increases echo cancellation performance improves; Loop#12.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>( N_M )</th>
<th>( N_r )</th>
<th>ECP in dB</th>
<th>( \delta_s ) in dB</th>
<th>([N_G/2])</th>
<th>( f_p ) in mHz</th>
<th>COMPLEXITY_{MEC}</th>
</tr>
</thead>
<tbody>
<tr>
<td>371</td>
<td>10</td>
<td>361</td>
<td>57.81</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>20</td>
</tr>
<tr>
<td>376</td>
<td>15</td>
<td>361</td>
<td>59.45</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>30</td>
</tr>
<tr>
<td>381</td>
<td>20</td>
<td>361</td>
<td>61.17</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>40</td>
</tr>
<tr>
<td>386</td>
<td>25</td>
<td>361</td>
<td>62.39</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>50</td>
</tr>
<tr>
<td>393</td>
<td>32</td>
<td>361</td>
<td>64.12</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>60</td>
</tr>
<tr>
<td>401</td>
<td>40</td>
<td>361</td>
<td>65.88</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>80</td>
</tr>
<tr>
<td>411</td>
<td>50</td>
<td>361</td>
<td>67.09</td>
<td>35.75</td>
<td>23</td>
<td>15.23</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.10: Echo Cancellation Performance (ECP) vs \( N_M \) (specific \( G(Z) \)): Loop#13. The main canceller multiplier size is \( 3 \times r \). The \( COMPLEXITY_{AIFIR} \) of the tail echo canceller is \( 45 \times r \) multipliers.

![Graph](image.png)

Figure 5.19: A graph to illustrate the effects of Main Echo C canceller length \( N_M \) on performance. The same \( G(Z) \) is used for each value of \( N_M \). As \( N_M \) increases echo cancellation performance improves: Loop#13.
The conclusion based on these results is that the echo cancellation performance of the simulated system improves as the length of the Main Echo Canceller $N_M$ increases. This is clearly illustrated in Figure 5.12 to Figure 5.19. This is due to the complete cancellation of the irregular section and the pseudo exponential region of the echo path by an AIFIR filter. As a result, the tail canceller is applied to a totally exponential region. The AIFIR filter performs best when applied to exponential region. The price paid for this improvement in performance is the increase in complexity (e.g., see last column of Table 5.3 to Table 5.10). The $COMPLEXITY_{AIFIR}$ of the tail echo canceller (from Eq.(4.20)) is $45 \times r$ multipliers since the interpolator parameters are fixed. For a given $N_M$, the complexity of the first stage canceller is calculated from Eq.(4.19) and included in those tables. The higher the complexity, the higher the performance. Obviously, the complexity of the two stage echo canceller increases dramatically as the order of the first stage echo canceller increases. For an exact cancellation of all the main pulse samples, the length of the first stage echo canceller $N_M$ must equal the length of the main pulse. Therefore, for a minimum complexity two-stage echo canceller, the order of the first stage canceller is kept as low as possible while still meeting the 60 db echo cancellation accuracy. From these results, it can be observed that the 60 dB echo cancellation requirement was attained when $N_M$ is greater or equal to 20 taps. Moreover, for $N_M = 10$, an increase of $N_M$ by 10 more taps results in an improvement in echo cancellation performance of at least 3 dB. For $N_M = 20$, an improvement in echo cancellation performance of at least 2 dB is gained by increasing $N_M$ by 12 more taps. From Figure 5.12 to Figure 5.19, it is seen that the increase in the main canceller length $N_M$ has less significant effect on the performance of the overall canceller when $N_M$ is greater than or equal to 40.

17. Thus, to minimize system complexity, there is no need to increase $N_M$ beyond 20 taps.
18. Except Loop#7
5.3.2 Investigation#2: Effects of Interpolator Passband $f_p$

In order to explore the limitations and performance improvement/degradation of the proposed system, a study of the effects of using various values of the passband edge frequency $f_p$ is vital. For $L$ equals to 15 for all the loops, the sparse filter order is 22 non-zero coefficients (from Eq.(4.16)). From the previous investigation, $N_M$ is equal to 20 taps. For a given loop and a given value of $f_p$, an interpolator was designed and the various values of $N_G$ and the stopband attenuation $\delta_s$ resulted from the DFDP are included in Table 5.11 to Table 5.18. Only the values of $N_G$ up to 54 were considered, therefore if more than this resulted from the DFDP the entry is left blank. In this study, an interpolator was designed for various values of passband edge frequency $f_p$ for a given loop. The complete canceller was then simulated. The following observations apply to the results in Table 5.11 to Table 5.18. Firstly, it can be noted that if the value of $f_p$ increases, the stopband regions of the interpolator increase in width. Also, the transition bandwidth of the interpolator decreases. As a result, the order of the interpolator $N_G$ increases. The opposite is also true. Consequently, a substantial decrease in the length of the interpolator is clearly observed. The same stopband attenuation that was used in investigation#1 is also used here. Therefore, the effects of changing $f_p$ on the performance can be singled out. To show how the echo cancellation performance varies with the changes of $f_p$ for a given $N_M$, results of the echo canceller performance (with various values of $f_p$) are tabulated in Table 5.11 to Table 5.18. From these results, we note the following:

1. In all cases, 60 dB echo cancellation performance was attained.

2. As $f_p$ decreases, the echo cancellation performance improves until a certain $f_p$ (deflection point) at which the system performance is degraded. Figure 5.20 to Figure 5.27 show echo cancellation performance versus interpolator passband edge frequency $f_p$ for each of the eight considered lines, for $N_M = 20$. It was also
seen that the best echo canceller performance is obtained with a fairly narrow passband frequency \( f_p \) (near the *deflection point*). It is seen that the *deflection point* is between 8 and 10 mHz.

More complex interpolators were used in investigation\#1 (their design were based on Table 5.1). According to the above results, it can be seen that interpolators designed based on Table 5.1 are of higher order than necessary. Lower order and thus less complex interpolators can be designed by careful selection of the \( f_p \) parameter. In the next part of the investigation, it is desired to find effects of the main canceller length on the system performance when these efficient interpolators are utilized. Therefore, simulations of investigation\#1 should be repeated with these better interpolators. Also, the findings of investigation\#2 can facilitate the design of a *compromise* interpolator to be used on all lines. Echo cancellation performance versus interpolator passband edge frequency \( f_p \) is plotted to aid in the selection of one *compromise* interpolator.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_G/2]$</th>
<th>$\delta_v$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.53</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>15.63</td>
<td>60.91</td>
<td>20</td>
<td>23</td>
<td>35.05</td>
</tr>
<tr>
<td>13.67</td>
<td>62.44</td>
<td>20</td>
<td>21</td>
<td>34.62</td>
</tr>
<tr>
<td>11.72</td>
<td>63.18</td>
<td>20</td>
<td>20</td>
<td>35.34</td>
</tr>
<tr>
<td>9.76</td>
<td>63.95</td>
<td>20</td>
<td>18</td>
<td>35.88</td>
</tr>
<tr>
<td>7.81</td>
<td>64.30</td>
<td>20</td>
<td>13</td>
<td>37.26</td>
</tr>
<tr>
<td>6.84</td>
<td>63.81</td>
<td>20</td>
<td>13</td>
<td>35.99</td>
</tr>
</tbody>
</table>

Table 5.11: Echo Cancellation Performance (ECP) vs $f_p$: Loop#2.

Figure 5.20: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#2. For each passband edge, a specific $G(Z)$ is designed.
Table 5.12: Echo Cancellation Performance (ECP) vs $f_p$: Loop#5.

<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_G/2]$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.53</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>16.95</td>
<td>61.47</td>
<td>20</td>
<td>25</td>
<td>36.00</td>
</tr>
<tr>
<td>15.00</td>
<td>62.60</td>
<td>20</td>
<td>23</td>
<td>36.00</td>
</tr>
<tr>
<td>13.05</td>
<td>63.69</td>
<td>20</td>
<td>21</td>
<td>35.75</td>
</tr>
<tr>
<td>11.09</td>
<td>64.11</td>
<td>20</td>
<td>20</td>
<td>35.66</td>
</tr>
<tr>
<td>9.14</td>
<td>64.91</td>
<td>20</td>
<td>23</td>
<td>36.14</td>
</tr>
<tr>
<td>7.19</td>
<td>62.09</td>
<td>20</td>
<td>12</td>
<td>35.51</td>
</tr>
</tbody>
</table>

Figure 5.21: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#5. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$\lfloor N_G/2 \rfloor$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.53</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>15.62</td>
<td>62.74</td>
<td>20</td>
<td>29</td>
<td>35.04</td>
</tr>
<tr>
<td>13.67</td>
<td>65.05</td>
<td>20</td>
<td>21</td>
<td>34.59</td>
</tr>
<tr>
<td>11.72</td>
<td>64.29</td>
<td>20</td>
<td>20</td>
<td>35.87</td>
</tr>
<tr>
<td>9.76</td>
<td>64.93</td>
<td>20</td>
<td>17</td>
<td>34.41</td>
</tr>
<tr>
<td>7.81</td>
<td>65.17</td>
<td>20</td>
<td>13</td>
<td>34.13</td>
</tr>
<tr>
<td>5.86</td>
<td>65.90</td>
<td>20</td>
<td>12</td>
<td>35.99</td>
</tr>
</tbody>
</table>

Table 5.13: Echo Cancellation Performance (ECP) vs $f_p$: Loop#6.

Figure 5.22: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#6. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_G/2]$</th>
<th>$\delta$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.00</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>16.60</td>
<td>60.92</td>
<td>20</td>
<td>24</td>
<td>35.67</td>
</tr>
<tr>
<td>14.65</td>
<td>61.37</td>
<td>20</td>
<td>22</td>
<td>34.71</td>
</tr>
<tr>
<td>12.70</td>
<td>61.59</td>
<td>20</td>
<td>21</td>
<td>34.21</td>
</tr>
<tr>
<td>10.74</td>
<td>62.47</td>
<td>20</td>
<td>19</td>
<td>34.09</td>
</tr>
<tr>
<td>8.79</td>
<td>63.68</td>
<td>20</td>
<td>13</td>
<td>36.04</td>
</tr>
<tr>
<td>6.84</td>
<td>62.09</td>
<td>20</td>
<td>12</td>
<td>36.00</td>
</tr>
</tbody>
</table>

Table 5.14: Echo Cancellation Performance (ECP) vs $f_p$: Loop#7.

Figure 5.23: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#7. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$\lceil N_G/2 \rceil$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.59</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>15.62</td>
<td>61.05</td>
<td>20</td>
<td>23</td>
<td>35.05</td>
</tr>
<tr>
<td>13.67</td>
<td>62.87</td>
<td>20</td>
<td>21</td>
<td>34.69</td>
</tr>
<tr>
<td>11.79</td>
<td>63.48</td>
<td>20</td>
<td>20</td>
<td>35.53</td>
</tr>
<tr>
<td>9.79</td>
<td>64.89</td>
<td>20</td>
<td>15</td>
<td>34.91</td>
</tr>
<tr>
<td>7.81</td>
<td>65.13</td>
<td>20</td>
<td>12</td>
<td>34.69</td>
</tr>
<tr>
<td>5.86</td>
<td>63.27</td>
<td>20</td>
<td>12</td>
<td>35.30</td>
</tr>
</tbody>
</table>

Table 5.15: Echo Cancellation Performance (ECP) vs $f_p$: Loop#9.

Figure 5.24: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#9. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_M/2]$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.23</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>16.60</td>
<td>60.72</td>
<td>20</td>
<td>24</td>
<td>35.63</td>
</tr>
<tr>
<td>14.65</td>
<td>62.13</td>
<td>20</td>
<td>22</td>
<td>34.75</td>
</tr>
<tr>
<td>12.70</td>
<td>62.49</td>
<td>20</td>
<td>21</td>
<td>33.72</td>
</tr>
<tr>
<td>10.70</td>
<td>63.81</td>
<td>20</td>
<td>18</td>
<td>34.71</td>
</tr>
<tr>
<td>8.79</td>
<td>64.59</td>
<td>20</td>
<td>13</td>
<td>35.00</td>
</tr>
<tr>
<td>6.84</td>
<td>63.11</td>
<td>20</td>
<td>12</td>
<td>35.97</td>
</tr>
</tbody>
</table>

Table 5.16: Echo Cancellation Performance (ECP) vs $f_p$: Loop#11.

Figure 5.25: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#11. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_C/2]$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.14</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>17.19</td>
<td>60.72</td>
<td>20</td>
<td>24</td>
<td>34.60</td>
</tr>
<tr>
<td>15.28</td>
<td>61.73</td>
<td>20</td>
<td>22</td>
<td>34.60</td>
</tr>
<tr>
<td>13.28</td>
<td>62.15</td>
<td>20</td>
<td>21</td>
<td>34.63</td>
</tr>
<tr>
<td>11.33</td>
<td>63.79</td>
<td>20</td>
<td>20</td>
<td>35.13</td>
</tr>
<tr>
<td>9.38</td>
<td>64.08</td>
<td>20</td>
<td>13</td>
<td>34.93</td>
</tr>
<tr>
<td>7.42</td>
<td>62.41</td>
<td>20</td>
<td>12</td>
<td>34.99</td>
</tr>
</tbody>
</table>

Table 5.17: Echo Cancellation Performance (ECP) vs $f_p$: Loop#12.

Figure 5.26: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$: Loop#12. For each passband edge, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$f_p$ in mHz</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_G/2]$</th>
<th>$\delta_s$ in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.94</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>16.99</td>
<td>59.20</td>
<td>20</td>
<td>24</td>
<td>34.92</td>
</tr>
<tr>
<td>15.04</td>
<td>60.49</td>
<td>20</td>
<td>22</td>
<td>34.97</td>
</tr>
<tr>
<td>13.08</td>
<td>61.01</td>
<td>20</td>
<td>21</td>
<td>35.46</td>
</tr>
<tr>
<td>11.13</td>
<td>62.39</td>
<td>20</td>
<td>20</td>
<td>34.73</td>
</tr>
<tr>
<td>9.17</td>
<td>63.81</td>
<td>20</td>
<td>13</td>
<td>35.33</td>
</tr>
<tr>
<td>7.22</td>
<td>62.09</td>
<td>20</td>
<td>12</td>
<td>35.45</td>
</tr>
</tbody>
</table>

Table 5.18: Echo Cancellation Performance (ECP) vs $f_p$; Loop#13.

Figure 5.27: A plot to show the two-stage echo canceller performance sensitivity to the choice of the passband edge $f_p$; Loop#13. For each passband edge, a specific $G(Z)$ is designed.
As a result of the first part of the investigation, lower order interpolators can be designed by careful selection of the passband edge frequency. For $f_p$ chosen in the vicinity of the deflection point, the complexity of the overall system had substantially decreased, since the interpolator's taps were reduced. As a consequence, the interpolator order was found to be between 24 and 26 taps for all the listed lines. The $COMPLEXITY_{AIFIR}$ of the tail canceller is either 34 or 35 multipliers of size $r \times r$. The generated interpolated response of the AIFIR has 341 samples (from Eq.(3.27)). The simulations detailed in Table 5.19 to Table 5.26 indicate that the echo cancellation performance is affected by the length of the main canceller $N_M$ (as found in subsection 5.3.1). Some observations could be made from these results. Firstly, echo cancellation performance using the new efficient interpolators was at least 2 dB better\textsuperscript{19} than that of the investigation\#1 interpolators. Thus, in the remainder of this thesis only the efficient interpolators are considered. Secondly, as found in investigation\#1, provided that the coarse region of the impulse response is completely canceled by an AFIR main canceller, the echo cancellation performance improves as $N_M$ increases. Finally, for $N_M \geq 15$, echo cancellation performance of 60 dB was achieved for all loops considered. Also, from Table 5.19 to Table 5.26 (for $N_M = 20$), an improvement of almost 5 dB in the echo cancellation performance is noted by increasing $N_M$ by 10 more taps. For a given efficient interpolator, Figure 5.28 to Figure 5.35 illustrate that as the length $N_M$ improves, the canceller performance increases. Basically, these figures possess the same general characteristics as Figure 5.20 to Figure 5.27.

\textsuperscript{19}This was observed when $N_M$ was equal to 20.
| Total Length | $N_M$ | $N_r$ | ECP in dB | $\delta_r$ in dB | $|N_G/2|$ | $f_p$ in mHz | COMPLEXITY_{MEC} |
|-------------|-------|-------|-----------|-----------------|----------|-------------|------------------|
| 350         | 10    | 340   | 60.45     | 34.98           | 13       | 7.81        | 20               |
| 355         | 15    | 340   | 63.68     | 34.98           | 13       | 7.81        | 30               |
| 360         | 20    | 340   | 64.30     | 34.98           | 13       | 7.81        | 40               |
| 365         | 25    | 340   | 65.94     | 34.98           | 13       | 7.81        | 50               |
| 370         | 32    | 340   | 67.19     | 34.98           | 13       | 7.81        | 64               |
| 380         | 40    | 340   | 68.40     | 34.98           | 13       | 7.81        | 80               |
| 390         | 50    | 340   | 70.10     | 34.98           | 13       | 7.81        | 100              |

Table 5.19: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#2. The main canceller multiplier size is $3 \times r$. The COMPLEXITY_{MEC} of the tail echo canceller is $35 r \times r$ multipliers.

Figure 5.28: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#2.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_s$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY&lt;sub&gt;MFC&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>351</td>
<td>10</td>
<td>341</td>
<td>61.58</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>20</td>
</tr>
<tr>
<td>361</td>
<td>20</td>
<td>341</td>
<td>64.91</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>40</td>
</tr>
<tr>
<td>366</td>
<td>25</td>
<td>341</td>
<td>65.31</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>50</td>
</tr>
<tr>
<td>373</td>
<td>32</td>
<td>341</td>
<td>68.12</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>64</td>
</tr>
<tr>
<td>381</td>
<td>40</td>
<td>341</td>
<td>69.71</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>80</td>
</tr>
<tr>
<td>391</td>
<td>50</td>
<td>341</td>
<td>71.05</td>
<td>36.14</td>
<td>13</td>
<td>9.14</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.20: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#5. The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35 \times r$ multipliers.

![Graph](image)

Figure 5.29: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#5.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>( N_M )</th>
<th>( N_r )</th>
<th>ECP in dB</th>
<th>( \delta_e ) in dB</th>
<th>( [N_G/2] )</th>
<th>( f_p ) in mHz</th>
<th>COMPLEXITY(_{MEC} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>350</td>
<td>10</td>
<td>340</td>
<td>60.94</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>20</td>
</tr>
<tr>
<td>355</td>
<td>15</td>
<td>340</td>
<td>62.53</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>30</td>
</tr>
<tr>
<td>360</td>
<td>20</td>
<td>340</td>
<td>65.17</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>40</td>
</tr>
<tr>
<td>365</td>
<td>25</td>
<td>340</td>
<td>66.02</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>50</td>
</tr>
<tr>
<td>372</td>
<td>32</td>
<td>340</td>
<td>68.01</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>64</td>
</tr>
<tr>
<td>380</td>
<td>40</td>
<td>340</td>
<td>70.26</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>80</td>
</tr>
<tr>
<td>390</td>
<td>50</td>
<td>340</td>
<td>72.30</td>
<td>34.13</td>
<td>13</td>
<td>7.81</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.21: Echo Cancellation Performance (ECP) vs \( N_M \) (efficient \( G(Z) \)): Loop\#6. The main canceller multiplier size is \( 3 \times r \). The COMPLEXITY\(_{MEC} \) of the tail echo canceller is \( 35 r \times r \) multipliers.

![Graph](image)

Figure 5.30: A graph to illustrate the effects of Main Echo Canceller length \( N_M \) on performance (efficient interpolator): Loop\#6.
Table 5.22: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#7. The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{MEX}$ of the tail echo canceller is $35 \times r$ multipliers.

<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_{tr}$</th>
<th>ECP in dB</th>
<th>$\delta_3$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_r$ in mHz</th>
<th>$COMPLEXITY_{MEX}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>350</td>
<td>10</td>
<td>340</td>
<td>61.87</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>20</td>
</tr>
<tr>
<td>355</td>
<td>15</td>
<td>340</td>
<td>63.25</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>30</td>
</tr>
<tr>
<td>360</td>
<td>20</td>
<td>340</td>
<td>63.68</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>40</td>
</tr>
<tr>
<td>365</td>
<td>25</td>
<td>340</td>
<td>65.81</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>50</td>
</tr>
<tr>
<td>372</td>
<td>32</td>
<td>340</td>
<td>66.74</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>64</td>
</tr>
<tr>
<td>380</td>
<td>40</td>
<td>340</td>
<td>68.51</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>80</td>
</tr>
<tr>
<td>390</td>
<td>50</td>
<td>340</td>
<td>70.41</td>
<td>36.40</td>
<td>13</td>
<td>8.79</td>
<td>100</td>
</tr>
</tbody>
</table>

Figure 5.31: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#7.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_s$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY_{MEC}</th>
</tr>
</thead>
<tbody>
<tr>
<td>339</td>
<td>10</td>
<td>339</td>
<td>62.37</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>20</td>
</tr>
<tr>
<td>359</td>
<td>15</td>
<td>339</td>
<td>64.69</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>30</td>
</tr>
<tr>
<td>359</td>
<td>20</td>
<td>339</td>
<td>65.13</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>40</td>
</tr>
<tr>
<td>364</td>
<td>25</td>
<td>339</td>
<td>66.43</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>50</td>
</tr>
<tr>
<td>371</td>
<td>32</td>
<td>339</td>
<td>68.48</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>64</td>
</tr>
<tr>
<td>379</td>
<td>40</td>
<td>339</td>
<td>69.21</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>80</td>
</tr>
<tr>
<td>389</td>
<td>50</td>
<td>339</td>
<td>72.96</td>
<td>34.69</td>
<td>12</td>
<td>7.81</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.23: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#9.
The main canceller multiplier size is $3 \times r$. The COMPLEXITY_{MEC} of the tail
echo canceller is $34 \times r$ multipliers.

Figure 5.32: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on
performance (efficient interpolator): Loop#9.
Table 5.24: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#11. The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{AIFIR}$ of the tail echo canceller is $35 \times r \times r$ multipliers.

![Graph showing the relationship between ECP in dB and MEC length](image)

Figure 5.33: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#11.
| Total Length | \( N_M \) | \( N_r \) | ECP in dB | \( a_s \) in dB | \( |N_G/2| \) | \( f_p \) in mHz | COMPLEXITY | MEC |
|-------------|-------|------|--------|------------|--------|-----------|------------|-----|
| 350         | 10    | 340  | 59.36  | 34.93      | 13     | 9.38      | 20         |     |
| 355         | 15    | 340  | 62.71  | 34.93      | 13     | 9.38      | 30         |     |
| 360         | 20    | 340  | 64.08  | 34.93      | 13     | 9.38      | 40         |     |
| 365         | 25    | 340  | 64.99  | 34.93      | 13     | 9.38      | 50         |     |
| 372         | 32    | 340  | 66.79  | 34.93      | 13     | 9.38      | 64         |     |
| 380         | 40    | 340  | 68.81  | 34.93      | 13     | 9.38      | 80         |     |
| 390         | 50    | 340  | 70.21  | 34.93      | 13     | 9.38      | 100        |     |

Table 5.25: Echo Cancellation Performance (ECP) vs \( N_M \) (efficient \( G(Z) \)): Loop\#12. The main canceller multiplier size is \( 3 \times r \). The \( COMPLEXITY_{METH} \) of the tail echo canceller is \( 35 \times r \) multipliers.

![Graph](image)

Figure 5.34: A graph to illustrate the effects of Main Echo Canceller length \( N_M \) on performance (efficient interpolator): Loop\#12.
<table>
<thead>
<tr>
<th>Total Length</th>
<th>$N_M$</th>
<th>$N_r$</th>
<th>ECP in dB</th>
<th>$\delta_s$ in dB</th>
<th>$[N_G/2]$</th>
<th>$f_p$ in mHz</th>
<th>COMPLEXITY$<em>{M</em>{EC}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>351</td>
<td>10</td>
<td>341</td>
<td>58.91</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>20</td>
</tr>
<tr>
<td>356</td>
<td>15</td>
<td>341</td>
<td>60.35</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>30</td>
</tr>
<tr>
<td>361</td>
<td>20</td>
<td>341</td>
<td>63.81</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>40</td>
</tr>
<tr>
<td>366</td>
<td>25</td>
<td>341</td>
<td>64.35</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>50</td>
</tr>
<tr>
<td>373</td>
<td>32</td>
<td>341</td>
<td>67.58</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>64</td>
</tr>
<tr>
<td>381</td>
<td>40</td>
<td>341</td>
<td>68.93</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>80</td>
</tr>
<tr>
<td>391</td>
<td>50</td>
<td>341</td>
<td>70.11</td>
<td>35.21</td>
<td>13</td>
<td>9.17</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 5.26: Echo Cancellation Performance (ECP) vs $N_M$ (efficient $G(Z)$): Loop#13. The main canceller multiplier size is $3 \times r$. The $COMPLEXITY_{M_{EC}}$ of the tail echo canceller is $35 \times r$ multipliers.

![Graph](image)

Figure 5.35: A graph to illustrate the effects of Main Echo Canceller length $N_M$ on performance (efficient interpolator): Loop#13.
5.3.3 Investigation#3: Performance of a Compromise Interpolator

In subsection 5.3.2, it was found that the value of $f_p$ has a significant effect on the performance of the proposed echo canceller. Here we investigate the effects of employing a compromise interpolator (i.e., select one value of $f_p$) on echo cancellation performance for all Bellcore lines considered. For $L$ equals to 15 for all the loops. the sparse filter order is 22 non-zero taps (from Eq.(1.16)). Thus, an interpolated response of 340 samples (from Eq.(3.27)) is generated by the AIFIR filter to cancel the tail pulse. In order to implement the AIFIR tail canceller, 47 taps are required (from Eq.(4.17)). Results for some of the Bellcore lines using the two-stage echo cancellation (with the length of $N_M = 20$) are given in Table 5.27. From sensitivity plots (Figure 5.20 to Figure 5.27), it was observed that each plot had a maximum point. Each maximum point corresponds to a value of $f_p$. The choice of $f_p$ must not increase the complexity of the tail canceller. Meanwhile, it must not severely degrade the system performance under any loop. With this criterion in mind, we propose the use of $f_p$ equals to 7.81 mHz (Loop#6) in the design of the compromise interpolator since it is the smallest $f_p$ value in the selection. Its corresponding $N_G$ and the stopband attenuation $\delta_s$ are 25 taps and 34.69 dB, respectively. Also, Loop#6 is one of the longest lines and thus with higher attenuation. Therefore, the performance of the two-stage echo canceller on longer lines should have the priority even, if this selection of $f_p$ resulted in degradation in system performance on shorter lines. Stated another way, the required echo cancellation accuracy must be met on all loops. In order to satisfy the above criterion, system might have below optimum performance on some of the loops. The following simulation is performed to investigate the effects of using the compromise interpolator on system performance. Also, in this investigation we introduce the use of Loop#10 and Loop#0. The simulation results are tabulated in Table 5.27.
<table>
<thead>
<tr>
<th>Loop#</th>
<th>ECP in dB</th>
<th>N_M</th>
<th>f_p in mHz</th>
<th>([N_G/2])</th>
<th>(\delta_s) in dB</th>
<th>Length</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>64.06</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>5</td>
<td>61.98</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>6</td>
<td>65.17</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>360</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>7</td>
<td>62.77</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>9</td>
<td>64.97</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>11</td>
<td>62.15</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>12</td>
<td>66.89</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>13</td>
<td>61.56</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>10</td>
<td>60.03</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
<tr>
<td>0</td>
<td>56.70</td>
<td>20</td>
<td>7.81</td>
<td>13</td>
<td>34.69</td>
<td>560</td>
<td>40 3 x r + 35 r x r</td>
</tr>
</tbody>
</table>

Table 5.27: Echo The effects of using a compromise interpolator in the tail canceller on the echo cancellation performance (ECP): various Bellcore lines are considered.
Some observations can be made from Table 5.27. First of all, it can be noted that by using one compromise interpolator for all the loops, the performance of the overall system is degraded for most loops if we compare with the simulations that have one particular interpolator designed for a specific loop frequency response. Secondly, the echo cancellation performance of all considered Belcore lines meet the accuracy requirement that was set earlier (60 dB) except the loop#0. However, this loop causes significantly less attenuation of the far end signal, so less than 60 dB echo cancellation accuracy would be acceptable. Therefore, we propose this system to be used for all lines with 60 dB echo cancellation requirement. From Eq.(4.19) and Eq.(4.20), the complexity of this system is $40 \times 3 \times r + 35 \times r \times r$ multipliers. An AFIR needed to meet this requirement requires 678 multiplier of size $3 \times r$ to perform filtering and adaptation. The proposed tail canceller requires approximately 27% of the hardware of the transversal AFIR canceller. The main attributes of the proposed system are listed in Table 5.28. For completeness, the frequency response of the simulated tail (e.g., Loop#6) is shown in Figure 5.36. In addition, the frequency response of the sparse filter $F(Z^L)$ and the frequency response of the compromise interpolator filter $G(Z)$ are illustrated in Figure 5.37 and Figure 5.38, respectively.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Attainment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy requirement</td>
<td>60 dB</td>
</tr>
<tr>
<td>Complexity</td>
<td>$40 \times 3 \times r + 35 \times r \times r$</td>
</tr>
<tr>
<td>Length of MEC ($N_M$)</td>
<td>20</td>
</tr>
<tr>
<td>Length of IFIR ($N_r$)</td>
<td>310 samples</td>
</tr>
<tr>
<td>Length of $F(Z^L)$ ($N_F$)</td>
<td>22</td>
</tr>
<tr>
<td>Length of $G(Z)$ ($N_G/2$)</td>
<td>13</td>
</tr>
<tr>
<td>$L$</td>
<td>15</td>
</tr>
<tr>
<td>$\delta_s$</td>
<td>3.4 dB</td>
</tr>
<tr>
<td>$f_p$</td>
<td>7.81 mHz</td>
</tr>
</tbody>
</table>

Table 5.28: Important specifications of the proposed two-stage echo canceller.

---

20 It is 0.5 kft 26 gauge line
Figure 5.36: An exploded view of Loop#6 simulated tail frequency response.

Figure 5.37: A view of the wanted passband of the sparse filter $F(Z^L)$ (Loop#6) and its spectral images (only showing seven of these images). All spectral images must be attenuated by the interpolator $G(Z)$. 
Figure 5.38: An exploded view of the compromise interpolator.

5.3.4 Investigation #4: Effects of Interpolator Stopband Attenuation $\delta_s$

In this subsection, we study the effects of the interpolator stopband attenuation $\delta_s$ of the interpolator on the system performance. For a given interpolator and a sufficient length of the first stage canceller $N_M$, extensive simulations\textsuperscript{21} have showed that the selected stopband attenuation $\delta_s$ had a dramatic effect on the performance of the echo canceller. Insufficient stopband attenuation adversely degraded the performance of the echo canceller. On the other hand, high values of $\delta_s$ improved the system performance. For the compromise interpolator of the previous subsection, the passband frequency is kept unchanged at 7.81 mHz. Only the value of the stopband ripple $\delta_s$ is varied. For each value of $\delta_s$, an interpolator is designed and used in the simulations. It is evident that the greater the 'dB' value of $\delta_s$, the higher the order of the interpolator $N_G$, and the higher the complexity. For $L$ equals to 15 for all the loops, the sparse filter order is 22 non-zero coefficients (from Eq.(4.16)). To show how the echo cancellation performance varies with the value of $\delta_s$, simulation results of the

\textsuperscript{21}Due to space limitations, only the simulations that used a compromise interpolator will be reported here.
proposed echo canceler are reported in Table 5.29 to Table 5.36. Some conclusions can be made from these results.

1. For low values of the stopband attenuation $\delta_s$, 60 dB echo cancellation performance was not attained.

2. Given that the main echo pulse is completely removed by the AFIR main canceler (with minimum length $N_M$), decreasing the level of attenuation (i.e., $\delta_s$) on the unwanted spectral images results in a significant degradation in the system performance. As the values of the stopband attenuation $\delta_s$ decreases, echo cancellation performance degrades. This degradation is partly due to the fact that insufficient suppression of the unwanted replicas of the sparse filter response introduces high-frequency content in the simulated tail response. The AIFIR filter approximation of the tail response is compromised by the addition of the high-frequency content to the IFIR filter response.

3. From results for other loops, it was noticed that the degradation is more noticeable for longer loops.

Figure 5.39 to Figure 5.46 illustrate the echo cancellation performance as a function of the stopband attenuation $\delta_s$ for each of the eight considered lines, for $N_M = 20$. 

157
Table 5.29: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#2.

<table>
<thead>
<tr>
<th>$\delta_s$ in dB</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$\lfloor N_G/2 \rfloor$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>57.93</td>
<td>20</td>
<td>11</td>
<td>$40 \times r + 33 \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>64.06</td>
<td>20</td>
<td>13</td>
<td>$40 \times r + 35 \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>65.19</td>
<td>20</td>
<td>17</td>
<td>$40 \times r + 39 \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>67.34</td>
<td>20</td>
<td>20</td>
<td>$40 \times r + 42 \times r$</td>
</tr>
</tbody>
</table>

Figure 5.39: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#2. For each $\delta_s$, a specific $G(Z)$ is designed.
\[
\begin{array}{|c|c|c|c|c|}
\hline
\delta, \\
in \text{dB} & ECP \\
in \text{dB} & N_M & \lceil N_G/2 \rceil & \text{Complexity} \\
\hline
28.64 & 56.69 & 20 & 11 & 40 \times r + 33 \times r \\
34.69 & 61.98 & 20 & 13 & 40 \times r + 35 \times r \\
37.28 & 63.03 & 20 & 17 & 40 \times r + 39 \times r \\
41.27 & 65.72 & 20 & 20 & 40 \times r + 42 \times r \\
\hline
\end{array}
\]

Table 5.30: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#5.

Figure 5.40: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#5. For each $\delta_s$, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$\delta_s$ in dB</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_c/2]$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>58.37</td>
<td>20</td>
<td>11</td>
<td>$40 \times r + 33 \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>65.17</td>
<td>20</td>
<td>13</td>
<td>$40 \times r + 35 \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>66.24</td>
<td>20</td>
<td>17</td>
<td>$40 \times r + 39 \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>68.11</td>
<td>20</td>
<td>20</td>
<td>$40 \times r + 42 \times r$</td>
</tr>
</tbody>
</table>

Table 5.31: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#6.

Figure 5.41: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband edge $\delta_s$: Loop#6. For each $\delta_s$, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$\delta_s$ in dB</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_G/2]$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>58.98</td>
<td>20</td>
<td>11</td>
<td>$40 \times r + 33 \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>62.77</td>
<td>20</td>
<td>13</td>
<td>$40 \times r + 35 \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>65.91</td>
<td>20</td>
<td>17</td>
<td>$40 \times r + 39 \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>67.70</td>
<td>20</td>
<td>20</td>
<td>$40 \times r + 42 \times r$</td>
</tr>
</tbody>
</table>

Table 5.32: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#7.

Figure 5.42: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#7. For each $\delta_s$, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$\delta_s$ in dB</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_M/2]$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>61.38</td>
<td>20</td>
<td>11</td>
<td>$40 \times r + 33 \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>64.97</td>
<td>20</td>
<td>13</td>
<td>$40 \times r + 35 \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>67.48</td>
<td>20</td>
<td>17</td>
<td>$40 \times r + 39 \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>70.10</td>
<td>20</td>
<td>20</td>
<td>$40 \times r + 42 \times r$</td>
</tr>
</tbody>
</table>

Table 5.33: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#9.

Figure 5.43: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#9. For each $\delta_s$, a specific $G(z)$ is designed.
Table 5.34: Echo Cancellation Performance (ECP) vs \( \delta_s \): Loop\#11.

<table>
<thead>
<tr>
<th>( \delta_s ) in dB</th>
<th>ECP in dB</th>
<th>( N_M )</th>
<th>([N_G/2])</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>57.14</td>
<td>20</td>
<td>11</td>
<td>40 ( 3 \times r + 33 ) ( r \times r )</td>
</tr>
<tr>
<td>34.69</td>
<td>62.15</td>
<td>20</td>
<td>13</td>
<td>40 ( 3 \times r + 35 ) ( r \times r )</td>
</tr>
<tr>
<td>37.28</td>
<td>63.43</td>
<td>20</td>
<td>17</td>
<td>40 ( 3 \times r + 39 ) ( r \times r )</td>
</tr>
<tr>
<td>41.27</td>
<td>66.46</td>
<td>20</td>
<td>20</td>
<td>40 ( 3 \times r + 42 ) ( r \times r )</td>
</tr>
</tbody>
</table>

Figure 5.44: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple \( \delta_s \): Loop\#11. For each \( \delta_s \), a specific \( G(Z) \) is designed.
<table>
<thead>
<tr>
<th>$\delta_s$ in dB</th>
<th>ECP in dB</th>
<th>$N_M$</th>
<th>$[N_o/2]$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>56.91</td>
<td>20</td>
<td>11</td>
<td>$40 \times 3 \times r + 33 \times r \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>60.89</td>
<td>20</td>
<td>13</td>
<td>$40 \times 3 \times r + 35 \times r \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>63.07</td>
<td>20</td>
<td>17</td>
<td>$40 \times 3 \times r + 39 \times r \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>64.83</td>
<td>20</td>
<td>20</td>
<td>$40 \times 3 \times r + 42 \times r \times r$</td>
</tr>
</tbody>
</table>

Table 5.35: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#12.

Figure 5.45: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#12. For each $\delta_s$, a specific $G(Z)$ is designed.
<table>
<thead>
<tr>
<th>$\delta_s$ (in dB)</th>
<th>ECP (in dB)</th>
<th>$N_M$</th>
<th>$\lceil N_G/2 \rceil$</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.64</td>
<td>57.62</td>
<td>20</td>
<td>11</td>
<td>$40 \times r + 33 \times r$</td>
</tr>
<tr>
<td>34.69</td>
<td>61.56</td>
<td>20</td>
<td>13</td>
<td>$40 \times r + 35 \times r$</td>
</tr>
<tr>
<td>37.28</td>
<td>63.99</td>
<td>20</td>
<td>17</td>
<td>$40 \times r + 39 \times r$</td>
</tr>
<tr>
<td>41.27</td>
<td>67.06</td>
<td>20</td>
<td>20</td>
<td>$40 \times r + 42 \times r$</td>
</tr>
</tbody>
</table>

Table 5.36: Echo Cancellation Performance (ECP) vs $\delta_s$: Loop#13.

Figure 5.46: A plot to show the two-stage echo canceller performance sensitivity to the choice of the stopband ripple $\delta_s$: Loop#13. For each $\delta_s$, a specific $G(Z)$ is designed.
Discussion

To conclude our observations on the simulation results, we present a brief discussion of the different effects of all the parameters considered in the investigations. For a given interpolator, it has been found that increasing the main canceller order $N_M$ improves the performance (increases complexity) of the echo canceller. With efficient interpolators, it appears that there is no need to increase $N_M$ beyond 20 taps. Extensive simulations have shown that the choice of $f_p$ had a significant effect on the canceller performance. The decrease of $f_p$ up to a certain point improved the performance. Choosing a value of $f_p$ near the deflection point offered the best performance. With this in mind, a compromise interpolator was used on all loops. As a result, it was apparent that the canceller performance is degraded. In addition, Loop#0 did not meet the 60 dB echo cancellation accuracy. However, that is not a problem since this loop causes less attenuation of the far end signal and less than 60 dB echo cancellation accuracy is acceptable. For a given interpolator and $N_M$, simulation results have indicated that insufficient stopband attenuation $\delta_s$ adversely degraded the canceller performance. Moreover, it was found that adequate performance can be attained if $\delta_s \geq 32\ dB$. With all the adverse/favorable effects of the investigation parameters and the various tradeoffs between complexity and performance, a two-stage echo canceller, with a compromise fixed interpolator meeting the 60 dB echo cancellation accuracy was designed in Table 5.28 and was shown to be far less complex than the AHR echo canceller.
5.4 Comparison With Orthonormal-Functions-Based EC

In this section, we present a comparison of the proposed two stage echo canceller utilizing AllFIR filter as a tail canceller with Davidson-Falconer EC. Unfortunately, there have been very few publications regarding echo tail cancellation for the 2B1Q line code. The echo tail cancellation methods proposed by the industrial community were summarized in section 2.4. The most common designs were using a simple IIR tail canceller, some type of a high pass receiving filter, and a precise compromise network. Therefore, we will limit this comparison to the orthonormal-functions-based (i.e., IIR filters) tail canceller scheme proposed in [100], [101]. The method offers good results. However, The use of IIR filters implies the possibility of problems due to finite wordlength representation. Moreover, to maintain the orthogonality of the IIR filters, very accurate representation of the pole value is required.

The basic principle in [100], [101] is somewhat similar with our technique, the similarity being that a two-stage echo canceller is used. In [100], the first stage echo canceller is an adaptive transversal-type filter that covers the coarse section of the impulse response. The second stage echo canceller is intended to approximate the tail section by using linear combination of orthonormal functions. The weighting factors of the orthonormal functions are adapted to approximate the tail response. Exponential functions and Laguerre orthonormal functions were used. It has been found that echo cancellation performance using the Laguerre orthonormal functions was 8 to 11 dB better than that of the Exponential functions. It was also noted that the echo cancellation performance improves as the length of first stage canceller (N_b) increases. Moreover, it was reported that the number of terms required in the orthonormal functions expansion (N_a) (i.e., the number of IIR filters) affected the performance and the complexity of two-stage echo canceller. Of course, the more IIR filters used, the better the performance, and the higher the complexity of the system. The two-
stage echo canceller complexity was calculated by using \( \text{complexity} = 2N_s + 20N_e \). For 80 kbaud, the complexity of this technique was compared to that of the AFIR implementation. For 70 dB echo cancellation accuracy, it was reported that this technique offers a reduction in complexity by 51\% compared to a transversal AFIR of at least 330 taps. In addition, simulations for extremely high bit rates such as 640 kb/s (320 kbaud) were performed for 70 dB echo cancellation accuracy. These simulations showed an even greater decrease in complexity\(^{22}\) (85\% less complex than an AFIR).

We compared both approaches for two cases in \([100]\) for the same ECP requirement and for Loop\#9. The length of the first stage canceller is 20 taps for both approaches.

**Example\#1**

For 60 dB accuracy, Davidson-Falconer tail canceller has a complexity\(^{23}\) of 20 multipliers of size 20 × 20 (with \(N_a = 10\) and \(ECP = 63.6 \, \text{dB}\)). The total number of bits for the Davidson-Falconer tail canceller is 8000 bits. To compare this to the complexity of the AIFIR filter approach, we design a tail canceller with an interpolator passband of 7.81 mHz, stopband attenuation of 30.4 dB, \(N_G = 22\), the interpolation factor \(L\) is equal to 15, and the non-zero taps of the sparse filter \(N_F = 22\). The echo cancellation performance achieved by the AIFIR filter approach is 63.8 dB. The AIFIR filter approach possessed a complexity of 33 multipliers of size 16 × 16. The total number of bits for the AIFIR tail canceller is 8448 bits. Based on the qualitative analysis discussed in chapter 4, the AIFIR filter will have a tap wordlength close to that of the AFIR filter. Therefore, 16 × 16 multiplier size is reasonable. As seen from the comparison, both approaches are close in complexity. However, due to the fact that our method is an FIR-based, that could be considered as an advantage over the orthonormal (IIR) functions.

\(^{22}\)This is a consequence of the fact that the number of the IIR filters in the tail canceller does not increase with the increase of the baud rate \([100]\).

\(^{23}\)This example is cited from table 2 on page 13 for Loop\#9.
Example #2

For 70 dB accuracy, the tail canceller complexity\(^{24}\) of Davidson-Falconer canceller is 28 multipliers of size 20 \(\times\) 20 \((N_a = 14)\). The total number of bits for the Davidson-Falconer tail canceller is 11200 bits. The results obtained so far in this thesis are applicable to 60 dB echo cancellation accuracy. In order to fairly compare the complexity of Davidson-Falconer canceller to that of the AIFIR filter approach, we have to redesign our canceller to meet the 70 dB requirement. There are some design parameters that need to be changed. First of all, the noise source variance\(^{25}\) is changed to \(\sigma_n^2 = 10^{-12}\). Secondly, it has been found in earlier simulations that the echo cancellation performance can be improved by two methods given that the main echo pulse is completely removed by the first stage canceller (with sufficient\(^{26}\) length \(N_M\)):

1. For a given interpolator and sufficient first stage canceller length \(N_M\), increasing \(N_M\) beyond the required sufficient value causes a considerable improvement in the overall system performance. As a side penalty, the complexity of AIFIR filter increases.

2. For a sufficient first stage canceller length \(N_M\), increasing the level of attenuation \(\delta_s\) (implies increasing \(N_O\)) on the unwanted spectral images results in a sufficient improvement in the system performance. This improvement is achieved at the cost of higher AIFIR filter complexity.

The second method was selected since it was the least complex alternative. This is due to the fact that the interpolator is non-adaptive. Thus, increasing the order of \(G(Z)\) is less complex than increasing the order of the first stage canceller \(N_M\). For the tail echo canceller, 41 multipliers of size 16 \(\times\) 16 are required to attain 70 dB

\(^{24}\)This example is cited from page 31 for Loop#9.

\(^{25}\)The same is used in [100].

\(^{26}\)It is the necessary length to achieve the required ECA.
accuracy (i.e., $N_F = 22$, $N_G = 38$, and $\delta_s = 40.29 \, dB$). The total number of bits for the AIFIR tail canceller is 10496 bits.

5.5 Conclusion

Due to the shape of the echo path impulse response, two-stage echo cancellation is used in this research. In this chapter, the proposed two-stage echo canceller utilizing an AIFIR filter as a tail canceller has been simulated. The AIFIR echo tail canceller design method was given. Sufficient echo cancellation was attained under various loop configurations. In this thesis, 60 dB echo cancellation accuracy was selected as the performance requirement to be met by the proposed two stage canceller. The simulation results for 60 dB echo cancellation accuracy were presented. Considering the simulated results, it is evident that the echo cancellation performance is affected by the design parameters such as $f_p$, $\delta_s$, and $N_M$. Provided that the first stage canceller has the minimum required length to cancel all of the main echo pulse, two methods can be used to improve the echo cancellation performance. As a penalty, the complexity of the two-stage canceller increased. The first method is to increase the length of the main canceller $N_M$ beyond the minimum required length. This method was dismissed in favour of the second method, that is to increase the stopband attenuation $\delta_s$ on the unwanted spectral images of the sparse filter. Since the $G(Z)$ is non-adaptive, it is less complex to increase the interpolator order than to increase the main canceller length. Insufficient stopband attenuation adversely degraded the performance of the echo canceller. By trial and error, the lowest order of the interpolator was found to ensure meeting the echo cancellation accuracy requirement. The attributes of one compromise tail canceller to be used on all loops was given. It was also found that a main canceller length of 20 taps and an AIFIR tail canceller with an interpolator of passband edge frequency $f_p$ of 7.81 mHz, stopband attenuation $\delta_s$ of 34 dB (implies $N_G = 25$), and an interpolation factor $L$ of 15 (implies $N_F = 22$), 60 dB cancellation
accuracy could be achieved on a wide range of Bellcore loops. Lastly, the proposed AIFIR filter approach is compared with the canceller of Davidson & Falconer. The results show that the suggested approach is quite efficient. This proposed tail canceller approximately requires 27% of the hardware of the AFIR canceller.
Chapter 6

Conclusions and Recommendations

6.1 Concluding Remarks

In this thesis, a technique for two-stage echo cancellation has been presented in which the tail impulse response is approximated by an adaptive interpolative FIR (AIFIR) filter. The analytical and simulation results given in this thesis demonstrate that the use of the AIFIR echo tail canceller can substantially reduce the complexity of the two-stage canceller by approximately 73% when compared to the direct form AFIR echo canceller. In this thesis, 60 dB echo cancellation accuracy was selected as the performance requirement to be met by the proposed two stage canceller. Simulation results indicate that 60 db echo cancellation accuracy can be attained on many Bellcore loops using the proposed two-stage canceller, with a compromise fixed interpolator \( G(z) \). It has been found that a main canceller length of 20 taps and an AIFIR tail canceller with an interpolator section having a passband edge \( f_{ps} \) of 7.81 mHz, stopband attenuation \( \delta_s \) of 34 dB (implies \( N_G = 25 \)), and an interpolation factor \( L \) of 15 (implies \( N_F = 22 \)), 60 dB cancellation accuracy could be achieved on a wide range of Bellcore loops. Lastly, the proposed AIFIR filter approach is compared with the canceller of Davidson & Falconer. The results show that the suggested approach is quite efficient. The relevant attributes of the suggested canceller are given
in Table 5.28. From the simulation results, we conclude that:

1. Provided that the first stage canceller has the sufficient required length to represent all of the main echo pulse, the performance of the two-stage echo canceller improves as the length of the first stage canceller \( N_M \) increases. As a penalty, the complexity of the system increases. It has been found that using efficient interpolators, length of the first stage canceller \( N_M \) does not need to exceed 20 taps. These interpolators were designed by careful selection of the \( f_p \) parameter.

2. The selected value of the interpolator stopband edge \( f_p \) had a significant effect on the system performance. The decrease of \( f_p \) up to a certain point (deflection point) improves the system performance. Past this point, degradation of performance was noted. Choosing a value of \( f_p \) near the deflection point offered the best performance. With this in mind, a compromise interpolator was designed for all loops and was shown to meet the 60 dB echo cancellation accuracy for all loops considered except Loop#0. However, this loop causes less attenuation of the far end signal, so less than 60 dB echo cancellation accuracy would be acceptable.

3. For a given \( G(Z) \) and a sufficient length of the first stage canceller \( N_M \), it was found that the selected stopband attenuation \( \delta_s \) had a dramatic effect on the performance of the echo canceller. Insufficient stopband attenuation adversely degraded the performance of the echo canceller.

### 6.2 Suggestions for Future Work

There are many interesting aspects of AIFIR filtering which could not be pursued in the context of this thesis. A few are outlined below.

1. A fully adaptive IFIR filter (i.e., the IFIR filter is simulated as a cascade of an adaptive sparse filter and an adaptive interpolator \( G(Z) \)). This could improve
the performance of the AIFIR filter.

2. Two implementations of $G(Z)$ presented in [90] should be considered for future research (a must for high bit rates such as 800 kb/s). Multistage interpolators are more efficient than single-stage interpolators. The second implementation is an efficient recursive running sum interpolators.

3. In view of the significant computational savings obtained in [90] for simulating narrow-band responses which can span about 2500 samples, there is good reason to believe that AIFIR tail canceller could be used in extremely high bit rate (e.g., 800 kb/s).

4. For such high bit rates, it is expected that the echo tail impulse response will span at least 2000 samples. Provided that main echo pulse is simulated by an AFIR canceller, the tail pulse could be partitioned into several tail pulses. Several of the AIFIR tail canceller (proposed for 160 kb/s) could be cascaded in series to simulate the total echo tail.

5. The results in this thesis clearly demonstrate a need for further analysis of the stopband attenuation effects on performance.

6. The digital wordlength properties of the AIFIR filter should be explored.

7. In the literature of the multiplier-efficient narrow-band digital filters, we found eight different methods. These methods could be explored for the use as tail echo cancellers. A very promising technique is the multiplicative FIR (MFIR) filters [106], [107] which can efficiently approximate a given IIR filter type.

8. There would as well be merit in exploring a new class of IIFIR (FIR-IIR) filters, where the interpolation is performed by an IIR filter. This may provide further decrease in the complexity of the IFIR filter and may yield better performance.
Appendix A

Loop Information

A.1 Bellcore Loop Configurations

The structures of the Bellcore loops that were generated in chapter 5 are illustrated in Figure A.1 to Figure A.3. (CO = Central Office, Cust = Customer).
Figure A.1: Bellcore loops: Loop#1 to Loop#5.
Figure A.2: Bellcore loops: Loop#6 to Loop#10.
Figure A.3: Bellcore loops: Loop#11 to Loop#15.
A.2 Analysis of Hybrid Systems

Subscriber loops in the telephone loop plant can have different line gauges and may have bridged taps. In order to determine the transfer function of such lines/echo paths, a model is required.

A transmission line or any two port network (e.g., hybrid circuit) can be modeled by the ABCD parameters (transmission matrix) (see [112], [113]). Now, the block diagram of Figure 1.2 can be thought of as the model shown in Figure A.4. Using the ABCD parameters, the input impedance viewed from port 1 is given by

\[ Z_L = \frac{V_1}{I_1} = \frac{AV_2 + BI_2}{CV_2 + DI_2} = \frac{AZ_2 + B}{CZ_2 + D} \]  \hspace{1cm} (A.1)

Let \( V_F \) (far-end voltage) equals zero, then a transfer function is obtained as

\[ H(s) = \frac{V_1(s)}{V_T(s)} = \frac{Z_L}{Z_1 + Z_L} = \frac{AZ_2 + B}{CZ_2 Z_1 + DZ_1 + AZ_1 + B} \]  \hspace{1cm} (A.2)

Let \( V_T \) (near-end voltage) equals zero, then we obtain the transfer function of the channel as

\[ C(s) = \frac{V_1(s)}{V_F(s)} = \frac{(AD - BC)Z_1}{CZ_2 Z_1 + DZ_1 + AZ_1 + B} \]  \hspace{1cm} (A.3)

The echo path transfer function \( E(s) \) can be readily obtained by multiplying \( H(s) \) and \( C'(s) \)

\[ E(s) = \frac{(AZ_2 + B)(AD - BC)Z_1}{[CZ_2 Z_1 + DZ_1 + AZ_1 + B]^2} \]  \hspace{1cm} (A.4)

We assume that the echo path transfer function can be represented by an equivalent rational function as [109]

\[ E(s) = \beta_0 + \sum_i \frac{\beta_i s^i}{s + \lambda} \]  \hspace{1cm} (A.5)
Where \( \lambda \) depends on transformer parameters. The above equation reveals that the echo path transfer function contains a pole and zeros. Therefore, in order to completely cancel the echo path T.F., a similar rational function must be applied.

\[
V_T \quad Z_1 \quad V_L \quad 1 \quad A \quad B \quad 2 \quad Z_2 \quad V_F
\]

\[\text{Subscriber Line}\]

\[V_T = \text{Near-end Voltage}\]
\[V_R = \text{Received Voltage}\]
\[V_F = \text{Far-end Voltage}\]
\[V_L = \text{Attenuated Far-end Voltage}\]
\[Z = \text{Terminating Impedance}\]

Figure A.4: Model of a hybrid connected with subscriber line.
A.3 Cable Characteristics

In this appendix, the four data sheets from [102] that give the attributes by frequency are tabulated (Table A.1 to Table A.4). These data sheets are used\(^1\) to generate the impulse response of the echo path and the transmission path. These data sheets are utilized in the loop generating programs. The characteristic impedance \(Z_0\) of the line is

\[ Z_0 = \Re(Z_0) + j\Im(Z_0) \tag{A.6} \]

---

\(^1\)Merely data for polyethylene insulated cable (PIC) lines @ 70° F is used. This is in rapport with T1D1 specifications.
<table>
<thead>
<tr>
<th>Frequency in Hz</th>
<th>$R(Z_0)$ in ohms</th>
<th>$\mathcal{Y}(Z_0)$ in ohms</th>
<th>Gamma dB/mi</th>
<th>Gamma Deg/mi</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>10000</td>
<td>-10000</td>
<td>.000001</td>
<td>.000001</td>
</tr>
<tr>
<td>1</td>
<td>9182.32</td>
<td>-9127.69</td>
<td>.041</td>
<td>.27</td>
</tr>
<tr>
<td>5</td>
<td>4034.62</td>
<td>-4084.14</td>
<td>.092</td>
<td>.61</td>
</tr>
<tr>
<td>10</td>
<td>2888.76</td>
<td>-2885.55</td>
<td>.131</td>
<td>.85</td>
</tr>
<tr>
<td>15</td>
<td>2359.07</td>
<td>-2355.63</td>
<td>.16</td>
<td>1.06</td>
</tr>
<tr>
<td>20</td>
<td>2034.38</td>
<td>-2039.68</td>
<td>.183</td>
<td>1.22</td>
</tr>
<tr>
<td>30</td>
<td>1669</td>
<td>-1664.81</td>
<td>.226</td>
<td>1.5</td>
</tr>
<tr>
<td>50</td>
<td>1293.72</td>
<td>-1288.64</td>
<td>.292</td>
<td>1.93</td>
</tr>
<tr>
<td>70</td>
<td>1094.17</td>
<td>-1088.33</td>
<td>.345</td>
<td>2.29</td>
</tr>
<tr>
<td>100</td>
<td>916.43</td>
<td>-909.6</td>
<td>.412</td>
<td>2.72</td>
</tr>
<tr>
<td>150</td>
<td>749.6</td>
<td>-741.37</td>
<td>.504</td>
<td>3.36</td>
</tr>
<tr>
<td>200</td>
<td>650.34</td>
<td>-640.91</td>
<td>.581</td>
<td>3.89</td>
</tr>
<tr>
<td>300</td>
<td>539.29</td>
<td>-521.46</td>
<td>.709</td>
<td>4.75</td>
</tr>
<tr>
<td>500</td>
<td>415.75</td>
<td>-401.08</td>
<td>.909</td>
<td>6.21</td>
</tr>
<tr>
<td>700</td>
<td>355.9</td>
<td>-336.59</td>
<td>1.068</td>
<td>7.4</td>
</tr>
<tr>
<td>1000</td>
<td>299.3</td>
<td>-278.66</td>
<td>1.263</td>
<td>8.94</td>
</tr>
<tr>
<td>1500</td>
<td>248.78</td>
<td>-223.59</td>
<td>1.52</td>
<td>11.15</td>
</tr>
<tr>
<td>2000</td>
<td>219.22</td>
<td>-190.3</td>
<td>1.795</td>
<td>13.1</td>
</tr>
<tr>
<td>3000</td>
<td>185.51</td>
<td>-150.15</td>
<td>2.041</td>
<td>16.63</td>
</tr>
<tr>
<td>5000</td>
<td>153.94</td>
<td>-108.88</td>
<td>2.467</td>
<td>22.99</td>
</tr>
<tr>
<td>7000</td>
<td>183.92</td>
<td>-86.5</td>
<td>2.744</td>
<td>29.05</td>
</tr>
<tr>
<td>10000</td>
<td>127.32</td>
<td>-66.53</td>
<td>3.015</td>
<td>36.07</td>
</tr>
<tr>
<td>15000</td>
<td>118.6</td>
<td>-48.35</td>
<td>3.287</td>
<td>52.15</td>
</tr>
<tr>
<td>20000</td>
<td>114.61</td>
<td>-38.26</td>
<td>3.469</td>
<td>68.49</td>
</tr>
<tr>
<td>30000</td>
<td>110.69</td>
<td>-27.71</td>
<td>3.77</td>
<td>99.22</td>
</tr>
<tr>
<td>50000</td>
<td>107.73</td>
<td>-19.14</td>
<td>4.342</td>
<td>160.94</td>
</tr>
<tr>
<td>70000</td>
<td>106.26</td>
<td>-15.51</td>
<td>4.927</td>
<td>222.94</td>
</tr>
<tr>
<td>100000</td>
<td>104.74</td>
<td>-12.69</td>
<td>5.701</td>
<td>312.95</td>
</tr>
<tr>
<td>150000</td>
<td>102.94</td>
<td>-10.32</td>
<td>7.033</td>
<td>461.97</td>
</tr>
<tr>
<td>200000</td>
<td>101.63</td>
<td>-8.94</td>
<td>8.127</td>
<td>607.35</td>
</tr>
<tr>
<td>300000</td>
<td>100.02</td>
<td>-7.31</td>
<td>9.968</td>
<td>896.55</td>
</tr>
<tr>
<td>500000</td>
<td>98.32</td>
<td>-5.67</td>
<td>12.901</td>
<td>1468.89</td>
</tr>
<tr>
<td>700000</td>
<td>97.38</td>
<td>-4.8</td>
<td>15.292</td>
<td>2036.81</td>
</tr>
<tr>
<td>1000000</td>
<td>96.52</td>
<td>-4.02</td>
<td>18.315</td>
<td>2893.98</td>
</tr>
<tr>
<td>1500000</td>
<td>95.69</td>
<td>-3.29</td>
<td>22.482</td>
<td>4282.87</td>
</tr>
<tr>
<td>2000000</td>
<td>95.18</td>
<td>-2.83</td>
<td>26.002</td>
<td>5638.11</td>
</tr>
<tr>
<td>3000000</td>
<td>94.56</td>
<td>-2.33</td>
<td>31.919</td>
<td>8476.66</td>
</tr>
<tr>
<td>5000000</td>
<td>93.92</td>
<td>-1.8</td>
<td>41.333</td>
<td>14031.41</td>
</tr>
</tbody>
</table>

Table A.1: Cable characteristics for 19 AWG PIC line @ 70° F.
<table>
<thead>
<tr>
<th>Frequency in Hz</th>
<th>$R(Z_o)$ in ohms</th>
<th>$jX(Z_o)$ in ohms</th>
<th>Gamma dB/mi</th>
<th>Gamma Deg/mi</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000000</td>
<td>-14.000</td>
<td>0.000001</td>
<td>0.000001</td>
<td>0.000001</td>
</tr>
<tr>
<td>1</td>
<td>12927.71</td>
<td>-12929.34</td>
<td>0.059</td>
<td>0.39</td>
</tr>
<tr>
<td>5</td>
<td>5781.8</td>
<td>-5779.59</td>
<td>0.131</td>
<td>0.86</td>
</tr>
<tr>
<td>10</td>
<td>4086.69</td>
<td>-4086.44</td>
<td>0.185</td>
<td>1.29</td>
</tr>
<tr>
<td>15</td>
<td>3336.69</td>
<td>-3336.28</td>
<td>0.227</td>
<td>1.5</td>
</tr>
<tr>
<td>20</td>
<td>2891.65</td>
<td>-2889.05</td>
<td>0.362</td>
<td>1.73</td>
</tr>
<tr>
<td>30</td>
<td>2361.43</td>
<td>-2358.49</td>
<td>0.521</td>
<td>2.12</td>
</tr>
<tr>
<td>50</td>
<td>1825.81</td>
<td>-1826.24</td>
<td>0.414</td>
<td>2.73</td>
</tr>
<tr>
<td>70</td>
<td>1577.02</td>
<td>-1542.91</td>
<td>0.469</td>
<td>3.24</td>
</tr>
<tr>
<td>100</td>
<td>1295.02</td>
<td>-1290.21</td>
<td>0.585</td>
<td>3.67</td>
</tr>
<tr>
<td>150</td>
<td>1058.12</td>
<td>-1052.52</td>
<td>0.715</td>
<td>4.74</td>
</tr>
<tr>
<td>200</td>
<td>917.35</td>
<td>-910.71</td>
<td>0.825</td>
<td>5.48</td>
</tr>
<tr>
<td>300</td>
<td>750.35</td>
<td>-742.28</td>
<td>1.009</td>
<td>6.79</td>
</tr>
<tr>
<td>500</td>
<td>583.3</td>
<td>-572.94</td>
<td>1.298</td>
<td>8.71</td>
</tr>
<tr>
<td>700</td>
<td>494.74</td>
<td>-482.54</td>
<td>1.53</td>
<td>10.35</td>
</tr>
<tr>
<td>1000</td>
<td>416.16</td>
<td>-401.58</td>
<td>1.819</td>
<td>12.43</td>
</tr>
<tr>
<td>1500</td>
<td>342.84</td>
<td>-325.02</td>
<td>2.209</td>
<td>15.36</td>
</tr>
<tr>
<td>2000</td>
<td>299.57</td>
<td>-279.02</td>
<td>2.528</td>
<td>17.90</td>
</tr>
<tr>
<td>3000</td>
<td>249</td>
<td>-223.89</td>
<td>3.043</td>
<td>22.32</td>
</tr>
<tr>
<td>5000</td>
<td>199.83</td>
<td>-167.56</td>
<td>3.796</td>
<td>29.95</td>
</tr>
<tr>
<td>7000</td>
<td>174.89</td>
<td>-136.92</td>
<td>4.342</td>
<td>36.58</td>
</tr>
<tr>
<td>10000</td>
<td>154</td>
<td>-109.09</td>
<td>4.942</td>
<td>46.01</td>
</tr>
<tr>
<td>15000</td>
<td>136.34</td>
<td>-82.53</td>
<td>5.609</td>
<td>61.1</td>
</tr>
<tr>
<td>20000</td>
<td>127.25</td>
<td>-66.72</td>
<td>6.045</td>
<td>76.04</td>
</tr>
<tr>
<td>30000</td>
<td>111.68</td>
<td>-48.55</td>
<td>6.6</td>
<td>106.1</td>
</tr>
<tr>
<td>50000</td>
<td>101.12</td>
<td>-24.81</td>
<td>7.871</td>
<td>228.24</td>
</tr>
<tr>
<td>100000</td>
<td>107.24</td>
<td>-19.27</td>
<td>8.732</td>
<td>320.43</td>
</tr>
<tr>
<td>150000</td>
<td>105.52</td>
<td>-14.99</td>
<td>10.193</td>
<td>472.95</td>
</tr>
<tr>
<td>200000</td>
<td>104.38</td>
<td>-12.77</td>
<td>11.575</td>
<td>623.45</td>
</tr>
<tr>
<td>300000</td>
<td>102.56</td>
<td>-10.38</td>
<td>12.124</td>
<td>919.38</td>
</tr>
<tr>
<td>500000</td>
<td>100.38</td>
<td>-8.05</td>
<td>13.25</td>
<td>1499.65</td>
</tr>
<tr>
<td>700000</td>
<td>99.16</td>
<td>-6.81</td>
<td>21.617</td>
<td>2074.02</td>
</tr>
<tr>
<td>1000000</td>
<td>98.04</td>
<td>-5.7</td>
<td>25.873</td>
<td>2929.37</td>
</tr>
<tr>
<td>1500000</td>
<td>96.95</td>
<td>-4.66</td>
<td>31.714</td>
<td>4345.42</td>
</tr>
<tr>
<td>2000000</td>
<td>96.28</td>
<td>-4.04</td>
<td>36.698</td>
<td>5753.82</td>
</tr>
<tr>
<td>3000000</td>
<td>95.48</td>
<td>-3.3</td>
<td>43.021</td>
<td>8358.6</td>
</tr>
<tr>
<td>5000000</td>
<td>94.64</td>
<td>-2.56</td>
<td>58.244</td>
<td>14139.61</td>
</tr>
</tbody>
</table>

Table A.2: Cable characteristics for 22 AWG PIC line @ 70° F.
<table>
<thead>
<tr>
<th>Frequency in Hz</th>
<th>Re(Z₀) in ohms</th>
<th>Im(Z₀) in ohms</th>
<th>Gamma dB/mi</th>
<th>Gamma Deg/mi</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19000</td>
<td>-19000</td>
<td>0.0000001</td>
<td>0.000001</td>
</tr>
<tr>
<td>1</td>
<td>16306.82</td>
<td>-16297.63</td>
<td>0.074</td>
<td>49</td>
</tr>
<tr>
<td>5</td>
<td>7292.58</td>
<td>-7288.57</td>
<td>0.165</td>
<td>1.09</td>
</tr>
<tr>
<td>10</td>
<td>5156.83</td>
<td>-5153.61</td>
<td>0.234</td>
<td>1.54</td>
</tr>
<tr>
<td>15</td>
<td>4210.72</td>
<td>-4207.71</td>
<td>0.286</td>
<td>1.89</td>
</tr>
<tr>
<td>20</td>
<td>3646.77</td>
<td>-3643.81</td>
<td>0.39</td>
<td>2.18</td>
</tr>
<tr>
<td>30</td>
<td>2977.88</td>
<td>-2974.85</td>
<td>0.404</td>
<td>2.67</td>
</tr>
<tr>
<td>50</td>
<td>2307.15</td>
<td>-2303.82</td>
<td>0.522</td>
<td>3.45</td>
</tr>
<tr>
<td>70</td>
<td>1950.32</td>
<td>-1946.67</td>
<td>0.617</td>
<td>4.05</td>
</tr>
<tr>
<td>100</td>
<td>1632.29</td>
<td>-1628.16</td>
<td>0.738</td>
<td>4.88</td>
</tr>
<tr>
<td>150</td>
<td>1333.5</td>
<td>-1328.66</td>
<td>0.903</td>
<td>5.98</td>
</tr>
<tr>
<td>200</td>
<td>1155.49</td>
<td>-1150.09</td>
<td>1.042</td>
<td>6.9</td>
</tr>
<tr>
<td>300</td>
<td>944.5</td>
<td>-937.95</td>
<td>1.275</td>
<td>8.45</td>
</tr>
<tr>
<td>500</td>
<td>243.87</td>
<td>-218.18</td>
<td>4.95</td>
<td>36.43</td>
</tr>
<tr>
<td>700</td>
<td>621.1</td>
<td>-611.32</td>
<td>1.939</td>
<td>12.99</td>
</tr>
<tr>
<td>1000</td>
<td>521.4</td>
<td>-509.77</td>
<td>2.31</td>
<td>15.58</td>
</tr>
<tr>
<td>1500</td>
<td>428.12</td>
<td>-413.93</td>
<td>2.813</td>
<td>19.18</td>
</tr>
<tr>
<td>2000</td>
<td>372.85</td>
<td>-356.5</td>
<td>3.23</td>
<td>22.28</td>
</tr>
<tr>
<td>3000</td>
<td>307.87</td>
<td>-287.9</td>
<td>3.913</td>
<td>27.59</td>
</tr>
<tr>
<td>5000</td>
<td>243.87</td>
<td>-218.18</td>
<td>4.942</td>
<td>36.43</td>
</tr>
<tr>
<td>7000</td>
<td>210.75</td>
<td>-180.44</td>
<td>5.793</td>
<td>44.07</td>
</tr>
<tr>
<td>10000</td>
<td>182.25</td>
<td>-146.22</td>
<td>6.625</td>
<td>54.45</td>
</tr>
<tr>
<td>15000</td>
<td>157.04</td>
<td>-113.58</td>
<td>7.706</td>
<td>70.38</td>
</tr>
<tr>
<td>20000</td>
<td>143.29</td>
<td>-93.49</td>
<td>8.472</td>
<td>85.59</td>
</tr>
<tr>
<td>30000</td>
<td>128.79</td>
<td>-69.79</td>
<td>9.467</td>
<td>115.44</td>
</tr>
<tr>
<td>50000</td>
<td>117.22</td>
<td>-46.91</td>
<td>10.628</td>
<td>175.12</td>
</tr>
<tr>
<td>70000</td>
<td>112.62</td>
<td>-35.78</td>
<td>11.351</td>
<td>235.56</td>
</tr>
<tr>
<td>100000</td>
<td>109.5</td>
<td>-27</td>
<td>12.935</td>
<td>327.15</td>
</tr>
<tr>
<td>150000</td>
<td>107.21</td>
<td>-20.1</td>
<td>13.665</td>
<td>480.48</td>
</tr>
<tr>
<td>200000</td>
<td>105.96</td>
<td>-16.69</td>
<td>15.131</td>
<td>633.3</td>
</tr>
<tr>
<td>300000</td>
<td>104.31</td>
<td>-13.22</td>
<td>17.975</td>
<td>935.01</td>
</tr>
<tr>
<td>500000</td>
<td>102.12</td>
<td>-10.17</td>
<td>23.048</td>
<td>1525.69</td>
</tr>
<tr>
<td>700000</td>
<td>100.67</td>
<td>-8.6</td>
<td>27.284</td>
<td>2103.33</td>
</tr>
<tr>
<td>1000000</td>
<td>99.33</td>
<td>-7.2</td>
<td>32.639</td>
<td>2967.99</td>
</tr>
<tr>
<td>1500000</td>
<td>98.03</td>
<td>-5.88</td>
<td>40.026</td>
<td>4393.66</td>
</tr>
<tr>
<td>2000000</td>
<td>97.23</td>
<td>-5.1</td>
<td>46.267</td>
<td>5810.54</td>
</tr>
<tr>
<td>3000000</td>
<td>96.26</td>
<td>-4.17</td>
<td>56.753</td>
<td>8628.67</td>
</tr>
<tr>
<td>5000000</td>
<td>95.26</td>
<td>-3.23</td>
<td>73.405</td>
<td>14231.83</td>
</tr>
</tbody>
</table>

Table A.3: Cable characteristics for 24 AWG PIC line @ 70° F.
<table>
<thead>
<tr>
<th>Frequency in Hz</th>
<th>$\Re(Z_e)$ in ohms</th>
<th>$\Im(Z_e)$ in ohms</th>
<th>Gamma dB/mi</th>
<th>Gamma Deg/mi</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>25000</td>
<td>-25000</td>
<td>0.0000001</td>
<td>0.0000001</td>
</tr>
<tr>
<td>1</td>
<td>20562.46</td>
<td>-20551.04</td>
<td>0.093</td>
<td>0.61</td>
</tr>
<tr>
<td>5</td>
<td>9195.5</td>
<td>-9190.92</td>
<td>0.906</td>
<td>1.37</td>
</tr>
<tr>
<td>10</td>
<td>6502.38</td>
<td>-6498.86</td>
<td>1.294</td>
<td>1.94</td>
</tr>
<tr>
<td>15</td>
<td>5309.3</td>
<td>-5306.16</td>
<td>3.61</td>
<td>2.38</td>
</tr>
<tr>
<td>20</td>
<td>4398.12</td>
<td>-4395.14</td>
<td>4.16</td>
<td>2.75</td>
</tr>
<tr>
<td>30</td>
<td>3754.58</td>
<td>-3751.69</td>
<td>5.1</td>
<td>3.36</td>
</tr>
<tr>
<td>50</td>
<td>2908.66</td>
<td>-2905.67</td>
<td>6.58</td>
<td>4.34</td>
</tr>
<tr>
<td>70</td>
<td>2458.6</td>
<td>-2455.41</td>
<td>7.79</td>
<td>5.14</td>
</tr>
<tr>
<td>100</td>
<td>2057.93</td>
<td>-2053.99</td>
<td>9.31</td>
<td>6.15</td>
</tr>
<tr>
<td>150</td>
<td>1680.47</td>
<td>-1676.45</td>
<td>1.139</td>
<td>7.53</td>
</tr>
<tr>
<td>200</td>
<td>1455.83</td>
<td>-1451.35</td>
<td>1.315</td>
<td>8.7</td>
</tr>
<tr>
<td>300</td>
<td>1189.51</td>
<td>-1184.2</td>
<td>1.61</td>
<td>10.66</td>
</tr>
<tr>
<td>500</td>
<td>922.69</td>
<td>-916</td>
<td>2.075</td>
<td>13.75</td>
</tr>
<tr>
<td>700</td>
<td>780.91</td>
<td>-783.09</td>
<td>2.452</td>
<td>16.33</td>
</tr>
<tr>
<td>1000</td>
<td>654.74</td>
<td>-654.46</td>
<td>2.924</td>
<td>19.56</td>
</tr>
<tr>
<td>1500</td>
<td>536.48</td>
<td>-525.19</td>
<td>3.569</td>
<td>24.04</td>
</tr>
<tr>
<td>2000</td>
<td>466.25</td>
<td>-453.25</td>
<td>4.107</td>
<td>27.86</td>
</tr>
<tr>
<td>3000</td>
<td>383.39</td>
<td>-367.51</td>
<td>4.995</td>
<td>34.36</td>
</tr>
<tr>
<td>5000</td>
<td>301.19</td>
<td>-280.76</td>
<td>6.36</td>
<td>44.99</td>
</tr>
<tr>
<td>7000</td>
<td>258.16</td>
<td>-234.04</td>
<td>7.422</td>
<td>59.99</td>
</tr>
<tr>
<td>10000</td>
<td>229.58</td>
<td>-191.85</td>
<td>8.692</td>
<td>63.9</td>
</tr>
<tr>
<td>15000</td>
<td>186.46</td>
<td>-151.47</td>
<td>10.249</td>
<td>83.56</td>
</tr>
<tr>
<td>20000</td>
<td>167.08</td>
<td>-126.94</td>
<td>11.503</td>
<td>99.83</td>
</tr>
<tr>
<td>30000</td>
<td>145.71</td>
<td>-97.35</td>
<td>13.232</td>
<td>130.60</td>
</tr>
<tr>
<td>50000</td>
<td>127.25</td>
<td>-67.47</td>
<td>15.285</td>
<td>190.1</td>
</tr>
<tr>
<td>70000</td>
<td>119.21</td>
<td>-52.05</td>
<td>16.509</td>
<td>249.33</td>
</tr>
<tr>
<td>100000</td>
<td>113.56</td>
<td>-39.12</td>
<td>17.726</td>
<td>339.29</td>
</tr>
<tr>
<td>150000</td>
<td>109.61</td>
<td>-28.32</td>
<td>19.253</td>
<td>491.26</td>
</tr>
<tr>
<td>200000</td>
<td>107.8</td>
<td>-22.81</td>
<td>20.676</td>
<td>644.17</td>
</tr>
<tr>
<td>300000</td>
<td>105.92</td>
<td>-17.35</td>
<td>23.590</td>
<td>949.49</td>
</tr>
<tr>
<td>500000</td>
<td>103.87</td>
<td>-12.91</td>
<td>29.264</td>
<td>1551.77</td>
</tr>
<tr>
<td>700000</td>
<td>102.45</td>
<td>-10.86</td>
<td>34.177</td>
<td>2142.83</td>
</tr>
<tr>
<td>1000000</td>
<td>100.9</td>
<td>-9.09</td>
<td>37.189</td>
<td>3014.78</td>
</tr>
<tr>
<td>1500000</td>
<td>99.35</td>
<td>-7.42</td>
<td>50.493</td>
<td>4452.71</td>
</tr>
<tr>
<td>2000000</td>
<td>98.39</td>
<td>-6.43</td>
<td>58.348</td>
<td>5879.86</td>
</tr>
<tr>
<td>3000000</td>
<td>97.23</td>
<td>-5.26</td>
<td>71.551</td>
<td>8715.35</td>
</tr>
<tr>
<td>5000000</td>
<td>96.02</td>
<td>-4.08</td>
<td>92.543</td>
<td>14345.97</td>
</tr>
</tbody>
</table>

Table A.4: Cable characteristics for 26 AWG PIC line @ 70° F.
Appendix B

Simulations Information

B.1 Interpolator Specifications

We have designed diverse interpolator filters which were used in the simulations of chapter 5. As explained before, the design method of the interpolator filters is based on the tail frequency response. It is pointless to list the design of all interpolators. In this appendix, we provide the necessary informations that were used in the interpolator design. As an example, we give the vital characteristics of the multiband *compromise* interpolator in Figure B.1. The passband width and the various stopbands are provided in Table B.1.

<table>
<thead>
<tr>
<th>Passband edge</th>
<th>7.81</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stopband#1</td>
<td>58.86 → 74.48</td>
</tr>
<tr>
<td>Stopband#2</td>
<td>125.52 → 141.14</td>
</tr>
<tr>
<td>Stopband#3</td>
<td>192.19 → 207.89</td>
</tr>
<tr>
<td>Stopband#4</td>
<td>258.86 → 274.48</td>
</tr>
<tr>
<td>Stopband#5</td>
<td>325.52 → 341.14</td>
</tr>
<tr>
<td>Stopband#6</td>
<td>392.19 → 407.81</td>
</tr>
<tr>
<td>Stopband#7</td>
<td>458.86 → 474.48</td>
</tr>
</tbody>
</table>

Table B.1: The passband and stopband widths of compromise interpolator.
### Characteristics of Designed Filter

<table>
<thead>
<tr>
<th></th>
<th>Band 1</th>
<th>Band 2</th>
<th>Band 3</th>
<th>Band 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lower Band Edge</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0001</td>
<td>0.0002</td>
</tr>
<tr>
<td>Upper Band Edge</td>
<td>0.0000</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0002</td>
</tr>
<tr>
<td>Nominal Gain</td>
<td>1.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Nominal Ripple</td>
<td>1.700</td>
<td>1.700</td>
<td>1.700</td>
<td>1.700</td>
</tr>
<tr>
<td>Maximum Ripple</td>
<td>0.0196</td>
<td>0.0200</td>
<td>0.0196</td>
<td>0.0196</td>
</tr>
<tr>
<td>Ripple in dB</td>
<td>-16.90</td>
<td>-33.9887</td>
<td>-34.1333</td>
<td>-34.1333</td>
</tr>
<tr>
<td>Lower Band Edge</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0004</td>
<td>0.0005</td>
</tr>
<tr>
<td>Upper Band Edge</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0004</td>
<td>0.0005</td>
</tr>
<tr>
<td>Nominal Gain</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Nominal Ripple</td>
<td>1.700</td>
<td>1.700</td>
<td>1.700</td>
<td>1.700</td>
</tr>
<tr>
<td>Maximum Ripple</td>
<td>0.0196</td>
<td>0.0196</td>
<td>0.0197</td>
<td>0.0196</td>
</tr>
<tr>
<td>Ripple in dB</td>
<td>-34.0521</td>
<td>-34.1332</td>
<td>-34.1318</td>
<td>-34.1322</td>
</tr>
</tbody>
</table>

**Figure B.1**: Compromise interpolator characteristics and magnitude response.
B.2 Learning Curves

The convergence curves of the two-stage echo canceller is given here. Figure B.2 illustrates the graph of the mean squared error as a function of the baud period. The adaptation step-size $\mu = \alpha = 5.56 \times 10^{-4}$. 
Figure B.2: Convergence curves of the two-stage echo canceller: MSE vs Baud periods.
Appendix C

Proofs

C.1 Is R Diagonal Matrix?

Here we provide a pseudo proof that $R$ is a diagonal matrix for high values of $L$.

Since the input process $X$ is white (i.e., uncorrelated), then in order for $R$ to be a diagonal matrix, the following must be fulfilled

$$E[m_im_j] = \begin{cases} \sigma_i^2\|G\|^2 & i = j \\ 0 & i \neq j \end{cases} \quad \text{(C.1)}$$

There could be a situation when the matrix $R$ has non-zero off diagonal terms if the samples of the process $m_i$ overlap (i.e., correlate or non-orthogonal) with the samples of the process $m_j$. Therefore, there will be correlation between $m_i$ and $m_j$ as $i \to j$ and no correlation if $i \ll j$. Consequently, the value of $L$ will decide if $i \to j$ or $i \ll j$. Therefore, the possibility of correlation is highest in the samples of the product $m_0m_L$. The samples of the product $m_0m_{BL}$ will far apart and as a result uncorrelated. Thus, by selecting high values of the parameter $L$, we can be sure that the samples of the product $m_0m_L$ are far apart; therefore uncorrelated.
Bibliography


192


197


