Simultaneous Multi Curve Approximation with NURBS

by

Toni Prahasto

A thesis

presented to the University of Waterloo

in fulfilment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Mechanical Engineering

Waterloo, Ontario, Canada, 1998

©Toni Prahasto 1998

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

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

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

0-612-38292-3

Canada

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

The design of wing and turbine blades requires simultaneous approximation of sectional airfoils. These airfoils are specified as a collection of data points. From the perspective of flow performance the airfoils should be approximated to within a prespecified tolerance. Furthermore, to facilitate skinning, the curves must share a mutual degree and a mutual knot vector, i.e., they must be compatible.

This thesis aims to provide a method for creating a Non Uniform Rational B-spline (NURBS) skeleton of wing/blade from the sectional data to facilitate the skinning process and to serve as an aid in the design process. The goal of this method is to simulatneously fit sectional data with NURBS curves that approximate it to a pre-specified tolerance with a minimum number of parameters (knots, weights and control points).

This is achieved in two steps. First a skeleton of compatible curves that approximate the sectional data is constructed. Second, the curve approximation is modified to satisfy the tolerance requirement. The generation of the skeletal curves is based on nonlinear least square optimization. The method uses the BFGS descent direction to overcome the lethargic property and uses the condition number of the least square matrix to ensure the good behaviour of control points. The method for satisfying prespecified tolerance is based on identifying the knot span, that contains the most number of data points outside the pre-specified tolerance, and inserting additional knots in them. The process is continued until all points are within the prespecified tolerance. This method is called the tolerance based knot insertion method. This two step process is designed to ensure that the compatibility of the

skeletal curves is maintained at all times.

The method for fitting skeletal curves is tested on three single-section cases and three multi-section cases. The tests showed that, for the single-section cases, the maximum error decreased by a factor of a least five , and up to ten, from the initial maximum error. For multi-section cases, the maximum error decreased by a factor between two, and up to sixteen, from the initial maximum error. Based on these tests it is concluded that parametization of data does not contribute to reduction in least square error. On the other hand, knots contribute the most to the reduction of least square error.

The tolerance-based knot insertion was tested on three skeletons and proved to be successful in satisfying the tolerance with relatively low number of parameters. The tests showed that proposed method reduced the numbers of control points of the compatible curves that satisfy the prespecified tolerances between 20% to 40% in comparison to the existing methods.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professors Sanjeev Bedi, who supported me intellectually, financially, and emotionally throughout the period of my study. I hope that someday I can provide the same precious gift to someone else. Sincere thanks go to my examination committee: Professors H. El-Maraghy, Professors R.H. Bartels, Professors J.H.G. Howard, Professors F. Ismail, and Professors K. Ponnambalam, who donated their time and energy to perform a valuable function in my education.

Special acknowledgements go to the Islamic Development Bank of Saudi Arabia which provided the greatest part of the financial support for this study. My acknowledgements also goes to the Department of Cultural and Education of Republic of Indonesia for the financial support from September 1996 to April 1997.

I would like to convey special thanks to om Charlie and tante Yelli for their support, generosity, and hospitality, during the final period of my study in Waterloo. The company of Alexander Soegiarso during the last two years of my study is cherished. I greatly thank my friend, Dr. Sultan Siddique, for the valuable discussions during my study.

To my family: I would like to thank my parents for their love and support throughout all of those years of my education. Finally, I would like to express my greatest thanks to my wife Amie, my daughter Kania, and my son Satrio, for their great amount of love and support without which this achievement can never be realized. I am indebted forever.

Contents

1 Introduction 1

1.1 Definitions of Airfoil and Tolerance 3

1.2 Motivation 5

1.2.1 Airfoil Design Process 5

1.2.2 Wing Design Process 9

1.3 Proposed Approach 13

1.4 Objective of Research 15

1.5 Organization of Thesis 17

2 Curve Fitting: Basics and Literature Review 20

2.1 Least Squares Approximation 20

2.2 Parameters of NURBS Least Squares 22

2.3 Elimination of Control Points 24

2.4 Derivatives of Residual 27

2.5 Literature Survey 32

2.5.1 Free Knot Approach 33

2.5.2 Optimization of NURBS 35

2.5.3 Separation of Weights 37
2.5.4 Optimization of Parameterization 40
2.5.5 Survey on Parameterization of Data 42
2.6 Summary 44

3 Nonlinear Least Squares Method **45**
3.1 Constraints 46
3.1.1 Sequence of Data 46
3.1.2 Constraints Due To The Use of The B-spline Basis Functions 48
3.1.3 Condition Number of Matrix **R** 49
3.1.4 Summary of Constraints 58
3.2 Method To Satisfy Constraints 59
3.2.1 Method To Satisfy The Linear Constraint 60
3.2.2 Method To Satisfy The Nonlinear Constraint 61
3.3 Descent Direction 65
3.4 Dropping Linear Constraint from Active Set 67

4 Nonlinear NURBS Fitting **69**
4.1 NACA 2415 Airfoil 71
4.1.1 Optimization of Knots 71
4.1.2 Optimization of Weights 77
4.1.3 Optimization of Parameterization for Data 81
4.1.4 Optimization of Knots and Weights 85
4.1.5 Optimization of Knots, Weights and Parameterization 90
4.2 WTEA Airfoil 94

4.2.1 Optimization of Knots 95
4.2.2 Optimization of Weights 100
4.2.3 Optimization of Parameterization 104
4.2.4 Optimization of Knots and Weights 108
4.2.5 Optimization of Knots, Weights and Parameterization 113
4.3 Axial Compressor 117
4.3.1 Optimization of Knots 117
4.3.2 Optimization of Weights 123
4.3.3 Optimization of Parameterization for Data 127
4.3.4 Optimization of Knots and Weights 131
4.3.5 Optimization of Knots, Weights and Parameterization 136
4.4 Discussion 140

5 Multicurve Least Squares Method **145**
5.1 Introduction 145
5.1.1 Definition of Objective Function 146
5.1.2 Optimization Parameters 147
5.1.3 Linear and Nonlinear Constraints 147
5.1.4 Initialization of Knots 149
5.2 Wing Airfoils 153
5.2.1 Optimization of Knots 153
5.2.2 Optimization of Weights 159
5.2.3 Optimization of Parameterization 163
5.2.4 Optimization of Knots and Weights 168

5.2.5 Optimization of Knots, Weights, and Parameterization . . . 173
5.3 Compressor Blades . . . 178
5.3.1 Optimization of Knots . . . 178
5.3.2 Optimization of Weights . . . 187
5.3.3 Optimization of Parameterization . . . 195
5.3.4 Optimization of Knots and Weights . . . 203
5.3.5 Optimization of Knots, Weights, and Parameterization . . . 211
5.4 Turbine Blades . . . 219
5.4.1 Optimization of Knots . . . 220
5.4.2 Optimization of Weights . . . 228
5.4.3 Optimization of Parameterization . . . 236
5.4.4 Optimization of Knots and Weights . . . 244
5.4.5 Optimization of Knots, Weights, and Parameterization . . . 252
5.5 Discussion . . . 252

6 Tolerance Based Knot Insertion 265

6.1 Literature Survey . . . 266
6.1.1 Hierarchical Fitting . . . 266
6.1.2 Method Based on Knot Removal . . . 271
6.2 Tolerance Based Knot Insertion . . . 275
6.3 Experiment . . . 279
6.3.1 Wing Skeletal Curves . . . 279
6.3.2 Compressor Skeletal Curves . . . 280
6.3.3 Turbine Skeletal Curves . . . 282

6.4 Discussion 288
6.5 Attempt of Smoothing 291
6.6 Skinning 293
6.6.1 Integral B-spline Case 293
6.6.2 Rational B-spline Case 297

7 Conclusions and Future Work **299**
7.1 Achievements 301
7.2 Future Work 305

Bibliography **307**

A Non Uniform Rational B-Spline Basics **318**
A.1 B-spline Basis Functions 319
A.1.1 Definition of B-spline Basis 319
A.1.2 Properties of B-spline Basis 320
A.1.3 Derivatives of B-spline Basis 322
A.2 NURBS Curves and Surfaces 324
A.2.1 Rational B-spline Basis Function 324
A.2.2 NURBS Curves 326
A.2.3 NURBS Surfaces 327
A.3 Construction of Curves and Surfaces 328

B Curve Smoothing **331**
B.1 Partial Smoothing 335

C Basics of Skinning 336

List of Figures

1.1 Blade Construction 2

1.2 Simplified Procedure of Airfoil Design 7

1.3 Two Dimensional Flow Around an Airfoil 8

1.4 Simplified Procedure of Wing Design 10

1.5 Wing Geometry 11

2.1 Error Vector and Correction Coefficient 42

3.1 Significance of Topology 46

3.2 Vanishing Weight and Rational B-spline Basis 52

3.3 The B-spline Basis Function $N_5^4(t;\mathbf{u})$ and Its Supporting Data t_{11} and t_{12} 54

3.4 Region of High Condition Number due to $\mathbf{t}$ 56

3.5 Region of High Condition Number due to $\mathbf{u}$ 57

3.6 Reduction of Critical Step To Satisfy The Nonlinear Constraint . . 62

4.1 NACA 2415 with 12 mm chordlength 72

4.2 Reduction of Error of NACA airfoil - optimization of $\mathbf{u}$ 73

4.3 Error distribution of NACA airfoil - optimization of **u** 74
4.4 Knots of NACA airfoil - optimization of **u** 75
4.5 Control Polygon of NACA airfoil - optimization of **u** 76
4.6 Reduction of Error of NACA airfoil - optimization of **w** 78
4.7 Error distribution of NACA airfoil - optimization of **w** 79
4.8 Control Polygon of NACA airfoil - optimization of **w** 80
4.9 Reduction of Error of NACA airfoil - optimization of **t** 82
4.10 Error distribution of NACA airfoil - optimization of **t** 83
4.11 Control Polygon of NACA airfoil - optimization of **t** 84
4.12 Reduction of Error of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 86
4.13 Error distribution of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 87
4.14 Knots of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 88
4.15 Control Polygon of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 89
4.16 Reduction of Error of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ . . . 91
4.17 Error distribution of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ 92
4.18 Control Polygon of NACA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ 93
4.19 WTEA airfoil 94
4.20 Reduction of Error of WTEA airfoil - optimization of **u** 96
4.21 Error distribution of WTEA airfoil - optimization of **u** 97
4.22 Knots of WTEA airfoil - optimization of **u** 98
4.23 Control Polygon of WTEA airfoil - optimization of **u** 99
4.24 Reduction of Error of WTEA airfoil - optimization of **w** 101
4.25 Error distribution of WTEA airfoil - optimization of **w** 102

4.26 Control Polygon of WTEA airfoil - optimization of **w** 103
4.27 Reduction of Error of WTEA airfoil - optimization of **t** 105
4.28 Error distribution of WTEA airfoil - optimization of **t** 106
4.29 Control Polygon of WTEA airfoil - optimization of **t** 107
4.30 Reduction of Error of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 109
4.31 Error distribution of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 110
4.32 Knots of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 111
4.33 Control Polygon of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 112
4.34 Reduction of Error of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ . . . 114
4.35 Error distribution of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ 115
4.36 Control Polygon of WTEA airfoil - optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ 116
4.37 Blade Section of Concepts ETI 118
4.38 Reduction of Error of compressor blade's airfoil - optimization of **u** 119
4.39 Error distribution of compressor blade's airfoil - optimization of **u** . 120
4.40 Knots of compressor blade's airfoil - optimization of **u** 121
4.41 Control Polygon of compressor blade's airfoil - optimization of **u** . . 122
4.42 Reduction of Error of compressor blade's airfoil - optimization of **w** 124
4.43 Error distribution of compressor blade's airfoil - optimization of **w** . 125
4.44 Control Polygon of compressor blade's airfoil - optimization of **w** . . 126
4.45 Reduction of Error of compressor blade's airfoil - optimization of **t** . 128
4.46 Error distribution of compressor blade's airfoil - optimization of **t** . 129
4.47 Control Polygon of compressor blade's airfoil - optimization of **t** . . 130
4.48 Reduction of Error of compressor blade's airfoil - optimization of $(\mathbf{u}, \mathbf{w})$ 132

4.49 Error distribution of compressor blade's airfoil - optimization of (u, w) 133
4.50 Knots of compressor blade's airfoil - optimization of (u, w) 134
4.51 Control Polygon of compressor blade's airfoil - optimization of (u, w) 135
4.52 Reduction of Error of compressor blade's airfoil - optimization of (u, w, t) 137
4.53 Error distribution of compressor blade's airfoil - optimization of (u, w, t) 138
4.54 Control Polygon of compressor blade's airfoil - optimization of (u, w, t) 139
4.55 Reduction of Maximum Error After Optimization 142

5.1 Failure of Knot Initialization 150
5.2 Data of NACA 2415 and WTEA Airfoils 154
5.3 Wing airfoils with an optimization of u: Reduction of error 155
5.4 Wing airfoils with an optimization of u: Error distribution of NACA airfoil 156
5.5 Wing airfoils with an optimization of u: Error distribution of WTEA airfoil 157
5.6 NACA and WTEA airfoils after optimization of u 158
5.7 Wing airfoils with an optimization of w: Reduction of error 159
5.8 Wing airfoils with an optimization of w: Error distribution of NACA airfoil 160
5.9 Wing airfoils with an optimization of w: Error distribution of WTEA airfoil 161
5.10 NACA and WTEA Airfoils After Optimization of w 162

5.11 Wing airfoils with an optimization of **t**: Reduction of error 164
5.12 Wing airfoils with an optimization of **t**: Error distribution of NACA airfoil 165
5.13 Wing airfoils with an optimization of **t**: Error distribution of WTEA airfoil 166
5.14 NACA and WTEA Airfoils After Optimization of **t** 167
5.15 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w})$: Reduction of error . . 169
5.16 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w})$: Error distribution of NACA airfoil 170
5.17 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w})$: Error distribution of WTEA airfoil 171
5.18 NACA and WTEA airfoils after optimization of **u** 172
5.19 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Reduction of error . 174
5.20 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Error distribution of NACA airfoil 175
5.21 Wing airfoils with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Error distribution of WTEA airfoil 176
5.22 NACA and WTEA airfoils after optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$ 177
5.23 Data of Compressor 179
5.24 Compressor blade's airfoils with an optimization of **u**: Reduction of error 180
5.25 Hub airfoil of compressor blade with an optimization of **u**: Error distribution 181

5.26 Mid section airfoil of compressor blade with an optimization of u: Error distribution 182
5.27 Tip airfoil of compressor blade with an optimization of u: Error distribution 183
5.28 Hub airfoil of compressor blade with an optimization of u: fitted curves 184
5.29 Mid section airfoil of compressor blade with an optimization of u: fitted curves 185
5.30 Tip airfoil of compressor blade with an optimization of u: fitted curves 186
5.31 Compressor blade's airfoils with an optimization of w: Reduction of error 188
5.32 Hub airfoil of compressor blade with an optimization of w: Error distribution 189
5.33 Mid section airfoil of compressor blade with an optimization of w: Error distribution 190
5.34 Tip airfoil of compressor blade with an optimization of w: Error distribution 191
5.35 Hub airfoil of compressor blade with an optimization of w: fitted curves 192
5.36 Mid section airfoil of compressor blade with an optimization of w: fitted curves 193
5.37 Tip airfoil of compressor blade with an optimization of w: fitted curves 194
5.38 Compressor blade's airfoils with an optimization of t: Reduction of error 196

5.39 Hub airfoil of compressor blade with an optimization of t: Error distribution 197
5.40 Mid section airfoil of compressor blade with an optimization of t: Error distribution 198
5.41 Tip airfoil of compressor blade with an optimization of t: Error distribution 199
5.42 Hub airfoil of compressor blade with an optimization of t: fitted curves 200
5.43 Mid section airfoil of compressor blade with an optimization of t: fitted curves 201
5.44 Tip airfoil of compressor blade with an optimization of t: fitted curves 202
5.45 Compressor blade's airfoils with an optimization of (u, w): Reduction of error 204
5.46 Hub airfoil of compressor blade with an optimization of (u, w): Error distribution 205
5.47 Mid section airfoil of compressor blade with an optimization of (u, w): Error distribution 206
5.48 Tip airfoil of compressor blade with an optimization of (u, w): Error distribution 207
5.49 Hub airfoil of compressor blade with an optimization of (u, w): fitted curves 208
5.50 Mid section airfoil of compressor blade with an optimization of (u, w): fitted curves 209

5.51 Tip airfoil of compressor blade with an optimization of (u, w): fitted curves 210
5.52 Compressor blade's airfoils with an optimization of (u, w, t): Reduction of error 212
5.53 Hub airfoil of compressor blade with an optimization of (u, w, t): Error distribution 213
5.54 Mid section airfoil of compressor blade with an optimization of (u, w, t): Error distribution 214
5.55 Tip airfoil of compressor blade with an optimization of (u, w, t): Error distribution 215
5.56 Hub airfoil of compressor blade with an optimization of (u, w, t): fitted curves 216
5.57 Mid section airfoil of compressor blade with an optimization of (u, w, t): fitted curves 217
5.58 Tip airfoil of compressor blade with an optimization of (u, w, t): fitted curves 218
5.59 Data of Turbine Blades 219
5.60 Turbine blade's airfoils with an optimization of u: Reduction of error 221
5.61 Hub airfoil of turbine blade with an optimization of u: Error distribution 222
5.62 Mid section airfoil of turbine blade with an optimization of u: Error distribution 223

5.63 Tip airfoil of turbine blade with an optimization of u: Error distribution 224
5.64 Hub airfoil of turbine blade with an optimization of u: fitted curves 225
5.65 Mid section airfoil of turbine blade with an optimization of u: fitted curves 226
5.66 Tip airfoil of turbine blade with an optimization of u: fitted curves 227
5.67 Turbine blade's airfoils with an optimization of w: Reduction of error 229
5.68 Hub airfoil of turbine blade with an optimization of w: Error distribution 230
5.69 Mid section airfoil of turbine blade with an optimization of w: Error distribution 231
5.70 Tip airfoil of turbine blade with an optimization of w: Error distribution 232
5.71 Hub airfoil of turbine blade with an optimization of w: fitted curves 233
5.72 Mid section airfoil of turbine blade with an optimization of w: fitted curves 234
5.73 Tip airfoil of turbine blade with an optimization of w: fitted curves 235
5.74 Turbine blade's airfoils with an optimization of t: Reduction of error 237
5.75 Hub airfoil of turbine blade with an optimization of t: Error distribution 238
5.76 Mid section airfoil of turbine blade with an optimization of t: Error distribution 239
5.77 Tip airfoil of turbine blade with an optimization of t: Error distribution 240

5.78 Hub airfoil of turbine blade with an optimization of **t**: fitted curves 241
5.79 Mid section airfoil of turbine blade with an optimization of **t**: fitted curves 242
5.80 Tip airfoil of turbine blade with an optimization of **t**: fitted curves . 243
5.81 Turbine blade's airfoils with an optimization of $(\mathbf{u}, \mathbf{w})$: Reduction of error 245
5.82 Hub airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: Error distribution 246
5.83 Mid section airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: Error distribution 247
5.84 Tip airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: Error distribution 248
5.85 Hub airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: fitted curves 249
5.86 Mid section airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: fitted curves 250
5.87 Tip airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w})$: fitted curves 251
5.88 Turbine blade's airfoils with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Reduction of error 253
5.89 Hub airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Error distribution 254

5.90 Mid section airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Error distribution 255
5.91 Tip airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: Error distribution 256
5.92 Hub airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: fitted curves 257
5.93 Mid section airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: fitted curves 258
5.94 Tip airfoil of turbine blade with an optimization of $(\mathbf{u}, \mathbf{w}, \mathbf{t})$: fitted curves 259

6.1 The Shaded Regions are Out-Of-Tolerance Region 267
6.2 Line Segments of Out-Of-Tolerance Region 270
6.3 Merging of Out-Of-Tolerance Line Segments 271
6.4 Error Distribution of Integral Wing Skeleton 280
6.5 Error Distribution of Rational Wing Skeleton 281
6.6 Control Points of Integral Wing Skeleton 281
6.7 Control Points of Rational Wing Skeleton 282
6.8 Error Distribution of Integral Compressor Skeleton 283
6.9 Error Distribution of Rational Compressor Skeleton 284
6.10 Control Points of Integral Compressor Skeleton 284
6.11 Control Points of Rational Compressor Skeleton 285
6.12 Error Distribution of Integral Turbine Skeleton 286
6.13 Error Distribution of Rational Turbine Skeleton 287

6.14 Control Points of Integral Turbine Skeleton 287
6.15 Control Points of Rational Turbine Skeleton 288
6.16 Curvature Distribution at Knots of NACA 2415. Changes of Signs at u_{14} and u_{18} causing Inflection Points 292
6.17 Smoothed NACA 2415 294
6.18 Partially Smoothed NACA 2415 295
6.19 Integral B-spline Surface of Wing 296
6.20 Integral B-spline Surface of Compressor blade 296
6.21 Integral B-spline Surface of Turbine blade 297
6.22 Rational B-spline Surface of Wing 297
6.23 Rational B-spline Surface of Compressor blade 298
6.24 Rational B-spline Surface of Turbine blade 298

7.1 Comparison of Existing and Proposed Approaches 300
7.2 Turbine Design Procedure proposed by Hourmouziadis[46] 306

A.1 Double Knots, $u_j = u_{j+1}$ 324
A.2 Triple Knots 324

C.1 Control points of Section Curve 337

List of Tables

4.1 Least Square Error 140

4.2 Maximum Distance Between Data and Curves 140

4.3 Number of Iteration for Single Curves 141

5.1 Least Square Error 254

5.2 Maximum Distance Between Data and Curves 260

5.3 Number of Iteration for Skeletal Curves 260

5.4 Reduction of Maximum Distance 262

6.1 Sources of skeleton 279

6.2 Comparison between Proposed Method and Piegl's Method 288

6.3 Results of Individual Curve Fitting 289

Nomenclature

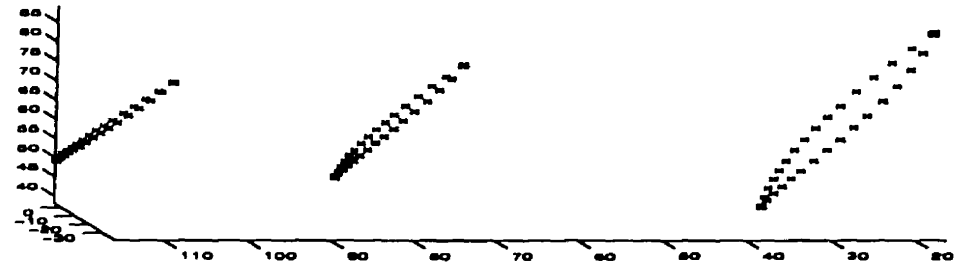
ε	least-squares error
N_i^k	the i-th B-spline basis function of order k
R_i^k	the i-th rational B-spline basis function of order k
$\mathbf{I}$	identity matrix
$\mathbf{R}$	matrix of rational B-spline basis function
$\mathbf{R}^T$	pseudoinverse matrix of matrix $\mathbf{R}$
$\mathbf{e}$	vector of Euclidean distances
$\mathbf{c}$	control points
$\mathbf{p}$	data points
$\mathbf{u}$	knot vector
$\mathbf{t}$	vector of parameterization of data points
$\mathbf{w}$	vector of weights of control points

Chapter 1

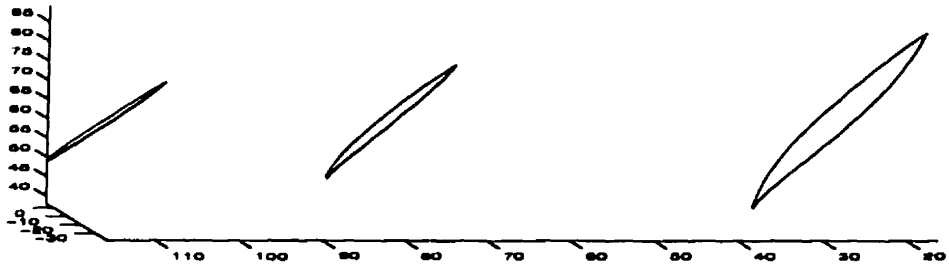
Introduction

The goal of this thesis is to present a new approach of constructing a set of Non-Uniform Rational B-spline (NURBS) curve/curves that approximate a given data set to within a prespecified tolerance while satisfying the compatibility condition. The compatibility condition refers to a constraint that the curves share a mutual degree and a mutual knot vector. compatibility condition is required to allow skinning of the curves without merging their knots. This goal is motivated by a need to develop a set of skeletal curves that can be skinned into a surface. Figure 1.1(a) shows an example of a data set that has been fitted with curves shown in Figure 1.1(b) which approximate the data set to within a prespecified accuracy while satisfying the compatibility condition between the curves. Figure 1.1(c) shows the surface constructed by skinning the skeletal curves in Figure 1.1(b). The goal of eliminating the knot merging process prior to skinning is to reduce the number of control points of the skinned surface.

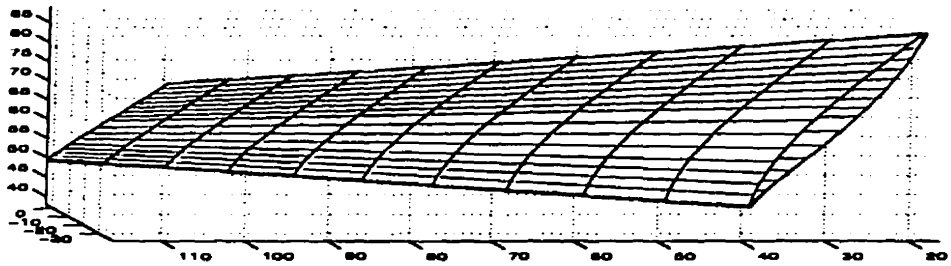
The existing approaches to this problem ignore the compatibility requirement



(a) data



(b) fitted curves



(c) skinned surface

Figure 1.1: Construction of Turbomachine Blades and Aircraft Wings: (a) section curves given as data, (b) approximation curve to the data, (c) skinned surface over the approximation curves

during the approximation process. The approximation is geared toward satisfying the prespecified tolerance. This neglect raises the need to make the curves compatible after the tolerance is satisfied. Since compatibility is realized by knot merging, the number of control points of the compatible curves may be excessive. This disadvantage will be carried over to the skinned surface that is constructed over the curves.

This research approaches the problem with a different scheme. Compatibility is enforced first by constraining the curves to share a mutual degree and a mutual knot vector. This constraint is then applied within the nonlinear least squares problems to generate a set of compatible curves. The resulting curves are compatible but do not yet satisfy the prespecified tolerance. These curves are then subjected to a repetitive knot insertion process that is bound to maintain the compatibility. This insertion is geared to satisfy the prespecified tolerance.

1.1 Definitions of Airfoil and Tolerance

This section gives the definitions of the airfoil and the tolerance that are used in this thesis.

- **Airfoil** is defined as a thin plate whose shape is designed to generate aerodynamic forces when the airfoil is in motion relative to the surrounding fluid[1]. The curve defining the shape of airfoil can be broken apart into four subcurves, namely: leading edge, trailing edge, pressure side, and suction side. Stream of fluid makes first contact with the airfoil at the leading edge, separates into

two streams each of which traverses the pressure side and the suction side respectively, and leaves the airfoil at the trailing edge. The stream of fluid that traverses the pressure side experiences an increase of static pressure, whereas that traversing the suction side experiences a decrease of static pressure.

- The definition of tolerance used in this thesis is different from the definition of tolerance defined by the American National Standards Institute (ANSI). The definition of tolerance by ANSI can be obtained in the work by Zeid[88].

  In this thesis, the **tolerance** *is defined as the maximum Eucledian distance that a pair of points is allowed to be apart from each other. In other words, a pair of points* $\mathbf{x}_1$ *and* $\mathbf{x}_2$ *is said to satisfy a tolerance* ϵ *if* $||\mathbf{x}_1 - \mathbf{x}_2||_2 \leq \epsilon$. *Otherwise, these points are said to be out-of-tolerance.*

  When applied to a set of data points and the approximation curve obtained from least-square fitting of the data points, the tolerance is defined as the largest Eucledian distance that each of the data points is allowed to be apart from the associated point on the fitted curve. This definition of tolerance between a set of data points and the approximation curve can be considered as the stricter version of the ANSI profile tolerance.

  This distinctive definition is introduced to allow a simple expression of tolerance with respect to the parameters of the approximation curve obtained from least-square fitting.

1.2 Motivation

This research is aimed at reducing the geometric difficulties[46] faced during aerodynamic design of turbomachine blades and aircraft wings. The developed geometric methods improve the construction of blade and wing surfaces from discrete data as illustrated in Figure 1.1.

This section presents a brief description of the current airfoil and wing design procedures and proposes a method of compatible NURBS curves to improve the wing design process. Simplified design procedures of airfoils and wings that serve a single flight condition are used as the vehicle to describe the source of inefficiencies in the existing wing design process. The description focuses on the aerodynamic design of wings with fixed geometry.

1.2.1 Airfoil Design Process

Figure 1.2 illustrates a simplified airfoil design procedure. Existing airfoil design procedure starts with a specification of a flight condition in term of the Reynolds number of the mainstream flow, the desired pressure distribution along the upper and lower parts of the airfoil, and the chord of the airfoil. These data are used to select an initial airfoil from a catalog. The criteria of selection is to find an airfoil whose pressure distribution approximates the desired pressure distribution. The selected airfoil and the desired pressure distribution are then used as input to an inviscid incompressible flow-based inverse design[20, 53]. This geometric design, however, lacks the influence of boundary-layer and vortex phenomena. Figure 1.3 illustrates most of the possible two dimensional flow phenomena that may occur on

subsonic airfoils.

The airfoil geometry resulting from the existing inverse-design procedure are represented as a set of points. This airfoil geometry is subsequently subjected to realistic two dimensional flow analysis which accounts for boundary-layer separation and vortex phenomena. If unfavorable performace occurs along the curves of the airfoil, the designer locally perturbs the adverse regions in order to improve the flow along the airfoil. It is known that the change of realistic flow with respect to local perturbation is extremely nonlinear and difficult to anticipate. Prediction of necessary geometric perturbance to adjust the pressure distribution and boundary layer development is a very difficult and time consuming task. This is due to the complex boundary layer behavior that defines the overall section performance. Besides this objective of favorable flow, there exists another objective to minimize *drag-to-lift* ratio in order to reduce the fuel consumption. *In short, performing modification of airfoil geometry to obtain healthy and steady flow and minimum drag to lift ratio is a tedious and complex task.*

Existing point representation of the airfoil further worsens the situation encountered by the designer. Modification of geometry is performed by translating one or more points in the adverse region. This method lacks control over the continuity and smoothness of airfoil geometry which are very critical to the quality of flow along the airfoil. The lack of control over smoothness and the associated unpredictable behavior of the boundary layer development results in a the trial-and-error method for the design of airfoils.

From the above discussion, it is fair to conclude that the existing airfoil design

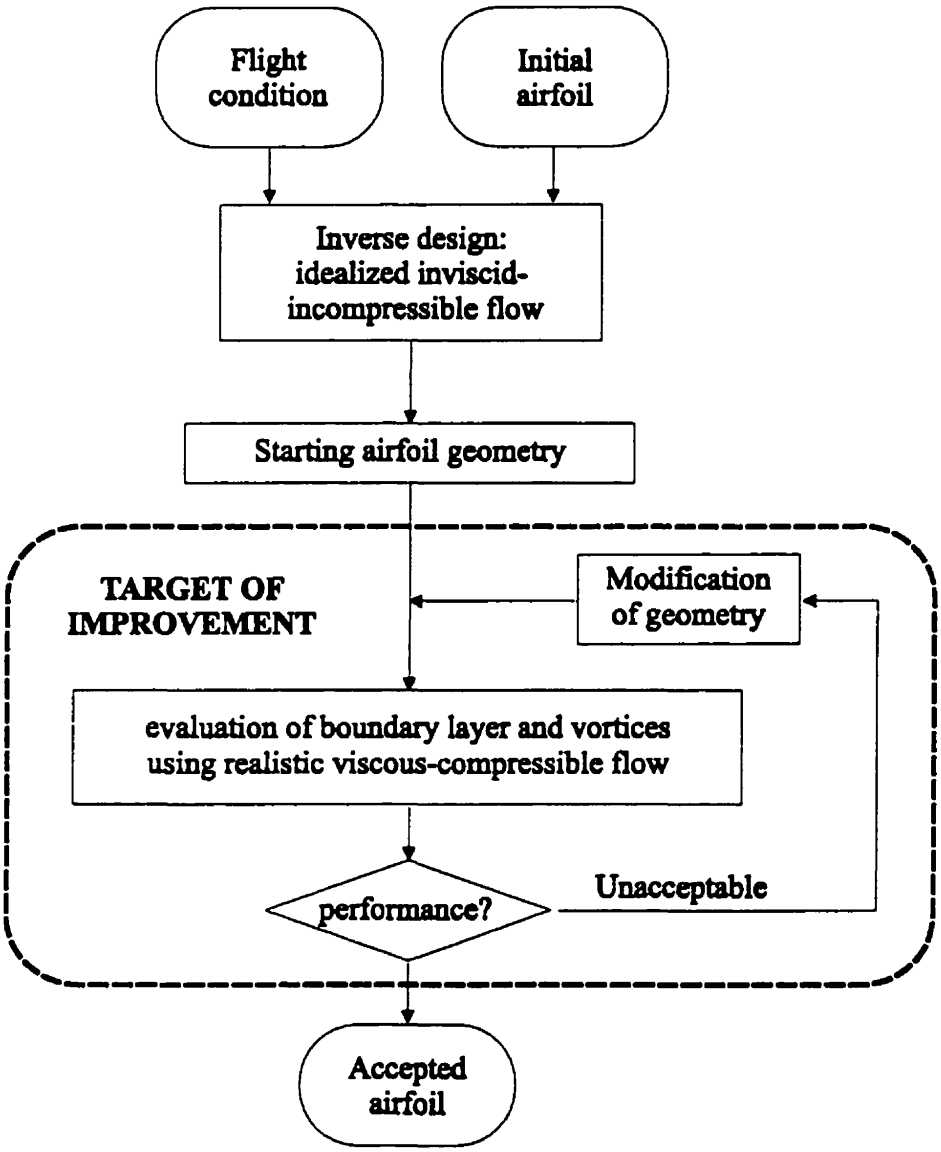


Figure 1.2: Simplified Procedure of Airfoil Design

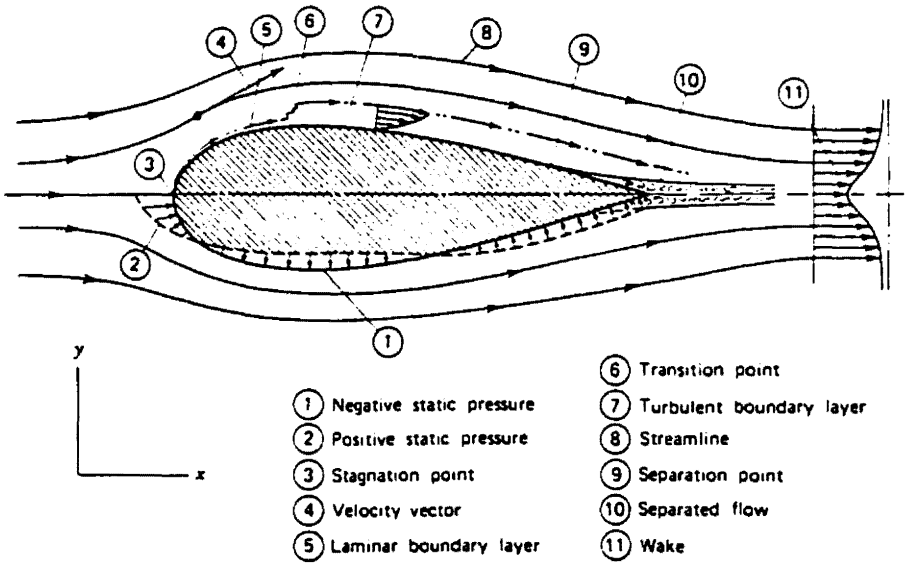


Figure 1.3: Two Dimensional Flow Around an Airfoil[64]

paradigm suffers significant inefficiency due to the use of point representation. Replacing this representation with NURBS representation will instantly provide the airfoil designer with a set of NURBS shape modification tools, e.g. control point repositioning, weight modification, warping, flattening, bending, constraint-based curve modification, smoothing, etc[5, 6, 22, 27, 29, 44, 52, 57, 70, 71, 74, 77]. These tools are much superior to the modification tools for points. NURBS modification tools allow exceptional control over the continuity, smoothness, and locality of changes, of the modified region. The tasks inside the dashed box in Figure 1.2 are those with the most potential for improvement of performance through the use of NURBS representation.

1.2.2 Wing Design Process

The aerodynamic design of a fixed geometry wing results in a wing geometry that serves a prespecified flight condition specified by the Reynolds number of the mainstream flow, angle of attack, and the overall lift force that the wing must generate. The objectives of wing's aerodynamic design is to achieve a healthy and stable flow that possesses minimum drag to lift ratio.

The simplified design procedure is shown in Figure 1.4. The design starts with a prespecified flight condition. This typically includes the payload of the aircraft, the flight altitude, and the cruising speed. Based on this specification, wing geometry is estimated; this typically includes chord, wing span, and sweep angle, as shown in Figure 1.5. These estimates of wing geometry and the prespecified flight condition are used as inputs to the airfoil design process.

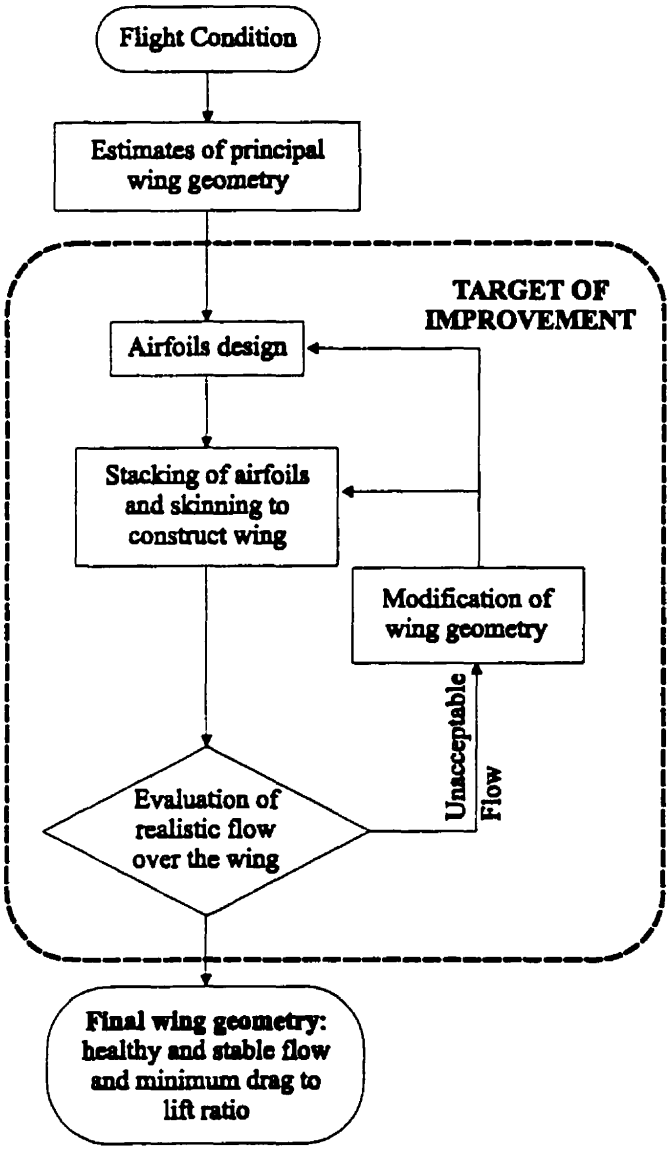


Figure 1.4: Simplified Procedure of Wing Design

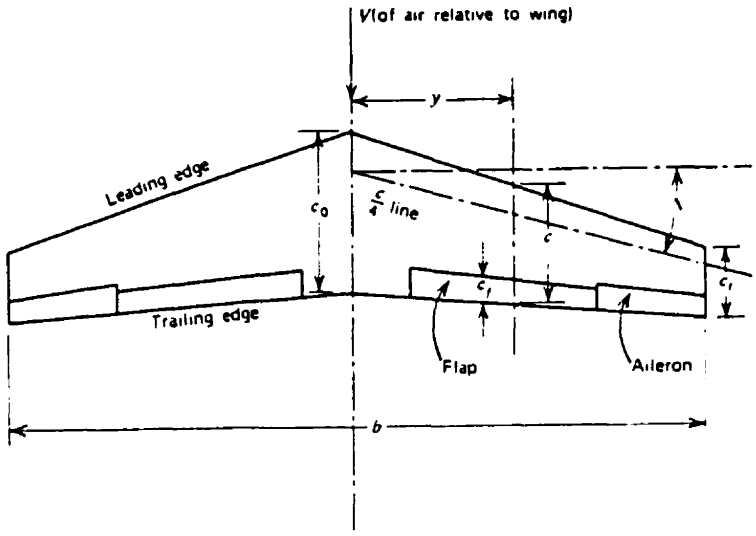


Figure 1.5: Wing Geometry[64]

After a number of airfoils have been designed, they are stacked at prespecified positions and orientations along the wing span. Following this stacking, the wing surface is generated by skinning over these airfoils. Then the wing surface is subjected to a realistic three-dimensional viscous flow to evaluate the quality and stability of the flow. This evaluation is particularly important to inspect the flow between the airfoils because the actual flow behavior in this region may significantly differ from the behavior of the designed airfoils. When adverse flow behavior occurs, the wing geometry is modified by rearranging the stacking of airfoils and/or redesigning the airfoils, as shown in Figure 1.4. Stacking rearrangement leaves the airfoil geometry unchanged, and it is performed by one or more of the following: repositioning airfoils along the wing span, reorienting the airfoil with respect to

fluid flow oncoming direction, and inserting an additional airfoil in between two consecutive airfoils.

The modification of the airfoil is performed using the method that was explained in the previous subsection, see Figure 1.2. Therefore, this modification carries over the inefficiencies found in the airfoil design procedure. Improvement of airfoil design procedure will automatically improve the wing design procedure.

This section has presented the disadvantage of using the point representation in the airfoil and wing design and identified the resulting inefficiencies. Point representation lacks robust tools for controlled modification. When this is linked with the unpredictable behavior of boundary layer development, it results in an expensive and time consuming aerodynamic design processes.

The design procedures for airfoils and wings, illustrated in Figures 1.2 and 1.4, show that the flow analysis and the expertise of the aerodynamicist are merged in a closed loop identified by the dashed boxes labeled TARGET OF IMPROVEMENT. In these loops, the geometric modeler serves as the tool of communication between the aerodynamicist and the flow analysis package. The instruction from the aerodynamicist is in the form of modification to the geometry of the airfoils/wings. The flow analysis package receives this instruction and subsequently produces the flow performance for the modified geometry. These loops are potential candidates for automation. In the automated system the input will be the data set modeling the airfoil(s). The automated system interface will take this data and prepare it for a flow analysis. The results of the flow analysis may be analyzed by either an aerodynamicist, an artificial-based system, a heuristic system, a mathematical model,

or etc. The outcome of the analysis will be in the form of a modification of the geometry of the airfoil. This loop will continue until the geometry of the airfoil produces the desired pressure distribution. Realization of this automation needs integration of geometric modeler and flow analysis.

The magnitude and complexity of the automation depends on the number of geometric parameters in the representation. If point representation is the basis of the geometry, the automation may not be feasible due to the large number of degrees of freedom. A concise geometric representation is needed. Nonuniform Rational B-splines offer such a concise representation. For NURBS modelers, the parameters are: control points, knots, weights, and degree.

An infinite number of NURBS curves can be created to approximate a given airfoil data set. Each of these curves is based on a different parameter set. From the perspective of complexity, it will be ideal to select that NURBS representation which results in a minimum number of parameters. The additional requirement of minimum number of control points does not affect the ability to do controlled modification of the NURBS curve.

1.3 Proposed Approach

This research aims to investigate the representation of geometry of airfoils using NURBS curves. Having NURBS geometry throughout the design cycle allows controlled modification of wing and airfoil geometry with much fewer degrees of freedom.

In this thesis a new method of geometric construction of airfoils and wings is

presented. In this proposed approach the different airfoil data set, see Figure 1.1, which make up the skeleton of the wing, are simultaneously fitted with NURBS curves that share a mutual knot vector and a mutual degree.

Using Golub's expression for the least squares error of B- spline approximation[33, 32], the sum of errors of an individual curve is a nonlinear function of knots, weights, and parameterization for data. For multicurve simultaneous approximation, the sum of errors for all curves is selected as the objective function, and it is a nonlinear function of mutual knots, weights, and parameterization for data. The solution of this problem of nonlinear least squares is obtained using an optimization scheme. This scheme must consider linear constraints due to the monotonically increasing property of the knots and the parametrization for data, and positivity of the weights. The latter is to avoid singularity and to preserve the convex hull property of the resulting curves. The goal of solving this nonlinear least squares problem is to reduce the least squares error simultaneously in all the skeleton curves. This, however, may still fail to satisfy the prespecified tolerance.

For satisfying the prespecified tolerance, the approach of knot insertion is taken in this research. The insertion serves as a mechanism to adjust (increase) the degree of freedom. The constraint that the B-spline basis are defined on a mutual knot and have a mutual degree guarantees compatibility of the curves throughout the knot insertion process. The final results are a set of approximation NURBS curves that are compatible and satisfy the prespecified tolerance. The compatibility among the NURBS curves eliminates the need for knot merging prior to skinning.

In conclusion, the proposed approach attempts to produce a set of NURBS

curves that satisfy a set of various prespecified tolerances and the compatibility requirement with reasonably number of degrees of freedom (control points). Success of this attempt is measured by how few basis are required to satisfy these tolerances.

1.4 Objective of Research

The objective of this research is to simultaneously fit one or more skeletal curves, each represented by a set of discrete points, with NURBS curves such that they have a reasonably low number of parameters (knots, weights, and control points) and the distance between the discrete point set and the point on the curve is less than a prespecified value.

- In order to achieve this goal, a method to fit a NURBS curve to a discrete data set is developed. To address this problem the least squares method is used. This method is a function of many parameters: knots, weights, control points, and parameterization of data. The proposed method optimizes all or a combination of these parameters to obtain a minimal representation.

- To study the sensitivity of the least squares error with respect to the various combinations of parameters an investigation was conducted by performing nonlinear least squares fitting of a single curve. Three data set were subjected to this investigation. These data sets represent NACA 2415 airfoil, WTEA airfoil, and an axial compressor airfoil. Each data set was subjected to five sets of combination of adjustable parameters. These sets are:

- knots
- knots and weights
- knots, weights, and parameterization for the data
- weights
- parameterization for the data

This investigation served two objectives. The first objective was to measure the usefulness, in an engineering sense, of using the parameters as adjustable parameters. The usefulness was measured by the decrease of least squares error resulting from each of these five sets of adjustable parameters. The second objective was to measure the effectiveness of incorporation of the non-linear constraint in maintaining the accuracy and good behavior of the control points.

- The single curve method is then extended to a multicurve method.
- Similar to the previous case, an investigation to determine the sensitivity to the adjustable parameters was also conducted on multicurve (skeleton) cases. This investigation was conducted on three sets of skeleton: a two-section skeleton of a wing, a three-section skeleton of an axial compressor blade, and a three-section skeleton of an axial turbine blade. Each skeleton was subjected to five sets of combination of parameters identical to the combinations in the previous investigation. The goals of this investigation were the same as in the previous investigation.

- The methods developed above fit the discrete data representing the skeletal curves with NURBS curves based on the user selected degree and number of control points. Such a curve may not be within a prespecified tolerance to all the discrete data points. An iterative method, called the Tolerance based knot insertion method, is developed to increase the number of control points and to meet the prespecified tolerance requirement. This is done by adding knots in specific regions of the curve.

- An investigation of knot insertion was conducted to study its effectiveness in satisfying the prespecified tolerance. The resulting curves from the previous investigation were subjected to Tolerance Based Knot insertion and the reduction of the number of control points to represent compatible sections that satisfy a prespecified tolerance was measured. The resulting number of control points were compared with those obtained from the knot removal based method[70, 82].

1.5 Organization of Thesis

Chapter 2 presents the theoretical background and literature review on the least squares approximation with NURBS curves. The first part of this chapter covers the derivation of expression of least squares problems with rational B- splines; it is adopted from Golub's work[33, 32]. Identification of parameters and elimination of control points from the parameters are also presented. The elimination of control points naturally transforms this problem into a nonlinear least square problem.

Next the derivation of gradient is presented as its availability and properties are critical for solving the nonlinear least square optimization problem. The second part of this chapter presents a literature review on this subject. The review focuses on NURBS approximation with fixed number of basis functions.

Chapter 3 presents the proposed approach to solve the NURBS nonlinear least squares problem. The chapter starts with a definition of constraints on the feasible domain of the parameters. A new definition of nonlinear constraint is introduced to improve the accuracy of the control points. A new approach to satisfy this constraint will also be presented. This chapter ends with a discussion on the descent direction for the nonlinear least squares optimization problem.

Chapter 4 presents the implementation of the method proposed in Chapter 3. It also presents the investigation of sensitivity as discussed earlier. The implementation serves as a check of numerical behavior of the proposed approach presented in the previous chapter. In particular, this chapter highlights the distinctions that knots, weights, and parameterization for data respectively, are capable of reducing the least squares error. It will be shown that the least squares error does not suffer if weights and parameterization for data are eliminated from the list of adjustable parameters. This chapter will also highlight the effectiveness of the proposed nonlinear constraint in maintaining the accuracy of the control points.

Chapter 5 presents an extension of the implementation of the proposed approach to multicurve problems. This chapter's highlight is the compatibility enjoyed by the skeleton curves. Similar to the previous chapter, it will be shown that the least squares error does not suffer if weights and parameterization for data are eliminated

from the list of adjustable parameters. This chapter also shows the effectiveness of the proposed nonlinear constraint in maintaining the accuracy of the control points.

Chapter 6 presents the Tolerance Based Knot Insertion method of providing sufficient number of basis in order to satisfy the prespecified accuracy. The number of basis are increased via a knot insertion method. Comparison with existing methods for constructing compatible section curves is also provided. The comparison clearly demonstrates that our approach is superior in the sense that the number of control points required to produce compatible section curves that satisfy a prespecified accuracy are much less than the existing method. This chapter also presents an attempt to smooth the curves, and ends with a visualization of the skinned surface.

Finally, this thesis is closed with a chapter that summarizes the achievement in this thesis. Possible future research directions are also presented.

Chapter 2

Curve Fitting: Basics and Literature Review

This chapter presents a theoretical background and literature review of the least squares problems for parametric Nonuniform Rational B-Spline (NURBS) curves. The coverage focuses on approximation problems where the number of basis functions is constant. This chapter assumes familiarity with basics of NURBS; otherwise, readers may refer to Appendix A for theoretical exposition on NURBS curves and surfaces.

2.1 Least Squares Approximation

Approximation is suitable for constructing a curve from a set of data when the number of data is very large and/or the data contains measurement errors[70, 73]. The curve constructed by approximation does not exactly satisfy the data, instead

some disagreement between the data and the curve exists. The disagreement is measured by a vector norm. The alternatives of vector norms are one-norm (L_1), two-norm (L_2), and infinity norm (L_∞). Theoretically, the selection of norm should be based on the statistical significance of the measurement error. In practice, however, it is common to use the (L_2)-norm unless the norm has some statistical significance. Approximation with two-norm is known as *least-squares problems*. The use of one-norm and infinity-norm leads to problem with discontinuity in derivatives. Although special methods exist for dealing with these problems, the complexity of the problem is significantly increased compared with the least-squares problem[31].

In this thesis, the disagreement between the data and the approximation curve is measured using a two-norm measure. The two-norm measure is selected for three reasons: first to allow elimination of linear parameters (the control points for NURBS curves) from the parameter set, second to simplify the expression for first derivative of the measure of disagreement with respect to its parameters, and third to provide the continuity of the first derivative of measure of disagreement.

The use of two-norm as measure of the disagreement between the curve and the data yields a simple formulation of measure of disagreement, i.e. $\varepsilon = \sqrt{\mathbf{e}^T\mathbf{e}}$ where $\mathbf{e}$ is a vector whose elements are Euclidean distances between the data and the corresponding points on the curve. To simplify the derivative of the measure of disagreement, it is common to modify the measure of disagreement to $\varepsilon = \mathbf{e}^T\mathbf{e}$, simply the square of the original formulation. The modified expression yields a simpler first derivative,

$$\frac{\partial \varepsilon}{\partial \alpha_i} = 2\mathbf{e}^T \frac{\partial \mathbf{e}}{\partial \alpha_i} \tag{2.1}$$

The expression of a NURBS approximation curve is $\mathbf{p}_a = \mathbf{R}\mathbf{c}$, where $\mathbf{p}_a$ is the value of NURBS curve at the parameterization for the data, $\mathbf{R}$ is an overdetermined matrix of rational B-spline basis, and $\mathbf{c}$ are the control points; the expression for the elements of $\mathbf{R}$ can be found in Appendix A. The least squares error for the curve can be expressed as

$$\begin{aligned} \varepsilon &= (\mathbf{p} - \mathbf{p}_a)^T (\mathbf{p} - \mathbf{p}_a) \\ &= (\mathbf{p} - \mathbf{R}\mathbf{c})^T (\mathbf{p} - \mathbf{R}\mathbf{c}) \qquad (2.2) \\ \text{where} \quad & \mathbf{R} = \mathbf{R}(\mathbf{u}, \mathbf{w}, \mathbf{t}) \\ & \mathbf{p} \text{ is the data set} \end{aligned}$$

The objective of curve approximation is to minimize the measure of disagreement, that is to minimize ε. The equation above clearly shows that minimization of ε can be performed by adjusting $\mathbf{R}$ and $\mathbf{c}$.

2.2 Parameters of NURBS Least Squares

Approximation with NURBS curve requires the following set of parameters: degree of basis, number of basis functions, knots, weights, coordinates of control

points, and parameterization for data; the last parameter applies only for parametric NURBS curves. The first two parameters are discrete, and their values are normally specified by the users instead of being computed. Degree of a curve is application dependent and is very rarely considered as an adjustable parameter. The degree is specified based on continuity requirement. Most engineering applications use cubic curves since cubic is the lowest possible degree that provide curvature continuity and capability of representing space curve. Although curves of higher degree are also capable of representing space curves, their tendency to oscillate limits their scope of applications[44]. The number of basis constitutes the degrees of freedom of the problems. The number of basis determines the lowest possible disagreement between the data and the approximation curve; a greater number of basis functions allows a smaller disagreement, i.e. a more accurate approximation. At best, the determination can be performed heuristically. Existing methods of B-spline curve approximation that facilitate "heuristic adjustment" of number of basis in order to satisfy a prespecified accuracy can be classified into two groups. The first group is based on repeated knot insertion. Bartels et. al.[26, 28] and Dierckx[17] developed their own methods that fall into this group. Bartels' method is known as *hierarchical spline fitting* whereas Dierckx's method does not have any particular name. Tiller[82] and Piegl[70] developed a method based on knot removal. This method falls into the second group. Complete exposition on these heuristic methods will be presented in Chapter 6.

After the degree and the number of basis functions have been specified, the next step is to obtain the knots, weights, coordinates of control points, and parameter-

ization for data, that yields a minimum ϵ. Since the ϵ is a nonlinear function of knots, weights, and parameterization for data, the solution of these parameters is obtained using optimization.

The next section will present an adaptation of Golub's work which demonstrates that control points are redundant parameters. They can be eliminated from the parameters without affecting the solution.

2.3 Elimination of Control Points

Eqn. (2.2) clearly shows the separation of linear parameters (control points) from nonlinear parameters (knots, weights, and parameterization of data). Nonlinear variables are implicitly embedded in the elements of matrix **R** whereas the linear parameters are explicitly shown in Eqn. (2.2). This type of least square problems is known as *the least square problem with parameter separation.*

Golub[32, 33] shows that this least square problem can be simplified by eliminating the linear parameters from the domain of the parameters so that the actual domain of the problem is reduced to the domain of nonlinear parameters only. Therefore, adjustment of parameters is performed only on the nonlinear parameters. However, elimination of linear parameters can only be performed if the measure of disagreement uses two-norm measure. The author is not aware of methods of elimination of linear parameters for approximation problems that use one-norm or infinity-norm measure of disagreement.

The process of elimination of control points starts with the formulation of the explicit expression of control points. For a given matrix **R**, the control points can

be explicitly written as a function of the data and the pseudo inverse of matrix $\mathbf{R}$. The expression of the control point then becomes $\mathbf{c} = \mathbf{R}^{+}\mathbf{p}$ where $\mathbf{R}^{+}$ denotes the pseudo inverse of matrix $\mathbf{R}$.

The *pseudo inverse* of an $m \times n$ matrix $\mathbf{R}$ is a unique $n \times m$ matrix such that $\mathbf{c} = \mathbf{R}^{+}\mathbf{p}$ is the vector of minimum Euclidean length that minimizes $||\mathbf{p} - \mathbf{R}\mathbf{c}||_2$. There are several mathematically equivalent expressions for the pseudo inverse of $\mathbf{R}$. When $\mathbf{R}$ is nonsingular, $\mathbf{R}^{+} = \mathbf{R}^{-1}$. When $\mathbf{R}$ has full column rank, the pseudo inverse may be written as $\mathbf{R}^{+} = \left(\mathbf{R}^{T}\mathbf{R}\right)^{-1}\mathbf{R}^{T}$. This expression of $\mathbf{R}^{+}$, however, is useful only for descriptive purposes. For computational purposes, $\mathbf{R}^{+}$ should be computed using matrix factorization. When $\mathbf{R}$ has full column rank, QR decomposition can be used to compute $\mathbf{R}^{+}$ using $\mathbf{R}^{+} = \mathbf{R}_{*}^{-1}\mathbf{Q}_{1}^{T}$; symbol $\mathbf{R}_{*}$ and $\mathbf{Q}$ denotes the nonsingular upper triangular matrix and the lower triangular matrix in QR decompositions respectively. When $\mathbf{R}$ is rank deficient, the most convenient form of the pseudo inverse is based on the singular value decomposition (SVD). By performing SVD to the rank deficient matrix whose column rank is r, the decomposition will produce matrices $\mathbf{U}$, $\mathbf{W}$, and $\mathbf{V}$ satisfying $\mathbf{R} = \mathbf{U}\mathbf{W}\mathbf{V}^{T}$. The pseudo inverse can then be calculated as $\mathbf{R}^{+} = \mathbf{V}\mathbf{M}\mathbf{U}^{T}$; $\mathbf{M}$ is a diagonal matrix whose elements are the reciprocal of the elements of diagonal matrix $\mathbf{W}$, except when the elements of $\mathbf{W}$ are zero at which the elements of $\mathbf{M}$ are set to zero. Methods to compute the pseudo inverse can be found in many textbooks[31, 34, 67].

By substituting $\mathbf{c} = \mathbf{R}^{+}\mathbf{p}$ into Eqn. (2.2), the measure of disagreement can be rewritten as

$$\varepsilon = \left(\mathbf{p} - \mathbf{R}\mathbf{R}^{+}\mathbf{p}\right)^{T}\left(\mathbf{p} - \mathbf{R}\mathbf{R}^{+}\mathbf{p}\right) \tag{2.3}$$

Eqn. (2.3) shows that the control points have been eliminated from the expression of measure of disagreement. The proof that the minimum of the functional in Eqn. (2.3) is exactly identical to the minimum of the original functional in Eqn. (2.2) can be found in Golub's articles [32, 33]; the principal requirement of this proof is that the measure of disagreement must use the two-norm.

Equation (2.3) is the measure of disagreement of approximation in one-dimensional problems. Three-dimensional problems can be viewed as consisting of three one-dimensional problems. By restricting the curves in all coordinate directions to share a mutual set of knots, a mutual set of weights, and a mutual set of parameterization of data, the measure of disagreement in each coordinate can then be written as

$$\varepsilon_x = \left(\mathbf{p}_x - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_x\right)^{T}\left(\mathbf{p}_x - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_x\right) \tag{2.4}$$

$$\varepsilon_y = \left(\mathbf{p}_y - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_y\right)^{T}\left(\mathbf{p}_y - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_y\right) \tag{2.5}$$

$$\varepsilon_z = \left(\mathbf{p}_z - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_z\right)^{T}\left(\mathbf{p}_z - \mathbf{R}\mathbf{R}^{+}\mathbf{p}_z\right) \tag{2.6}$$

These measures of disagreement can be summarized in a simple expression

$$\varepsilon = \varepsilon_x + \varepsilon_y + \varepsilon_z \tag{2.7}$$

The use of addition operators is due to simplicity of derivatives of the overall measure of disagreement, that is

$$\frac{\partial \varepsilon}{\partial} = \frac{\partial \varepsilon_x}{\partial} + \frac{\partial \varepsilon_y}{\partial} + \frac{\partial \varepsilon_z}{\partial} \tag{2.8}$$

Equations (2.2) to (2.7) have indicated clearly that NURBS least squares is a nonlinear least squares problem. Solution of this problem is obtained by an optimization method. Selection of an optimization method depends on the continuity of objective function, availability of gradient, and its continuity. Golub has proven that the objective function of this nonlinear least squares problem and the gradient are continuous if matrix **R** has a constant rank within the domain of the parameters[32, 33]. Golub has also derived the expression of the gradient. An adaptation of his work will be presented in the next section.

2.4 Derivatives of Residual

The elimination of linear parameters from the formulation of measure of disagreement reduces the parameters that need to be adjusted to nonlinear parameters only. As already mentioned, adjustment of nonlinear parameters in the least square problems is performed using an optimization method. Good descent direction can be obtained using the gradient of the measure of disagreement. Therefore, first derivatives of measure of disagreement with respect to nonlinear parameters are important in the adjustment of nonlinear parameters to minimize the measure of disagreement.

The measure of disagreement in Eqn. (2.3) can be written as a function of $\mathbf{RR}^+$

$$\varepsilon = \mathbf{p}^T \left(\mathbf{I} - \mathbf{RR}^+\right)^T \left(\mathbf{I} - \mathbf{RR}^+\right) \mathbf{p} \tag{2.9}$$

Derivative of measure of disagreement with respect to a nonlinear parameter, say α_i, is

$$\frac{\partial \varepsilon}{\partial \alpha_i} = -2\mathbf{p}^T \left(\mathbf{I} - \mathbf{RR}^+\right)^T \left[\frac{\partial \left(\mathbf{RR}^+\right)}{\partial \alpha_i}\right] \mathbf{p} \tag{2.10}$$

The explicit expression of gradient of measure of disagreement can be obtained by deriving the term inside the square bracket. The procedure to derive the term requires identification of properties of $\mathbf{RR}^+$ and $\left(\mathbf{I} - \mathbf{RR}^+\right)$. Golub[32, 33] shows that the properties of the $\mathbf{RR}^+$ term that are relevant to the first derivative of measure of disagreement are

$$\left(\mathbf{RR}^+\right)^T = \mathbf{RR}^+ \tag{2.11}$$

$$\left(\mathbf{RR}^+\right)\mathbf{R} = \mathbf{R} \tag{2.12}$$

$$\mathbf{RR}^+ = \left(\mathbf{RR}^+\right)\left(\mathbf{RR}^+\right) \tag{2.13}$$

$$\left(\mathbf{I} - \mathbf{RR}^+\right)^T \left(\mathbf{I} - \mathbf{RR}^+\right) = \left(\mathbf{I} - \mathbf{RR}^+\right) \tag{2.14}$$

$$\left(\mathbf{I} - \mathbf{RR}^+\right)^T \left(\mathbf{R}^+\right)^T = \mathbf{0} \tag{2.15}$$

Golub[33] starts by differentiating both sides of Eqn. (2.13); the result is

$$\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i} = \frac{\partial\left(\left(\mathbf{R}\mathbf{R}^{+}\right)\left(\mathbf{R}\mathbf{R}^{+}\right)\right)}{\partial\alpha_i} \tag{2.16}$$

The right hand side of Eqn. (2.16) can be arranged to

$$\frac{\partial\left(\left(\mathbf{R}\mathbf{R}^{+}\right)\left(\mathbf{R}\mathbf{R}^{+}\right)\right)}{\partial\alpha_i} = \frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\left(\mathbf{R}\mathbf{R}^{+}\right) + \left(\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i} \tag{2.17}$$

The following procedure derives the expressions of terms on the right hand side of Eqn. (2.17). The first term can be rearranged to

$$\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\left(\mathbf{R}\mathbf{R}^{+}\right) = \left(\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\mathbf{R}\right)\mathbf{R}^{+} \tag{2.18}$$

The term in the bracket in Eqn. (2.18) can be obtained by differentiating both sides of Eqn. (2.12); the differentiation yields

$$\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\mathbf{R} + \left(\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i} = \frac{\partial\mathbf{R}}{\partial\alpha_i} \tag{2.19}$$

Rearranging Eqn. (2.19) yields

$$\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\mathbf{R} = \left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i} \tag{2.20}$$

Substituting Eqn. (2.20) into Eqn. (2.18) yields

$$\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\left(\mathbf{R}\mathbf{R}^{+}\right) = \left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+} \tag{2.21}$$

Eqn. (2.21) is the final expression of the first term in the right hand side of Eqn. (2.17). Using Eqn. (2.11), the second term of the right hand side of Eqn. (2.17) can be written as

$$\left(\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i} = \left(\mathbf{R}\mathbf{R}^{+}\right)^{T}\left(\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\right)^{T} \tag{2.22}$$

$$= \left(\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i}\left(\mathbf{R}\mathbf{R}^{+}\right)\right)^{T} \tag{2.23}$$

Substituting Eqn. (2.21) into Eqn. (2.23) yields

$$\left(\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\left(\mathbf{R}\mathbf{R}^{+}\right)}{\partial\alpha_i} = \left(\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+}\right)^{T} \tag{2.24}$$

Substituting Eqn. (2.21) and Eqn. (2.24) into Eqn. (2.17) yields

$$\frac{\partial\left(\left(\mathbf{R}\mathbf{R}^{+}\right)\left(\mathbf{R}\mathbf{R}^{+}\right)\right)}{\partial\alpha_i} = \left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+} + \left(\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+}\right)^{T} \tag{2.25}$$

Substituting Eqn. (2.25) into Eqn. (2.16), followed by substituting Eqn. (2.16) Eqn. (2.10) yields

$$\begin{aligned}\frac{\partial\varepsilon}{\partial\alpha_i} &= -2\mathbf{p}^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)^{T}\left[\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+} + \right.\\ &\qquad \left.\left(\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+}\right)^{T}\right]\mathbf{p}\end{aligned} \tag{2.26}$$

Expanding terms yields

$$\begin{aligned}\frac{\partial\varepsilon}{\partial\alpha_i} &= -2\left[\mathbf{p}^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+}\mathbf{p}\right] + \\ &\quad -2\left[\mathbf{p}^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)^{T}\left(\mathbf{R}^{+}\right)^{T}\left(\frac{\partial\mathbf{R}}{\partial\alpha_i}\right)^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)^{T}\mathbf{p}^{T}\right]\end{aligned} \tag{2.27}$$

The term inside the second square bracket vanishes due to Eqn. (2.15) so that Eqn. (2.27) simplifies to

$$\frac{\partial\varepsilon}{\partial\alpha_i} = -2\mathbf{p}^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)^{T}\left(\mathbf{I}-\mathbf{R}\mathbf{R}^{+}\right)\frac{\partial\mathbf{R}}{\partial\alpha_i}\mathbf{R}^{+}\mathbf{p} \tag{2.28}$$

Substituting Eqn. (2.14) into Eqn. (2.28) yields a simple expression

$$\frac{\partial \varepsilon}{\partial \alpha_i} = -2\mathbf{p}^T \left(\mathbf{I} - \mathbf{R}\mathbf{R}^+\right) \frac{\partial \mathbf{R}}{\partial \alpha_i} \mathbf{R}^+ \mathbf{p} \tag{2.29}$$

Eqn. (2.29) is the final expression of the first derivative of measure of disagreement with respect to a nonlinear parameter.

Equations (2.2) and (2.29) have shown the proper expressions for the objective function and its gradient. These equations are the fundamental forms of the NURBS nonlinear least squares problem. The next section will present literature review on this subject.

2.5 Literature Survey

This section presents literature survey in the field of curve approximation with B-splines. The survey focuses on approximation problems where least squares norm is used and the degree of curve and the number of basis are fixed. Referring to Equation (2.3), the general form of least square curve approximation with rational B-spline is

$$\min_{\text{over (u,w,t)}} \quad \mathbf{p}^T \left(\mathbf{I} - \mathbf{R}\mathbf{R}^+\right)^T \left(\mathbf{I} - \mathbf{R}\mathbf{R}^+\right) \mathbf{p} \tag{2.30}$$

2.5.1 Jupp: The Free Knot Approach

Jupp is probably the first researcher who investigated the above objective function[49, 50, 51] where **R** contains B-spline basis. His work is limited on the functional integral B-spline, thus **w** and **t** are dropped from the parameter set. Knots become the only parameters that need to be adjusted. The most important finding in Jupp's work is *the lethargy theorem* that shows poor convergence of adjustment when two or more knots become nearly coincident. The theorem is based on Jupp's observation that the gradient lies on the null space of active constraint. In other words, the gradient does not have any component in the range space of the active constraint. Jupp argues that the lack of component in the range space will trap the knots in the null space; once two or more knots coincide, it is impossible for the knots to be separated. When the knots are *almost* coincident, the components of the gradient in the range space are not zero but they are very small. Such small components, according to Jupp, will cause the knots to wander around the active constraint. The net result is poor convergence as stated in Jupp's lethargy theorem. In summary, Jupp's characterization of knot adjustment is largely based on his argument that descent direction is solely a function of projection of the gradient in the range space of the active constraints, not a function of the gradient itself. Based on his observation, Jupp concludes that, whenever one or more of knot constraints are active, the descent direction always lies in the null space of the constraint due to the lack of component in the range space of the constraints.

Jupp's argument, however, contradicts the characteristic of Newton and quasi-Newton descent directions. These descent directions always lie between gradient

and line of constant objective[31, 73, 15] provided that the Hessian or its approximation is well-conditioned, positive definite, and is not dominated by its diagonal. The key strategy is to keep the approximate Hessian, besides its projected matrix, throughout the optimization. When two or more knots coincide, the quasi Newton descent direction is calculated using the approximate Hessian instead of its projected matrix. If the descent direction has component into the feasible domain, the constraints on those knots are deleted. Therefore, it becomes possible for those knots to separate. Experiments conducted by the author shows that the descent direction obtained from the BFGS method do have components in the range space of the constraints. These components allow the current iterate to leave the constraints. This observation is opposite to Jupp's characterization of the behavior of knot adjustment.

Jupp concludes that this lethargic property seems to make fitting splines with free knots a most unattractive problem. His supposition is supported by scarcity of published articles on fitting splines with free knots. Jupp proposes the variable transformation to overcome the lethargic property. The knots **u** are transformed to artificial variables **h** using a logarithmic function

$$h_i = \log \frac{u_{i+1} - u_i}{u_i - u_{i-1}} \tag{2.31}$$

Jupp states that such transformation is smooth and transforms the problem into an unconstrained one. The combination of the transformation and the lethargic property implies that the gradient will become arbitrarily small at large distances

from the origin in the transformed space. The overall effect is that the nonlinear least squares algorithm always converges (numerically) a finite distance from the origin even when the exact solution has multiple knots. However, Jupp does not address issues commonly found in variable transformation methods such as inadvertent exclusion of the desired minimum, significant increase of the degree of nonlinearity, adverse scaling of variables, inadvertent presence of singularities and discontinuities of derivatives, singularity or ill-conditioning of Hessian matrix, and inadvertent presence of additional local minima and stationary points.

Finally, our three comments on Jupp's result are: (1) gradient's lack of, or its insignificance of, components in the range space of knots' active constraints does not always mean that the descent direction obtained from the gradient has no or insignificant components in the range space of the active constraint, (2) transformation of variables (knots) are unnecessary whenever the descent direction has significant components in the range space of the active constraints, and (3) ill-conditioning prevention of $\mathbf{R}$ is not addressed in spite of its significant influence on the accuracy of the solution of the observation equation $\mathbf{Rc} = \mathbf{p}$, in particular when computation is performed with finite precision.

2.5.2 Gengoux and Mekhilef: Optimization of NURBS

Gengoux investigates nonlinear least squares with NURBS curves, in particular the optimization of knots and weights[30]. He uses a penalty method to handle the constraints. The penalty terms represent linear constraints on the weights and knots and nonlinear constraint on the condition number of $\mathbf{R}$. The linear constraints are

represented by $\mathbf{g}(\mathbf{u}, \mathbf{w}, \mathbf{t}) \leq 0$. The penalty term for these constraints is

$$\Phi_1 = \sum_i \alpha_i \sup(g_i, 0)^r \tag{2.32}$$

The power r is either 2 or 3; $r = 2$ yields once differentiable Φ whereas $r = 3$ yields twice differentiable Φ. This penalty terms are weighted by factor α_i. The penalty term representing the nonlinear constraint on the condition number of $\mathbf{R}$ is formulated indirectly as a function of control points. The term is defined as the sum of distances of two consecutive control points; the formulation of this term is

$$\Phi_2 = \beta \sum_i^{n-1} ||\mathbf{c}_{i+1} - \mathbf{c}_i||_2^2 \tag{2.33}$$

When the matrix $\mathbf{R}$ is ill-conditioned, the control points have wildly large components. These components will, in turn, raise the penalty term Φ_2. Therefore, Φ_2 functions as a signal and prevents the ill-conditioning of $\mathbf{R}$. However, Gengoux does not mention the analytical gradient of Φ_2 with respect to knots, weights, and parameterization of data. In the author's opinion, availability of this gradient is crucial in the optimization with respect to the nonlinear parameters. It appears that Φ_2 does not function as it is supposed to. Gengoux's report clearly shows that whenever weights are set as the only adjustable parameters, one or more control points have very large components. This phenomena is surely caused by ill-conditioning of $\mathbf{R}$ despite the incorporation of Φ_2. It is also unclear why Gengoux concludes that

the main cause of explosion of the coordinates of the control points is caused by insufficiency of number of basis.

The author's comments regarding Gengoux's experiments are: Φ_2 appears to be ineffective in maintaining the condition number of $\mathbf{R}$ under certain feasible maximum, and analytical gradient of every term in the objective function should be derived if possible.

2.5.3 Heidrich: Separation of Weights

Heidrich's approaches curve approximation with NURBS by fixing knots and parameterization of data. Weights are the only adjustable nonlinear parameters. Heidrich shows that weights can be obtained by solving a set of homogeneous equations[38].

One dimensional least squares problem with NURBS curve will be used to show Heidrich's approach; complete illustration for three-dimensional problems can be found in Heidrich's thesis. The evaluation of the approximation curve at parameter t_k can be written as

$$\frac{\sum_{i=1}^{n} w_i c_i N_i(t_k)}{\sum_{i=1}^{n} w_i N_i(t_k)} = p_k; \qquad \text{for } k = 1:m \tag{2.34}$$

By multiplying the right hand side with the denominator of the left hand side, the equation can be written as

$$\sum_{i=1}^{n} w_i c_i N_i(t_k) = p_k \sum_{i=1}^{n} w_i N_i(t_k); \qquad \text{for } k = 1:m \tag{2.35}$$

The matrix form of Equation (2.35) is

$$\mathbf{PBw} = \mathbf{BC}_w \tag{2.36}$$

where $\mathbf{P}$ is an $m \times m$ diagonal matrix whose diagonal elements are the data $\mathbf{p}$, $\mathbf{B}$ is the $m \times n$ matrix of B-spline basis, $\mathbf{w} \in \Re^n$ is the weight vector, and $\mathbf{C}_w$ is the homogeneous form of the control points, that is $c_{w,i} = w_i c_i$. Equation (2.36) can be rearranged to

$$\begin{bmatrix} \mathbf{B} & -\mathbf{PB} \end{bmatrix}_{m \times 2n} \begin{Bmatrix} \mathbf{c}_w \\ \mathbf{w} \end{Bmatrix}_{2n \times 1} = \{0\}_{m \times 1} \tag{2.37}$$

The solution of the equation can be obtained by premultiplying both sides with the transpose of the matrix in the square bracket. Since the right hand side elements are zeros, this term remains and the final equation becomes

$$\begin{bmatrix} [\mathbf{B}^T\mathbf{B}]_{n\times n} & [-\mathbf{B}^T\mathbf{PB}]_{n\times n} \\ [-\mathbf{B}^T\mathbf{PB}]_{n\times n} & [\mathbf{B}^T\mathbf{PPB}]_{n\times n} \end{bmatrix}_{2n\times 2n} \begin{Bmatrix} [\mathbf{c}_w]_{n\times 1} \\ [\mathbf{w}]_{n\times 1} \end{Bmatrix}_{2n\times 1} = \{0\}_{2n\times 1} \tag{2.38}$$

Eliminating the lower left terms of the diagonal yields

$$\begin{bmatrix} \mathbf{B}^T\mathbf{B} & -\mathbf{B}^T\mathbf{P}\mathbf{B} \\ \mathbf{0} & \left[\mathbf{B}^T\mathbf{P}\mathbf{P}\mathbf{B} - \left(\mathbf{B}^T\mathbf{P}\mathbf{B}\right)\left(\mathbf{B}^T\mathbf{B}\right)\right] \end{bmatrix} \begin{Bmatrix} \mathbf{c}_w \\ \mathbf{w} \end{Bmatrix}_{2n\times 1} = \mathbf{0} \tag{2.39}$$

The second row clearly shows that weights can be solved from the n-sets of homogeneous linear equations as

$$\left[\mathbf{B}^T\mathbf{P}\mathbf{P}\mathbf{B} - \left(\mathbf{B}^T\mathbf{P}\mathbf{B}\right)\left(\mathbf{B}^T\mathbf{B}\right)\right]\mathbf{w} = \mathbf{0} \tag{2.40}$$

Since the terms inside the square bracket depend only on the data $\mathbf{P}$ and the B-spline basis $\mathbf{B}$, the weights can generally be obtained by finding the nontrivial solution of the $n \times n$ sets of homogeneous equations. Heidrich points out that minimization algorithm must be used if nonnegative weights are to be obtained.

In Heidrich's methods, the starting equation in Equation (2.34) is not exactly satisfied in the approximation problem. Since the data is not exactly satisfied by the approximating curve, the form of Equation (2.34) should have been

$$\frac{\sum_{i=1}^{n} w_i c_i N_i\left(t_k\right)}{\sum_{i=1}^{n} w_i N_i\left(t_k\right)} = p_k + \epsilon_k; \qquad \text{for } k = 1:m \tag{2.41}$$

where ϵ_k is the disagreement between the data and the point on the approximating curve. It is not clear how Heidrich justifies the elimination of ϵ_k from the

equation.

2.5.4 Survey of Optimizing Parameterization of Data

Optimization of parameterization of data is popular in the problems of fitting with B-splines where nonlinear adjustment is performed. Adjustment of parameterization of data is encouraged, perhaps, by the simplicity of the derivative of a B-spline curve with respect to the parameter. This survey presents three approaches by Sarkar, Hoschek, and Rogers.

Sarkar: Optimization of Parameterization of Data

Sarkar and Biblap[79, 78] investigated the optimization of parameterization of data on a surface with constant number of control points. The data is approximated by the parametric integral B-spline. The set up starts with initialization of knots and parameterization of data. The control points are then solved from the overdetermined sets of equations $\mathbf{Rc} = \mathbf{p}$. Sarkar observes that the least squares error $||\mathbf{p}_k - \mathbf{s}(t_k)||_2$ does not represent the smallest distance between the data $\mathbf{p}_k$ and the approximation curve $\mathbf{s}(\mathbf{t})$. Using the current knots and the control points, the process proceeds with finding the parameter values that yield smallest distances between the data and the approximation B-spline curve. The Levenberg-Mardquardt method is employed to find such parameter values. After these best parameter values are obtained, they are used to construct new $\mathbf{R}$ and, subsequently, the new $\mathbf{R}$ is used to find the new set of control points. The steps are repeated until no further refinement of least square errors can be obtained.

Sarkar's work lacks theoretical justification to prove that the new control points obtained from the new parameter values will produce smaller least squares error. Sarkar's method, as reported in his articles, fails whenever knots are included in the optimization.

Hoschek: Parameter Optimization

Hoschek's approach[42] is almost identical to Sarkar's. The difference lies in the method to find the new parameter values. The refinement of parameter is performed using the formula

$$\bar{t_i} = t_i + \Delta c_i \frac{b-a}{\mu} \tag{2.42}$$

where a and b are the first and the last knots respectively, μ is the length of the polygon spanned by the data set $\mathbf{p}_i$'s, and Δc_i is the magnitude of the projected error $\mathbf{d}_i$ on the tangent line at point $\mathbf{y}_i$ on the approximation curve. Figure 2.1 illustrates the meaning of Δc_i.

Hoschek's work also lacks of theoretical background which proves that the new control points obtained from the new parameter values will always produce smaller least squares error.

Rogers: Parameter Optimization

Rogers' approach[76] is similar to the previous two approaches. Rogers uses the first-order approximation to calculate the gradient. Steepest descent is then used

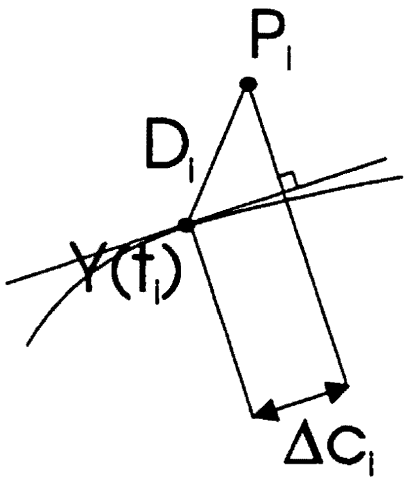


Figure 2.1: Projection of Error Vector $\mathbf{d}_i$ and the Correction Coefficient Δc_i

to optimize the least square error. The adjustable parameters are limited only to parameterization of data. This method is based on approximate values of gradient and the steepest descent method has poor convergence.

2.5.5 Survey on Parameterization of Data

This subsection presents a literature survey on B-spline curve approximation where nonlinear parameters are not adjusted. The nonlinear parameters are set up only once, followed by computation of control points from the overdetermined equation $\mathbf{Rc} = \mathbf{p}$. Although setting up the nonlinear parameters means the determination or computation of the knots, the weights, and the parameterization of data, most of existing methods to compute these three sets of parameters focus on the last set of parameters, i.e. the parameterization of data. Thus, this survey is focused on the assignment of parameterization of data.

The most widely known family of parameterizations of data has the form of

$$t_i = t_{i-1} + \frac{||\mathbf{p}_i - \mathbf{p}_{i-1}||_2^\beta}{\sum_{k=2}^{m} ||\mathbf{p}_k - \mathbf{p}_{k-1}||_2^\beta} \quad \text{for } i = 2, \cdots, m \tag{2.43}$$

The well known values of β are 0, 1, and $\frac{1}{2}$, each of which represent uniform, chord length, and centripetal, parameterization respectively[70, 22, 44]. Uniform parameterization, in general cases, produces poor results because this parameterization neglects the positions of the points. Chord length parameterization generally produces better results than the uniform parameterization. If corners are not present in the data, chord length parameterization usually produces "adequately pleasing" results. However, if corners are present, this parameterization usually fails to reproduce the corners. In his paper[54, 55, 56], Lee proposes the centripetal parameterization to overcome the inadequacy of chord length parameterization to produce curves with corners.

Parameterization of data is an important issue in interpolation problems. High curvature regions, corners, wiggles, and smoothness, are the main theme in the research of parameterization of data. Works on the parameterization of data, in particular for interpolation problems, are numerous[2, 12, 21, 25, 36, 38, 42, 54, 55, 56, 62, 61, 63]. In approximation problems, satisfying prespecified accuracy is the main issue. Thus, selecting a parameterization of data is usually less significant than finding the minimum degrees of freedom (number of basis functions). Most of the literature on approximation problems use the chord length or centripetal parameterization.

2.6 Summary

This chapter has presented the theoretical background and literature review of NURBS least squares. The theoretical exposition has shown the proper expression of the objective function and its gradient in Equations (2.3) and (2.29) respectively. The literature review has shown that there are two main obstacles in finding the solution for this problem, they are: the lethargic property of gradient with respect to coincident knots and the inaccuracy of control points. The next chapter will present a new approach to overcome these obstacles.

Chapter 3

Nonlinear Least Squares Method

The previous chapter has highlighted two obstacles in solving the NURBS least squares problem: the possible degradation of $\mathbf{R}$, that yields poor control points, and the lethargic property of gradient with respect to coincident knots. These obstacles motivate the development of a new approach that is presented in this chapter.

The degradation of $\mathbf{R}$ can be overcome by introducing a new constraint on the maximum permissible limit that the condition number of $\mathbf{R}$ may attain. This constraint is applied to the line minimization phase of the optimization. Detailed exposition and rationale of this constraint will be presented in this chapter. Jupp has pointed out that the main cause of the lethargic behavior of the optimization is due to the lack of orthogonal component in the gradient. The descent direction identified by Broyden-Fletcher-Goldfarb-Shanno, commonly called BFGS direction, has orthogonal component provided that the the diagonal elements of the approximate Hessian do not dominate the off-diagonal elements[15, 31, 73]. In this work

the BFGS descent direction is used to defeat the lethargic property.

In this thesis an airfoil data set is approximated with a NURBS curve using nonlinear least squares method. In the next step parameters in the approximation NURBS curve are optimized. This optimization is subject to both linear and nonlinear constraints. The linear constraint applies to knots, weights, and parameterization for data. The nonlinear constraint applies to the condition number of matrix **R**. The details of the constraints are described next.

3.1 Constraints

3.1.1 Sequence of Data

The shape of a curve, represented by a set of data points, is determined by the coordinates of the data and its sequence. Figure (3.1) shows two sets of data that have identical coordinates but different sequences. The figure clearly shows that the difference in the sequence results in different shapes.

Figure 3.1: Different Topology Resulting in Different Shapes

In this given problems the parameter values assigned to the data serve as sequence information. This can be realized by constraining the values of the parameters to be nondecreasing. This criterion must never be violated.

Parameterization for data can be done in many ways. One method of assignment of the parameter value to the i-th point is by incrementing the parameter value of the preceding point, i.e. the $(i-1)$-th point. The expression for the i-th parameter

value is $t_i = t_{i-1} + \delta t_i$ for all i, where δt_i is a positive real number. The positivity of δt_i guarantees that the resulting parameterization is nondecreasing.

The difference among existing methods of parameterization of data lies in the definition of δt_i.

If parameterization is considered as an adjustable parameter, the nondecreasing property should not be violated. The simplest way to achieve this is to introduce constraints on the parameterization. Since δt_i is positive, the expression for these constraints are $t_i - t_{i-1} > 0$. In our implementation, this expression is relaxed by first substituting the ">" with "$\geq$", and secondly by modifying the expression of the constraint to $t_i - t_{i-1} \geq \delta t_i$. The order of magnitude of δt_i must be greater than the computer precision to maintain the nondecreasing property of the parameters. This modification of constraint is done to facilitate the solution of the linearly constrained least square problems. In this work, the *Active-Set*[31] method is used for the adjustment of parameters.

The matrix form of these linear constraints is

$$\mathbf{A}_t \mathbf{t} \geq \delta \mathbf{t} \tag{3.1}$$

Subscript t refers to parameterization of data. The size of $\mathbf{A}_t$ is $m \times (m-1)$ where m is the number of point in the data set. Matrix $\mathbf{A}_t$ is a banded matrix with bandwidth equal to two. The nonzero elements of $\mathbf{A}_t$ are $a_{i,i} = -1$ and $a_{i,i+1}$ for $i = 1 : (m-1)$.

3.1.2 Constraints Due To The Use of The B-spline Basis Functions

In B-spline curves the knot sequence must be nondecreasing. Therefore, adjustment of knots must take this requirement into account. This requirement can be met by introducing constraints on the knots. The constraints on a knot sequence with q elements can be written as $u_i - u_{i-1} \geq 0$ for $i = 2 : q$. The matrix form of the constraints is

$$\mathbf{A}_u \mathbf{u} \geq \mathbf{0} \tag{3.2}$$

where subscript u refers to knots. Matrix $\mathbf{A}_u$ is banded with bandwidth equal to two, and the elements in the bandwidth are $a_{i,i} = -1$ and $a_{i,i+1} = 1$ for $i = 1 : (q-1)$.

For practical purposes, the end knots are clamped to force the curve to interpolate the first and the last control points; the clamping can be represented as a set of equality constraints $u_1 = u_2 = \cdots = u_k$ and $u_{q-k+1} = u_{q-k+2} = \cdots = u_q$ where k is the order of the B-spline basis and q is the number of knots. Since B-spline basis is invariant with respect to affine transformation of knots, end knots can be constrained further to $u_1 = u_2 = \cdots = u_k = 0$ and $u_{q-k+1} = u_{q-k+2} = \cdots = u_q = 1$.

Other constraints for practical purpose are to set the parameter values of the first and the last points equal to the first and the last domain knot respectively, that is $t_1 = 0$ and $t_m = 1$ as described in Appendix A. The purpose of these constraints

is to guarantee that the data is fitted by the whole curve instead of a portion of it.

To prevent singularity of the basis and portability of the approximation curve, weights are constrained to positive values, that is $w_i > 0$ for all i. Using the same argument as in the case of constraint on parameters, constraint on weights are modified to $w_i \geq \delta_w$ where δ_w is a positive real number that is very close to, but larger than, computer precision.

Since rational B-spline basis is invariant with respect to uniform scaling of weights, adjustment of weights must be able to prevent the chance of uniform scaling to occur. The prevention can be done by constraining one of the weight to remain during the adjustment. This is implemented by introducing an equality constraint $w_1 = 1$.

3.1.3 Condition Number of Matrix R

In this work a constraint is used to maintain the good behavior of the control points. Since control points, $\mathbf{c}$, are solved from the equation $\mathbf{Rc} = \mathbf{p}$, their uniqueness and accuracy depends on the reciprocal of the condition number of matrix $\mathbf{R}$. Unique $\mathbf{c}$ is obtained if the reciprocal of the condition number of $\mathbf{R}$ is not zero. Schoenberg-Whitney's rule is a standard method to verify uniqueness of $\mathbf{c}$[13]. Accuracy of $\mathbf{c}$ depends on computer precision in addition to the reciprocal of condition number of $\mathbf{R}$. Accuracy of $\mathbf{c}$ cannot be guaranteed if the order of magnitude of reciprocal of condition number is less than the computer precision. The strategy used to guarantee uniqueness and accuracy of $\mathbf{c}$ is explained in the following paragraphs.

Uniqueness is guaranteed if $\mathbf{R}$ has full column rank, which means that every

column of $\mathbf{R}$ contains one or more nonzero elements. In other words, $\mathbf{R}$ has full column rank if every support of B-spline basis contains at least one data point. For a given set of parameterization of data $\mathbf{t} \in \Re^m$, and a complete set of rational B-spline basis N_i^k defined along a knot sequence $\mathbf{u} \in \Re^{n+k}$, the Schoenberg-Whitney rule requires the existence of set $\mathbf{t}^* \subset \mathbf{t}$ to guarantee full column rank of matrix $\mathbf{R}$. The set $\mathbf{t}^*$ is defined as

$$\left\{\{\mathbf{t}^* \in \Re^n\} \subset \{\mathbf{t} \in \Re^m\} : u_j < t_j^* < u_{j+k} \text{ for } j = 1, \cdots, n\right\} \tag{3.3}$$

Since the interval $[u_j, u_{j+k})$ is the support of the B-spline basis N_j^k, existence to t_j^* means that every basis is supported at least one data point.

Existence of $\mathbf{t}^*$ guarantees that matrix $\mathbf{R}$ has full column rank and, therefore, guarantees uniqueness of the solution of the overdetermined equation $\mathbf{Rc} = \mathbf{p}$.

For computation of $\mathbf{Rc} = \mathbf{p}$ with finite precision, accuracy of $\mathbf{c}$ must be warranted. The accuracy of $\mathbf{c}$ can be assessed from the condition number of matrix $\mathbf{R}$. When the reciprocal of the condition number of matrix $\mathbf{R}$ reaches computer precision (or stated differently, the condition number of $\mathbf{R}$ becomes large), matrix $\mathbf{R}$ becomes ill-conditioned and the solution of $\mathbf{Rc} = \mathbf{p}$ will be inaccurate or will have wildly large components.

An ideal way to prevent ill-conditioning of matrix $\mathbf{R}$ is to introduce a constraint on the condition number of the matrix. The theoretical expression of constraint would be $\text{cond}(\mathbf{R}) \leq \Lambda$ where $\text{cond}(\mathbf{R})$ is a function of knots, weights, and parameterization of data, and Λ is a prespecified limit whose order of magnitude

must conform to computer precision. However, the mathematical expression of the condition number is difficult, if not impossible, to derive.

Fortunately, ill-conditioning of matrix **R** has a specific pattern with respect to feasible knots, weights and parameterization of data, if the number of data is greater than the number of basis. The following paragraph will explain how the pattern of condition number of **R** can be used to set up a strategy to prevent ill-conditioning of **R**.

Matrix **R** becomes ill-conditioned whenever the largest rational B-spline basis in one or more columns of **R** approaches zero. This happens if either the weights associated with the basis approaches zero or the support of the basis is almost devoid of data, that is the data within the support lie very close to the end of the support.

Figure 3.2 shows the monotonically decreasing behavior of the rational B-spline basis when the weight approaches zero; notice that the basis is exactly zero at zero weight. At this limit, the condition number of **R** reaches infinity. This phenomena is used to assume that the condition number of **R** behaves reciprocally to the behavior of the basis, i.e. the condition number of **R** is monotonically increasing as the weight approaches zero.

Matrix **R** may also become ill-conditioned whenever the data supporting the basis functions are concentrated near the ends of the support of the basis functions. Figure 3.3 is used to explain how the concentration of data can lead to ill- conditioning of **R**. The curve represents the $N_5^4(t)$ cubic B-spline basis function. The support of $N_5^4(t)$ is $[u_8, u_9)$. In this case this basis function is set to be supported

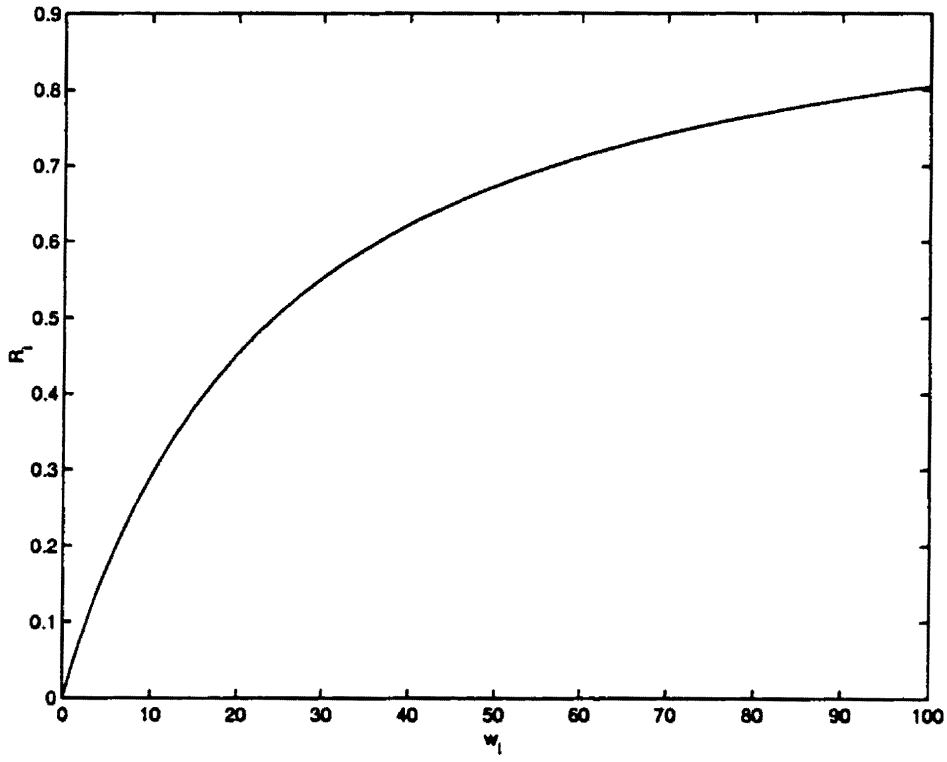


Figure 3.2: Vanishing Weight and Rational B-spline Basis

by two data whose parameter values are t_{11} and t_{12}. Because these data are the only support for the $N_5^4(t)$ B-spline basis function, the 11-th and the 12-th rows of $\mathbf{R}$ are those at which the $N_5^4(t)$ B-spline basis function are nonzero; the $N_5^4(t)$ for other rows are zero. Whenever t_{11} and t_{12} are getting closer to u_8 and/or u_9, the $N_5^4(t_{11})$ and $N_5^4(t_{12})$ will be decreasing, and at some point both of these basis functions will be close to the machine precision. At this point, the matrix $\mathbf{R}$ will become ill-conditioned. More severe case occurs when t_{11} and t_{12} moves away from $[u_8, u_9)$. In this case, the column of $\mathbf{R}$ corresponding to $N_5^4(t)$ are all zero, and the matrix $\mathbf{R}$ no longer enjoys full column rank. This column rank deficiency will result in nonunique solution of $\mathbf{Rc} = \mathbf{p}$.

To show the ill-conditioning of $\mathbf{R}$ with respect to the movement of parameterization for data to the ends of the support of a cubic B-spline basis function, the knot vector in Figure 3.3 are set to $\mathbf{u} = \{0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6\}$ and the parameterization for data are set to

$$\begin{aligned} \mathbf{t} = \{&0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, \\ &t_{11}, t_{12}, \\ &5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 6\} \end{aligned} \tag{3.4}$$

Therefore, the cubic B-spline basis function $N_5^4(t)$ are supported only if $(t_{11}, t_{12}) \in [u_5 = 1, u_9 = 5)$. A test case was run by varying t_{11} and t_{12} as follows

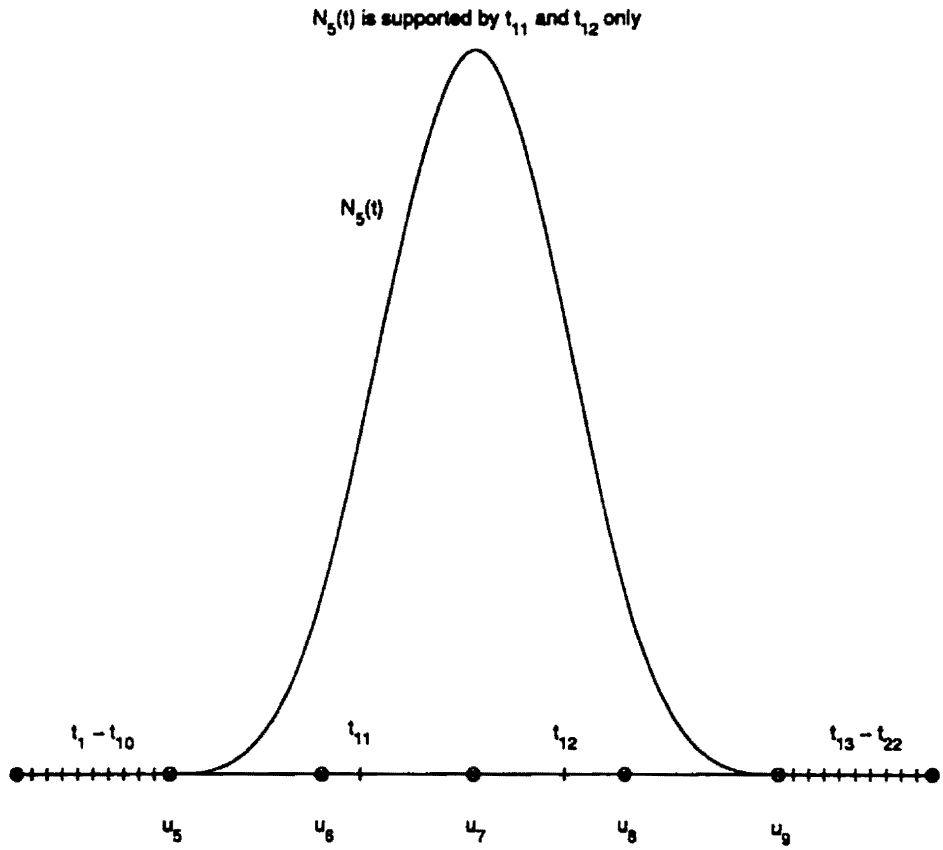


Figure 3.3: The B-spline Basis Function $N_5^4(t;\mathbf{u})$ and Its Supporting Data t_{11} and t_{12}

$$(t_{11}, t_{12}) \in [1, 5) \text{ subject to } t_{11} \leq t_{12} \tag{3.5}$$

The triangle in Figure 3.4 shows the domain of (t_{11}, t_{12}) defined by this equation. Based on the condition number of matrix $\mathbf{R}(t_{11}, t_{12})$, this triangular area are into two subareas: white subarea where $\text{cond}(\mathbf{R}) \leq 500$ and shaded subarea where $\text{cond}(\mathbf{R}) > 500$. The shaded subarea in the bottom left portion of the triangle represents the concentration of (t_{11}, t_{12}) near the knot u_5. The shaded subarea in the top right portion of the triangle represents the concentration of (t_{11}, t_{12}) near the knot u_9. The shaded subarea in the top left portion represents the case where t_{11} is near u_5 and t_{12} is near u_9.

Another way that the B-spline basis function can loose the supporting data is due to movement of knots. This knot movement may, again, result in concentration of data at the end of supports of the basis function. To illustrate this, the same knot vector and parameterization for data are used, except that t_{11} is set to 2.3 and t_{12} is set to 3.6. In this case, the knots u_8 and u_9 are varied to

$$(u_8, u_9) \in [3, 5) \text{ subject to } u_8 \leq u_9 \tag{3.6}$$

When $(u_8, u_9) \leq t_{11}$, the B-spline basis function $N_5^4(t)$ are not supported, and the column of $\mathbf{R}$ corresponding to $N_5^4(t)$ are all zero. Figure 3.5 shows the triangular area defined by $(u_8, u_9) \in [3, 5)$ subject to $u_8 \leq u_9$. The white sub-

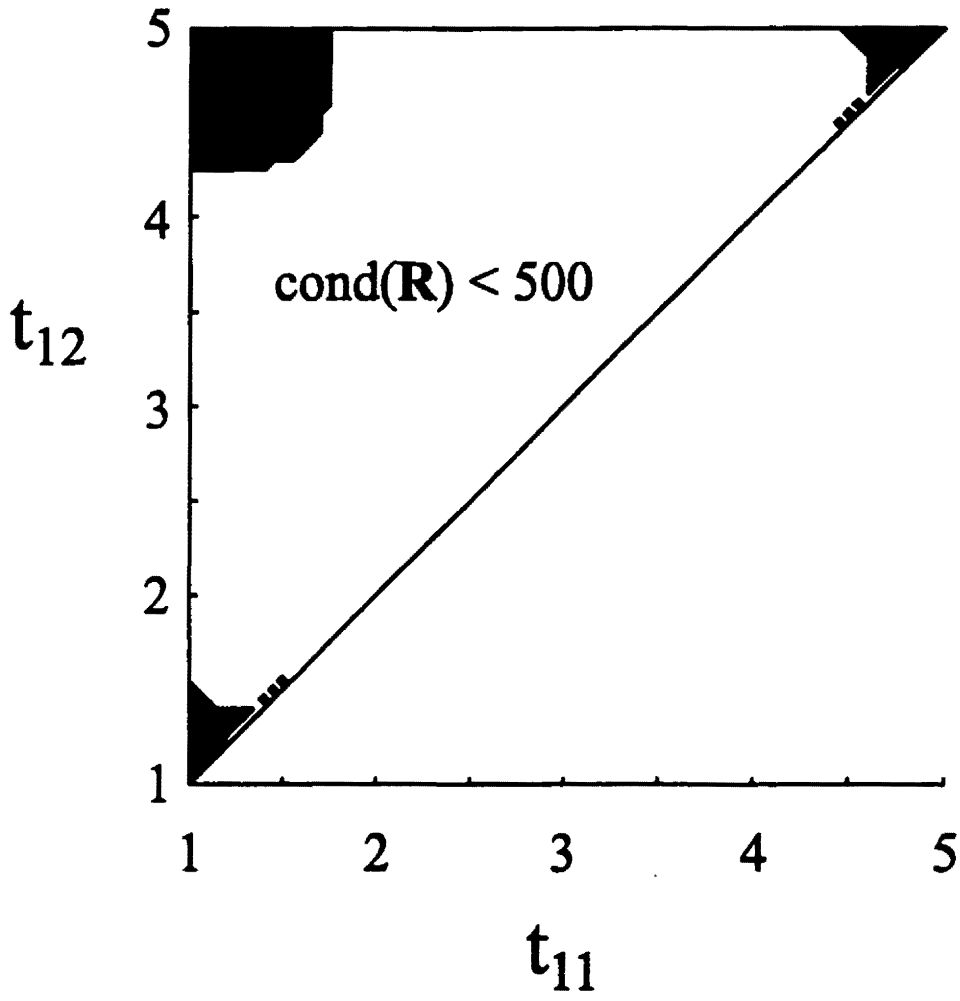


Figure 3.4: Triangular area defined by $(t_{11}, t_{12}) \in [1, 5)$ subject to $t_{11} \leq t_{12}$. Shaded subareas represent (t_{11}, t_{12}) resulting in cond $(\mathbf{R}) > 500$. White subarea represent (t_{11}, t_{12}) resulting in cond $(\mathbf{R}) \leq 500$.

area represents combinations of $(u_8, u_9) \in [3, 5)$ subject to $u_8 \leq u_9$ that yield cond $(\mathbf{R}) \leq 500$ whereas the shaded subarea represents conbination of $(u_8, u_9) \in [3, 5)$ subject to $u_8 \leq u_9$ that yield cond $(\mathbf{R}) > 500$.

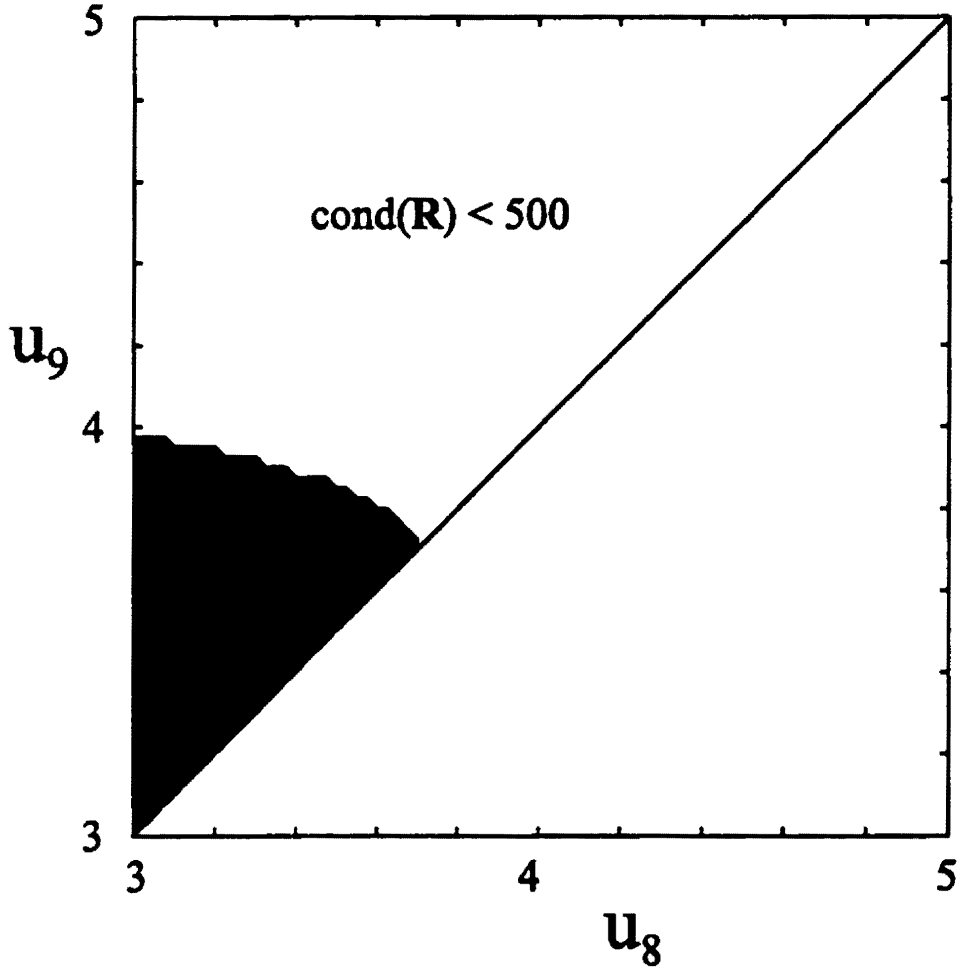


Figure 3.5: Triangular area defined by $(u_8, u_9) \in [3, 5)$ subject to $u_8 \leq u_9$. Shaded subareas represent (u_8, u_9) resulting in cond $(\mathbf{R}) > 500$. White subarea represent (t_{11}, t_{12}) resulting in cond $(\mathbf{R}) \leq 500$.

The examples shown in Figures 3.2, 3.4, and 3.5, demonstrate the existence of unfavorable regions within which the condition number of matrix $\mathbf{R}$ is large such

that the solution of $\mathbf{Rc} = \mathbf{p}$ may suffer inaccuracy or may be nonunique. The examples show that the locations of this unfavorable regions are close to the linear constraints, leaving the interior favorable.

3.1.4 Summary of Constraints

In summary, the feasible domain of nonlinear parameters (knots, weights, and parameterization of data) is given by the following constraints:

1. Major constraints:

   - Linear constraints:

     Feasible parameterization of data : $\mathbf{A}_t\mathbf{t} \geq \mathbf{0}$

     Feasible knot sequence : $\mathbf{A}_u\mathbf{u} \geq \mathbf{0}$

     Feasible weights : $\mathbf{A}_w\mathbf{w} \geq \mathbf{0}$

   - Nonlinear constraints:

     Accuracy of control points : $\text{cond}\,(\mathbf{R}) \leq \Lambda$

   These major constraints determines the type of optimization that is used to obtain the solution of the nonlinear least squares problem. This will be described in the next section.

2. Minor constraints:

Invariance of knots w.r.t. affine transformation	:	$u_1 = 0$ and $u_q = 1$
Invariance of weights w.r.t. uniform scaling	:	$w_1 = 1$
Coincident end knots	:	$u_1 = u_2 = \cdots u_k = 0$ and $u_{q-k+1} = u_{q-k+2} = \cdots = u_q = 1$
Bound of parameterization of data	:	$t_1 = 0$ and $t_m = 1$

Due to their simple expressions, all of these minor constraints can be satisfied by setting the variables and removing them from the list of adjustable parameters.

3.2 Method To Satisfy Constraints

This section presents the method to satisfy all of the constraints described in the previous section. The existence of nonlinear constraint, i.e. cond($\mathbf{R} \leq \Lambda$), significantly increases the complexity of satisfying the constraints[31]. The simplest way to deal with a mix of linear and nonlinear constraints is to apply the method of nonlinear constraints to the overall constraints. However, Gill suggests that it is almost always worthwhile to treat the linear constraints separately whenever possible[31].

In this work, the characteristic of the nonlinear constraint shown in Figures 3.2, 3.4, and 3.5, suggests that the infeasible domain with respect to the nonlinear constraint is concentrated near the linear constraints. Therefore, most of the interior of the feasible domain with respect to the linear constraints also satisfies the nonlinear constraint. This characteristic forms the justification to separate the method

of satisfying the linear constraints from that of the nonlinear constraint. The separation is performed by, firstly, computing the critical step length that maintains the feasibility with respect to the linear constraint, and secondly reducing this critical step if necessary in order to maintain the feasibility with respect to the nonlinear constraint. The detail of this strategy is presented in the following two subsections.

3.2.1 Method To Satisfy The Linear Constraint

Satisfying linear constraints is performed using the active-set method; adoption of Golub's procedure[31] is used in our codes.

To retain the feasibility of the subsequent iterations, it is necessary to ensure that the step length does not violate any linear constraints in the inactive set $\mathbf{A}_i$. Thus, the step along $\mathbf{v}$ to the nearest linear constraint (if any) becomes an upper bound for step length α. To calculate this bound, the vector $\psi = \mathbf{A}_i\mathbf{v}$ is needed. If ψ_j is nonnegative for all j, any positive movement along $\mathbf{v}$ will not add any active linear constraint; in this case, the upper bound is infinity. However, if one or more components of ψ are negative, there exists a critical step where the corresponding linear constraint becomes binding, i.e. the j-th inactive linear constraint will become active. For example, if several components of ψ are negative. Denoting $\mathbf{a}_m$ as the m-th row of $\mathbf{A}_i$ and $b_{i,m}$ as the m-th component of $\mathbf{b}_i$, vector γ whose components are defined as

$$\gamma = \min_{\text{over } i} \left(\frac{b_{i,m} - \mathbf{a}_m^T \mathbf{x}}{\psi_m} \right) \qquad \text{for all } m \text{ where } \psi_m < 0 \tag{3.7}$$

can be calculated. Note that for feasible x's, the denominator is always negative. Thus, for all m where ψ_m is negative, the term inside the braces is always positive since the equation is applied only to negative denominator. The upper bound of the step length that retains the feasibility of subsequent iterations with respect to the linear constraints is the smallest value of the term in the braces for all i.

3.2.2 Method To Satisfy The Nonlinear Constraint

After the critical step that maintains the linear constraints is obtained, i.e. γ in Equation 3.7, this step are verified further to ensure that nonlinear constraint is not violated. The verification of this critical step and the strategy of reducing this step to satisfy the nonlinear constraints is described next.

In Figure 3.6, the triangle represents the feasible domain with respect to the linear constraints. The white subarea in the triangle represents the feasible domain with respect to the nonlinear constraint, whereas the shaded subarea represents the infeasible domain with respect to the nonlinear constraint. The solid circle represent the position at the current iteration; this position satisfies both the linear and nonlinear constraints. The vectors $\mathbf{d}_1$ and $\mathbf{d}_2$ represent two possible descent directions. The figure also shows two critical steps γ_1 and γ_2 that are associated with $\mathbf{d}_1$ and $\mathbf{d}_2$ respectively; these critical steps are computed using Equation 3.7.

Line minimization along the descent $\mathbf{d}_1$ can be performed at full step γ_1 since the nonlinear constraint will be satisfied at this step. In this work, the condition number of matrix $\mathbf{R}$ is computed at the full step γ after it is obtained from the Equation 3.7.

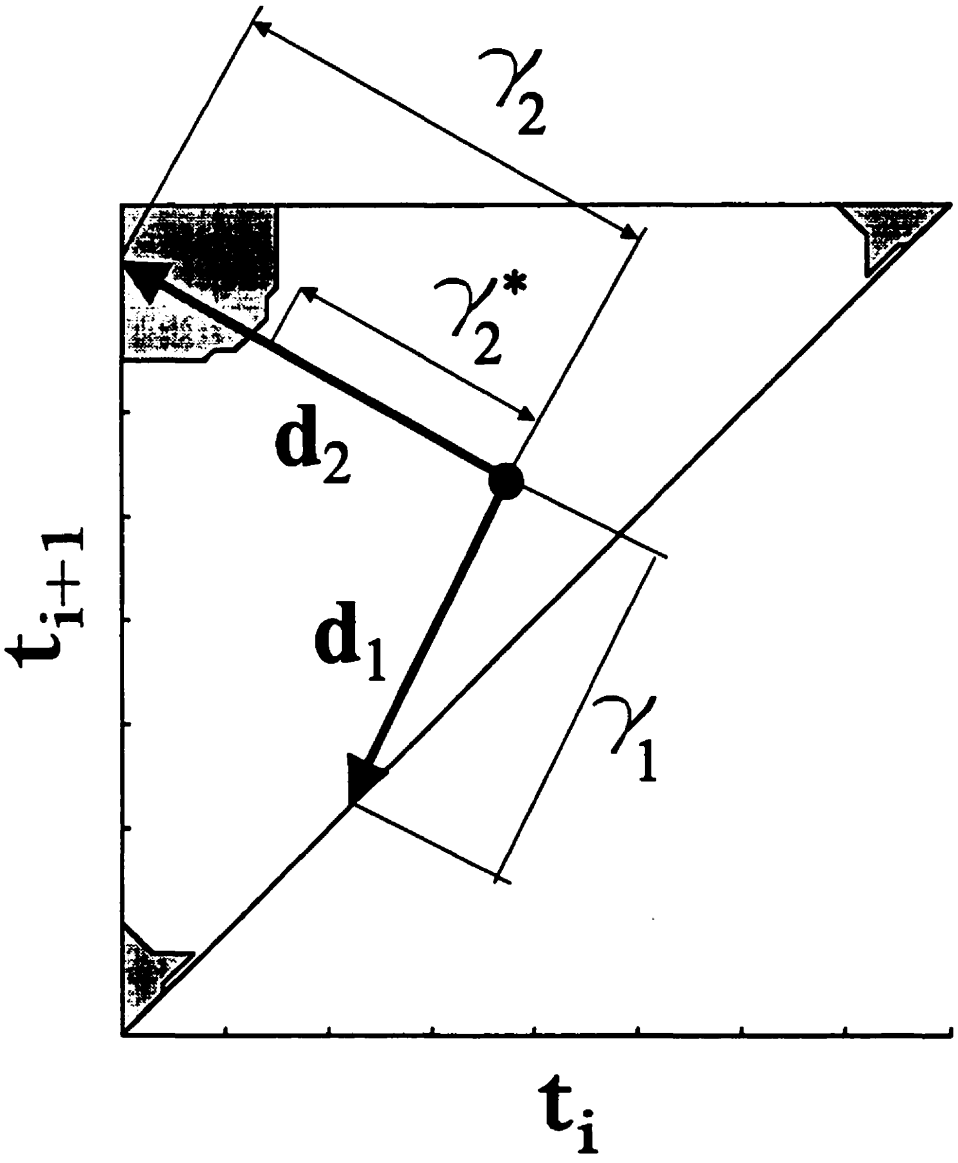


Figure 3.6: Reduction of Critical Step To Satisfy The Nonlinear Constraint

On the other hand, line minimization along the descent $\mathbf{d}_2$ cannot be performed at full step γ_2 because violation of the nonlinear constraint will occur. To maintain the satisfaction of the linear constraint, a reduced critical step, denoted by γ_2^* in Figure 3.6, is used instead of the γ_2. The method of computing the reduced critical step is presented next.

Ideally, the procedure to reduce γ must be developed with a consideration to conform to the behavior of the condition number of $\mathbf{R}$ with respect to γ. However, other than just monotonically increasing with respect to increasing step length, the behavior of the condition number of $\mathbf{R}$ is difficult to be described accurately. Thus, the simplest method of reducing the maximum feasible step is by reducing it with a constant factor. The crucial step of the development of such a method is to find the factor of reduction that produces the best compromise between adequate length of maximum feasible step and fast convergence of finding the step that yields feasible condition number of $\mathbf{R}$. If the factor is too big, fast convergence is obtained at the expense of excessively short maximum step length, which in turn will reduce the rate of convergence of the overall optimization. On the other hand, an overly small factor will reduce the convergence of finding feasible maximum step length. From trial and error experiment, a reduction factor of 20% yields the best compromise.

With the scheme of constant-factor reduction and the empirical reduction factor of 20%, the mechanism to maintain the feasibility of the condition number of $\mathbf{R}$ can be setup to reduce a given γ by 20% if the γ yields infeasible condition number of $\mathbf{R}$. If the reduced γ yields a feasible condition number, it is accepted. Otherwise, it is further reduced by 20% and the process is repeated until the condition number

at γ is satisfied.

From our experiment, incorporation of maximum allowable condition number of $\mathbf{R}$ into the determination of maximum feasible step proved to be effective to prevent inaccuracy of the solution of $\mathbf{Rc} = \mathbf{p}$.

This section has presented a setup to satisfy the nonlinear constraint by limiting the maximum permissible step such that the constraint will not be violated. Two useful features of this approach are:

- *Explicit expression of nonlinear constraint is unnecessary.* The concentration of the infeasible domain with respect to the nonlinear constraint at narrow area around the linear constraints (as shown in Figures 3.2, 3.4, and 3.5) allows the use of truncation of the critical step (as shown in Figure 3.6) as a mechanism to maintain feasibility with respect to the nonlinear constraint. This feature permits a simple method for satisfying this constraint.

- *Simpler objective function.* This feature is in contrast with the more complicated objective function due to Gengoux[30]. This feature's lack of penalty term yields simpler objective function such that the exact gradient can be derived. Availability of analytical gradient is very favorable for optimization problems.

This section ends the first part of this chapter which introduces the definition of constraints and the use of truncation of the critical step to satisfy the linear constraint. The next section will describe the descent direction to reduce the least square error.

3.3 Descent Direction

The nonlinear least squares problems are distinguished from the general optimization problems by the special structure of the Hessian matrix of the objective function. The expression of Hessian are

$$\mathbf{H} = \mathbf{J}^T\mathbf{J} + \sum_i e_i\mathbf{H}_i \tag{3.8}$$

where $\mathbf{J}$ represents the Jacobian of the vector $\mathbf{e} = \left(\mathbf{I} - \mathbf{R}\mathbf{R}^+\right)\mathbf{p}$, e_i represents the elements of $\mathbf{e}$, and $\mathbf{H}_i$ represents the Hessian of e_i. Most of algorithms for nonlinear least squares exploit this special structure.

Least-squares methods are typically based on the premise that eventually the first-order term $\mathbf{J}^T\mathbf{J}$ of (3.8) will dominate the second-order term $\sum_i e_i\mathbf{H}_i$. This is applicable whenever, at the solution, the residuals e_i are significantly smaller than the eigenvalues of $\mathbf{J}^T\mathbf{J}$. Least-squares methods that use this assumption are the Gauss-Newton and the Levenberg-Mardquardt methods. The Gauss-Newton method set the second-order term of (3.8) to zero. The Levenberg-Mardquardt method, on the other hand, set the second-order term to a scalar multiplication of identity matrix, i.e $\sum_i e_i\mathbf{H}_i = \lambda\mathbf{I}$. Generally, the general convergence of these method is linear, except for zero-residual problems where the method exhibits quadratic convergence.

The quasi-Newton method is aimed at least squares problem where the assumption of domination of $\mathbf{J}^T\mathbf{J}$-term over the $\sum_i e_i\mathbf{H}_i$-term does not apply, i.e. the

class of *large residual* problem. This method substitutes the $\sum_i e_i \mathbf{H}_i$-term with a quasi-Newton approximation M; an example of a quasi-Newton approximation is the update based on the BFGS update. However, it is important to note that the properties of hereditary positive-definiteness and n-step termination of the BFGS update in the general optimization problem do not apply in the least squares case. This is because the approximation is applied only to *part* of the Hessian. The blending the exact and the approximated curvature information is suspected to be the cause of the slower convergence of the quasi-Newton method in the least square problems than that in the general optimization problems.

In this research, we do not assume that, in general, the residuals e_i are significantly smaller than the eigenvalues of $\mathbf{J}^T\mathbf{J}$. This, combined with the possible slower convergence of the quasi-Newton for least squares problem, resulted in decision to treat this least square problem as a general minimization problem. Since the objective function of least squares fitting is smooth and its analytical derivative is available, second-order descent direction is used. Among existing second-order methods, variable metric methods (also known as quasi-Newton methods for general minimization problems) is well known for good convergence. The two main flavors of this method are: BFGS and DFP. The former has been acknowledged to be empirically superior than the latter. The BFGS descent direction is accompanied by the standard backtracking method. For this research, codes to compute the BFGS descent direction, the approximate Hessian, and backtracking are adopted from algorithms presented in Numerical Recipes[73], and modified to suit the active-set method. Restart is performed when the approximate Hessian becomes near-singular

or nonpositive definite due to buildup of roundoff errors.

3.4 Dropping Linear Constraint from Active Set

For general problems of linearly constrained optimization problems, the bookkeeping of gradient, approximate Hessian, and descent direction, are performed in the null space of active constraints. The purpose of performing the computation in the null space of the active constraints is to minimize the effect of rounding off error.

In this research, besides the bookkeeping mentioned earlier, an additional bookkeeping of gradient, approximate Hessian, and descent direction, are performed . This extra storage and computation is critically important when one or more knot constraints are active. Update of approximate of Hessian and computation of descent direction are performed in the unconstrained space when knot constraints are active. The goal is to identify whether the descent direction has components that are orthogonal to the active knot constraints and point into the feasible domain. If it does, removal of knot constraints from the active set is attempted. Otherwise, the currently active knot constraints remain in the active set. The Lagrange first order approximation was used for the determination to eliminate an active linear constraint from the active set[31].

This method of additional bookkeeping stems from the observation that the gradient is orthogonal to the range space of the active knot constraints. It must be noted, however, that this strategy may interfere with the standard anti zigzagging mechanism which relies on the gradient for constraint deletion from the active set. The proposed method, on the contrary, determines the deletion from the descent

direction. Nevertheless, the benefit of overcoming the lethargic property justifies the possible slower convergence due to zigzagging.

This chapter has presented a proposed method to overcome the main obstacles in the NURBS least squares problems. The next chapter will present the implementation of this method and its results.

Chapter 4

Nonlinear NURBS Fitting

This chapter presents implementation of the method that was explained in the previous chapter. The main purpose of the implementation is to measure the performance of the proposed optimization method to obtain a NURBS curve that approximates a given data set. The NURBS curve is a function of the optimized parameters, namely, knots, weights, and parameterization for data. These parameters can be optimized together or in various combinations such as knots, knots & weights, and knot & weights & parameterization for data. Another important purpose of this implementation is to measure the sensitivity of reduction of the least squares error with respect to these combinations of optimized parameters. This sensitivity analysis will be used in justifying which parameters can be dropped from the optimization. This results in an optimization with fewer degree of freedom. Also of particular importance is the effectiveness of the proposed nonlinear constraint in providing accurate control points.

Three sets of data representing a NACA 2415 airfoil, a WTEA airfoil, and an

axial compressor airfoil, were used to accomplish the above goals. Each data set was fitted with a NURBS curve using the proposed method while optimizing the following five combinations of parameters. These are:

1. knots;
2. weights;
3. parameterization for data;
4. knots and weights; and
5. knots, weights, and parameterization for data.

The first three tests were conducted to observe the behavior of each nonlinear parameter. The last two tests were conducted to observe the behavior of a combination of parameters. The first and the fourth tests allow a comparison of curve approximation of a given data set with an integral and a rational B-spline curves respectively. Results from the fifth test is used to justify whether parameterization for data can be dropped from the optimization. This is due to the large size of these parameters.

The proposed 15 test cases were fitted with the method described in Chapter 3. The termination of optimization was based on Lagrange multipliers and the standard termination criteria as suggested in[15, 73]. Arclength parameterization was used to initialize the parameterization for data. Weights were initialized to one. Knots were initialized using the averaging method[70]. Maximum permissible condition number for $\mathbf{R}$ was set to 500 based on a consideration that its reciprocal is

still much larger than the machine's precision (10^{-6}). Tolerances of activity for the linear constraints were set to 10^{-2} for knot and weight constraints, and to 10^{-5} for constraints of parameterization for the data.

4.1 NACA 2415 Airfoil

NACA 2415 is a planar airfoil and it is defined by two parts: upper and lower parts. The analytical definition of NACA 2415 can be found in Abbott's book[1]. The data curve for this airfoil was obtained by sampling 46 points on each of the upper and lower parts, totaling to 97 points after the identical points at the leading edge are merged into a single point.The data is sequenced by starting from the middle of the trailing edge, going up to the junction with the upper part and proceeding along the upper part to the junction between the upper part and the leading edge, going around the leading edge up to the junction between the leading edge and the lower part, proceeding along the lower part up to the junction of the lower part and the trailing edge, and finally proceeding to the middle of the trailing edge where the sequence of data starts. The chord length of the airfoil was set to 12 [mm]. Figure 4.1 shows the NACA 2415 airfoil.

The degree of approximation curve is set to cubic and the number of basis is set to 12.

4.1.1 Optimization of Knots

When the knots were optimized to fit the NACA 2415 data set with NURBS curve, the total least squares error, i.e. $\mathbf{e}_x^T\mathbf{e}_x + \mathbf{e}_y^T\mathbf{e}_y$, decreases from 0.186 [mm^2] to

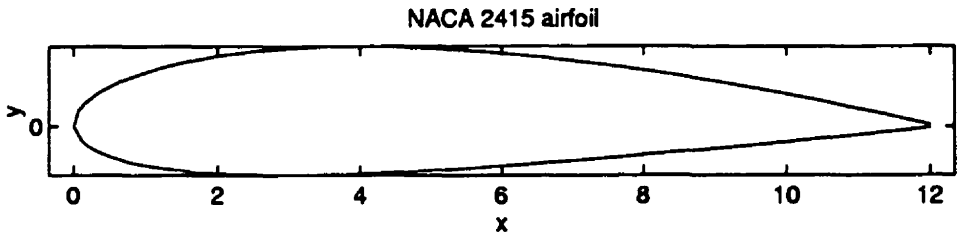


Figure 4.1: NACA 2415 with 12 mm chordlength

0.002 [mm^2] in 51 iterations; Figure 4.2 shows the decrease of error with respect to number of iteration. It is clear that the decrease of error beyond 10-th iteration is marginal (insignificant). The process of knot adjustment stops after 1225 (not shown in Figure 4.2 due to insignificance decrease of error) iteration due to gradient tolerance. Due to marginal decrease of error beyond 10-th iteration, the following discussion is focused on the first 10 iteration of this knot adjustment. Figure 4.3 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. This figure shows that, in general, the disagreement between the data and the approximation curves drops after the knot is adjusted. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.015 millimeters respectively.

Figure 4.4 shows the knot distributions, along the curves, before and after knot adjustment. The multiplicity of knots at the leading edge is 2, i.e. $u_8 = u_9$.

Figure 4.5 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of **R**.

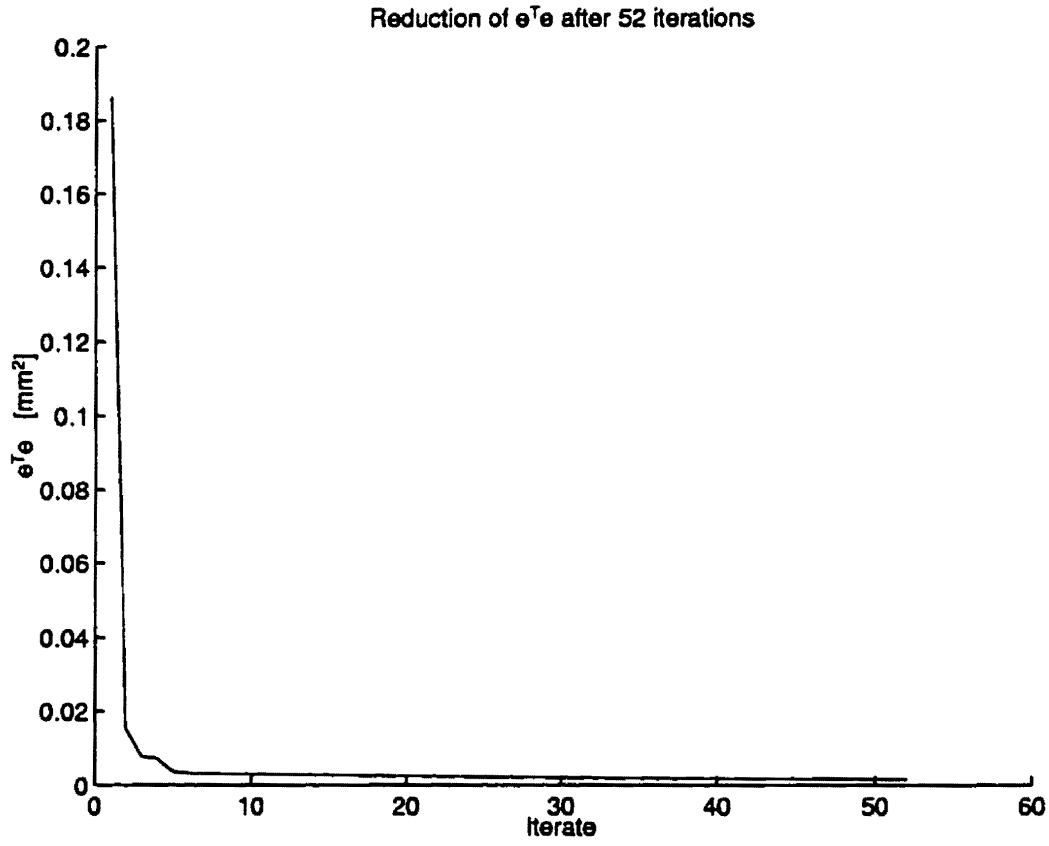


Figure 4.2: Reduction of least squares error of NACA airfoil vs. number of iteration obtained from optimizing the knots

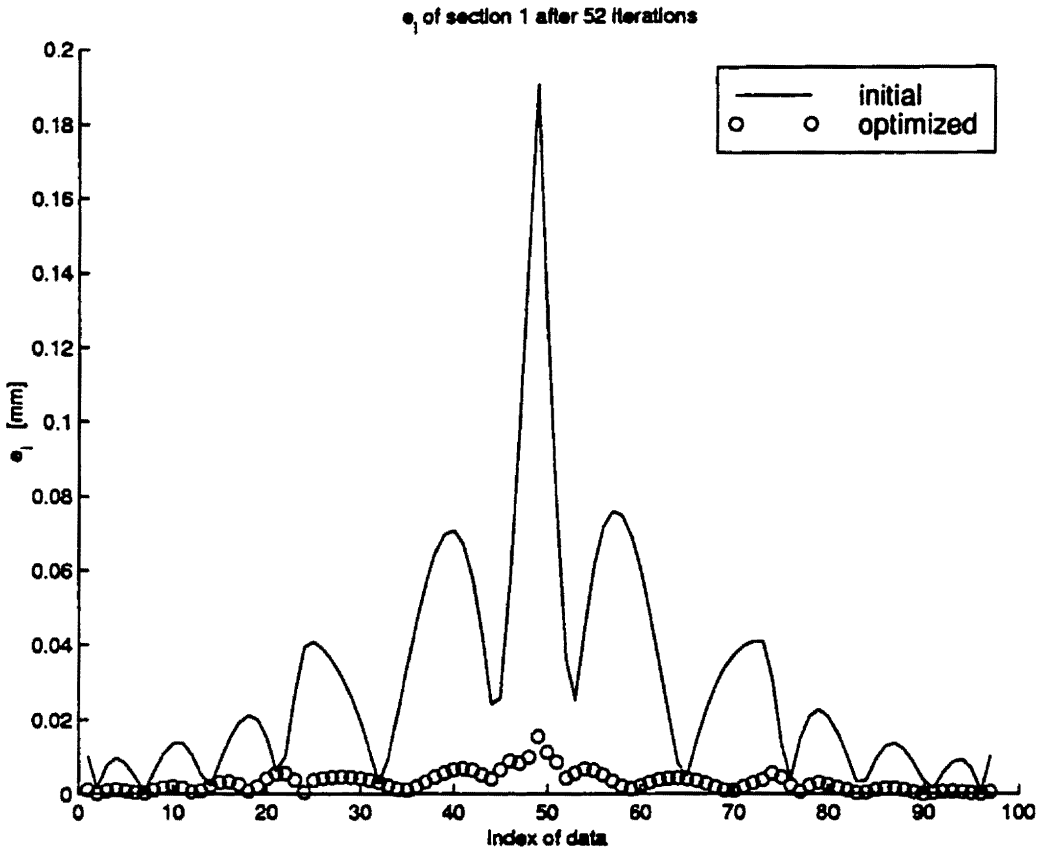


Figure 4.3: Error distributions of NACA airfoil before and after optimizing the knots

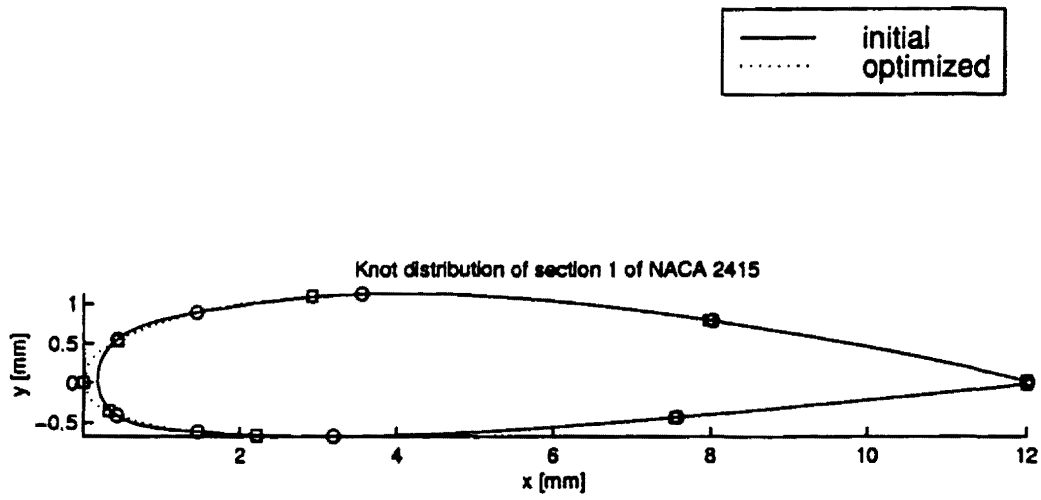


Figure 4.4: Distributions of Knots of NACA airfoil before and after optimizing the knots

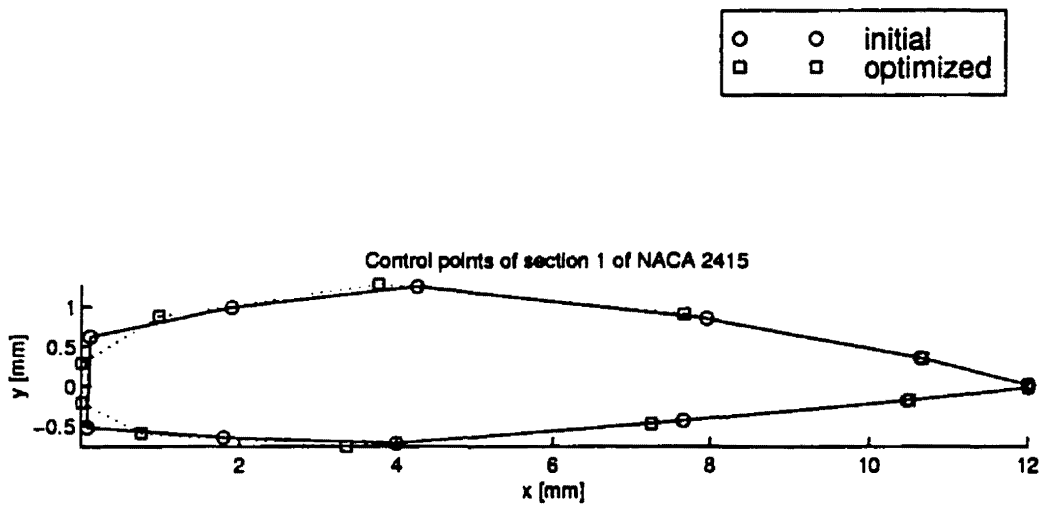


Figure 4.5: Control polygons of NACA airfoil before and after optimizing the knots

4.1.2 Optimization of Weights

The total least squares error decreases from 0.186 [mm^2] to 0.028 [mm^2] in 51 iterations; Figure 4.6 shows the decrease of error with respect to number of iteration. Figure 4.7 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.058 millimeters respectively. Figure 4.8 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

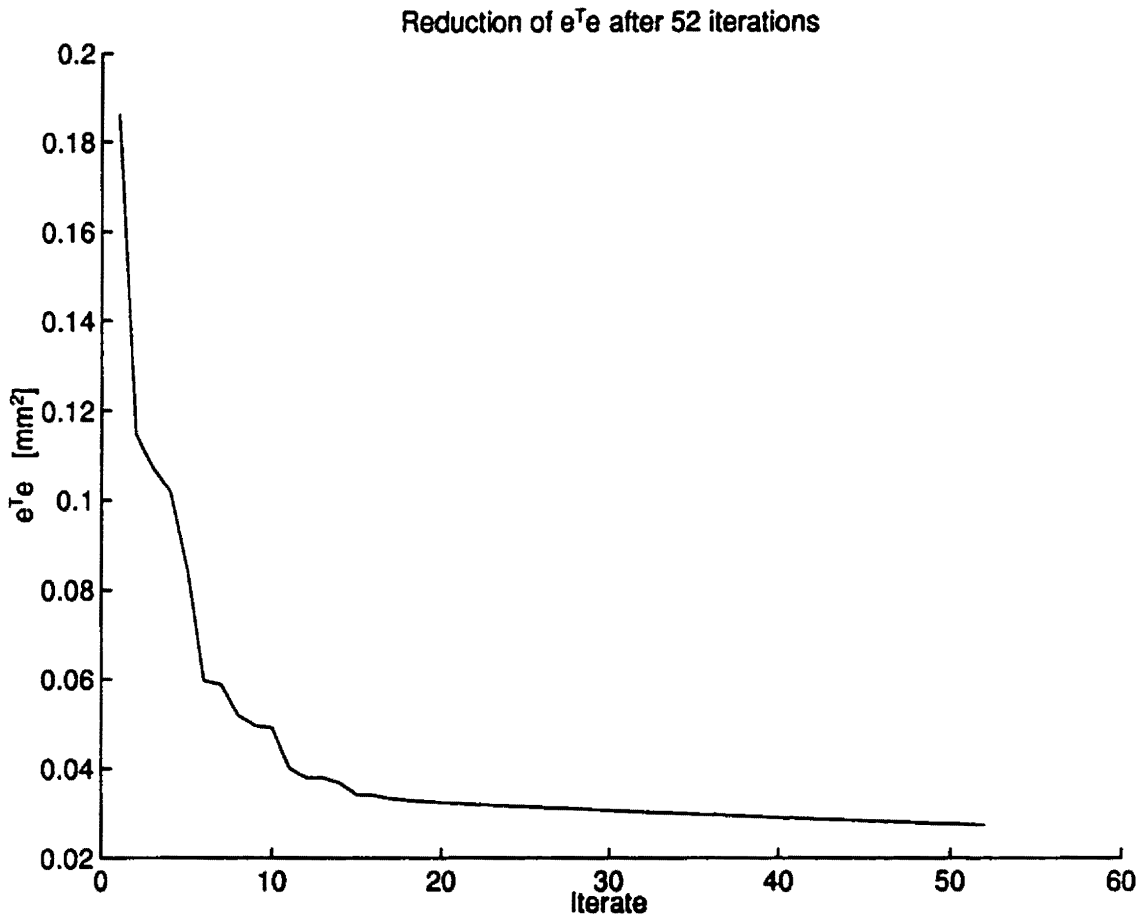


Figure 4.6: Reduction of least squares error of NACA airfoil vs. number of iteration obtained from optimizing the weights

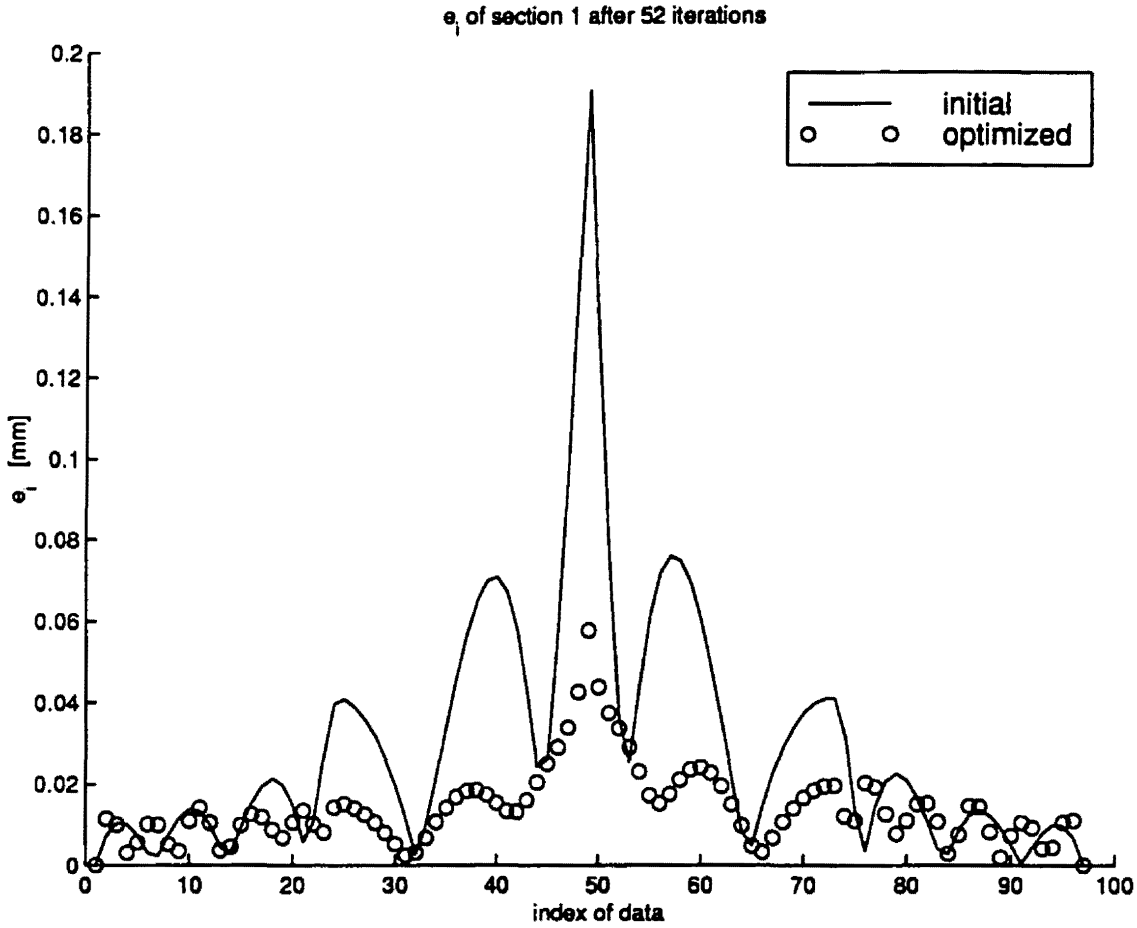


Figure 4.7: Error distributions of NACA airfoil before and after optimizing the weights

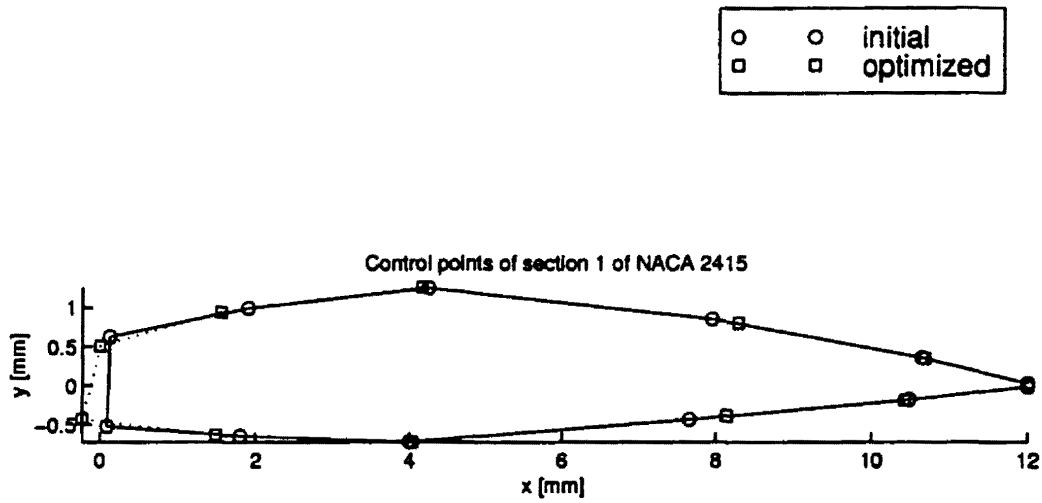


Figure 4.8: Control polygons of NACA airfoil before and after optimizing the weights

4.1.3 Optimization of Parameterization for Data

The total least squares error decreases from 0.186 [mm^2] to 0.068 [mm^2] in 3 iterations; Figure 4.9 shows the decrease of error with respect to number of iteration. Figure 4.10 shows the distributions of error along the airfoil before and after knot adjustment; solid line represents the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.113 millimeters respectively.

Figure 4.11 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of **R**.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. The backtrack algorithm failed to obtain adequate step length after three iteration.

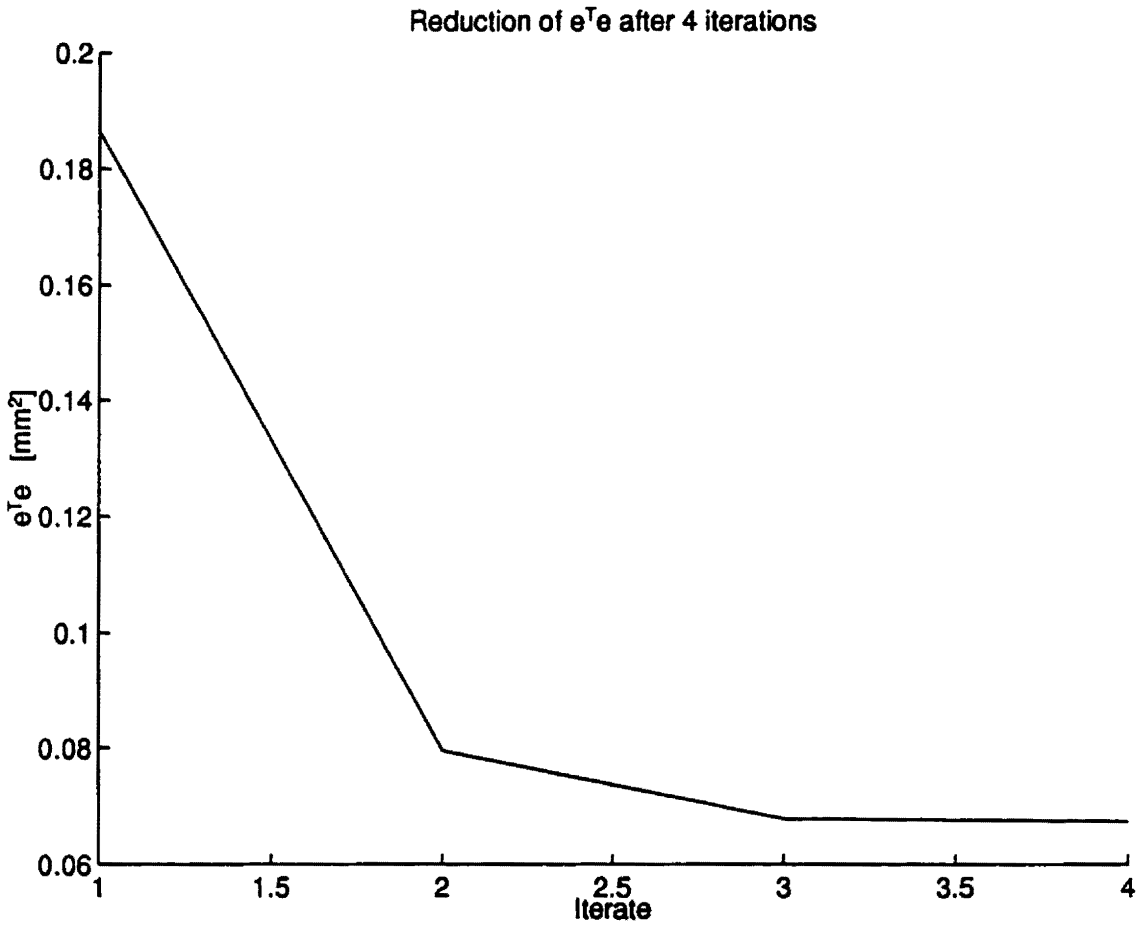


Figure 4.9: Reduction of least squares error of NACA airfoil vs. number of iteration obtained from optimizing the parameterization for data

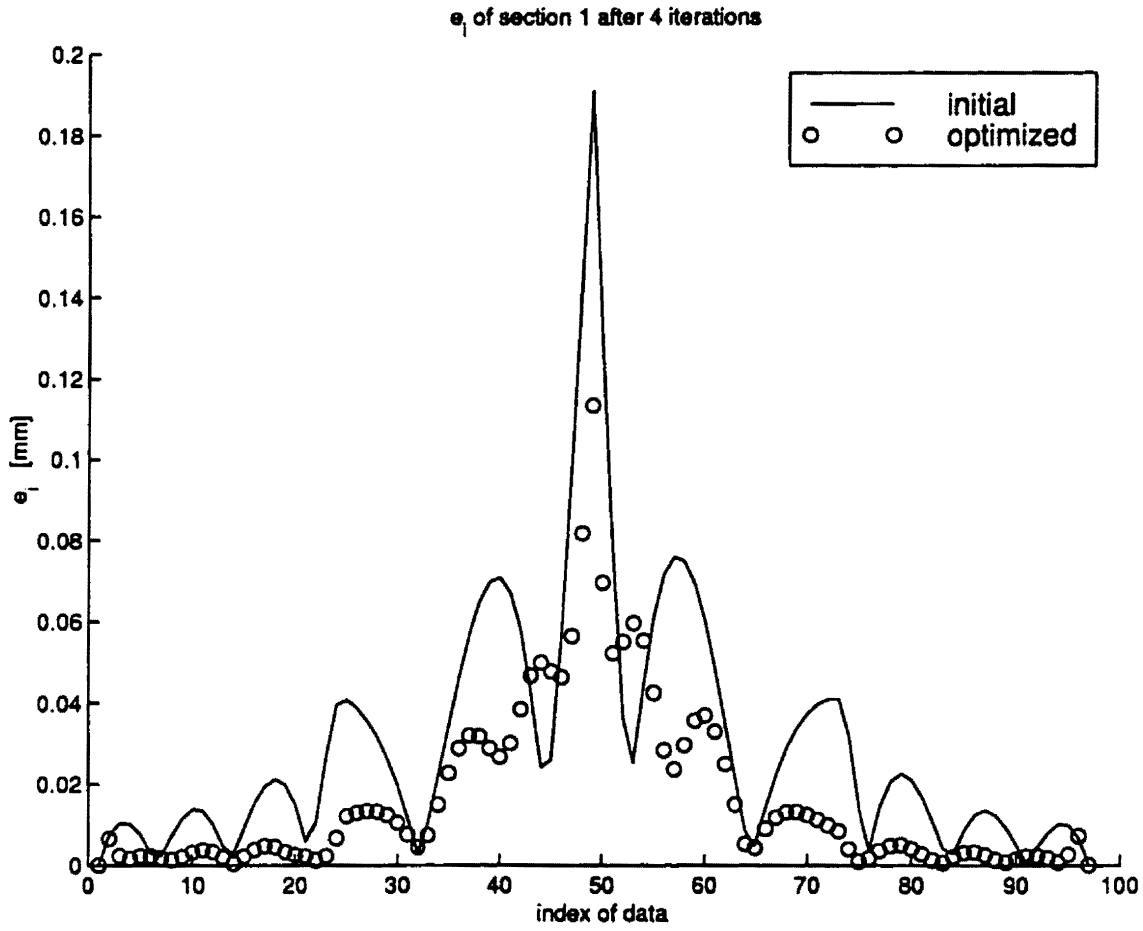


Figure 4.10: Error distributions of NACA airfoil before and after optimizing the parameterization for data

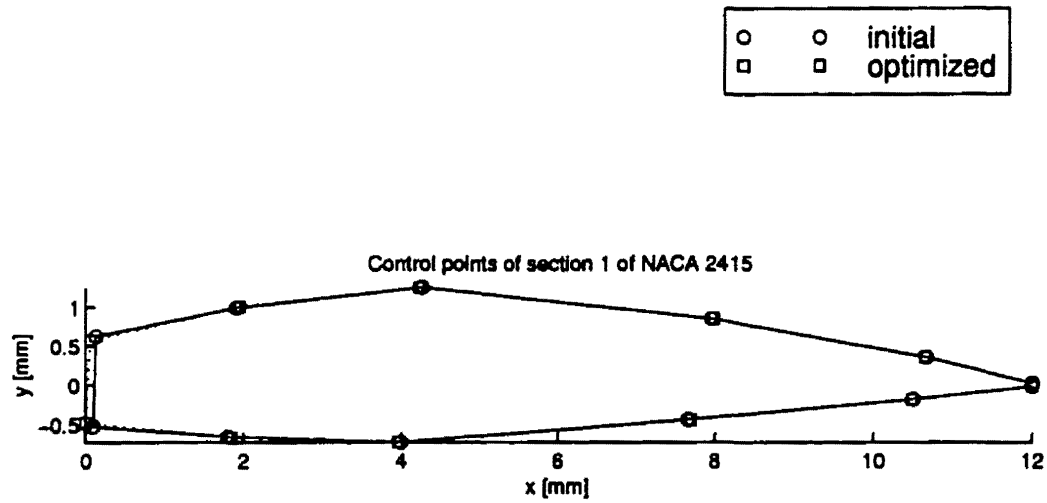


Figure 4.11: Control polygons of NACA airfoil before and after optimizing the parameterization for data

4.1.4 Optimization of Knots and Weights

The total least squares error decreases from 0.186 [mm^2] to 0.002 [mm^2] in 51 iterations; Figure 4.12 shows the decrease of error with respect to number of iteration. Figure 4.13 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.015 millimeters respectively. The weights practically did not change as indicated by extreme values of 1.0062 and 0.9924 from their initial values of unity.

Figure 4.14 shows the knot distributions, along the curves, before and after knot adjustment. The multiplicity of knots at the leading edge is 2, i.e. $u_8 = u_9$.

Figure 4.15 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

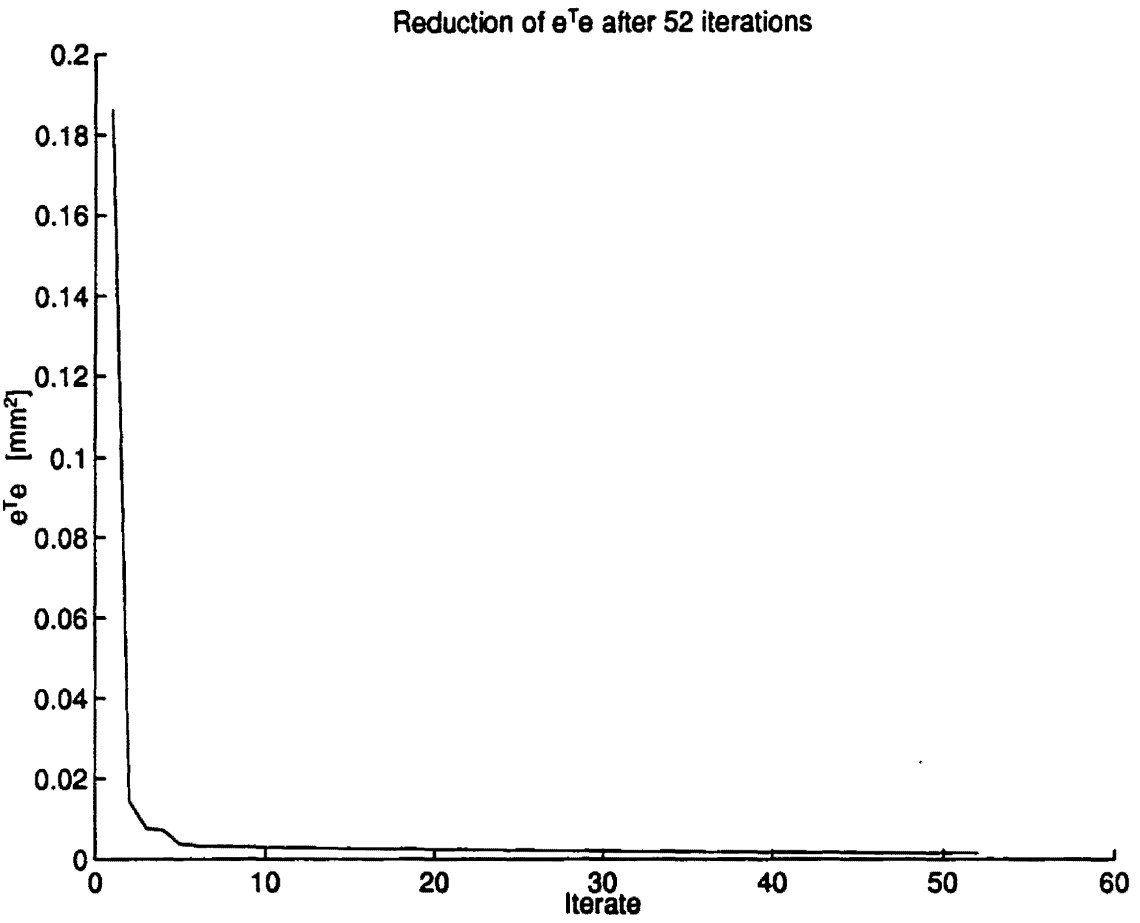


Figure 4.12: Reduction of least squares error of NACA airfoil vs. number of iteration obtained from optimizing the combination of knots and weights

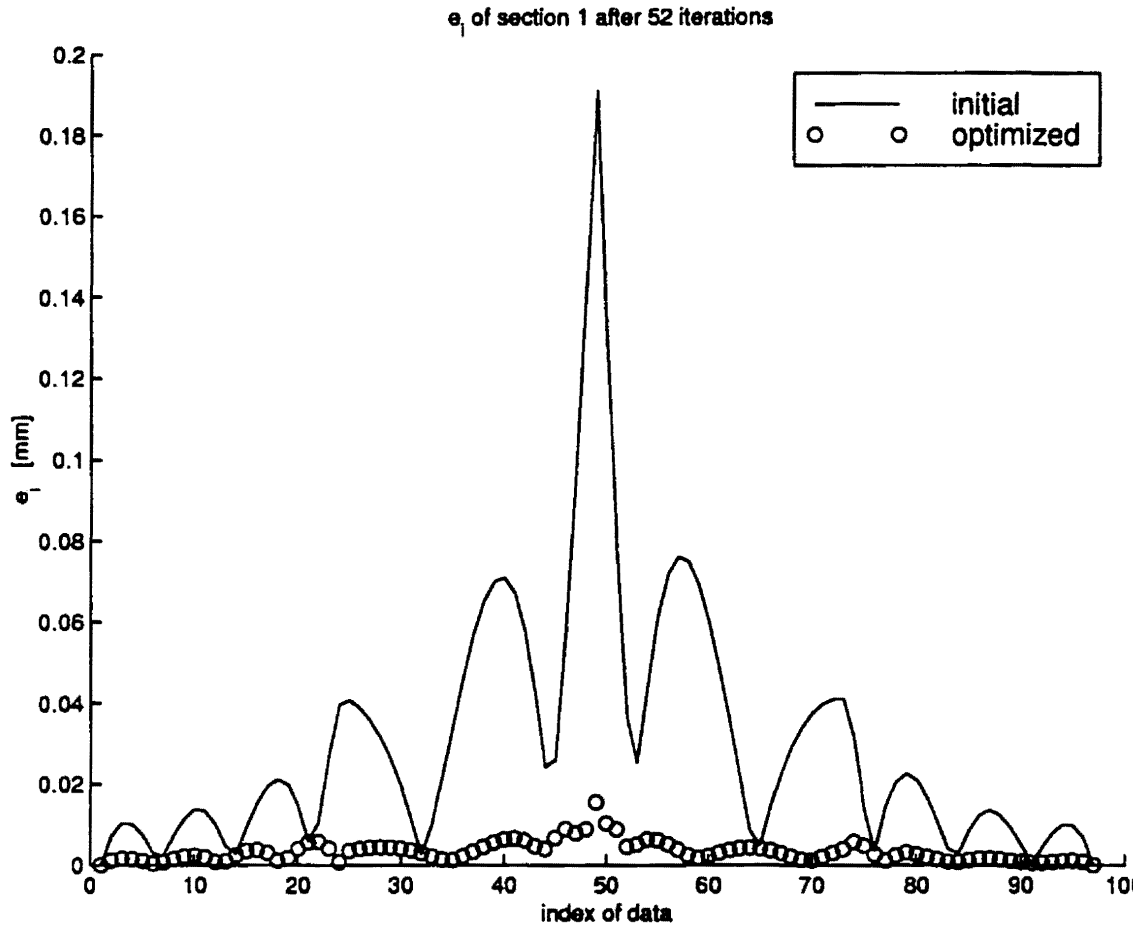


Figure 4.13: Error distributions of NACA airfoil before and after optimizing the combination of knots and weights

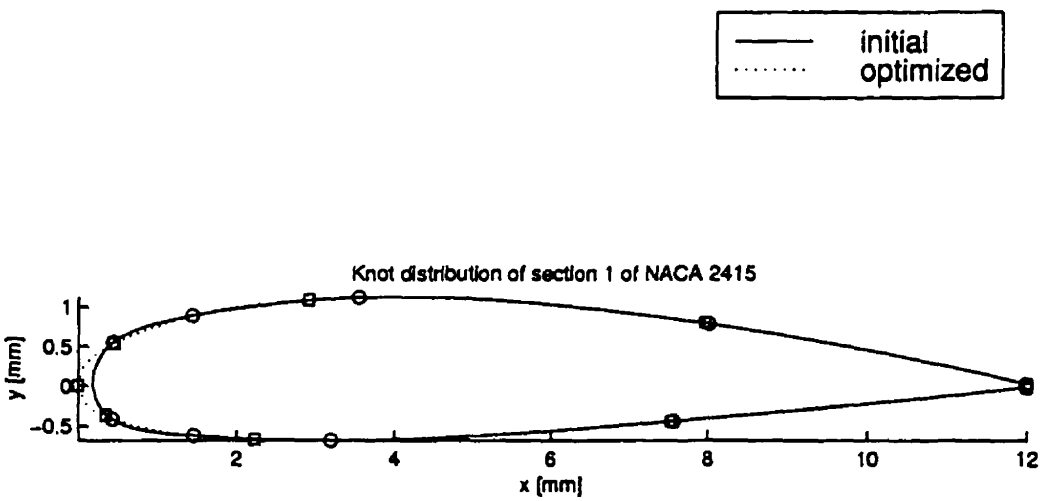


Figure 4.14: Distributions of Knots of NACA airfoil before and after optimizing the combination of knots and weights

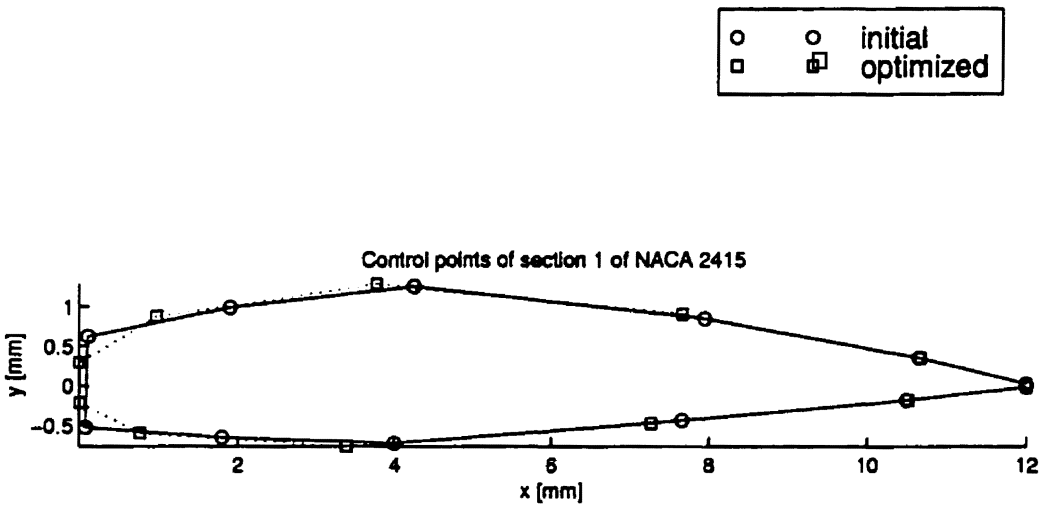


Figure 4.15: Control polygons of NACA airfoil before and after optimizing the combination of knots and weights

4.1.5 Optimization of Knots, Weights and Parameterization for Data

The total least squares error decreases from 0.186 [mm^2] to 0.061 [mm^2] in 2 iterations; Figure 4.16 shows the decrease of error with respect to number of iteration. Figure 4.17 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.107 millimeters respectively.

Figure 4.18 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. The backtrack algorithm failed to obtain adequate step length after three iteration.

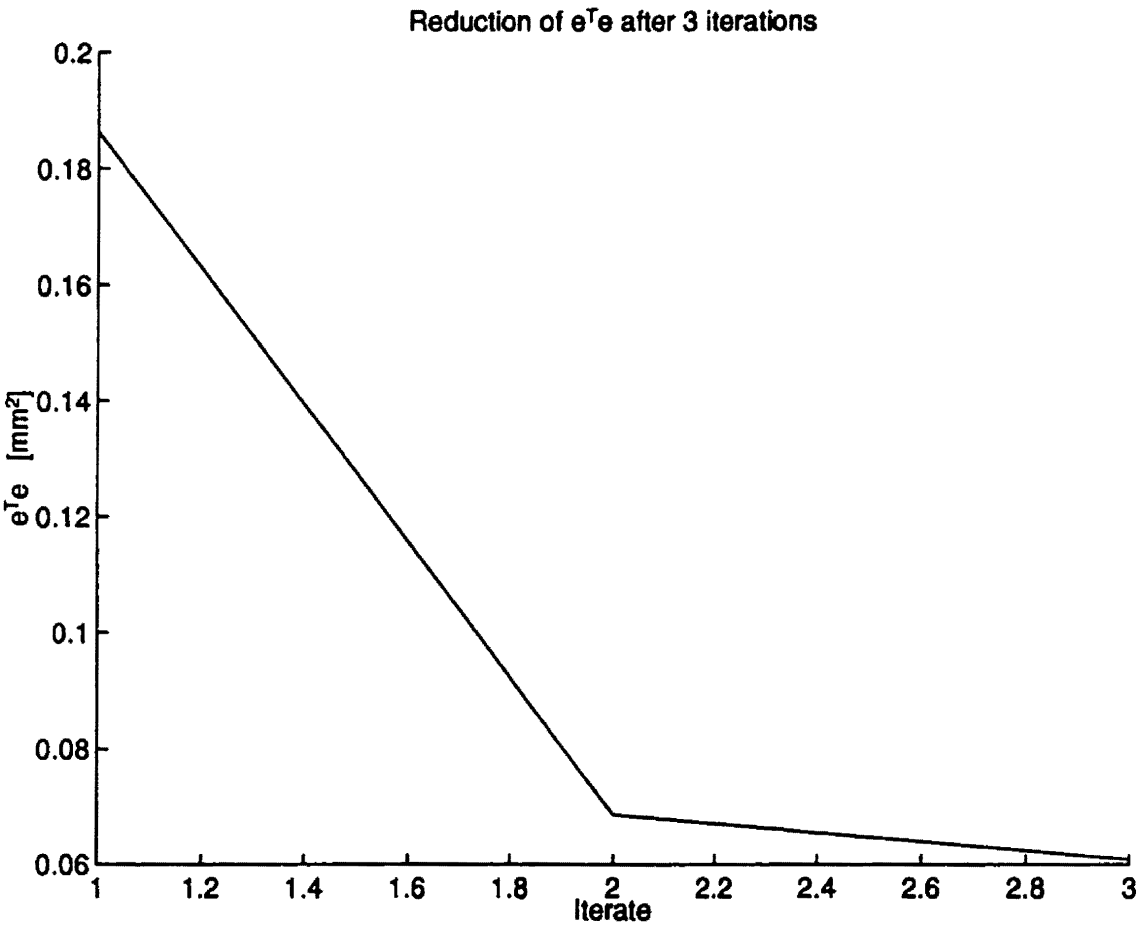


Figure 4.16: Reduction of least squares error of NACA airfoil vs. number of iteration obtained from optimizing the combination of knots, weights, and parameterization for data

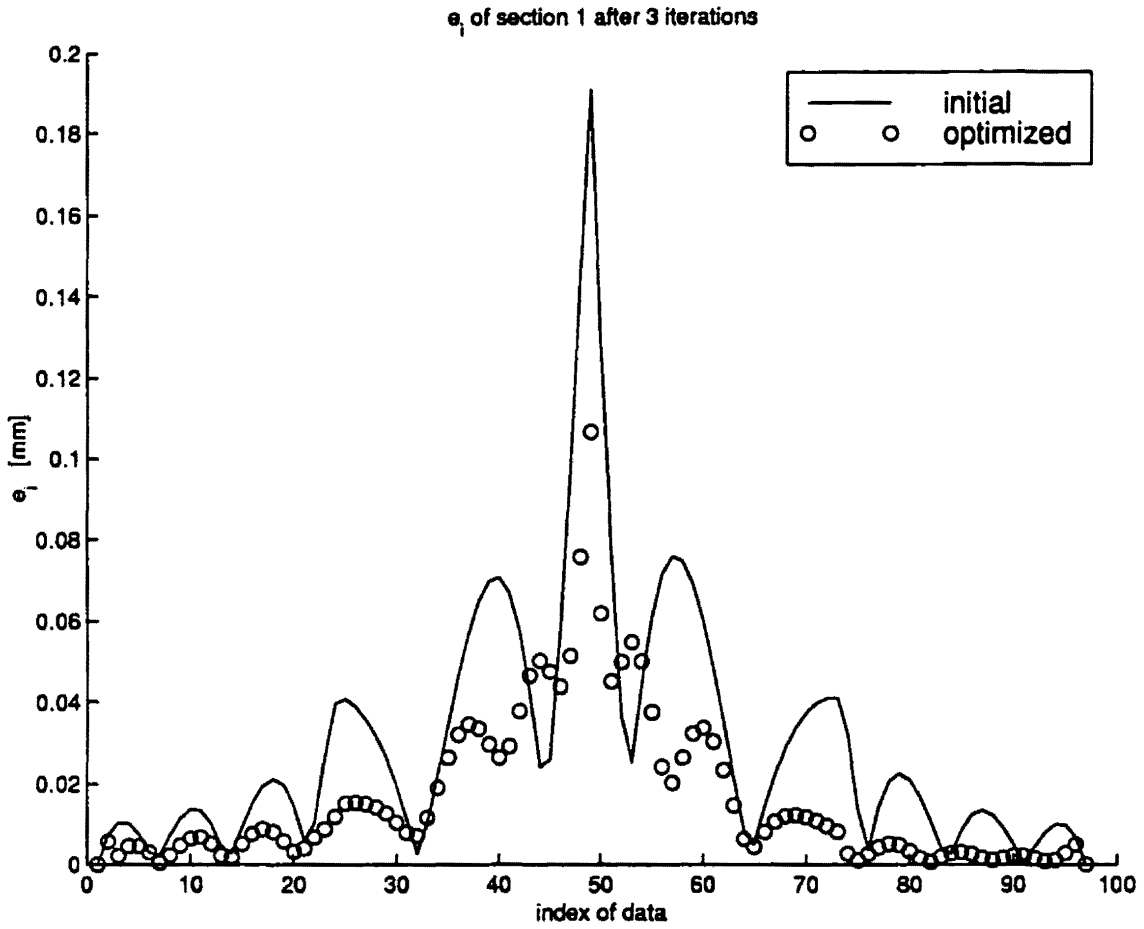


Figure 4.17: Error distributions of NACA airfoil before and after optimizing the combination of knots, weights, and parameterization for data

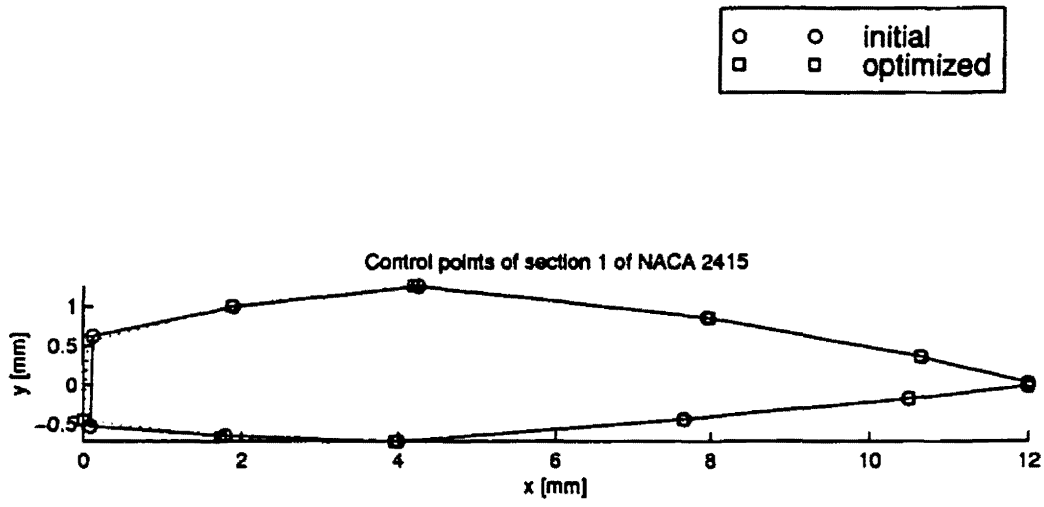


Figure 4.18: Control polygons of NACA airfoil before and after optimizing the combination of knots, weights, and parameterization for data

4.2 WTEA Airfoil

WTEA airfoil is a planar airfoil proprietary of deHavilland. The data curve consists of 97 points. The data is sequenced by starting from the middle of the trailing edge, going up to the junction with the upper part and proceeding along the upper part to the junction between the upper part and the leading edge, going around the leading edge up to the junction between the leading edge and the lower part, proceeding along the lower part up to the junction of the lower part and the trailing edge, and finally proceeding to the middle of the trailing edge where the sequence of data starts. Figure 4.19 shows the shape of WTEA airfoil. The degree of approximation curve is set to cubic and the number of basis is set to 12.

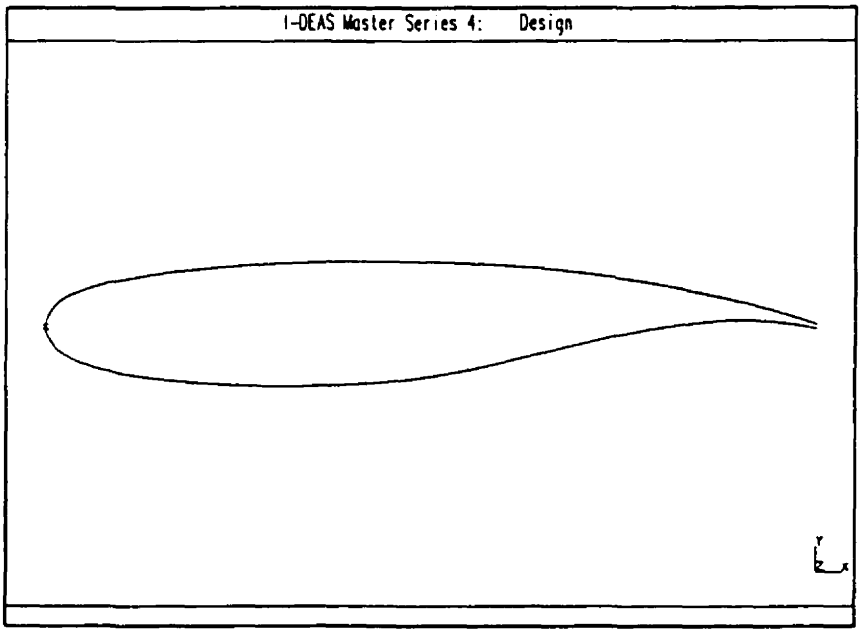


Figure 4.19: WTEA airfoil

4.2.1 Optimization of Knots

The total least squares error decreases from 0.737 [mm^2] to 0.023 [mm^2] in 51 iterations; Figure 4.20 shows the decrease of error with respect to number of iteration. Figure 4.21 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.211 millimeters, and 0.030 millimeters respectively.

Figure 4.22 shows the knot distributions, along the curves, before and after knot adjustment. Active knot constraints were $u_7 = u_8 = u_9$ and $u_{11} = u_{12}$.

Figure 4.23 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of **R**.

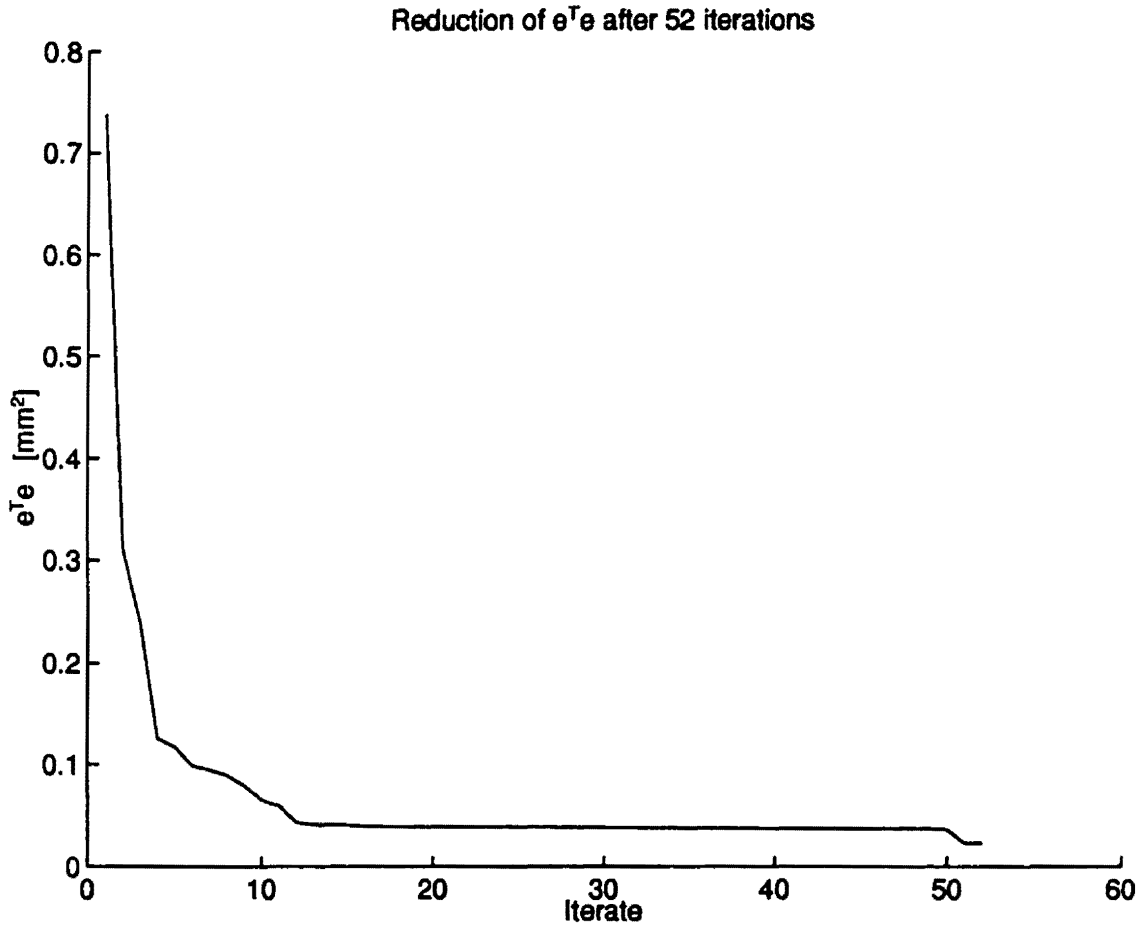


Figure 4.20: Reduction of least squares error of WTEA airfoil vs. number of iteration obtained from optimizing the knots

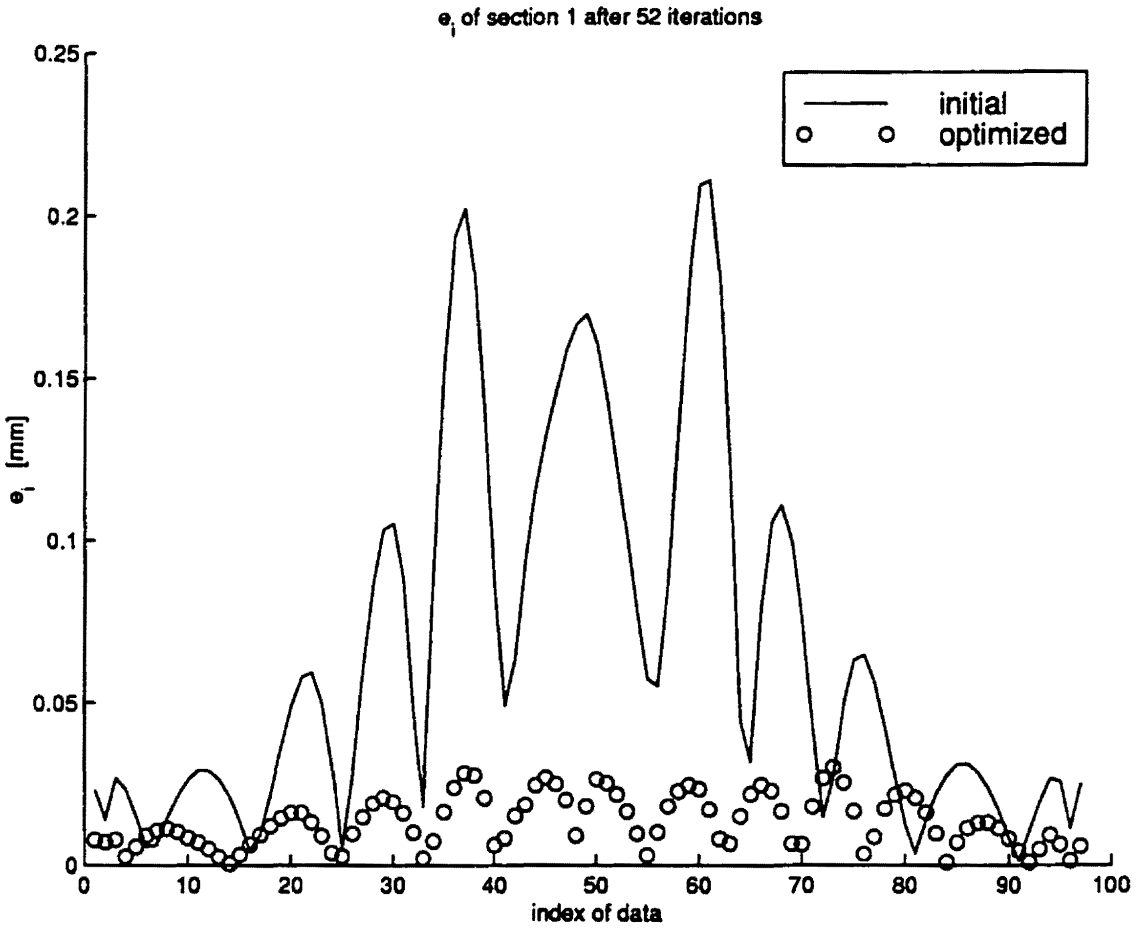


Figure 4.21: Error distributions of WTEA airfoil before and after optimizing the knots

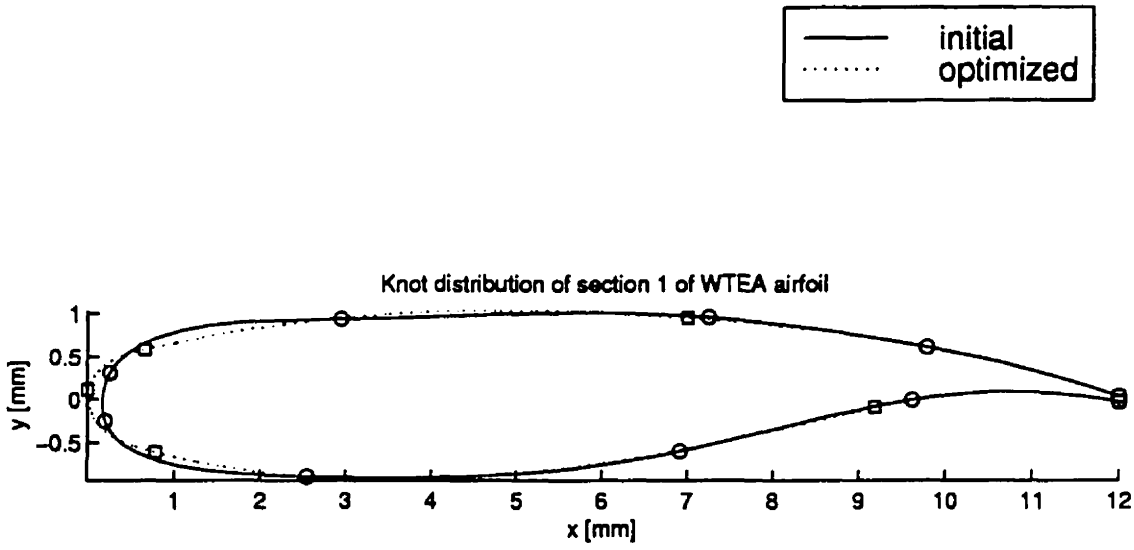


Figure 4.22: Distributions of Knots of WTEA airfoil before and after optimizing the knots

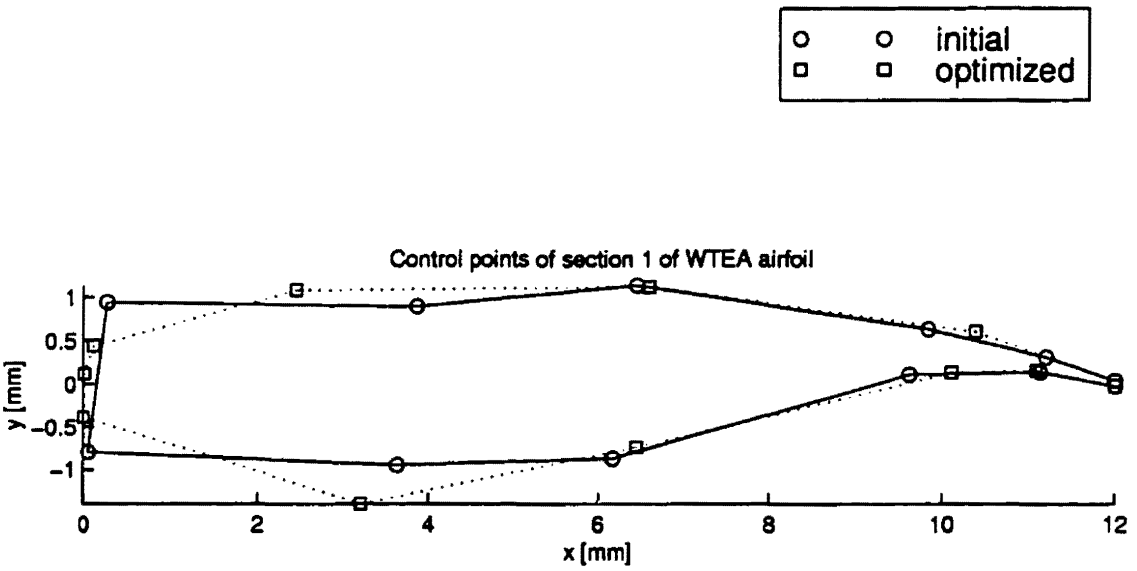


Figure 4.23: Control polygons of WTEA airfoil before and after optimizing the knots

4.2.2 Optimization of Weights

The total least squares error decreases from 0.737 [mm^2] to 0.059 [mm^2] in 51 iterations; Figure 4.24 shows the decrease of error with respect to number of iteration. Figure 4.25 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.211 millimeters, and 0.054 millimeters respectively. All weight constraints were inactive.

Figure 4.26 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

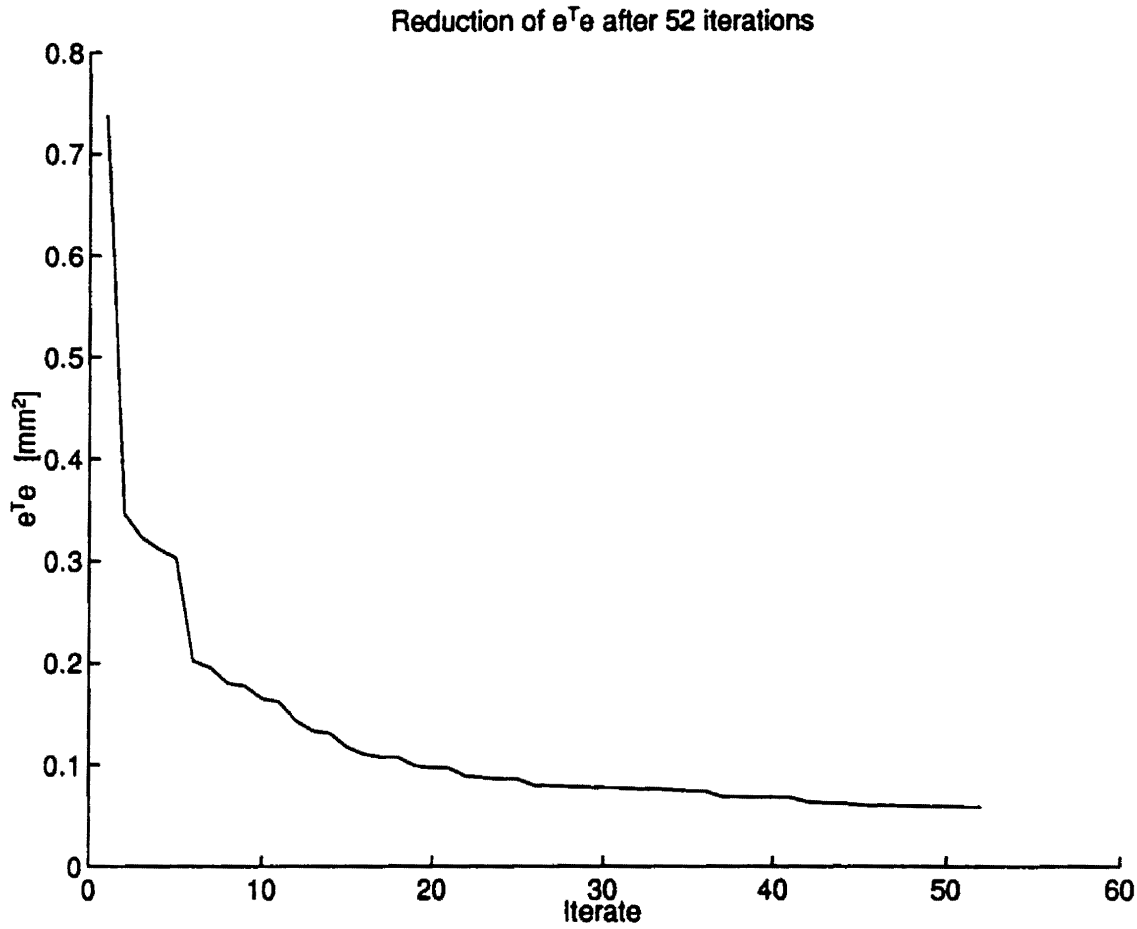


Figure 4.24: Reduction of least squares error of WTEA airfoil vs. number of iteration obtained from optimizing the weights

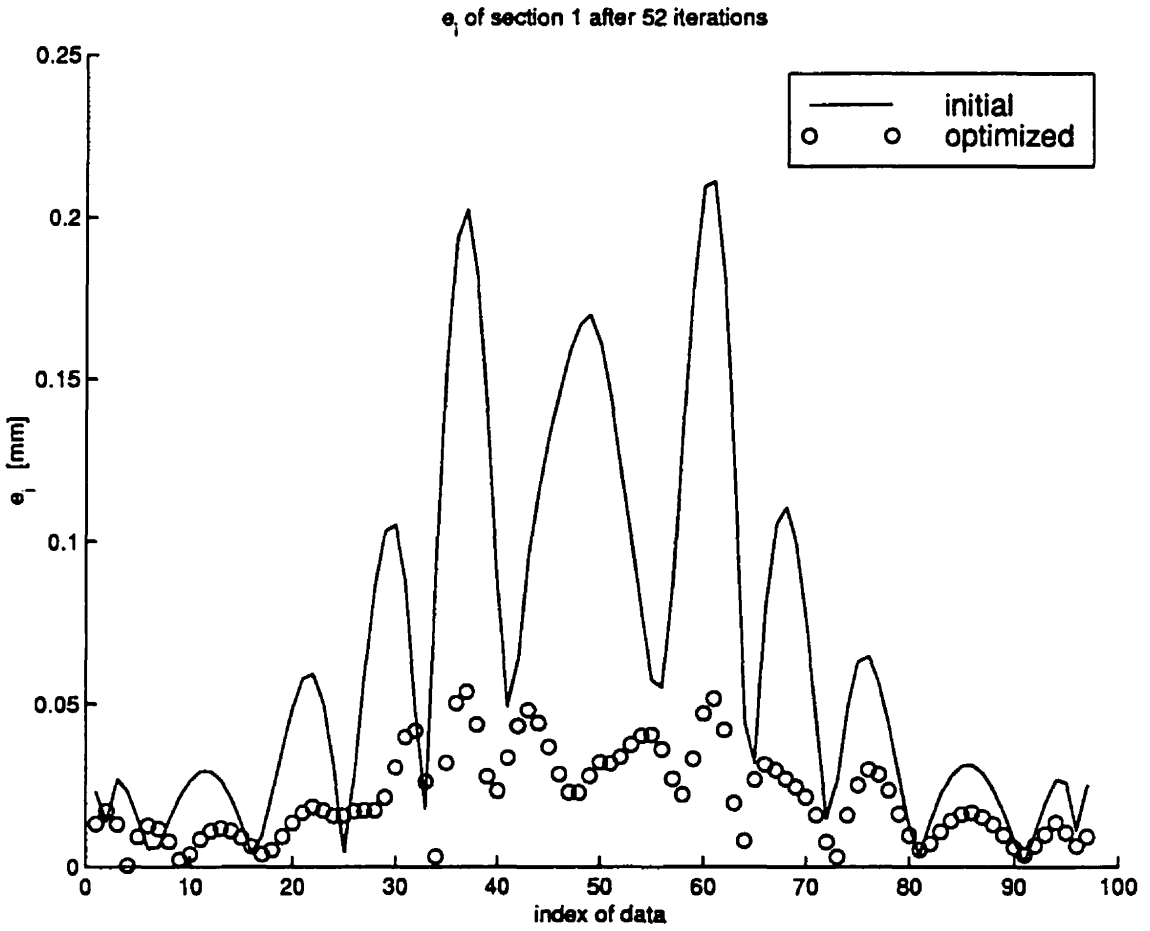


Figure 4.25: Error distributions of WTEA airfoil before and after optimizing the weights

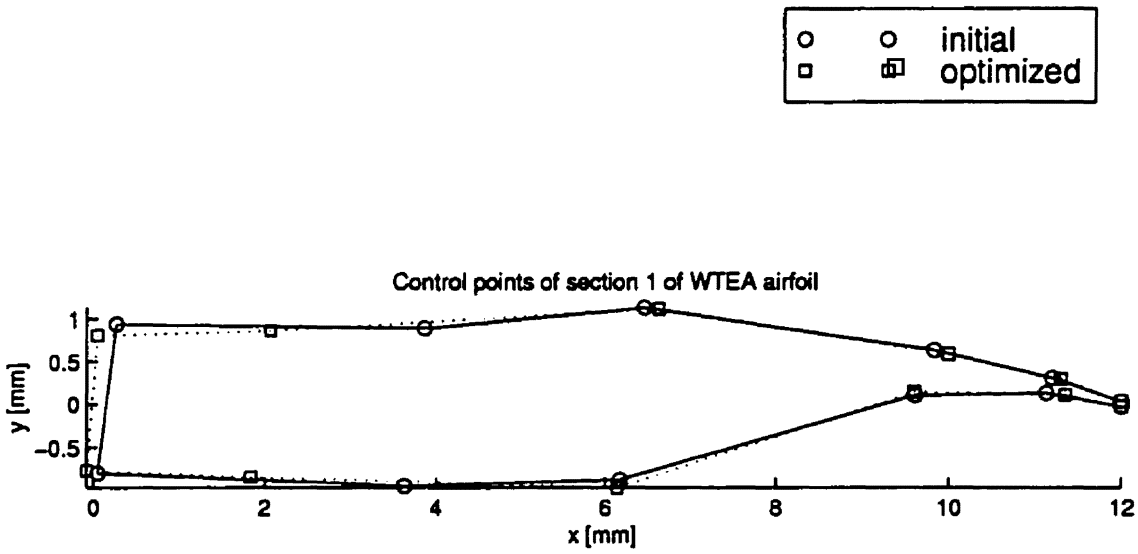


Figure 4.26: Control polygons of WTEA airfoil before and after optimizing the weights

4.2.3 Optimization of Parameterization

The total least squares error decreases from 0.737 [mm^2] to 0.193 [mm^2] in 2 iterations; Figure 4.27 shows the decrease of error with respect to number of iteration. Figure 4.28 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.191 millimeters, and 0.113 millimeters respectively.

Figure 4.29 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. The backtrack algorithm failed to obtain adequate step length after three iteration.

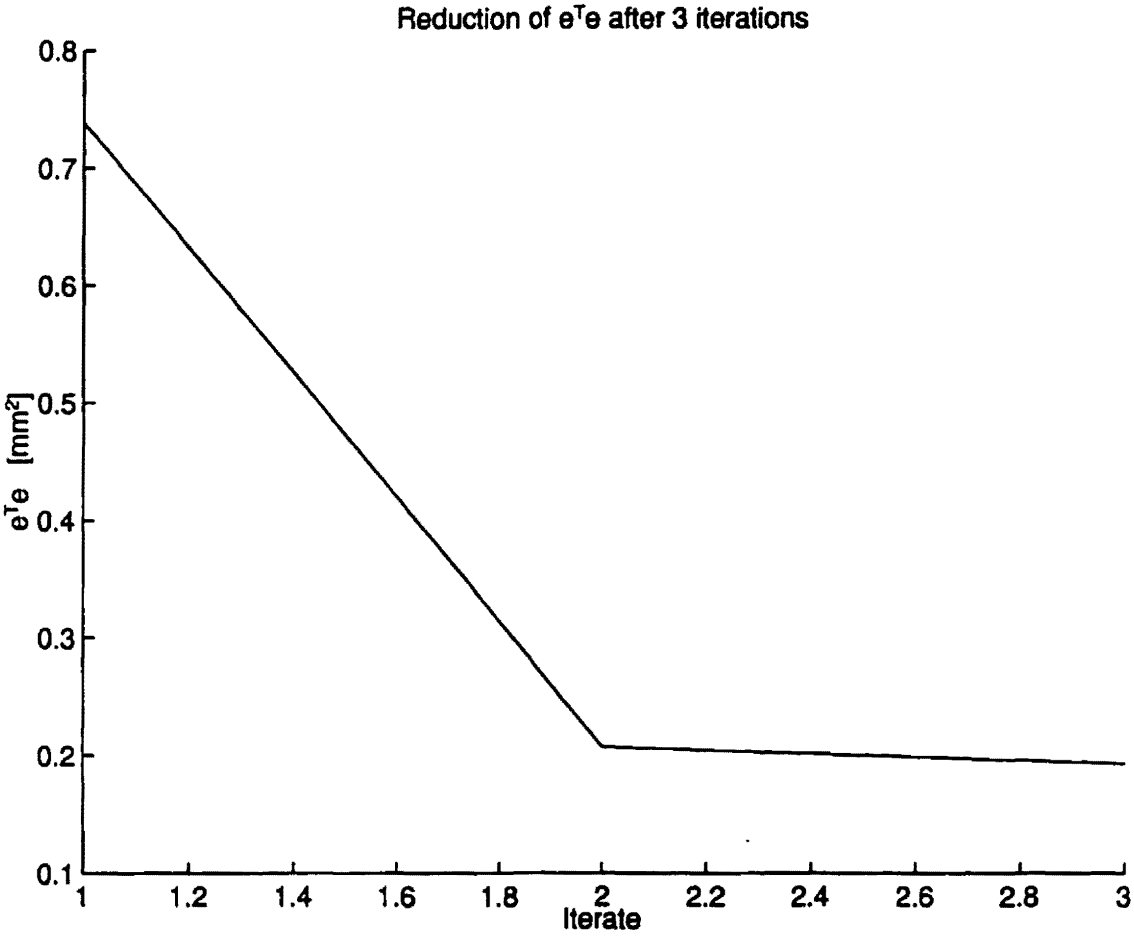


Figure 4.27: Reduction of least squares error of WTEA airfoil vs. number of iteration obtained from optimizing the parameterization for data

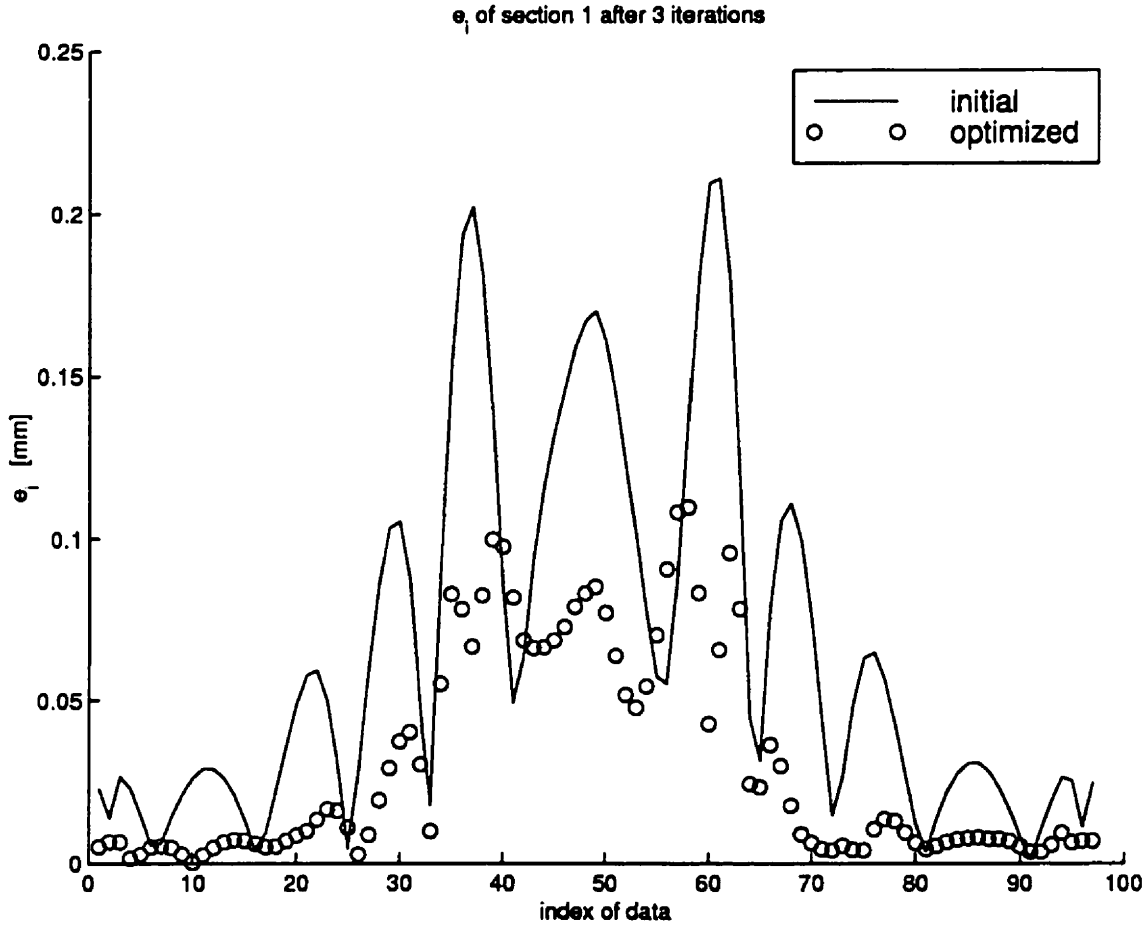


Figure 4.28: Error distributions of WTEA airfoil before and after optimizing the parameterization for data

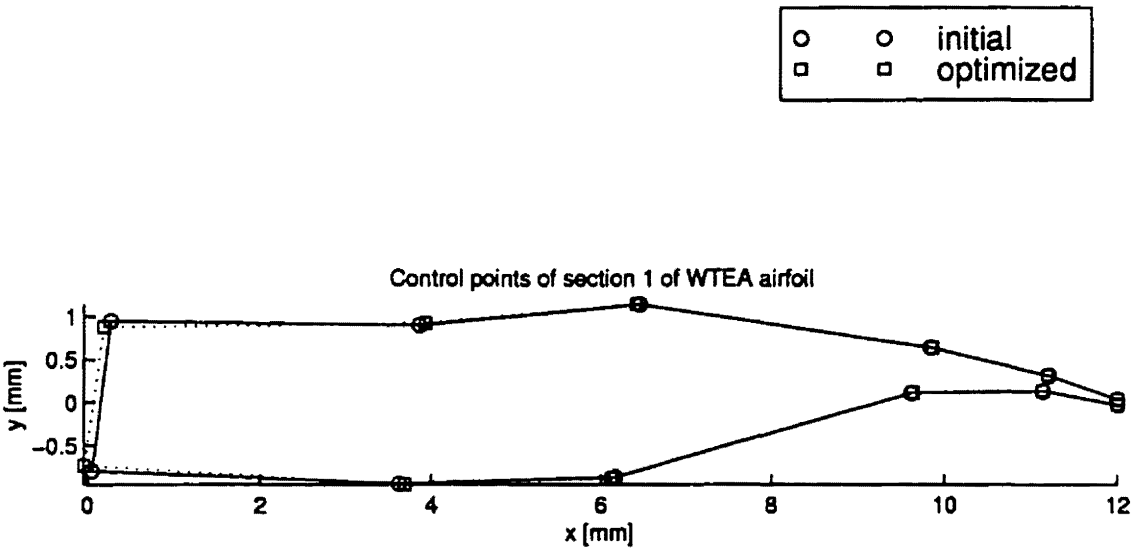


Figure 4.29: Control polygons of WTEA airfoil before and after optimizing the parameterization for data

4.2.4 Optimization of Knots and Weights

The total least squares error decreases from 0.737 [mm^2] to 0.012 [mm^2] in 51 iterations; Figure 4.30 shows the decrease of error with respect to number of iteration. Figure 4.31 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.211 millimeters, and 0.024 millimeters respectively. The weights practically did not change as indicated by extreme values of 1.0362 and 0.9278 from their initial values of unity.

Figure 4.32 shows the knot distributions, along the curves, before and after knot adjustment. Only one knot constraint was active, $u_7 = u_8$.

Figure 4.33 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

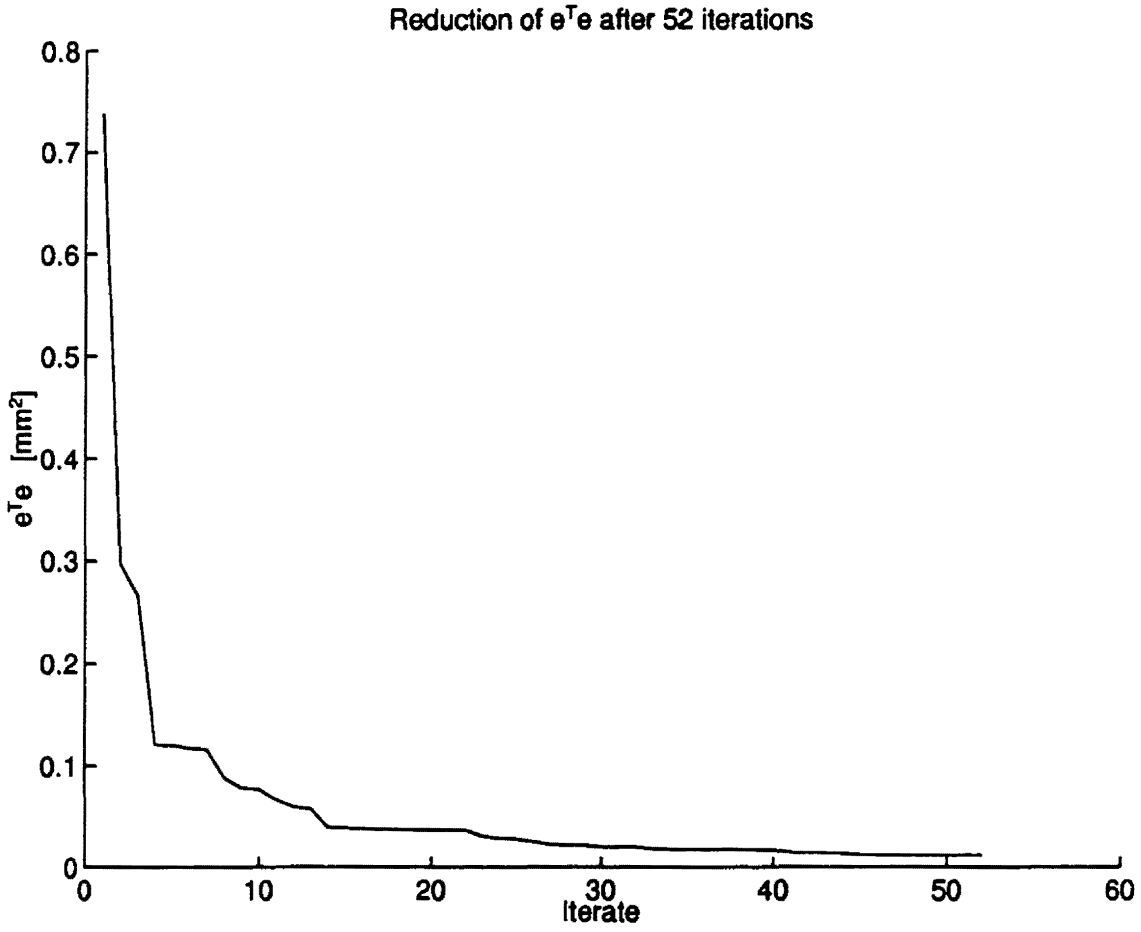


Figure 4.30: Reduction of least squares error of WTEA airfoil vs. number of iteration obtained from optimizing the combination of knots and weights

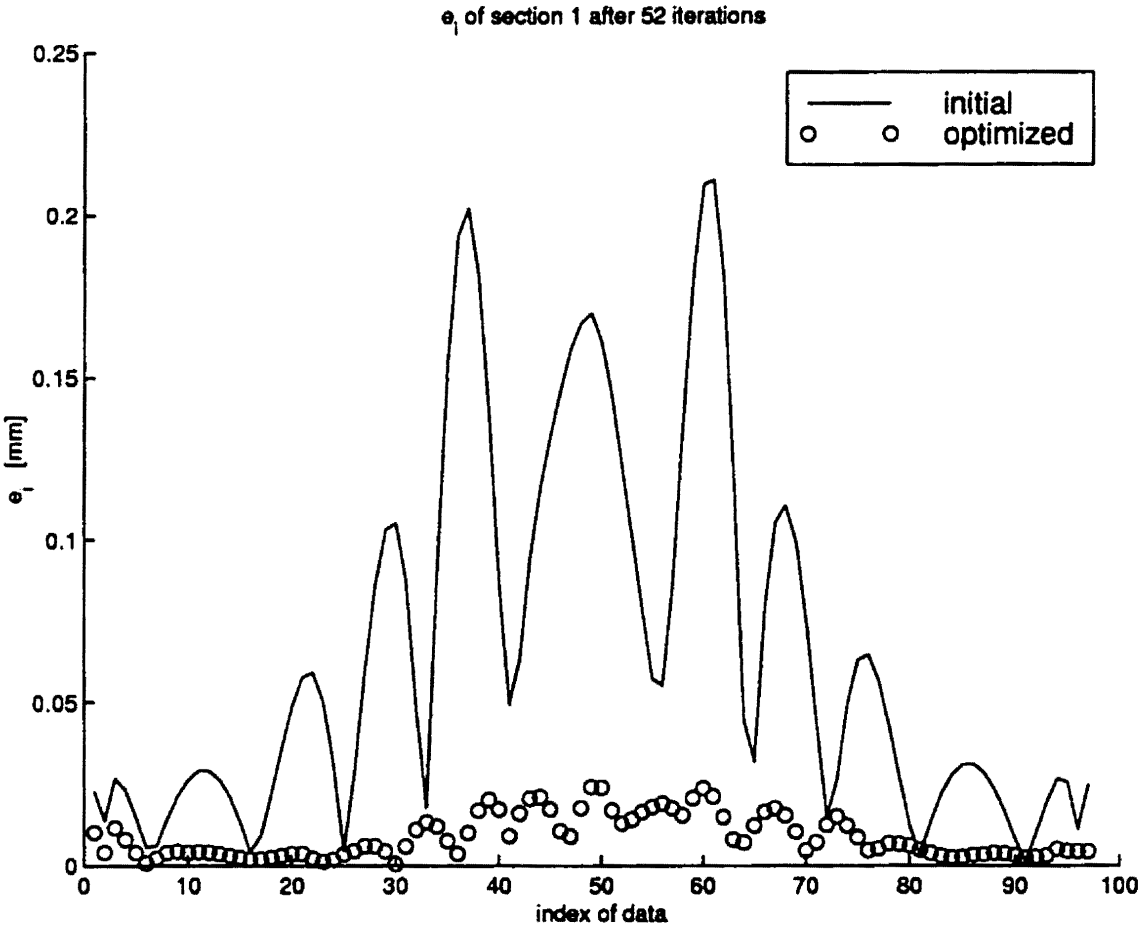


Figure 4.31: Error distributions of WTEA airfoil before and after optimizing the combination of knots and weights

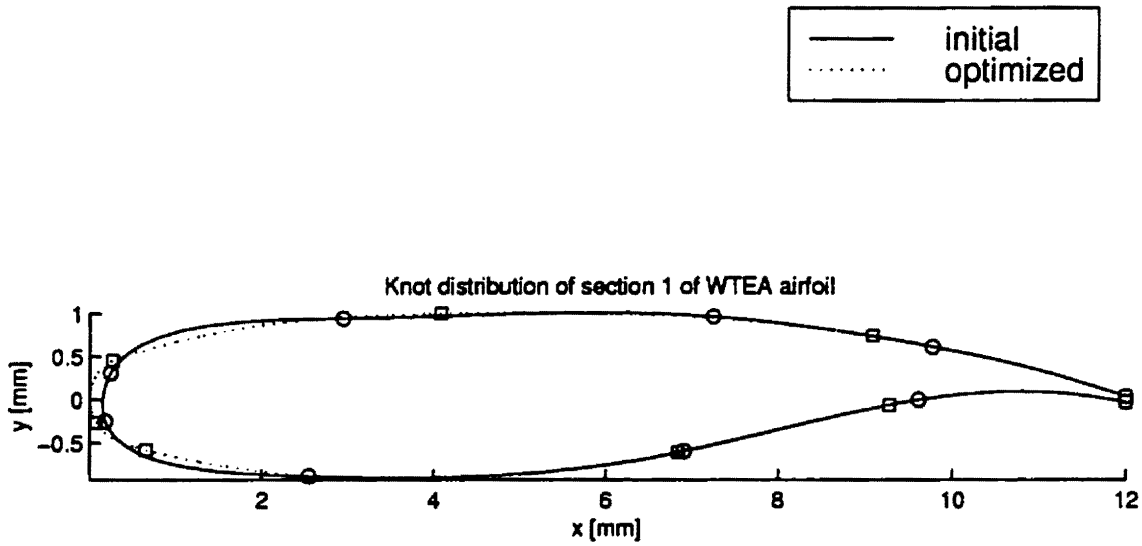


Figure 4.32: Distributions of Knots of WTEA airfoil before and after optimizing the combination of knots and weights

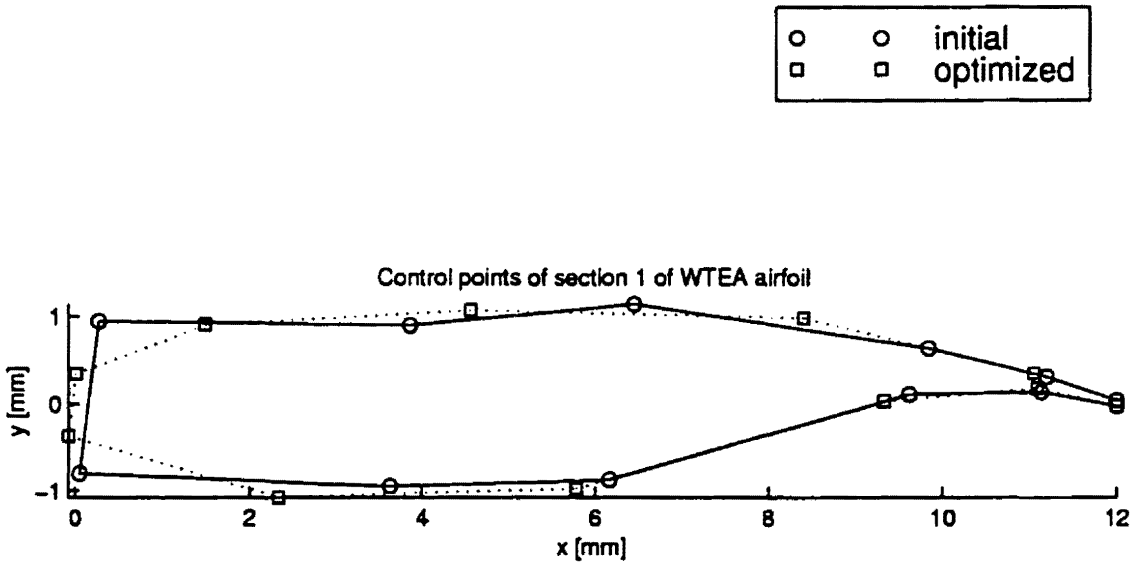


Figure 4.33: Control polygons of WTEA airfoil before and after optimizing the combination of knots and weights

4.2.5 Optimization of Knots, Weights and Parameterization

The total least squares error decreases from 0.737 [mm^2] to 0.105 [mm^2] in 2 iterations; Figure 4.34 shows the decrease of error with respect to number of iteration. Figure 4.35 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 0.211 millimeters, and 0.187 millimeters respectively. Figure 4.36 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. The backtrack algorithm failed to obtain adequate step length after three iteration.

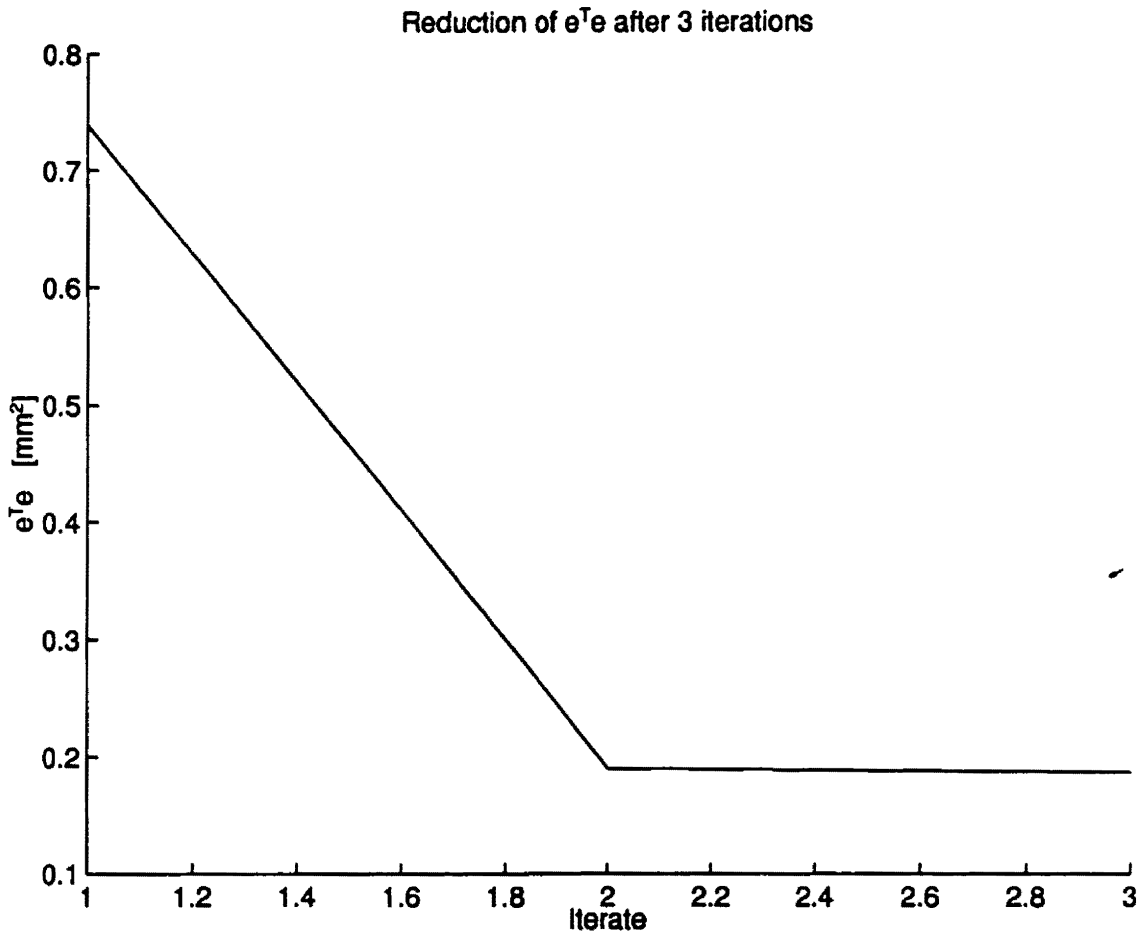


Figure 4.34: Reduction of least squares error of WTEA airfoil vs. number of iteration obtained from optimizing the combination of knots, weights, and parameterization for data

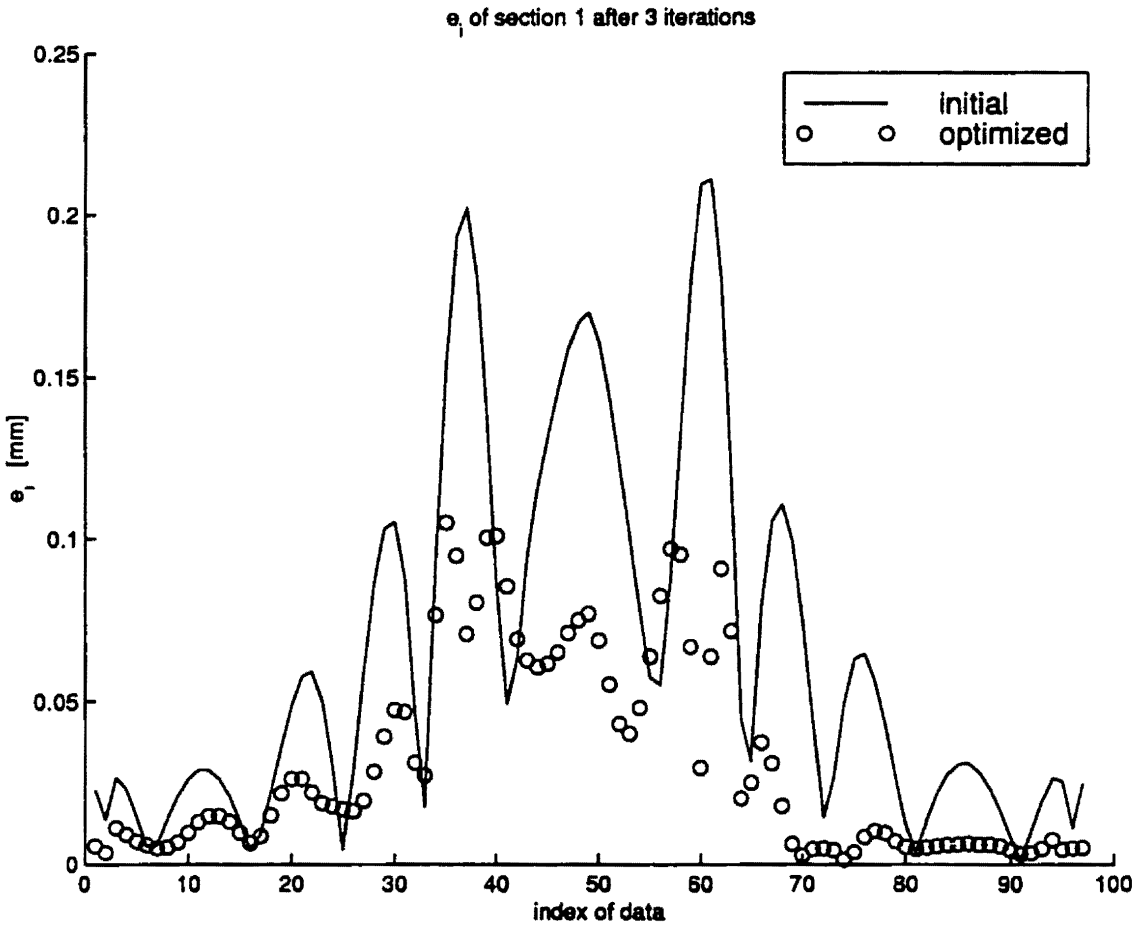


Figure 4.35: Error distributions of WTEA airfoil before and after optimizing the combination of knots, weights, and parameterization for data

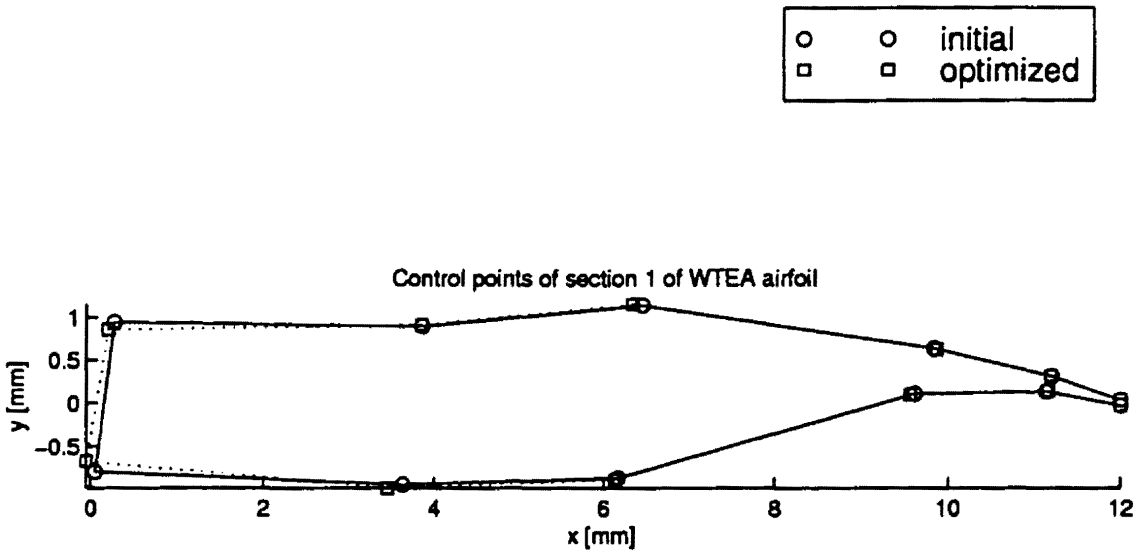


Figure 4.36: Control polygons of WTEA airfoil before and after optimizing the combination of knots, weights, and parameterization for data

4.3 Axial Compressor

This three-dimensional blade section is a proprietary of the Concepts ETI of Vermont. The data curve consists of 77 points. The data is sequenced by starting from the middle of the trailing edge, going up to the junction with the upper part and proceeding along the upper part to the junction between the upper part and the leading edge, going around the leading edge up to the junction between the leading edge and the lower part, proceeding along the lower part up to the junction of the lower part and the trailing edge, and finally proceeding to the middle of the trailing edge where the sequence of data starts. Figure 4.37 shows the shape of the blade section. The degree of approximation curve is set to cubic and the number of basis is set to 12.

4.3.1 Optimization of Knots

The total least squares error decreases from 9.09 [mm^2] to 0.327 [mm^2] in 51 iterations; Figure 4.38 shows the decrease of error with respect to number of iteration. Figure 4.39 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 1.21 millimeters, and 0.206 millimeters respectively.

Figure 4.40 shows the knot distributions, along the curves, before and after knot adjustment. Only one knot constraint was active, $u_8 = u_9$.

Figure 4.41 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points.

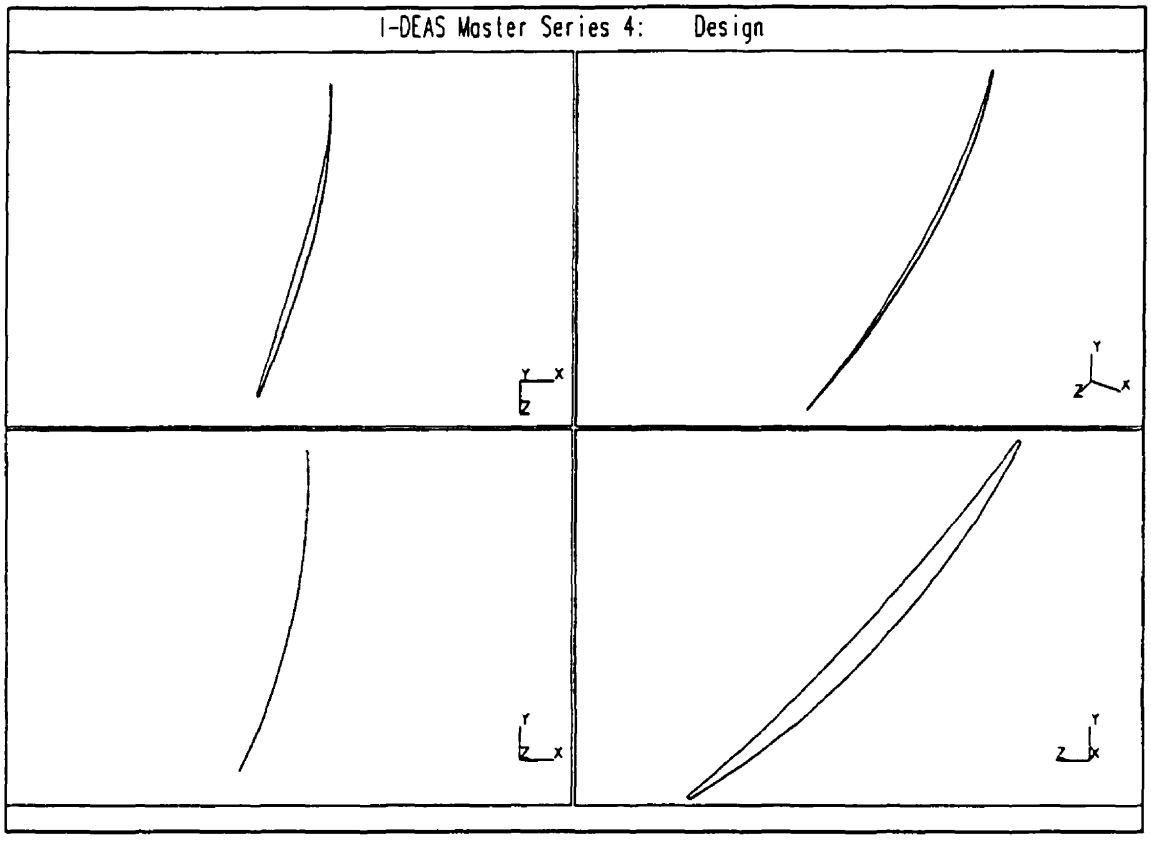


Figure 4.37: Blade Section of Concepts ETI

This is a result of constraining the condition number of **R**.

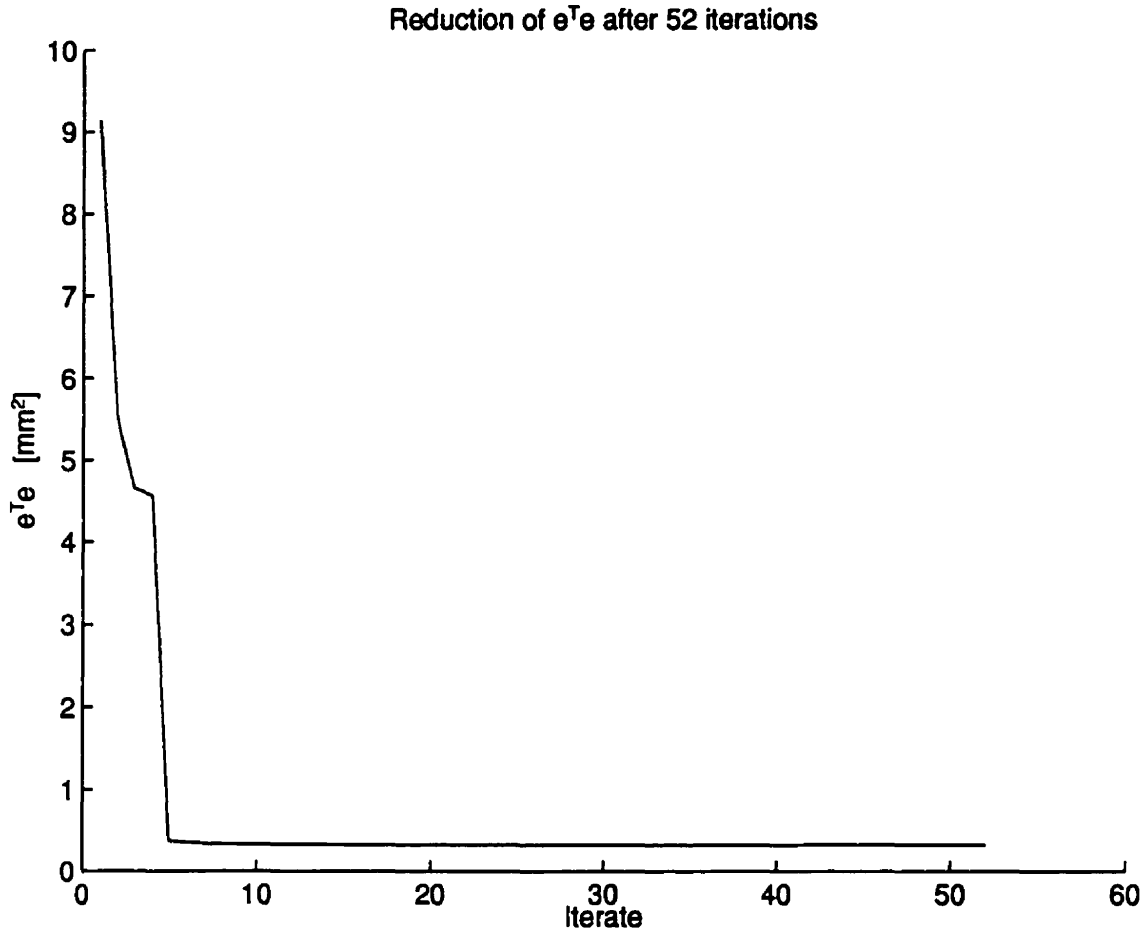


Figure 4.38: Reduction of least squares error of compressor blade's airfoil vs. number of iteration obtained from optimizing the knots

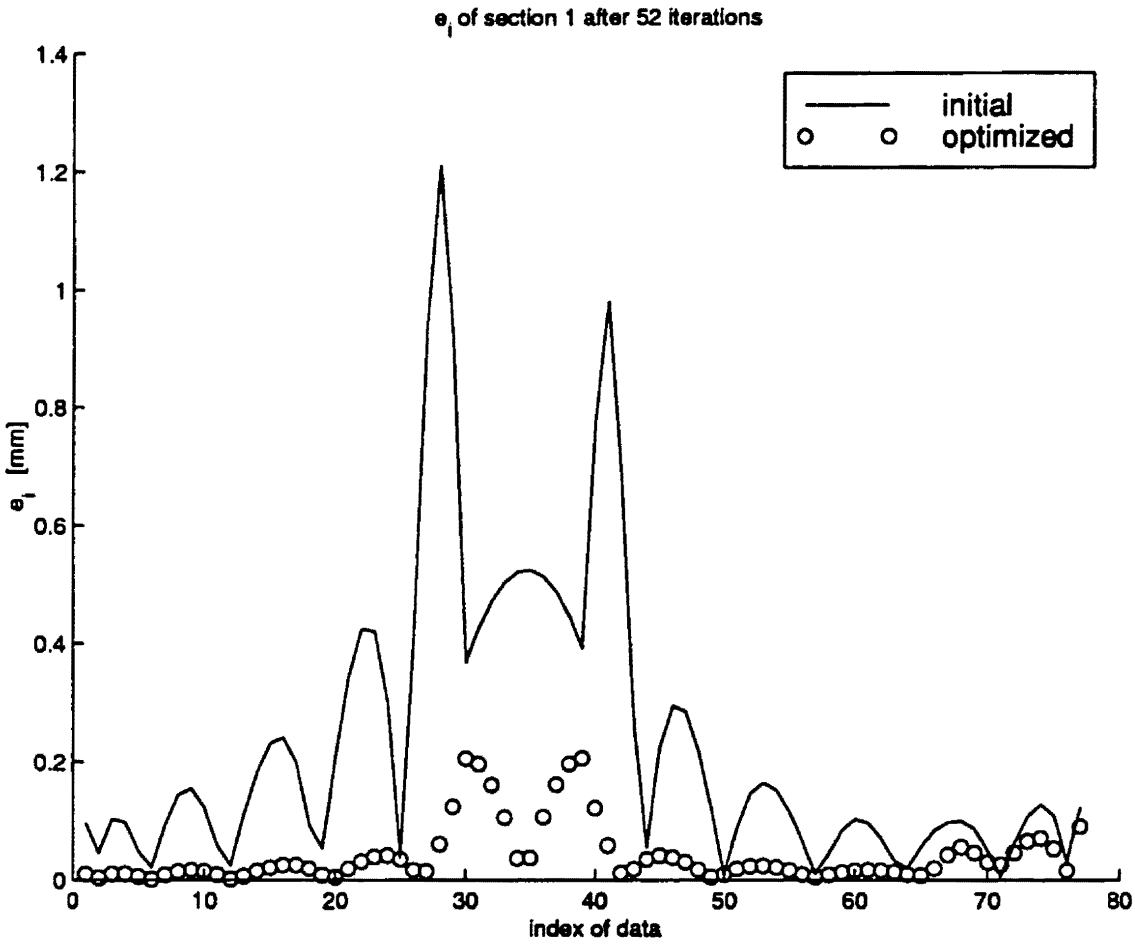


Figure 4.39: Error distributions of compressor blade's airfoil before and after optimizing the knots

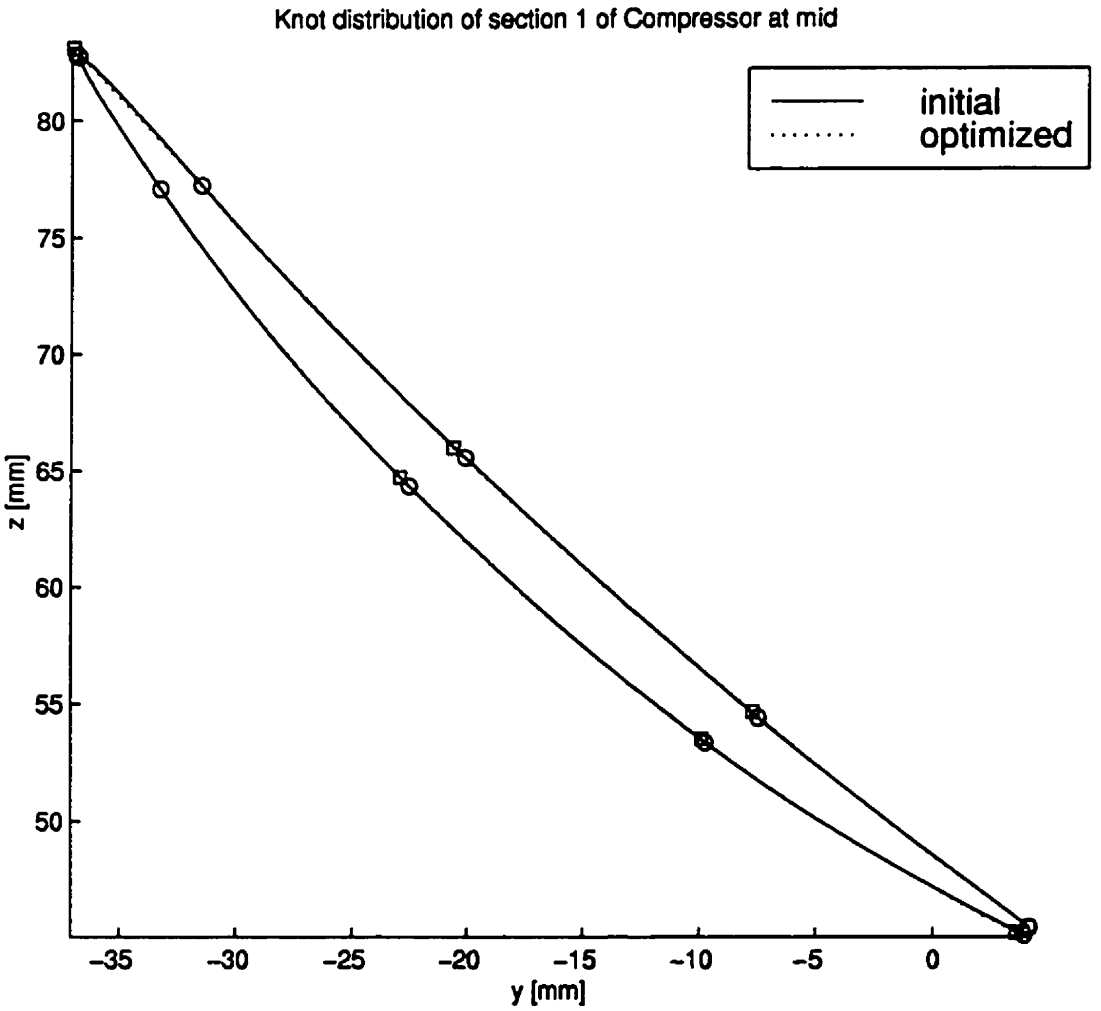


Figure 4.40: Distributions of Knots of compressor blade's airfoil before and after optimizing the knots

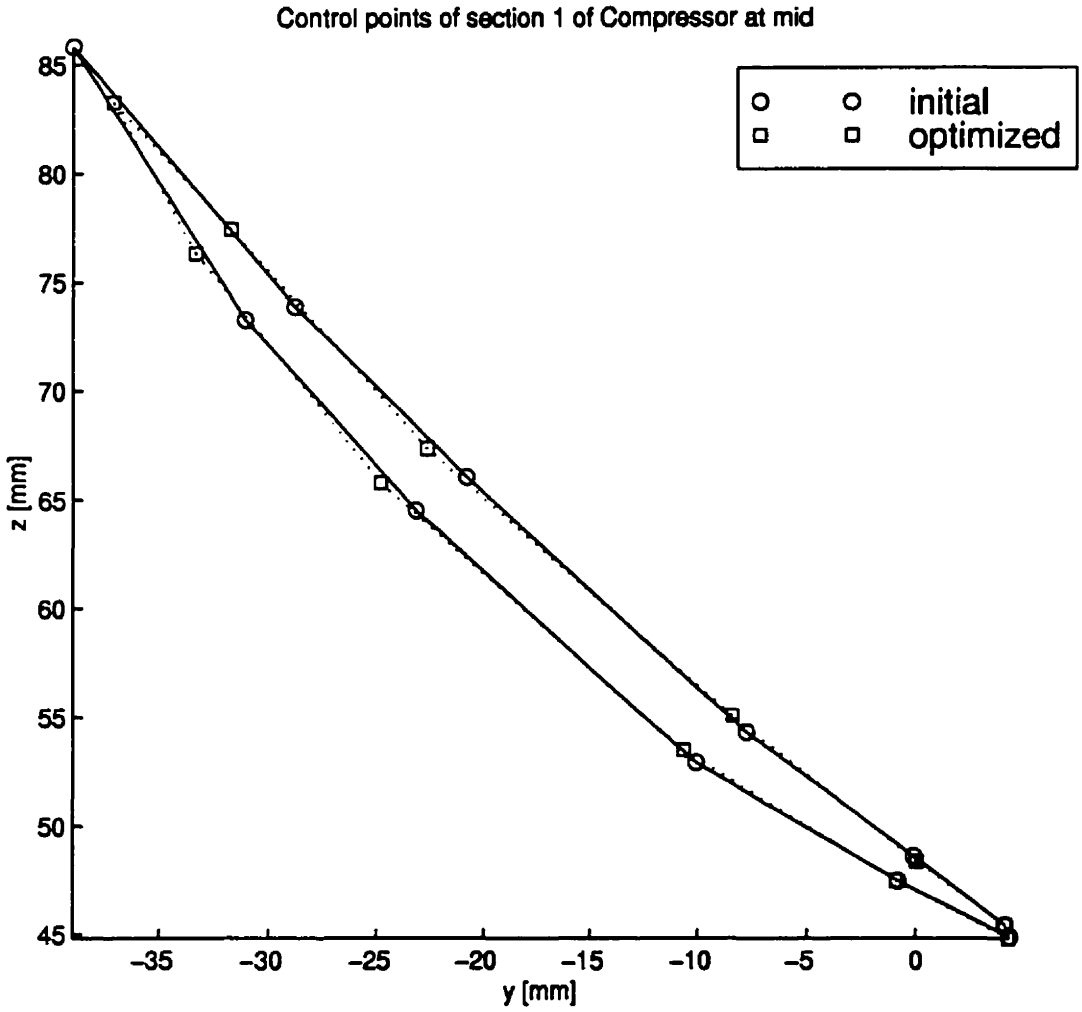


Figure 4.41: Control polygons of compressor blade's airfoil before and after optimizing the knots

4.3.2 Optimization of Weights

The total least squares error decreases from 9.09 [mm^2] to 0.78 [mm^2] in 51 iterations; Figure 4.42 shows the decrease of error with respect to number of iteration. Figure 4.43 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 1.21 millimeters, and 0.421 millimeters respectively. Active weight constraint were $w_6 = \varepsilon_w$. Figure 4.44 shows the control polygons of the approximation curves before and after knot adjustment.

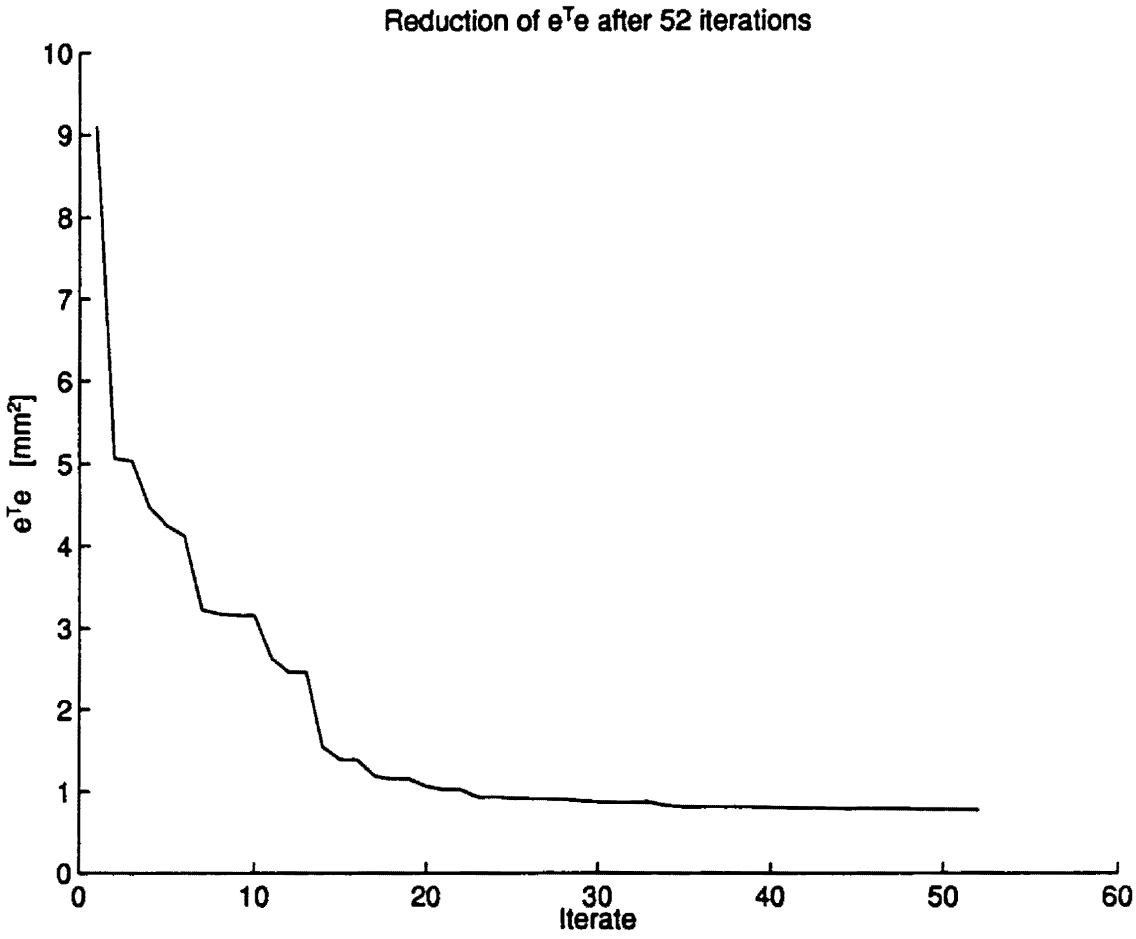


Figure 4.42: Reduction of least squares error of compressor blade's airfoil vs. number of iteration obtained from optimizing the weights

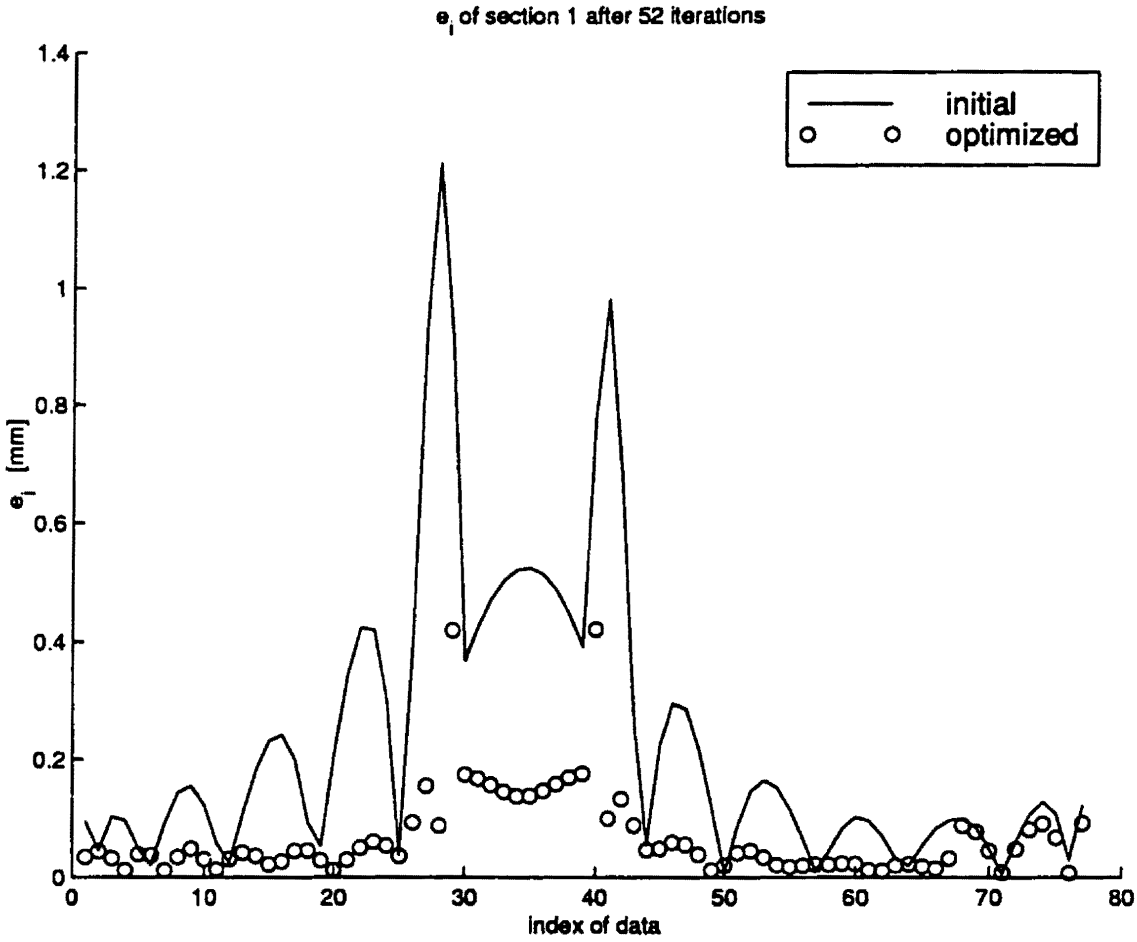


Figure 4.43: Error distributions of compressor blade's airfoil before and after optimizing the weights

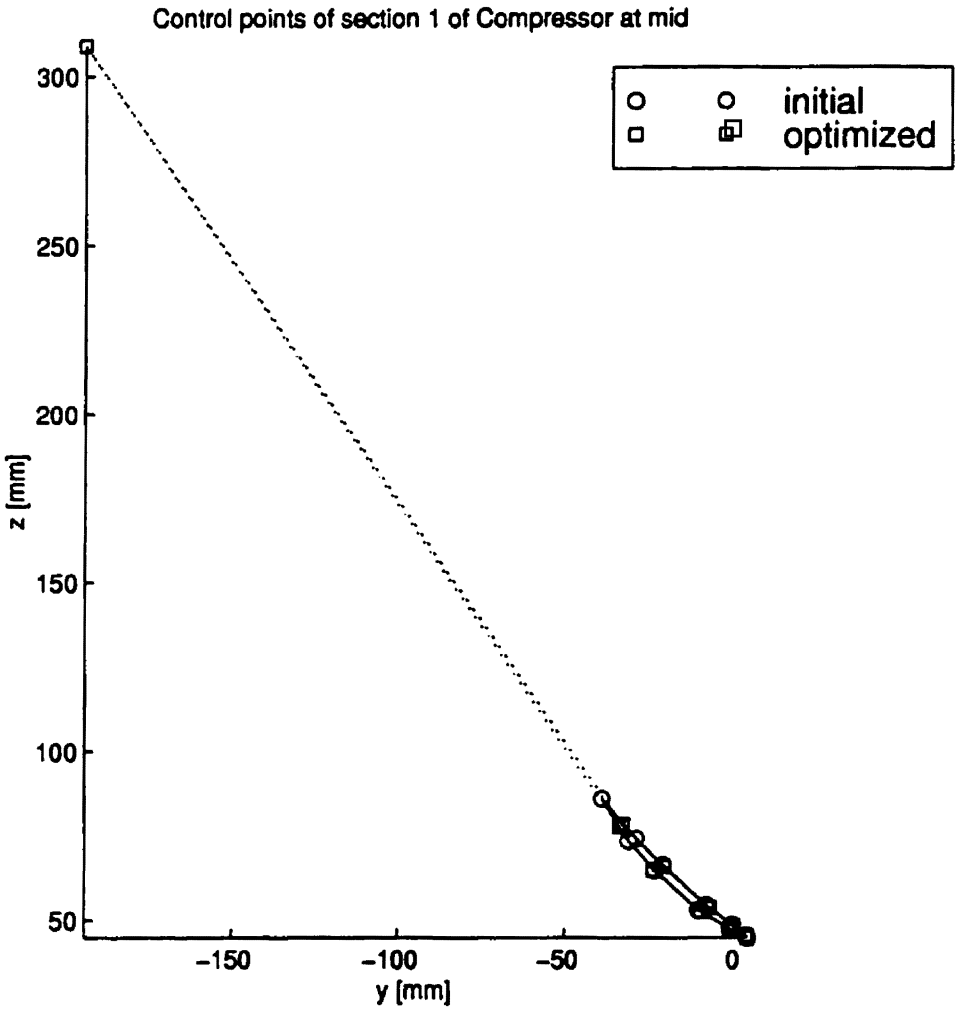


Figure 4.44: Control polygons of compressor blade's airfoil before and after optimizing the weights

4.3.3 Optimization of Parameterization for Data

The total least squares error decreases from 9.09 [mm^2] to 1.76 [mm^2] in 4 iterations; Figure 4.45 shows the decrease of error with respect to number of iteration. Figure 4.46(b) shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 1.21 millimeters, and 0.781 millimeters respectively.

Figure 4.47 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of **R**.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after three iteration.

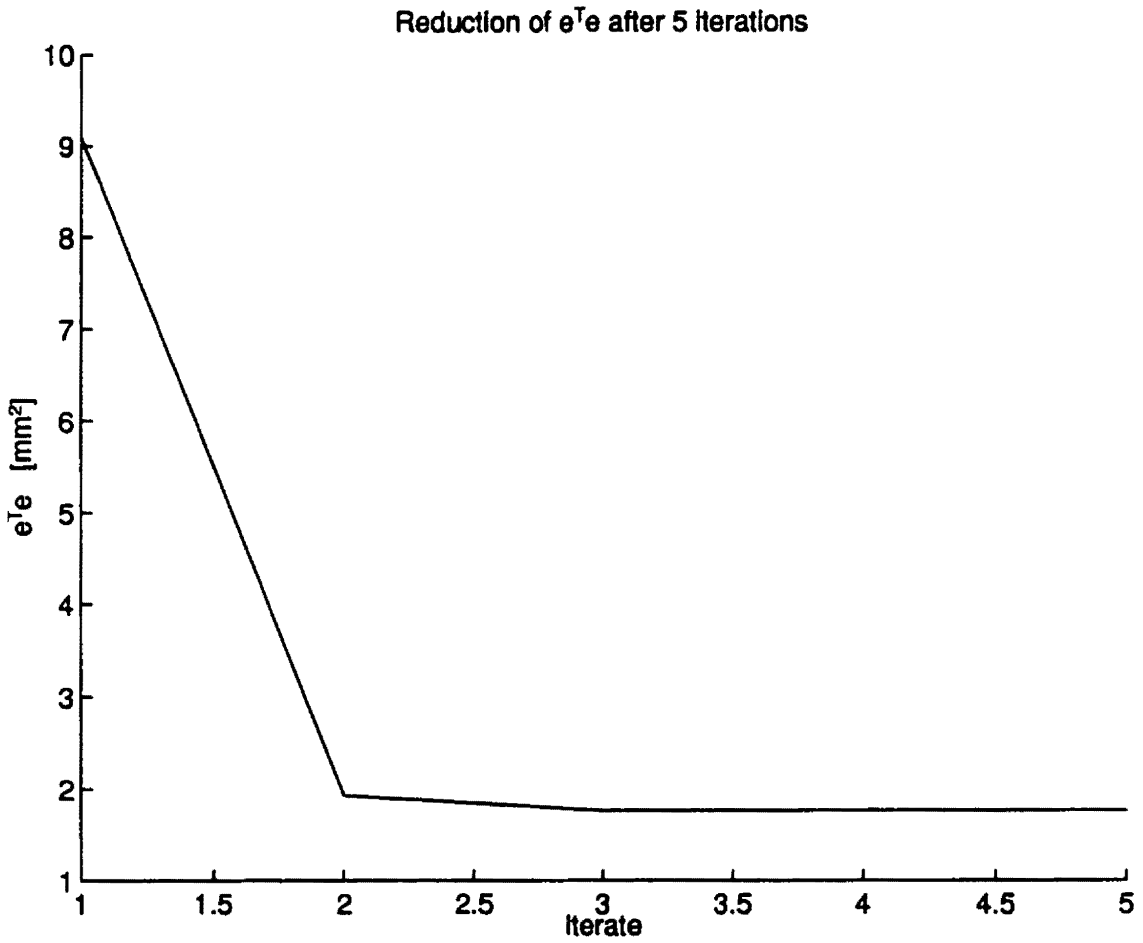


Figure 4.45: Reduction of least squares error of compressor blade's airfoil vs. number of iteration obtained from optimizing the parameterization for data

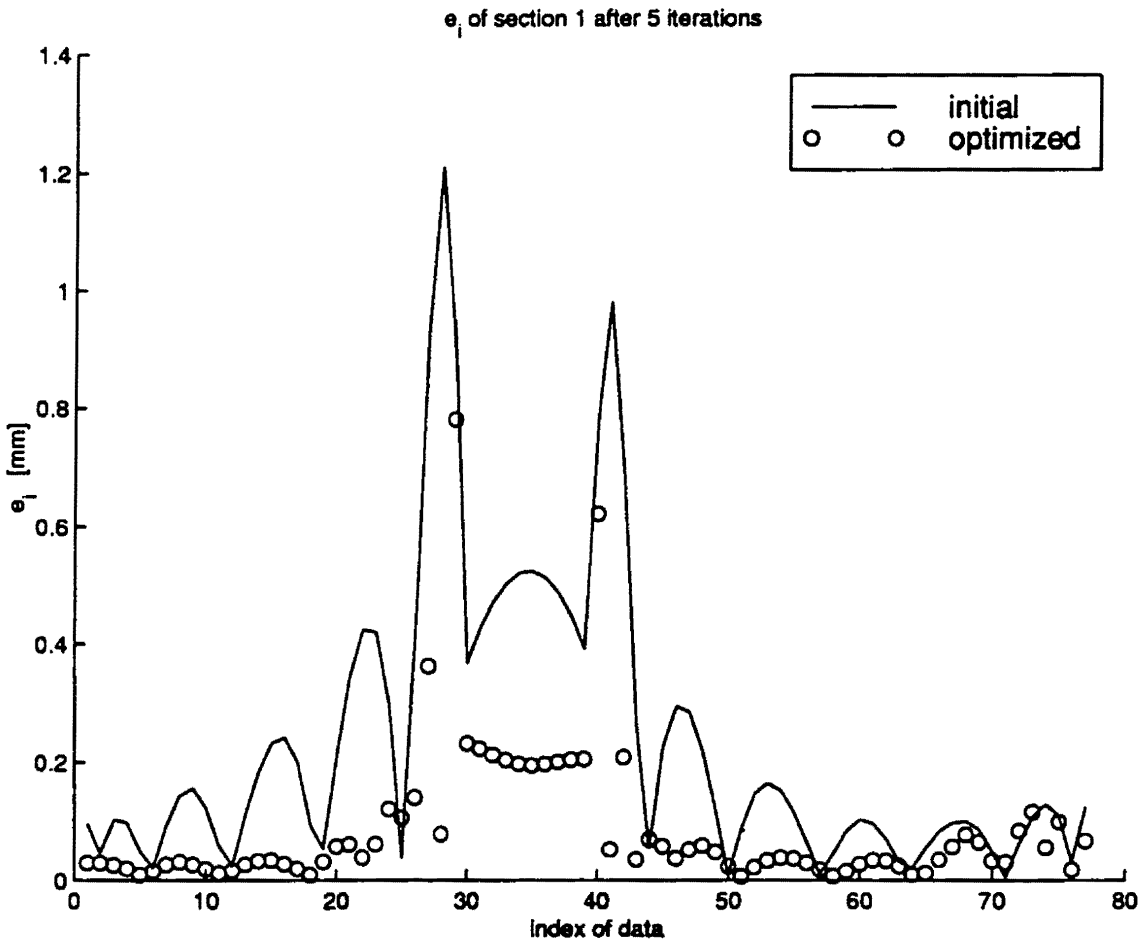


Figure 4.46: Error distributions of compressor blade's airfoil before and after optimizing the parameterization for data

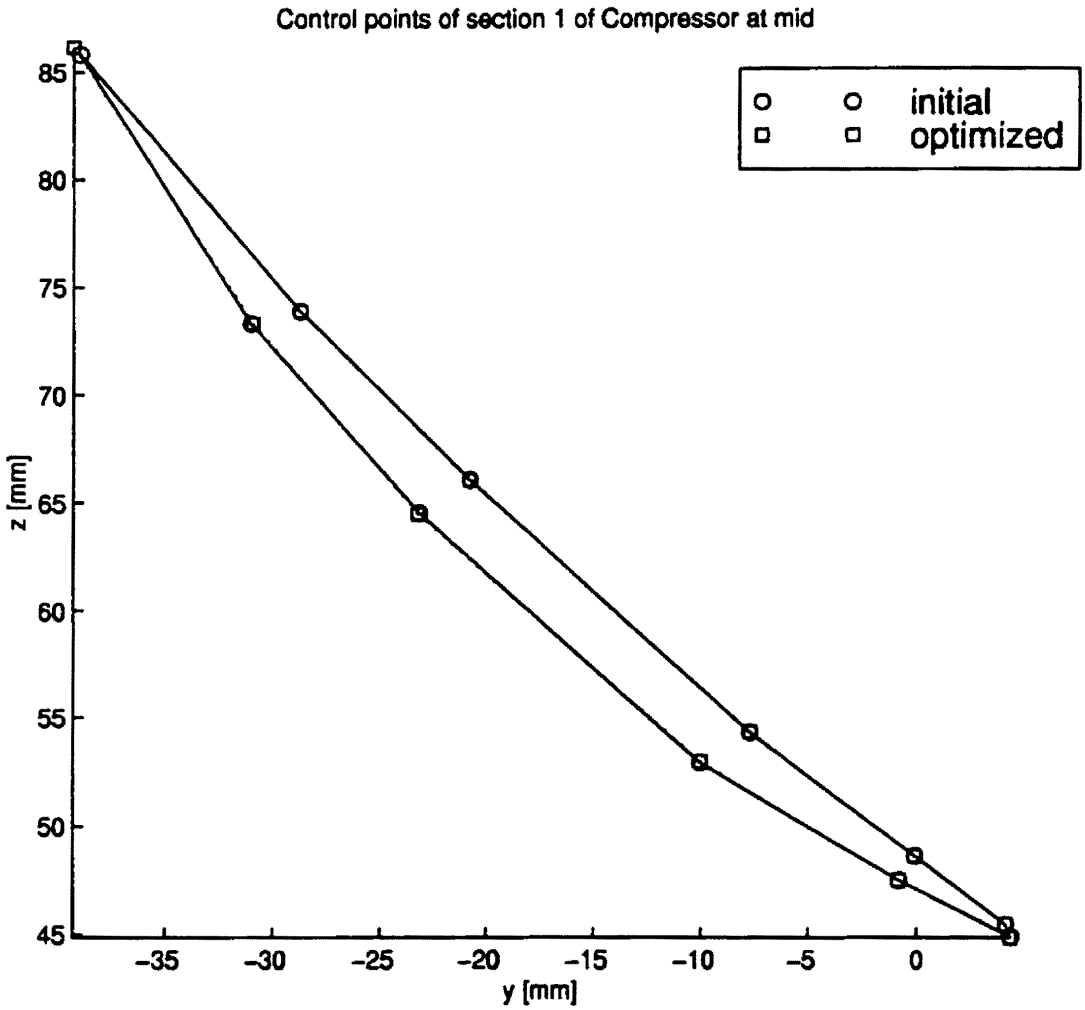


Figure 4.47: Control polygons of compressor blade's airfoil before and after optimizing the parameterization for data

4.3.4 Optimization of Knots and Weights

The total least squares error decreases from 9.09 [mm^2] to 0.335 [mm^2] in 51 iterations; Figure 4.48 shows the decrease of error with respect to number of iteration. Figure 4.49 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 1.21 millimeters, and 0.208 millimeters respectively. The weights practically did not change as indicated by extreme values of 1.0064 and 0.9932 from their initial values of unity.

Figure 4.50 shows the knot distributions, along the curves, before and after knot adjustment. None of the constraints was active.

Figure 4.51 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of $\mathbf{R}$.

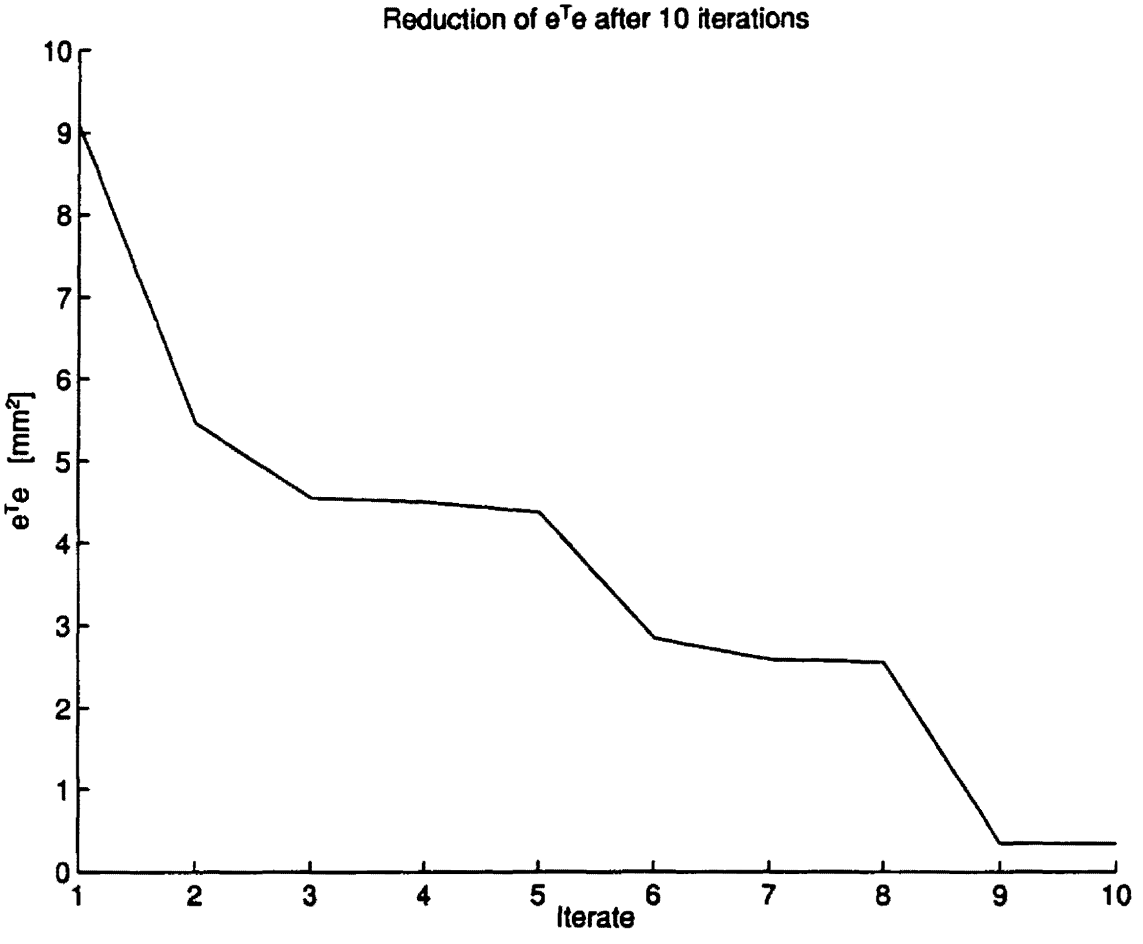


Figure 4.48: Reduction of least squares error of compressor blade's airfoil vs. number of iteration obtained from optimizing the combination of knots and weights

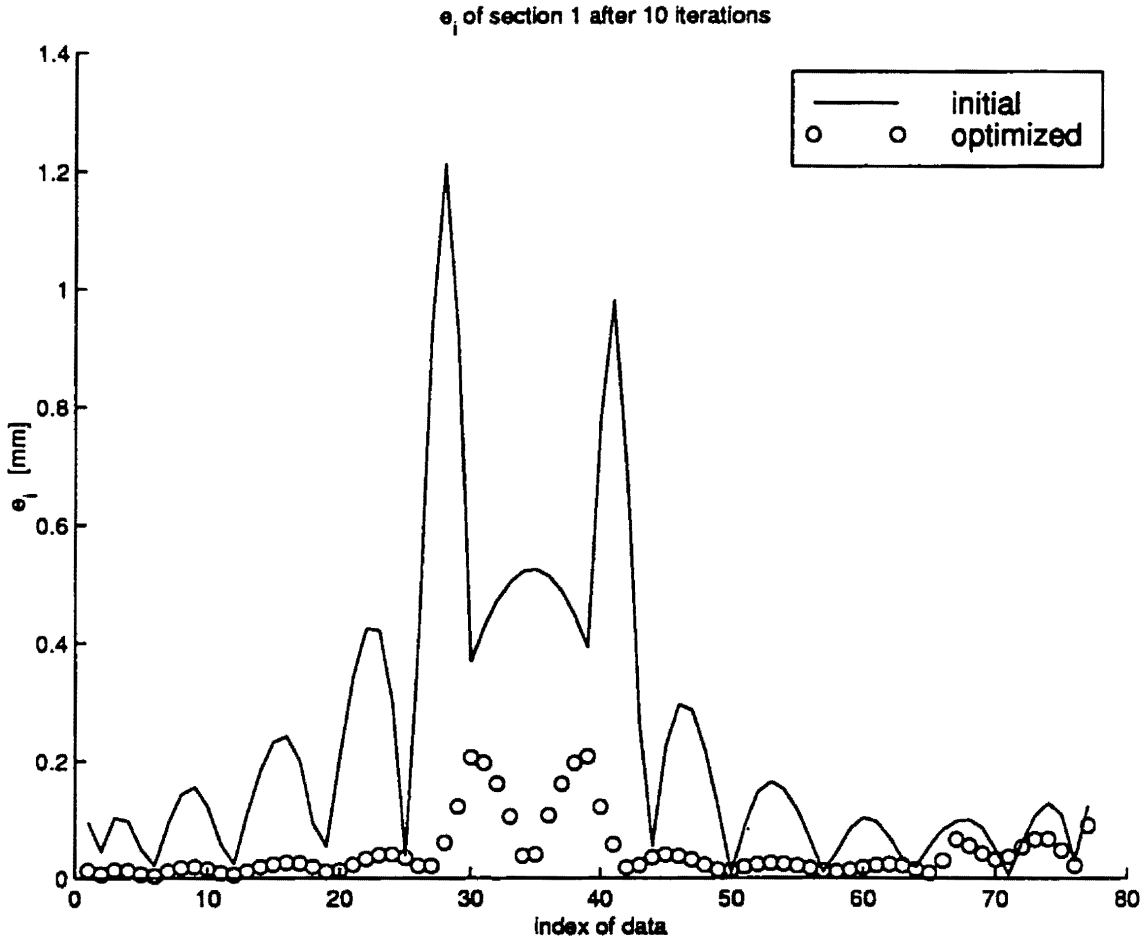


Figure 4.49: Error distributions of compressor blade's airfoil before and after optimizing the combination of knots and weights

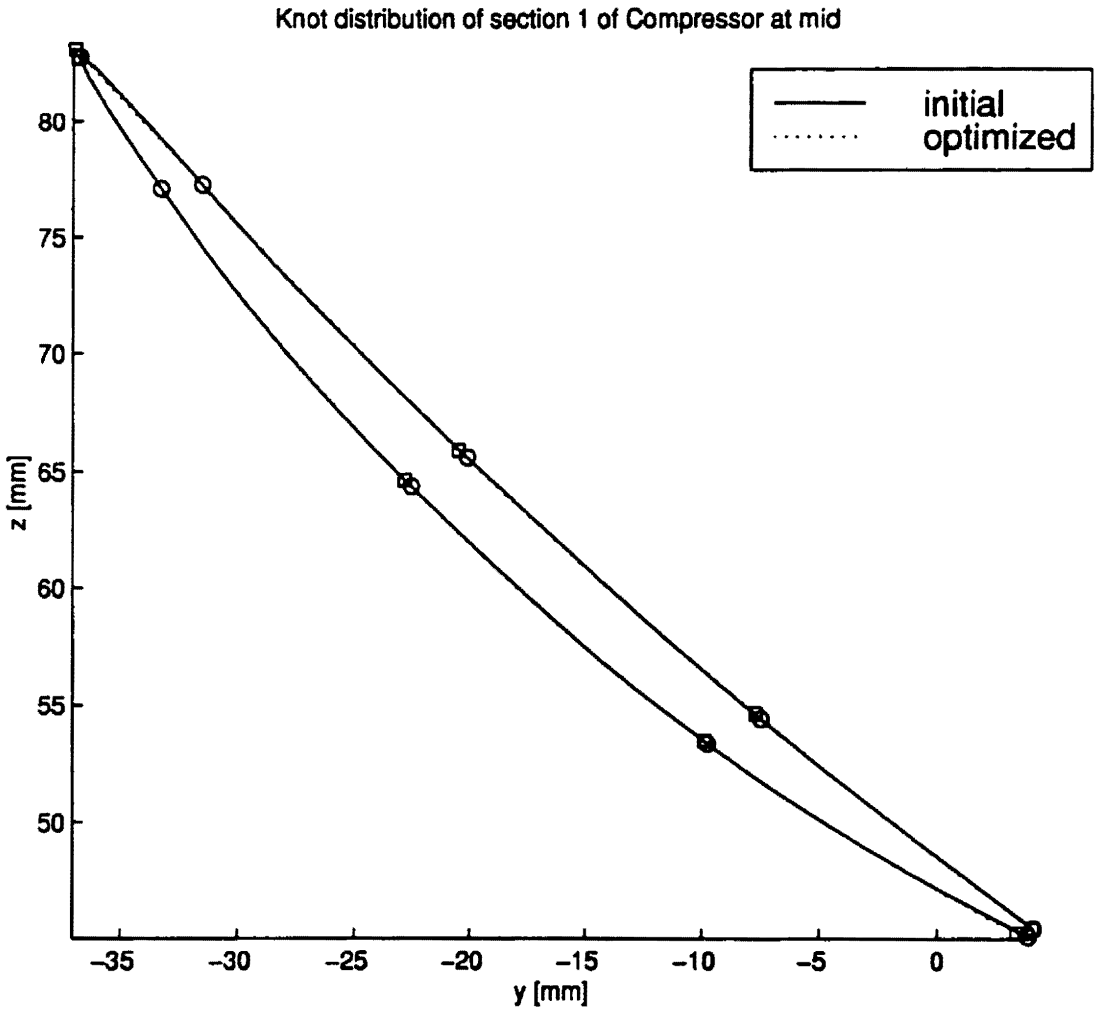


Figure 4.50: Distributions of Knots of compressor blade's airfoil before and after optimizing the combination of knots and weights

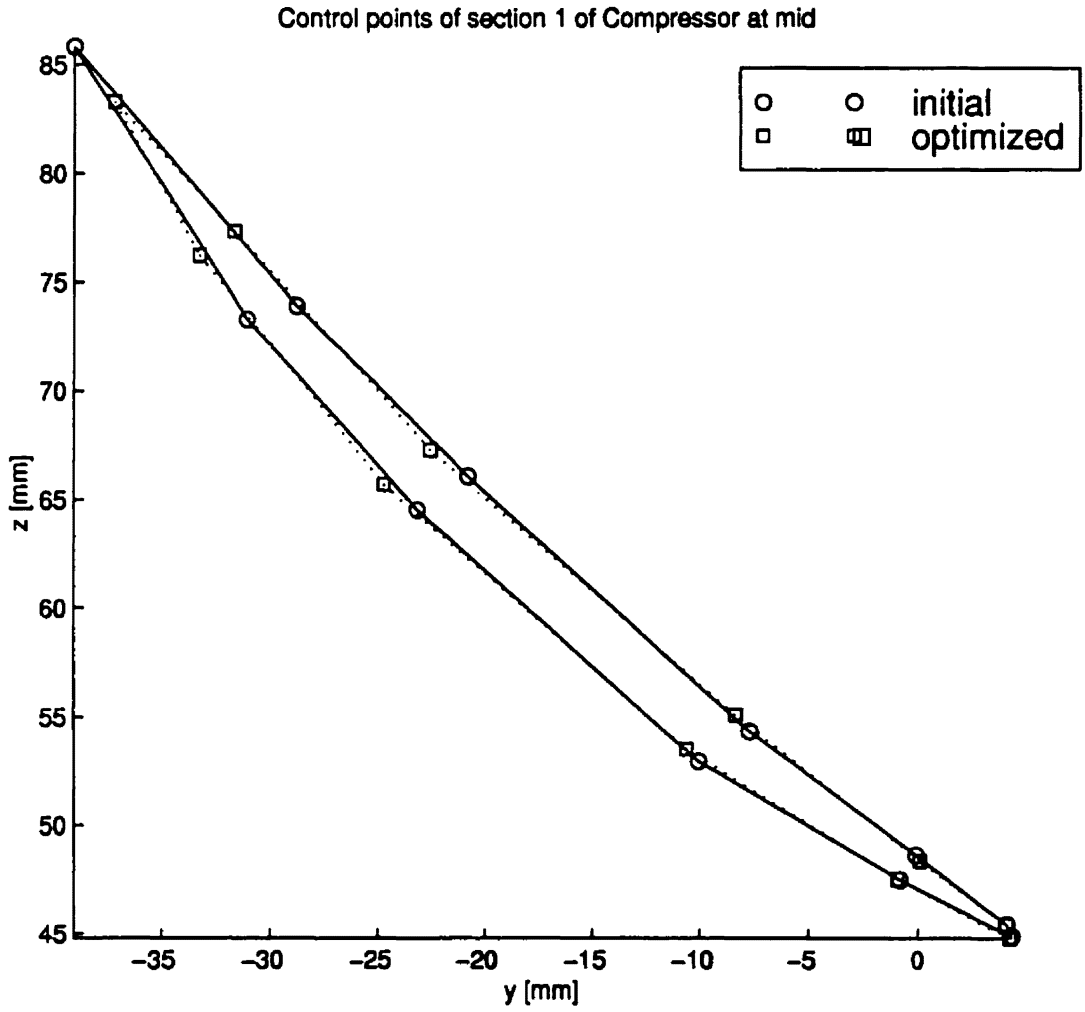


Figure 4.51: Control polygons of compressor blade's airfoil before and after optimizing the combination of knots and weights

4.3.5 Optimization of Knots, Weights and Parameterization

The total least squares error decreases from 9.09 [mm^2] to 3.98 [mm^2] in 1 iterations; Figure 4.52 shows the decrease of error with respect to number of iteration. Figure 4.53 shows the distributions of error along the airfoil before and after knot adjustment; solid lines represent the distribution before adjustment and circles represent the distribution after adjustment. The maximum errors before and after knot adjustment are about 1.21 millimeters, and 0.998 millimeters respectively.

Figure 4.54 shows the control polygons of the approximation curves before and after knot adjustment. This figure illustrates the good behavior of control points. This is a result of constraining the condition number of **R**.

Adjustment of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. The backtrack algorithm failed to obtain adequate step length after three iteration.

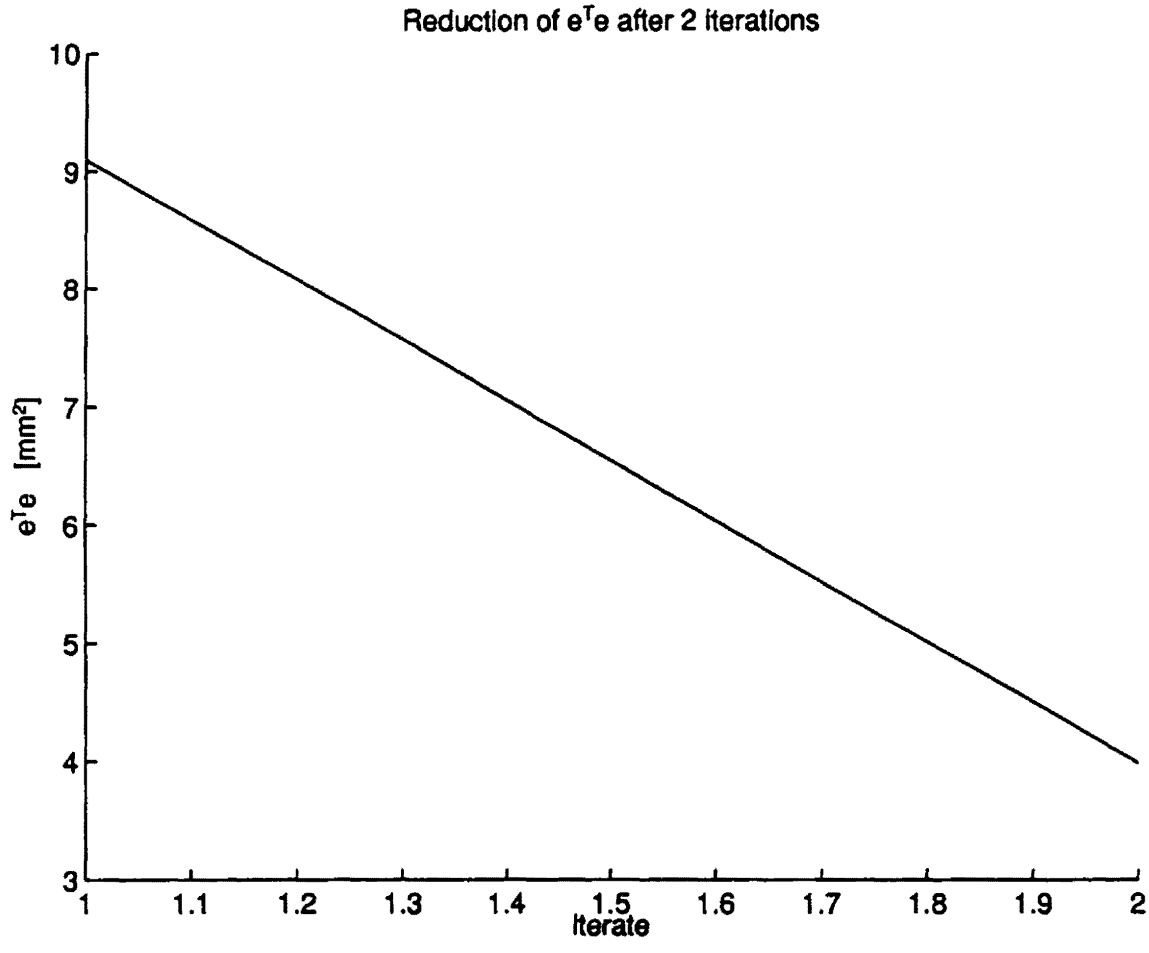


Figure 4.52: Reduction of least squares error of compressor blade's airfoil vs. number of iteration obtained from optimizing the combination of knots, weights, and parameterization for data

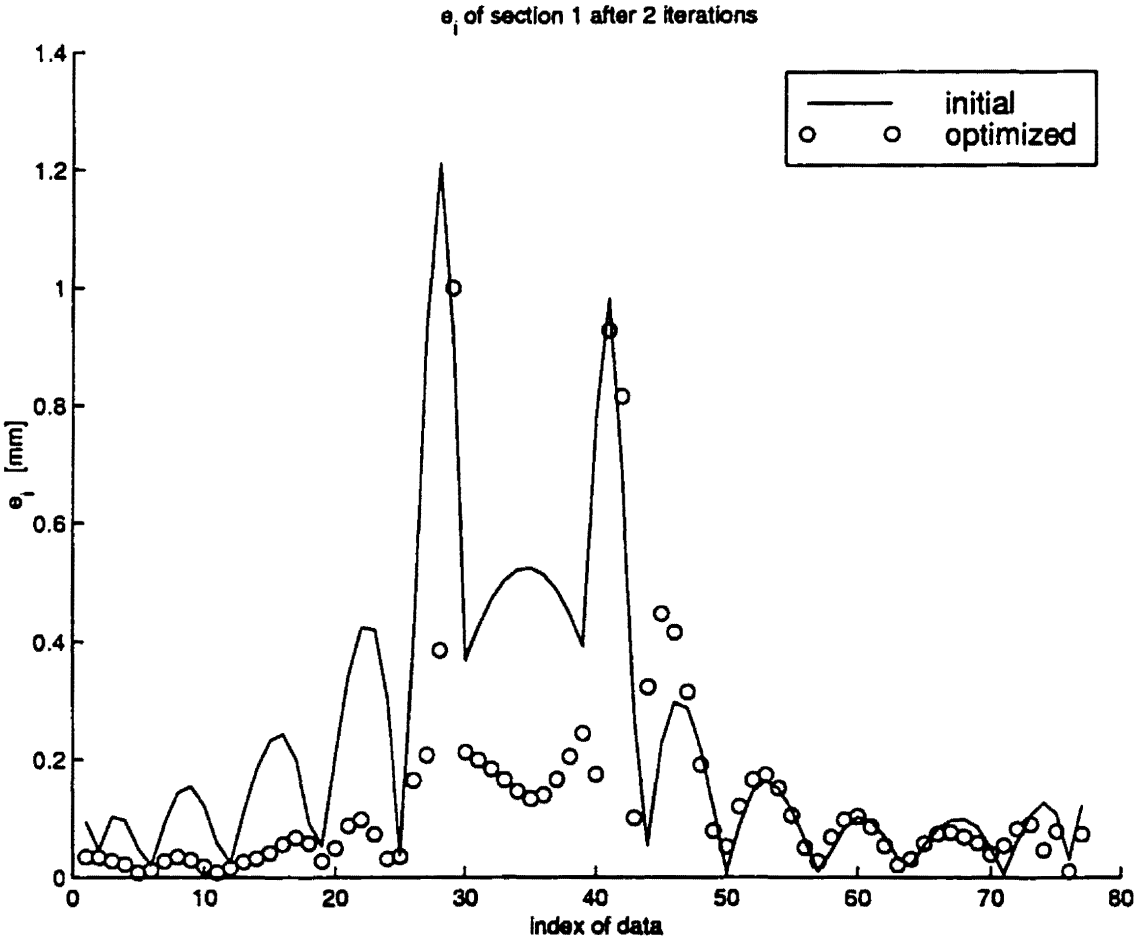


Figure 4.53: Error distributions of compressor blade's airfoil before and after optimizing the combination of knots, weights, and parameterization for data

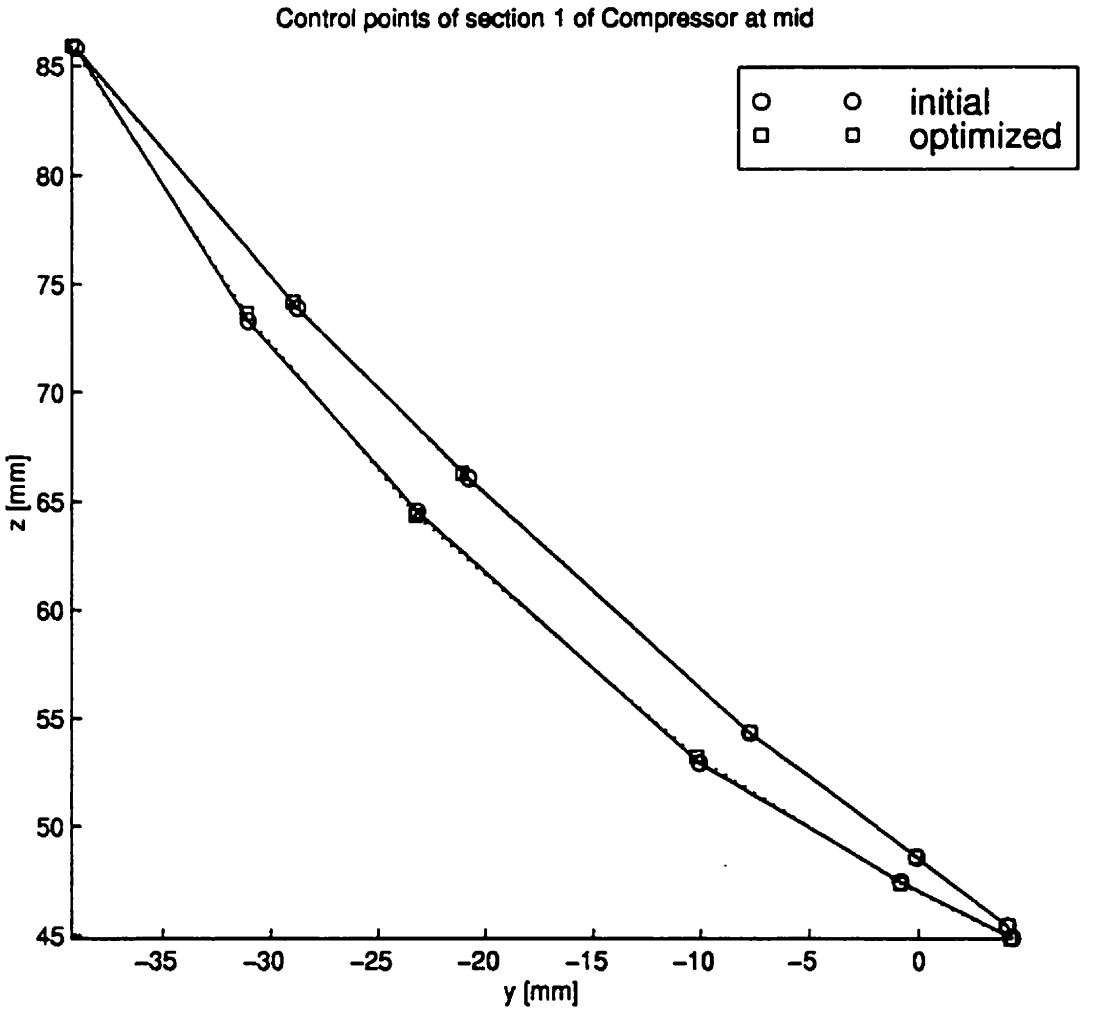


Figure 4.54: Control polygons of compressor blade's airfoil before and after optimizing the combination of knots, weights, and parameterization for data

4.4 Discussion

Results of the experiments are tabulated in Tables 4.1, 4.2, and 4.3, each of which lists the least-squares error, the maximum distance between data and their approximation, and the number of iteration for parameter adjustment respectively.

Table 4.1: Least Square Error of Three Sets of Data and Five Sets of Adjustable Parameters

	Residual $\mathbf{e}_x^T\mathbf{e}_x + \mathbf{e}_y^T\mathbf{e}_y + \mathbf{e}_z^T\mathbf{e}_z$ [mm^2]					
		Adjusted parameters				
Data	Initial	**u**	**w**	**t**	**(u, w)**	**(u, w, t)**
NACA 2415	0.186	0.002	0.028	0.068	0.002	0.061
WTEA	0.737	0.023	0.059	0.193	0.012	0.105
Compressor	9.09	0.327	0.78	1.76	0.335	3.98

Table 4.2: Maximum Distances Between Three Sets of Data and Their Approximation for Five Sets of Adjusted Parameters

	Maximum distance between data and curve [mm]					
		Adjusted parameters				
Data	Initial	**u**	**w**	**t**	**(u, w)**	**(u, w, t)**
NACA 2415	0.191	0.015	0.058	0.113	0.015	0.107
WTEA	0.211	0.030	0.054	0.113	0.024	0.187
Compressor	1.21	0.206	0.421	0.781	0.208	0.998

These results must be analyzed based on three criteria:

- performance of the approximation,
- the sensitivity of various combinations of optimized parameters, and
- the effect of constraining the condition number of matrix **R**.

Table 4.3: Number of Iteration for Three Sets of Data and Five Sets Adjusted Parameters

	Adjusted parameters				
Data	**u**	**w**	**t**	**(u, w)**	**(u, w, t)**
NACA 2415	51	51	3	51	2
WTEA	51	51	2	51	2
Compressor	51	51	4	51	1

The performance of the approximation showed that the least squares error is reduced by a factor of 93 for the NACA 2415 airfoil. This factor is obtained by dividing the initial error by the resulting error optimization of knots (under column **u**). The factor of error reduction is 61 and 27 for the WTEA and the axial compressor airfoils respectively. Table4.2 shows that the approximation also reduces the maximum distance between the curve and the data by factors of 12.7, 7, and 5.8, for the NACA 2415, the WTEA, and the axial compressor, airfoils respectively. A barchart constructed from Table4.2 is shown in Figure 4.55. This barchart shows the ratios of the maximum error before the optimization and the maximum error after the optimization for all combinations of parameters that were tested in this research. The unit of the ratios is in percent.

Table 4.1 and 4.2, and Figure 4.55, suggest that the best results (smallest residuals and smallest maximum error) are obtained from knot adjustment and knot-weight adjustment as shown in columns **u** and **(u, w)**. However, these columns also show that knot-weight adjustment produce only marginal improvement of accuracy over knot adjustment. This phenomena is consistent for the three different sets of data. Sections 4.1.4, 4.2.4, and 4.3.4, show that weights are practically unchanged

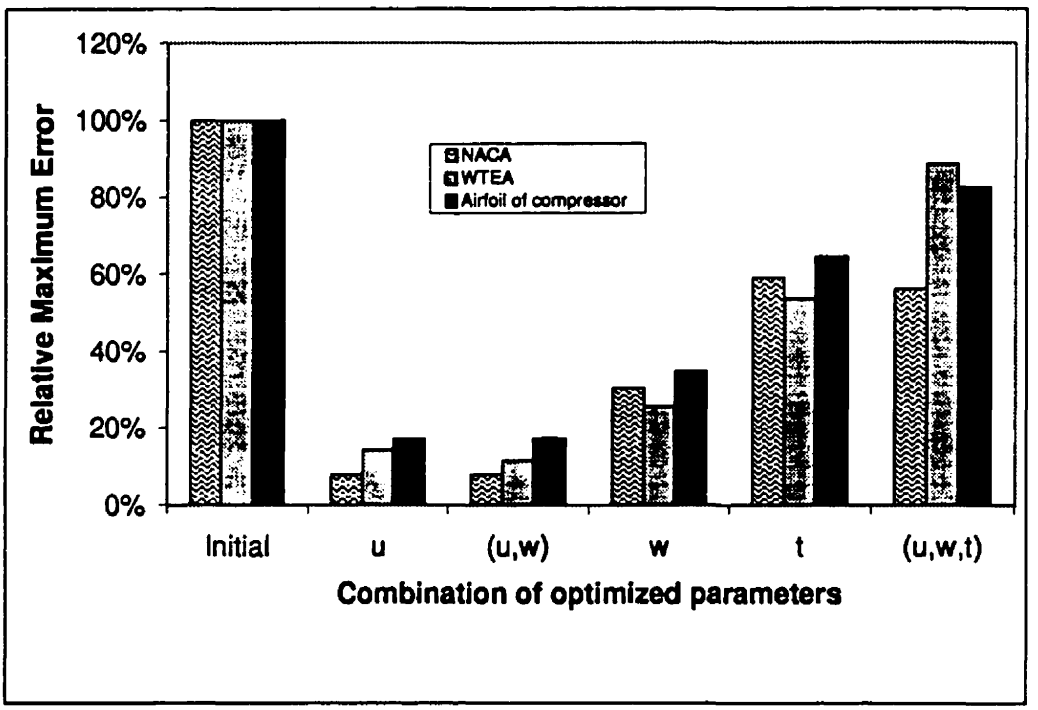


Figure 4.55: Barchart of the ratios of the maximum errors after the optimization and the maximum errors before the optimization.

on the knot- weight adjustments. The constancy of weights explains why knot-weight adjustment produce similar accuracy to one produced by knot adjustment.

This behavior can be used to justify the exclusion of weights from the adjustable parameters whenever knots are optimized. The effect of this exclusion for low order curves, whose orders are significantly smaller than their number of basis, is that the adjustable parameters will be halfed. This will result in faster computation.

Table 4.3 lists the number of iteration before our algorithm terminated the adjustment process. The table clearly shows that premature termination occured when adjustable parameters contain parameterization of data; this phenomena is shown in columns **t** and **(u, w, t)**.

This premature termination consistently occured in the backtracking part of the optimization. Standard texts of optimization suggest that this early termination is due to nonlinearity of the parameterization for the data[31], and cannot be handled easily. This finding can be used to justify dropping the parameterization for data from the adjustable parameters of NURBS least squares problem. The most significant benefit from this elimination is a significant reduction of number of adjustable parameters, which yields significantly lower computation time.

Columns **u**, **w**, and **(u, w)** in Tables 4.1 and 4.2 show the reduction of error and do not treat parameterization of data as an adjustable parameter. These columns clearly indicate that adjustment of knots produces significantly greater reduction of error than unadjusted knots, i.e. columns **w**. This finding can be used to justify that knots should be adjusted. This is the opposite of Jupp's[49, 50, 51] and Sarkar's[79, 78] works respectively, in which adjustment of knots produced

unfavorable reduction of least squares error. The favorable reduction of error in this research's knot adjustment can be attributed to the proposed method of elimination of active knot constraints as described in Section 3.3.

Focusing on columns $\mathbf{u}$ and $(\mathbf{u}, \mathbf{w})$ in Table 4.1 and 4.2 shows that adjusting knots and weights simultaneously yields only marginal improvement in the reduction of least squares error over the error obtained from adjusting only the knots. This finding, along with the findings in previous paragraphs, can be used to justify the elimination of weights and parameterization of data from the set of adjustable parameters. The elimination of these parameters results in insignificant loss of reduction of error. Moreover, this elimination reduces the number of adjustable parameters, which leads to significant reduction of computation time.

Chapter 5

Multicurve Least Squares Method

The previous two chapters have presented a method for optimally fitting a data set representing an airfoil with a NURBS curve. The optimization can be done over various combination of parameters. Chapter 4 studied the effectiveness of these combinations. In this chapter an implementation of a method to simultaneously fit many curves, represented by discrete data set, is presented.

5.1 Introduction

An application of this method can be found in turbine blade and wing design. In these application the wing/blade is defined by two or more airfoil sections each represented by a set of data points. These form the skeletal curves of the wing/blade skeleton and are skinned to generate the wing/blade surface.

The proposed method forces the skeletal curves to share a mutual knot vector and a mutual degree. This ensures compatibility between the different curves. In

order to extend the method developed for single curve application (see Chapter 3 and 4) a new objective function, a new list of parameters, a new list of constraints, and a method for knot initialization, are needed. This is discussed below.

5.1.1 Definition of Objective Function

Performing nonlinear least squares to a set of data curves simultaneously requires the definition of a new objective function. It must be a function of the least squares errors of the individual data curves and its gradient must be well defined and simple to derive. Based on this requirement, we propose the new objective function to be the sum of the least square errors of the individual data curves; denoting the new objective function as ε and the least squares error of a section curve as ε_i, the expression for the objective function is

$$\varepsilon = \sum_{i}^{l} \varepsilon_i \qquad \text{where } l \text{ is the number of curves} \tag{5.1}$$

The partial derivative of the objective function is simply

$$\frac{\partial \varepsilon}{\partial} = \sum_{i}^{l} \frac{\partial \varepsilon_i}{\partial} \tag{5.2}$$

The definition of $\partial \varepsilon_i / \partial$ is defined in Equation (2.29).

5.1.2 Optimization Parameters

The parameters of the objective function consist of the union of parameters of the individual skeletal curves. In this thesis, the union is constructed by stacking individual skeletal curve's parameters and can be written as

$$\mathbf{x} = \left\{ \begin{array}{cccccccccc} \mathbf{u} & \mathbf{w}_1^T & \mathbf{w}_2^T & \cdots & \mathbf{w}_n^T & \mathbf{t}_1^T & \mathbf{t}_2^T & \cdots & \mathbf{t}_n^T \end{array} \right\}^T \tag{5.3}$$

The number of elements of $\mathbf{w}_i$'s is identical for all i because of mutual $\mathbf{u}$ and degree. On the contrary, $\mathbf{t}_i$'s may have different number of elements depending on the size of the data. The linear parameters $\mathbf{c}_i$ are obtained from $\mathbf{R}_i^+\mathbf{p}_i$ where $\mathbf{R}_i = \mathbf{R}_i(\mathbf{u}, \mathbf{w}_i, \mathbf{t}_i)$ as described earlier in Equation 2.3.

5.1.3 Linear and Nonlinear Constraints

The matrix of linear constraints is obtained by combining the matrices of linear constraints of individual skeletal curve as follows

$$\begin{bmatrix} [\mathbf{A}_u] & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} \mathbf{A}_{w,1} & & \\ & \ddots & \\ & & \mathbf{A}_{w,n} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \begin{bmatrix} \mathbf{A}_{t,1} & & \\ & \ddots & \\ & & \mathbf{A}_{t,n} \end{bmatrix} \end{bmatrix} \tag{5.4}$$

where $\mathbf{A}_u$ is the matrix of linear constraints on the knots, $\mathbf{A}_{w,i}$ is the matrix of linear constraint on the weights of the i-th skeletal curve, and $\mathbf{A}_{t,i}$ is the matrix of linear constraint on the parameterization of the i-th skeletal curve.

The nonlinear constraints are formed by stacking the nonlinear constraints of individual curves as follows

$$\begin{Bmatrix} \text{cond}\,(\mathbf{R}_1) \\ \text{cond}\,(\mathbf{R}_2) \\ \vdots \\ \text{cond}\,(\mathbf{R}_n) \end{Bmatrix} \leq \begin{Bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{Bmatrix} \tag{5.5}$$

where $\mathbf{R}_i$ is the overdetermined least square matrix of the i-th curve and cond $(\mathbf{R}_i)$ is the condition number of $\mathbf{R}_i$. In this chapter, β_i's are set to 500 for all i.

5.1.4 Initialization of Knots

Initial knots are computed using a modified version of Piegl's averaging procedure[70]. The modification are the use of the union of the parametrization of the skeletal curves, i.e. $\mathbf{t} = \mathbf{t_1} \cup \mathbf{t_2} \cup \cdots \cup \mathbf{t_n}$, for computing the knots. The parameterization of data used to compute initial knots is obtained from the union of the parameterization of data of all curves; denoting the union by $\mathbf{t}$, its definition is

$$\mathbf{t} = \mathbf{t_1} \cup \mathbf{t_2} \cup \cdots \cup \mathbf{t_n} \tag{5.6}$$

The Schoenberg-Whitney condition most likely will be satisfied by this initialization of knots whenever the data curves consist of fairly uniform numbers of points. Care must be taken, however, when the numbers of data of the section curves are nonuniformly distributed. In this case knot initialization may fail to satisfy the Schoenberg-Whitney condition. When this happens, the problem can be overcome by reparameterization. This is done by computing a temporary knot vector using the parameterization values of the curves that failed the Schoenberg-Whitney condition. The knot vector has the same number of elements as the previously obtained mutual knot vector. The reparameterization is then performed as follows.

When knot initialization $\mathbf{u}$ fails to satisfy the Schoenberg-Whitney condition on the parameterization $\mathbf{t}$, there are one or more B-spline basis of order k not supported by the combination of $\mathbf{u}$, k, and $\mathbf{t}$. This is illustrated in Figure 5.1, where the B-spline basis, N_j^k, is not supported by the data (shown in $\Diamond$).

In this case, a temporary knot vector, $\mathbf{v}$ is computed using the averaging

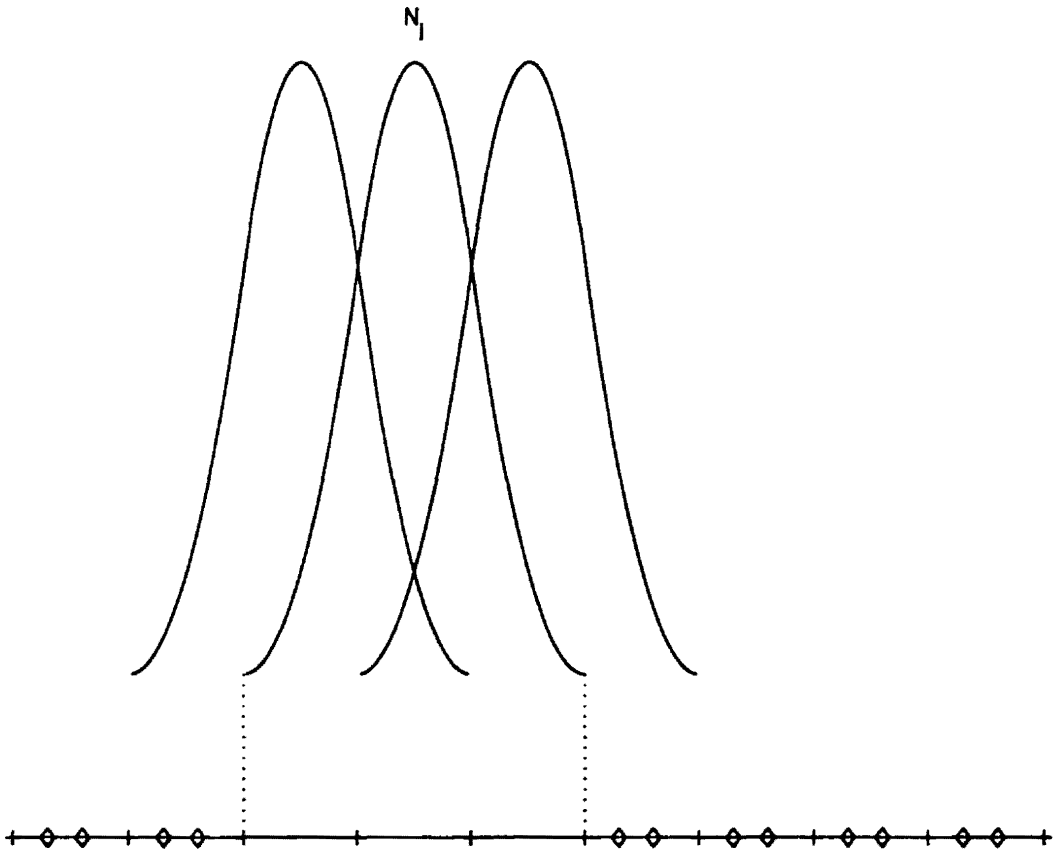


Figure 5.1: Failure of Knot Initialization due to Lack of Support for B-spline Basis N_j

method[70]. The vector **v** is based on **t** and the prespecified number of basis. The latter forces vectors **u** and **v** to have identical number of elements. The averaging method guarantees that all B-spline basis defined by **v**, **t**, and the prespecified number of basis, are supported.

A new parameterization for the data points is then computed by relocating elements of **t** in such a way that they occupy the same interval and that their relative positions within the interval remains the same. For $t_j \in [v_i, v_{i+1})$, its relative position within interval is denoted by α_j and it can be computed from the linear equation $\mathbf{t_j} = (1 - \alpha_j)\, v_i + \alpha_j v_{i+1}$. The new parameterization on **u** can then be computed using $\mathbf{t_j} = (1 - \alpha_j)\, u_i + \alpha_j u_{i+1}$.

This reparameterization will distribute the data uniformly along the mutual knot vector **u**. Thus, the problem of unsupported basis is eliminated. Furthermore, the definition of α_j preserves geometric information for the subset of data which lie on a single interval.

The proposed method was implemented to measure its performance in obtaining NURBS skeletal curves that approximate a given data set. The NURBS curves are functions of the optimized parameters, namely, knots, weights, and parameterization for data. These parameters can be optimized together or in various combinations such as knots, knots & weights, and knot & weights & parameterization for data. Another important purpose of this implementation is to measure the sensitivity of reduction of the least squares error with respect to these combinations of optimized parameters. This sensitivity analysis will be used in justifying which parameters can be dropped from the optimization. This results in an optimization

with fewer degree of freedom. Also of particular importance is the effectiveness of the proposed nonlinear constraint in providing accurate control points.

Three sets of data representing a skeleton of an airplane wing, a skeleton of an axial compressor blade, and a skeleton of a turbine blade were used to accomplish the above goals. Each data set was fitted simultaneously with NURBS skeletal curves curve using the proposed method while optimizing the following five combinations of parameters. These are:

1. knots;
2. weights;
3. parameterization for data;
4. knots and weights; and
5. knots, weights, and parameterization for data.

The first three tests were conducted to observe the behavior of each nonlinear parameters. The last two tests were conducted to observe the behavior of a combination of parameters. The first and the fourth tests allow a comparison of approximation with integral and rational B-spline methods respectively. Results from the fifth tests are used to justify the deletion of parameterization from the list of adjustable parameters.

The proposed 15 set of skeletal curves were fitted with the method described in this chapter. The termination criteria for the optimization was based on Lagrange multipliers and the standard termination criteria as suggested in[15, 73]. Arclength

parameterization was used to initialize the parameterization for data. Weights were initialized to one. Knots were initialized using the averaging method[70]. Maximum permissible condition number for $\mathbf{R}$ was set to 500 based on a consideration that its reciprocal is still much larger than the machine's precision (10^{-6}). Tolerances of activity for the linear constraints were set to 10^{-2} for knot and weight constraints, and to 10^{-5} for constraints of parameterization for the data.

5.2 Wing Airfoils

This section presents the decrease of objective function and the decrease of error distribution of a two-section skeleton. The curves forming the skeleton are the NACA 2415 and the WTEA, and the data are identical to ones used in the previous chapter. Figure 5.2 shows the three dimensional view of the airfoils and their relative locations.

5.2.1 Optimization of Knots

The sum of least-square error decreased from 0.771 mm^2 to 0.005 mm^2 in 51 iteration as shown in Figure 5.3. The largest distances between the data and the approximation curves decreased from 0.244 millimeters to 0.015 millimeters for NACA 2415 and from 0.158 millimeters to 0.011 millimeters for the WTEA as shown in Figure 5.4 and 5.5 respectively. The shape of the NACA 2415 and the WTEA, along with their junctions, are shown in Figure 5.6. Active knot constraint is $u_8 = u_9$.

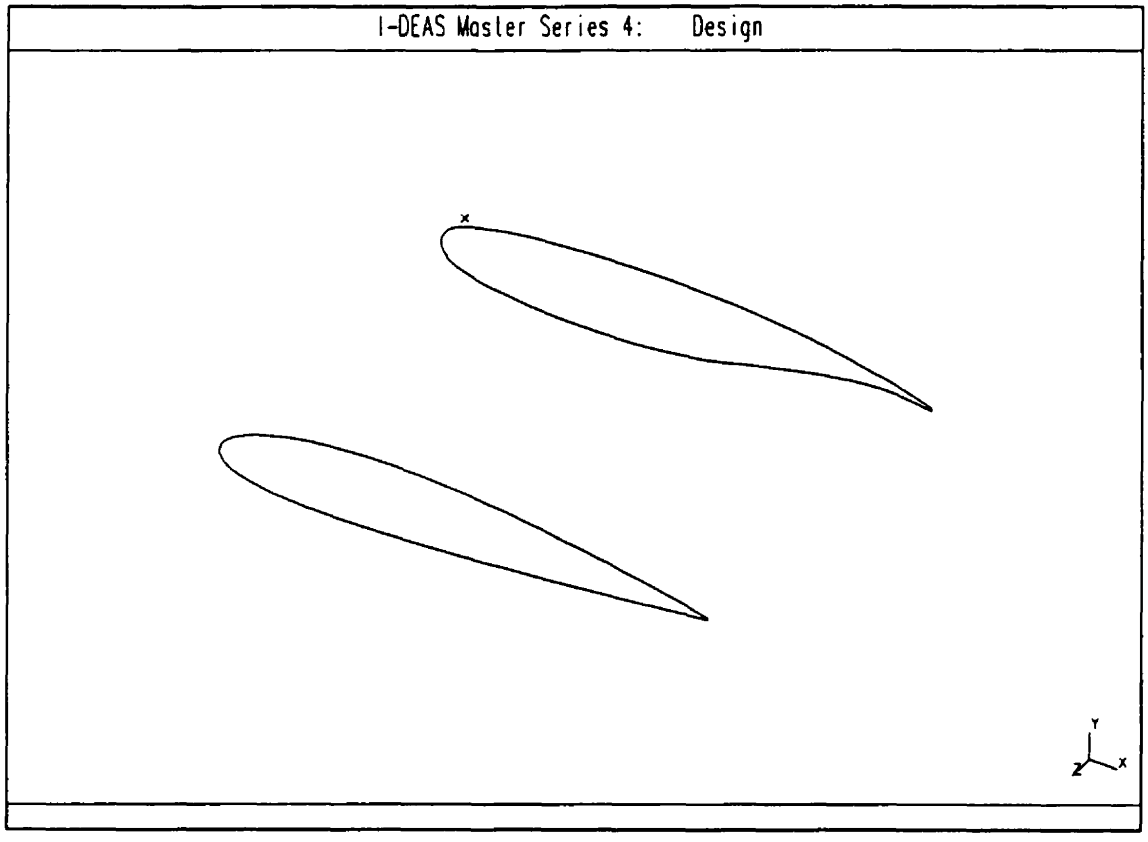


Figure 5.2: The NACA 2415 (left) and WTEA (right) Airfoils Form the Skeletal Curves of The Wing

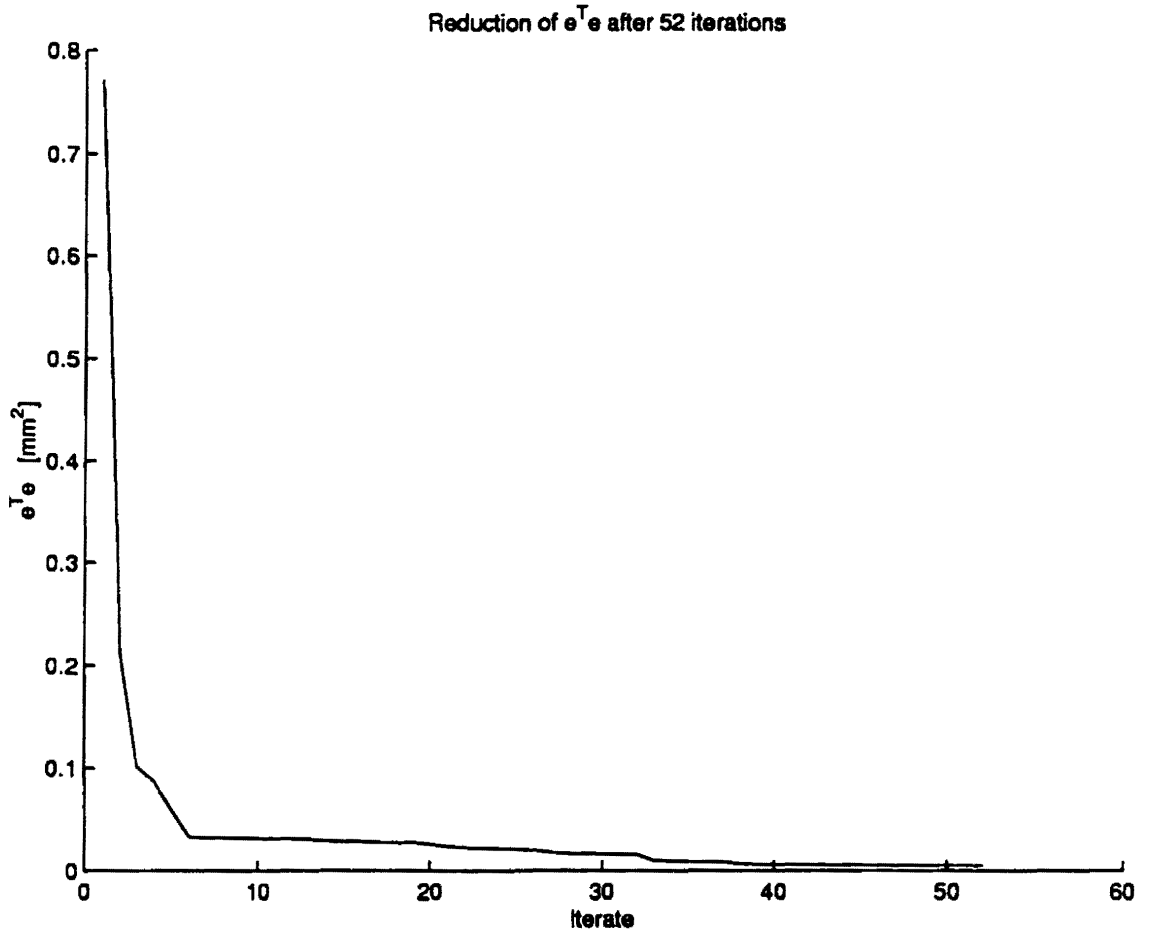


Figure 5.3: Reduction of error vs. number of iteration for two-curve wing airfoils obtained from optimizing the knots

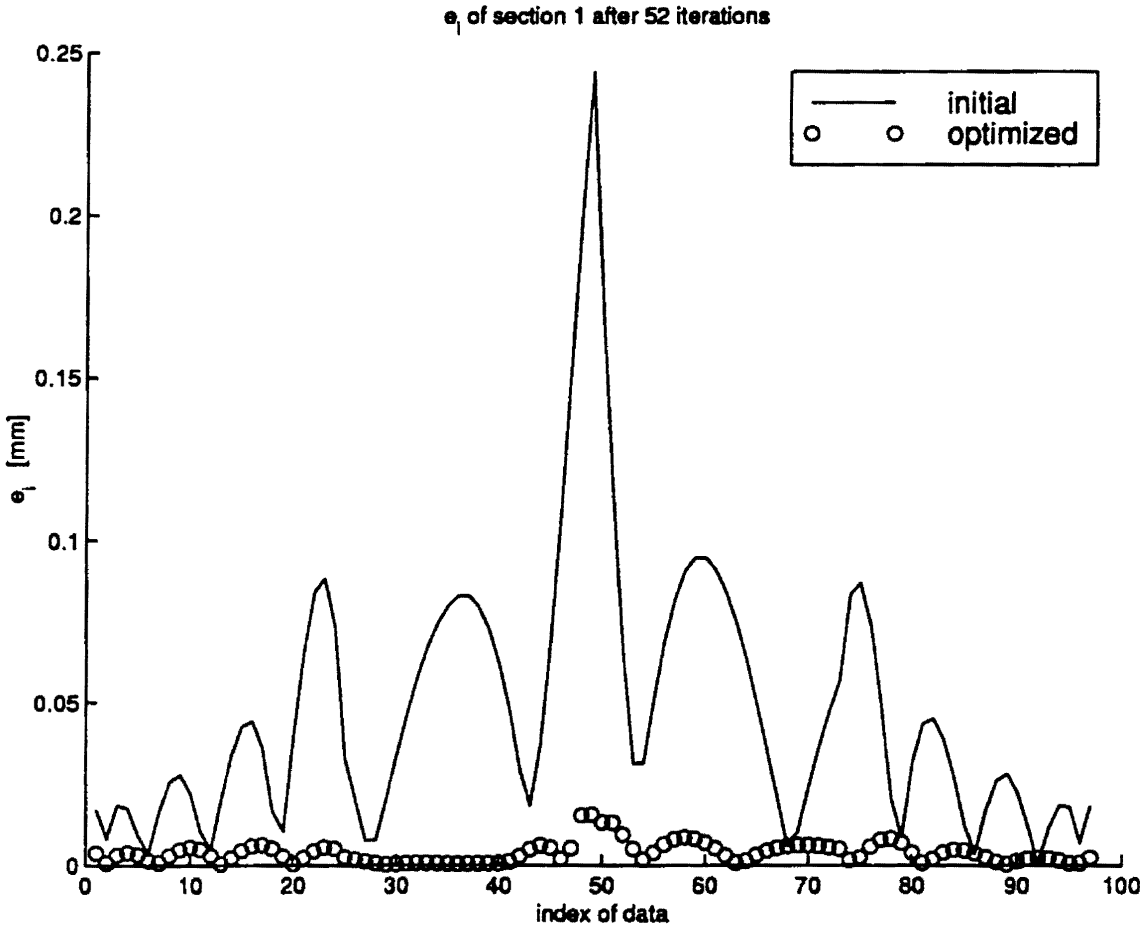


Figure 5.4: Error distribution of NACA airfoil of two-curve wing airfoils obtained from optimizing the knots.

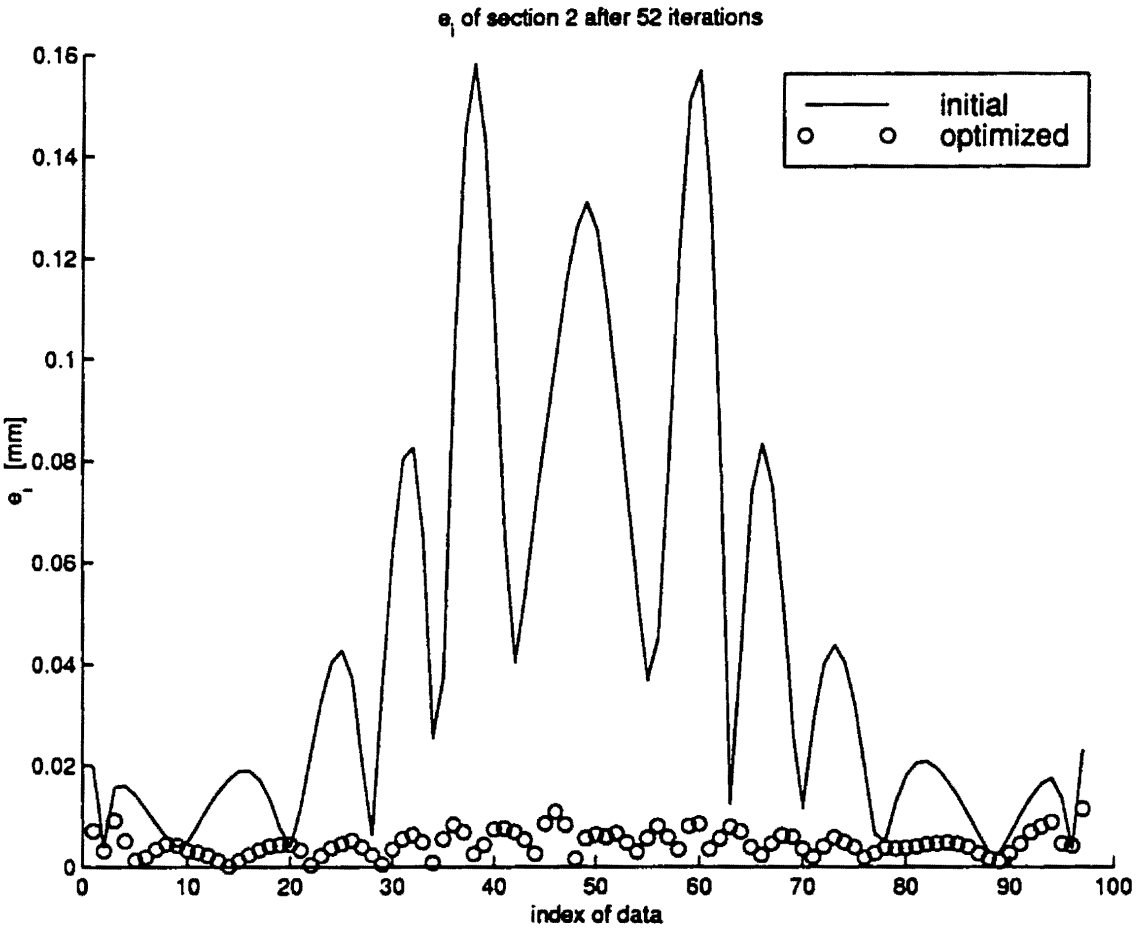


Figure 5.5: Error distribution of WTEA of two-curve wing airfoils obtained from optimizing the knots.

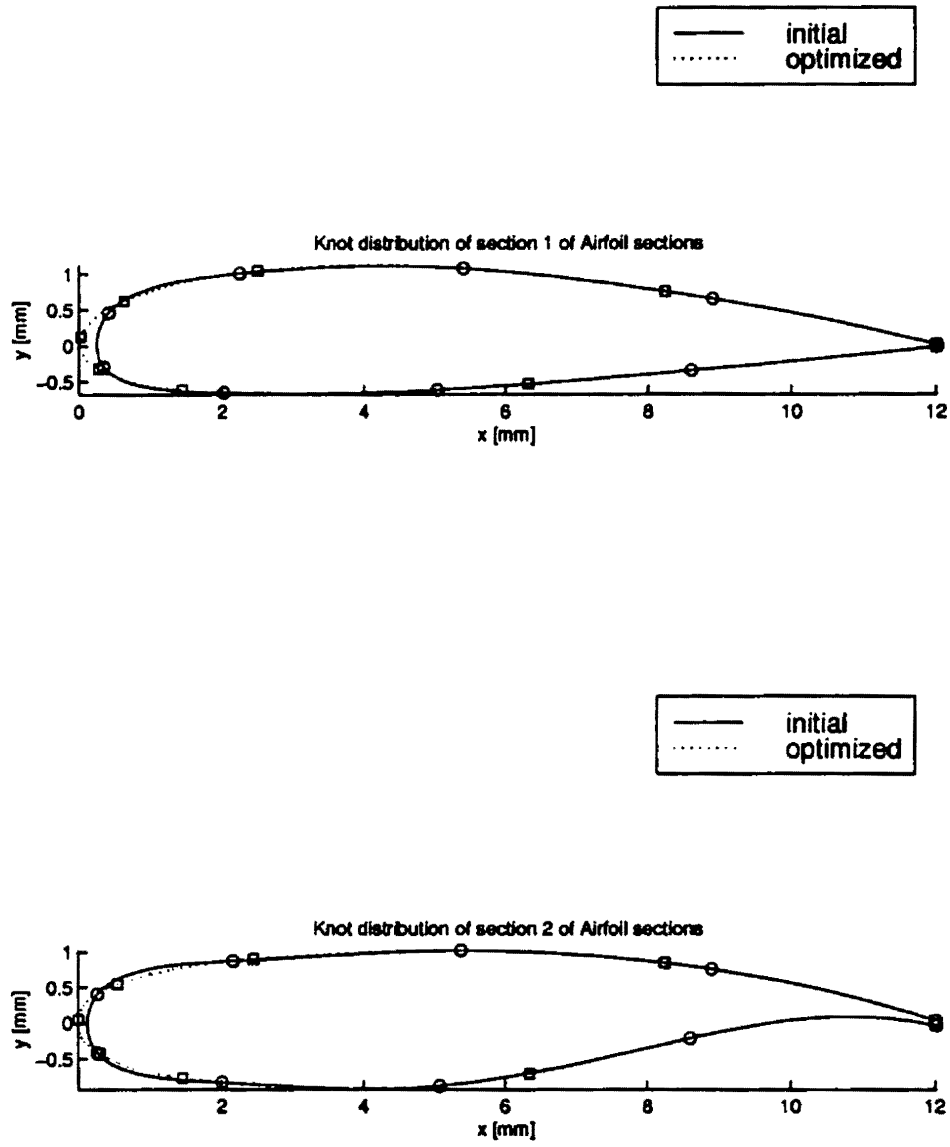


Figure 5.6: Shape of NACA 2415 (top) and WTEA (bottom) before and after optimization of knots.

5.2.2 Optimization of Weights

The sum of least-square error decreased from 0.771 [mm^2] to 0.073 [mm^2] in 51 iteration as shown in Figure 5.7. The largest distances between the data and the approximation curves decreased from 0.244 millimeters to 0.063 millimeters for NACA 2415 and from 0.158 millimeters to 0.038 millimeters for the WTEA as shown in Figure 5.8 and 5.9 respectively. The shape of the NACA 2415 and the WTEA, along with their junctions, are shown in Figure 5.10. All constraints are inactive.

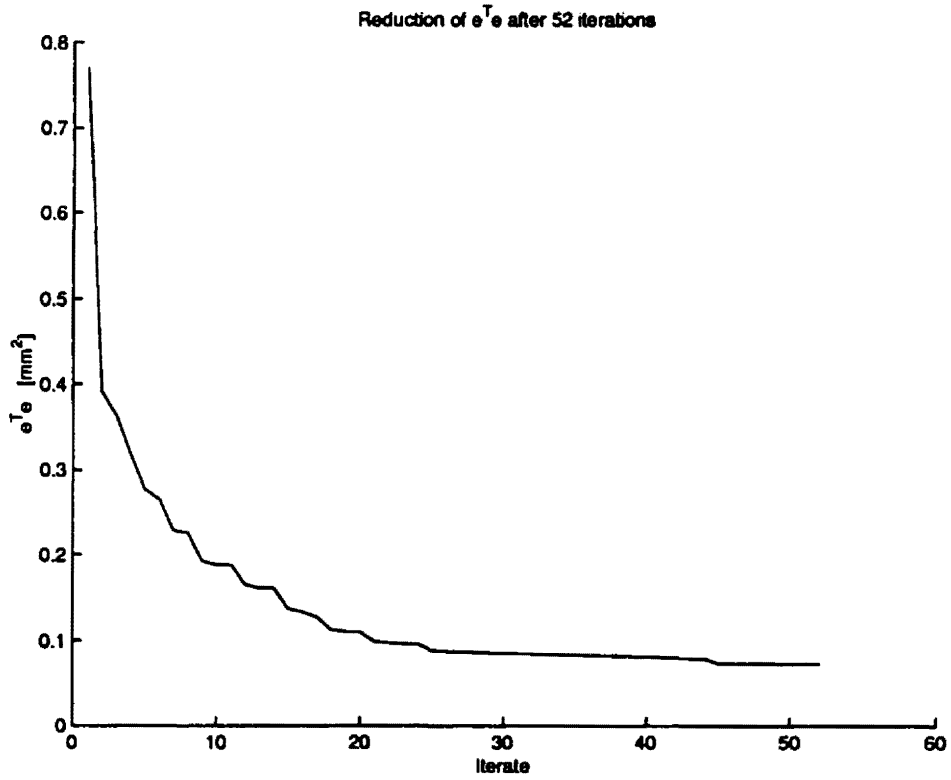


Figure 5.7: Reduction of error vs. number of iteration for two-curve wing airfoils obtained from optimizing the weights

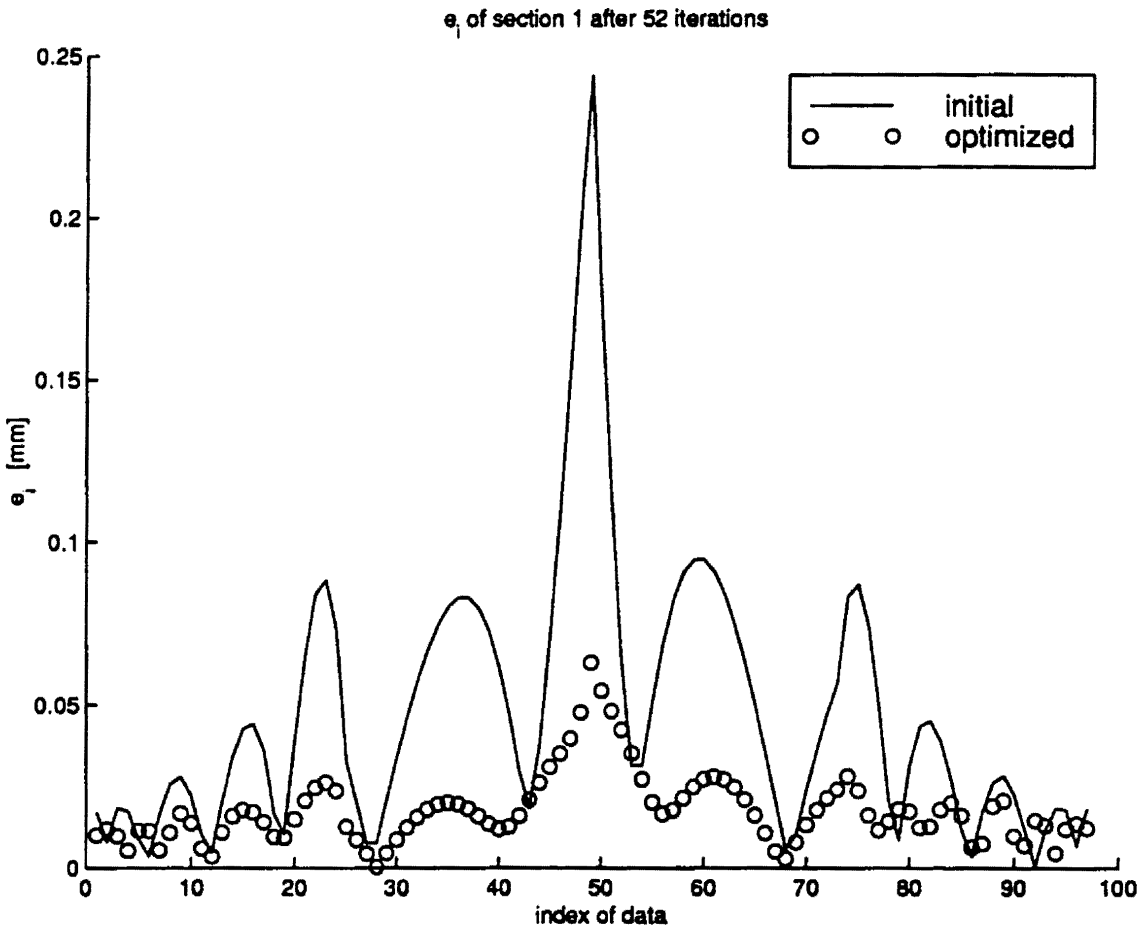


Figure 5.8: Error distribution of NACA airfoil of two-curve wing airfoils obtained from optimizing the weights.

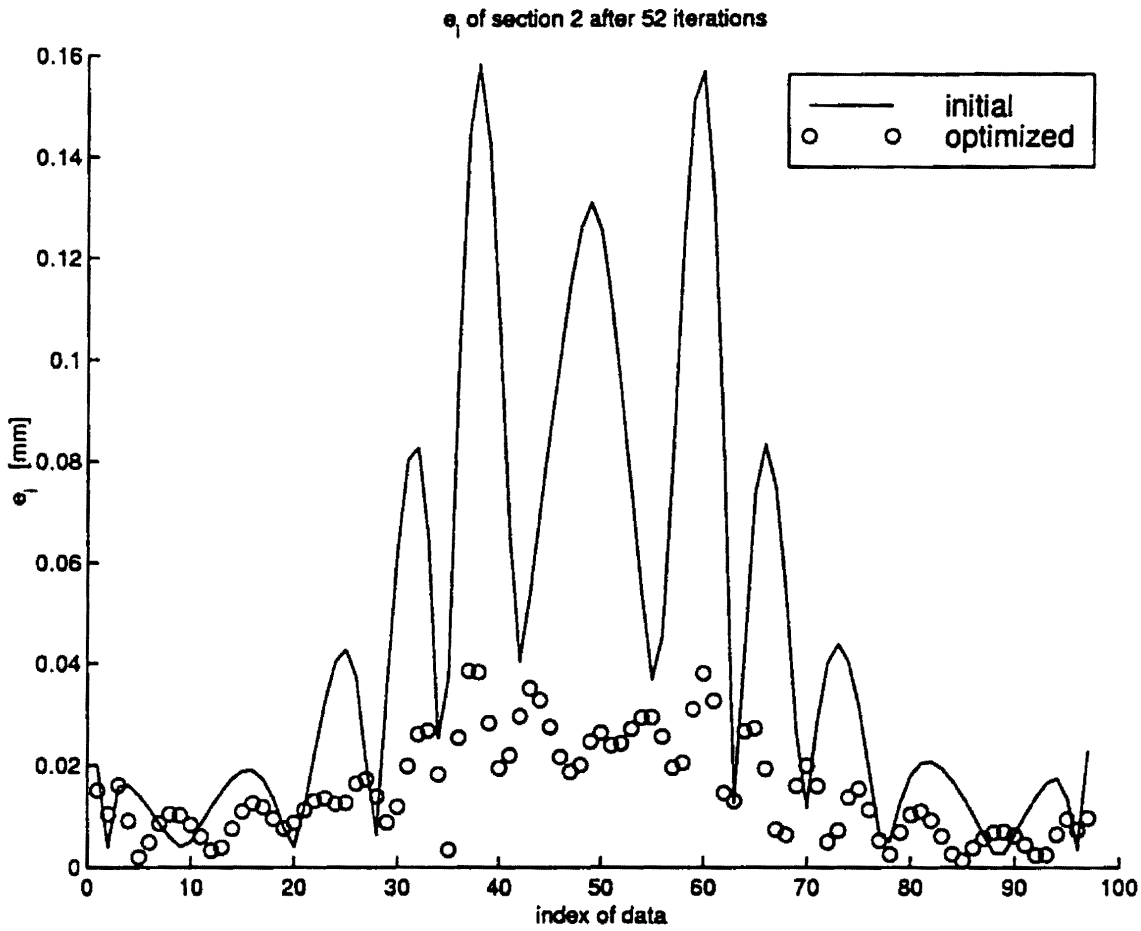


Figure 5.9: Error distribution of WTEA of two-curve wing airfoils obtained from optimizing the weights.

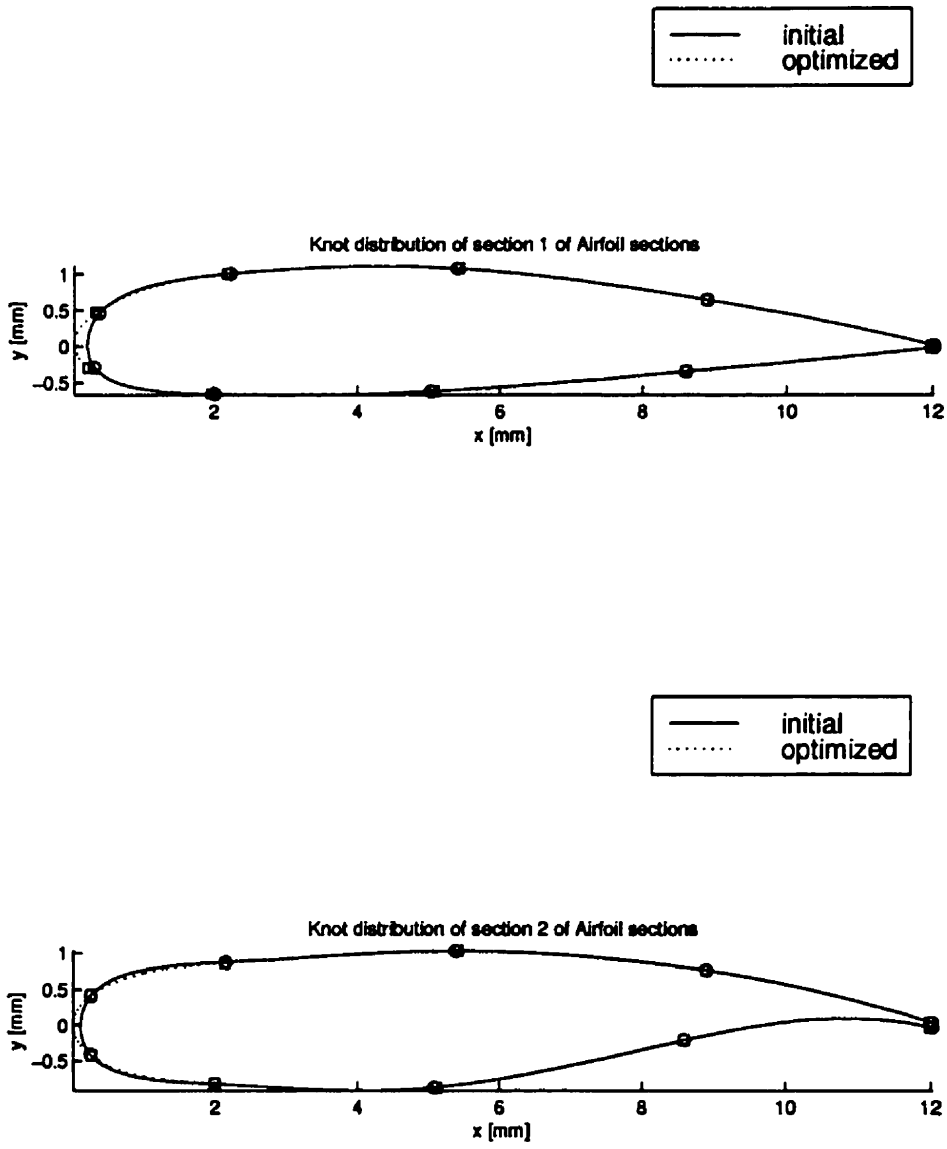


Figure 5.10: Shape of NACA 2415 (top) and WTEA (bottom) Before and After Optimization of Weights

5.2.3 Optimization of Parameterization

The sum of least-square error decreased from 0.771 [mm^2] to 0.240 [mm^2] in 2 iteration as shown in Figure 5.11. The largest distances between the data and the approximation curves decreased from 0.244 millimeters to 0.158 millimeters for NACA 2415 and from 0.158 millimeters to 0.080 millimeters for the WTEA as shown in Figure 5.12 and 5.13. The shape of the NACA 2415 and the WTEA, along with their junctions, are shown in Figure 5.14.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after two iteration.

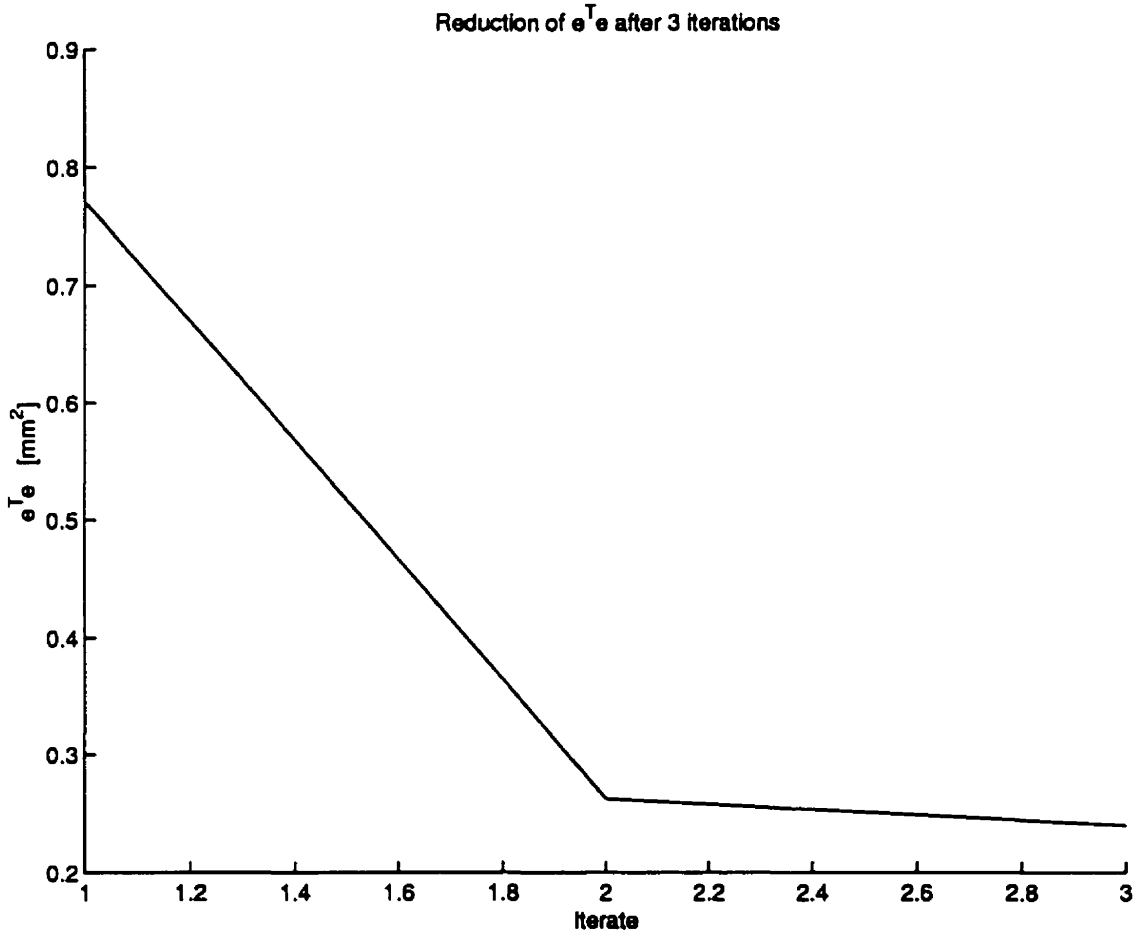


Figure 5.11: Reduction of error vs. number of iteration for two-curve wing airfoils obtained from optimizing the parameterization for data

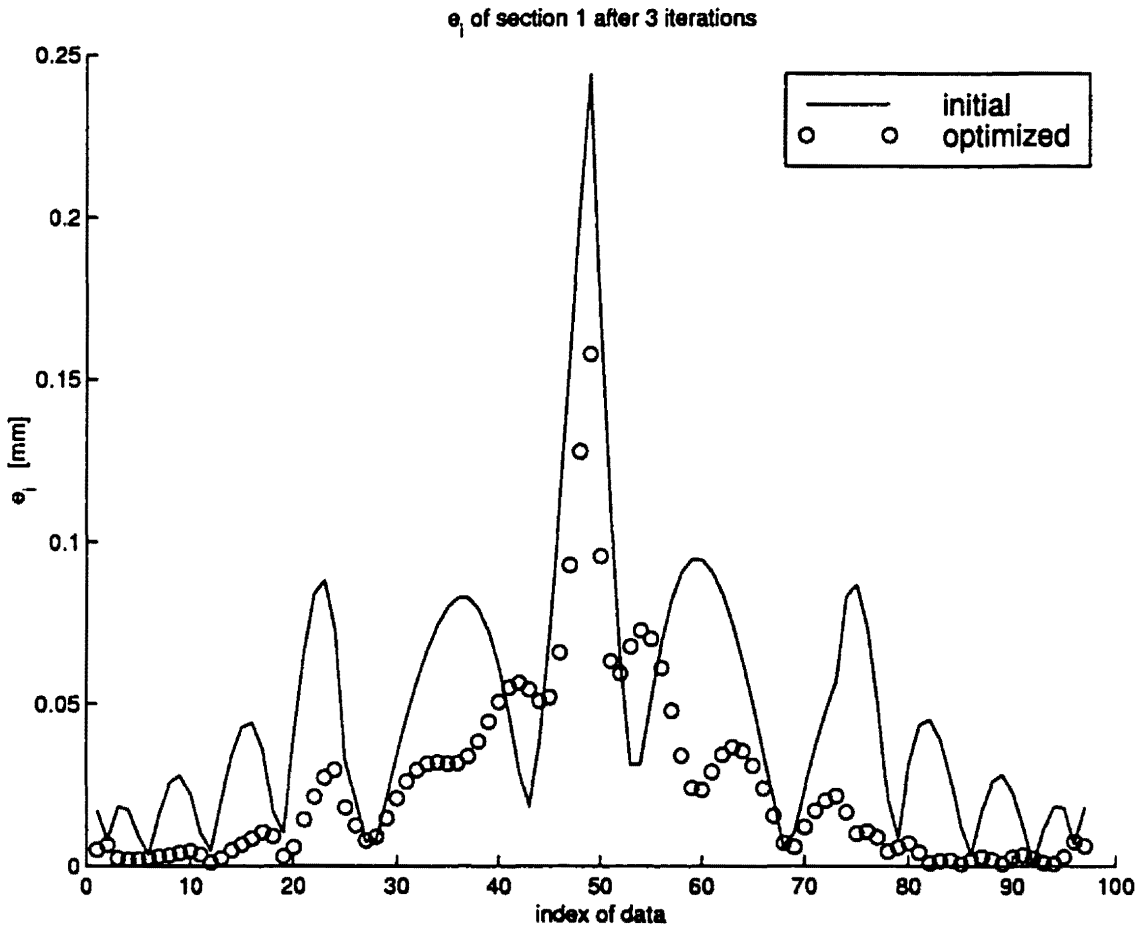


Figure 5.12: Error distribution of NACA airfoil of two-curve wing airfoils obtained from optimizing the parameterization for data.

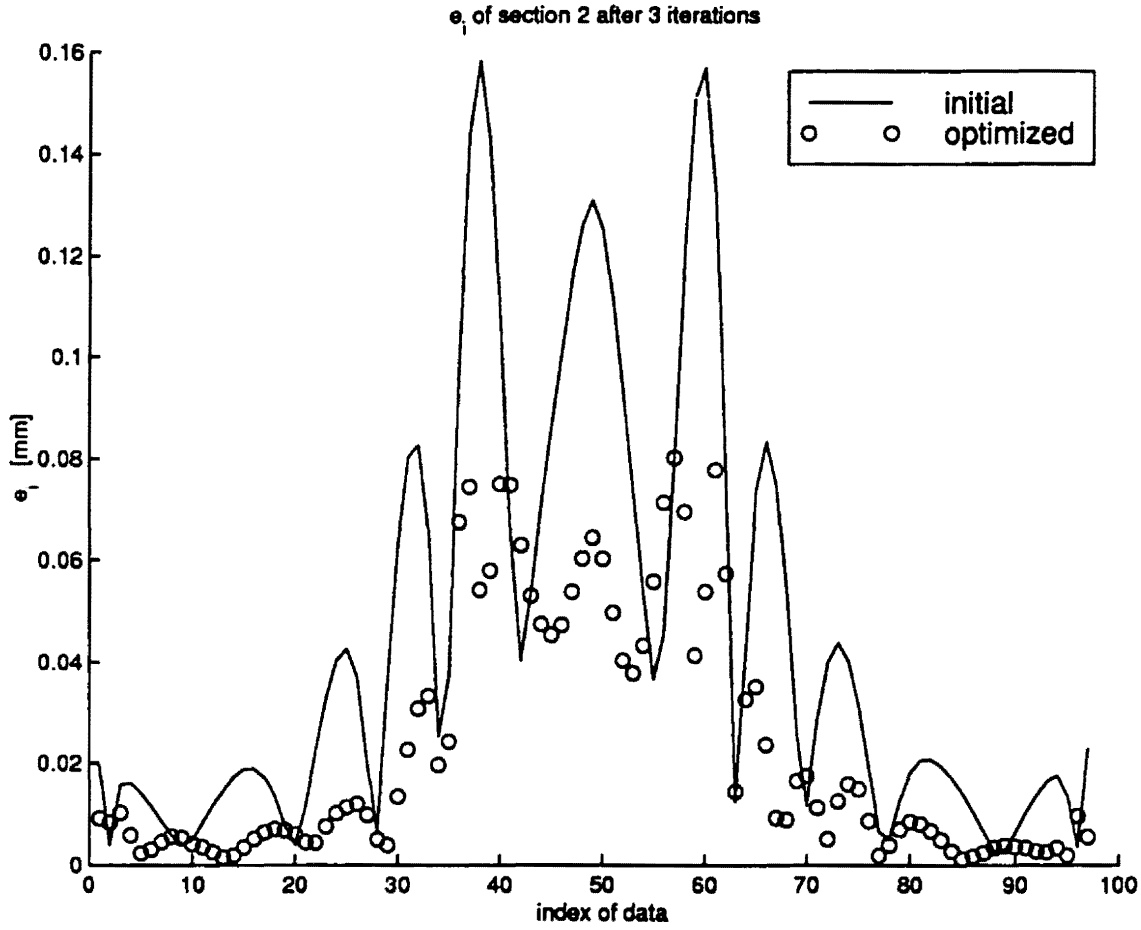


Figure 5.13: Error distribution of WTEA of two-curve wing airfoils obtained from optimizing the parameterization for data.

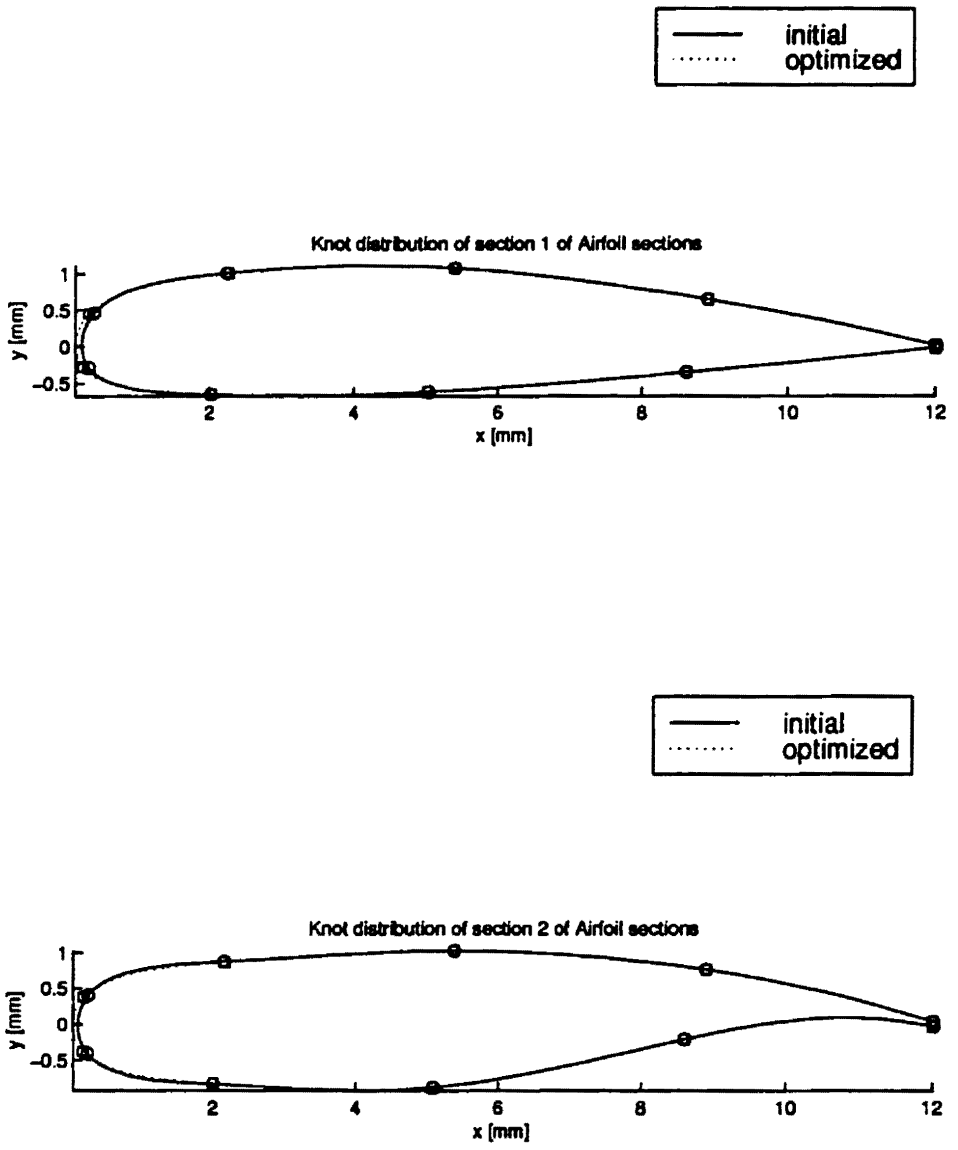


Figure 5.14: Shape of NACA 2415 (top) and WTEA (bottom) Before and After Optimization of Knots

5.2.4 Optimization of Knots and Weights

The sum of least-square error decreased from 0.771 [mm^2] to 0.006 [mm^2] in 42 iteration as shown in Figure 5.15. The largest distances between the data and the approximation curves decreased from 0.244 millimeters to 0.027 millimeters for NACA 2415 and from 0.158 millimeters to 0.014 millimeters for the WTEA as shown in Figure 5.16 and 5.17. The shape of the NACA 2415 and the WTEA, along with their junctions, are shown in Figure 5.18. Active knot constraint is $u_8 = u_9$. The change of weights before and after optimization is less than 5 percent.

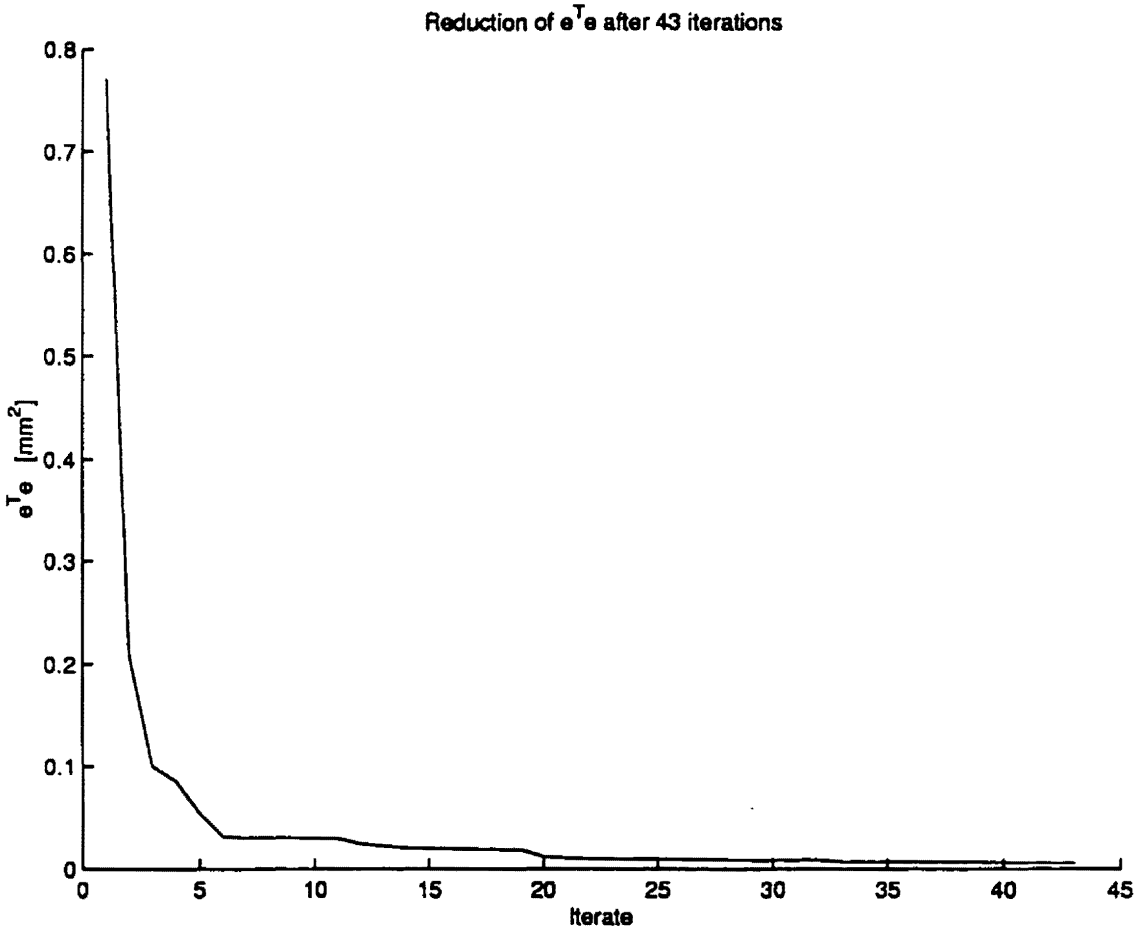


Figure 5.15: Reduction of error vs. number of iteration for two-curve wing airfoils obtained from optimizing the combination of knots and weights

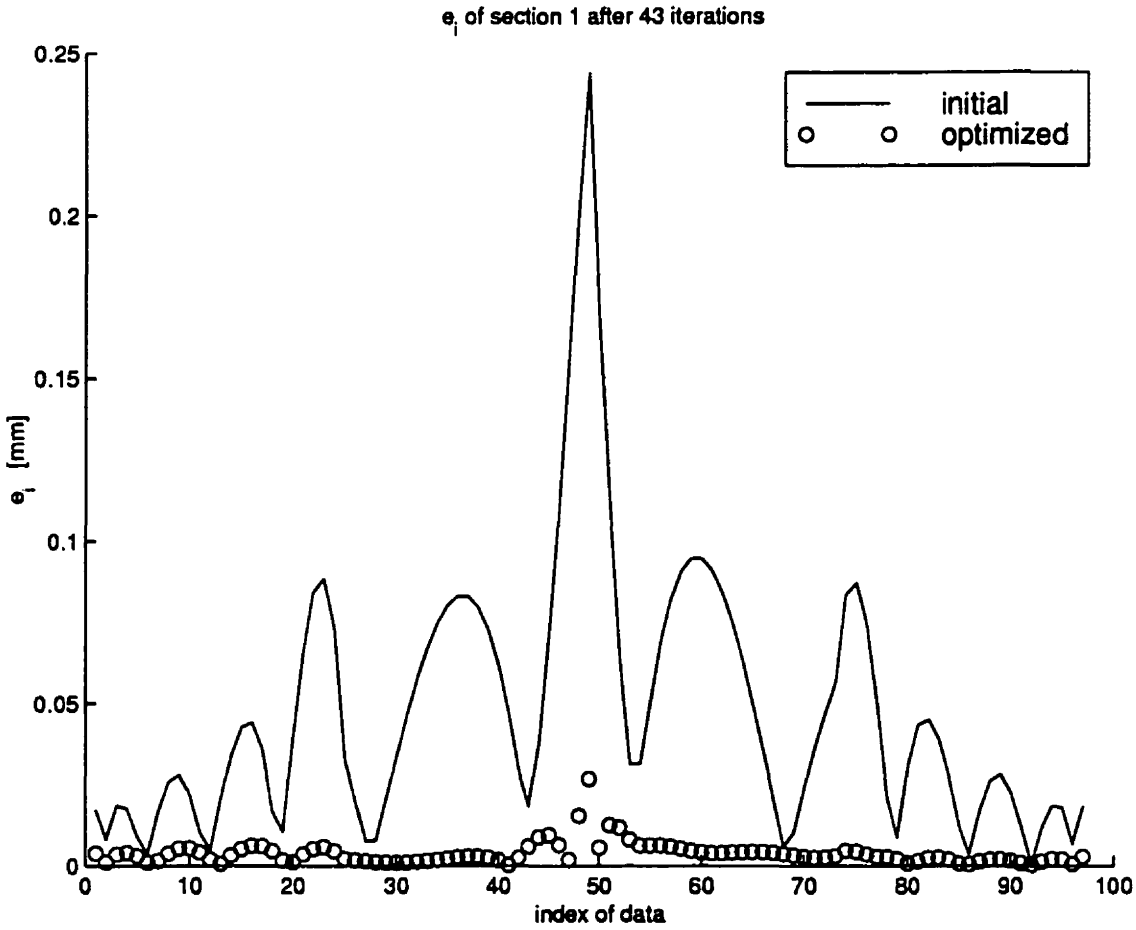


Figure 5.16: Error distribution of NACA airfoil of two-curve wing airfoils obtained from optimizing the combination of knots and weights.

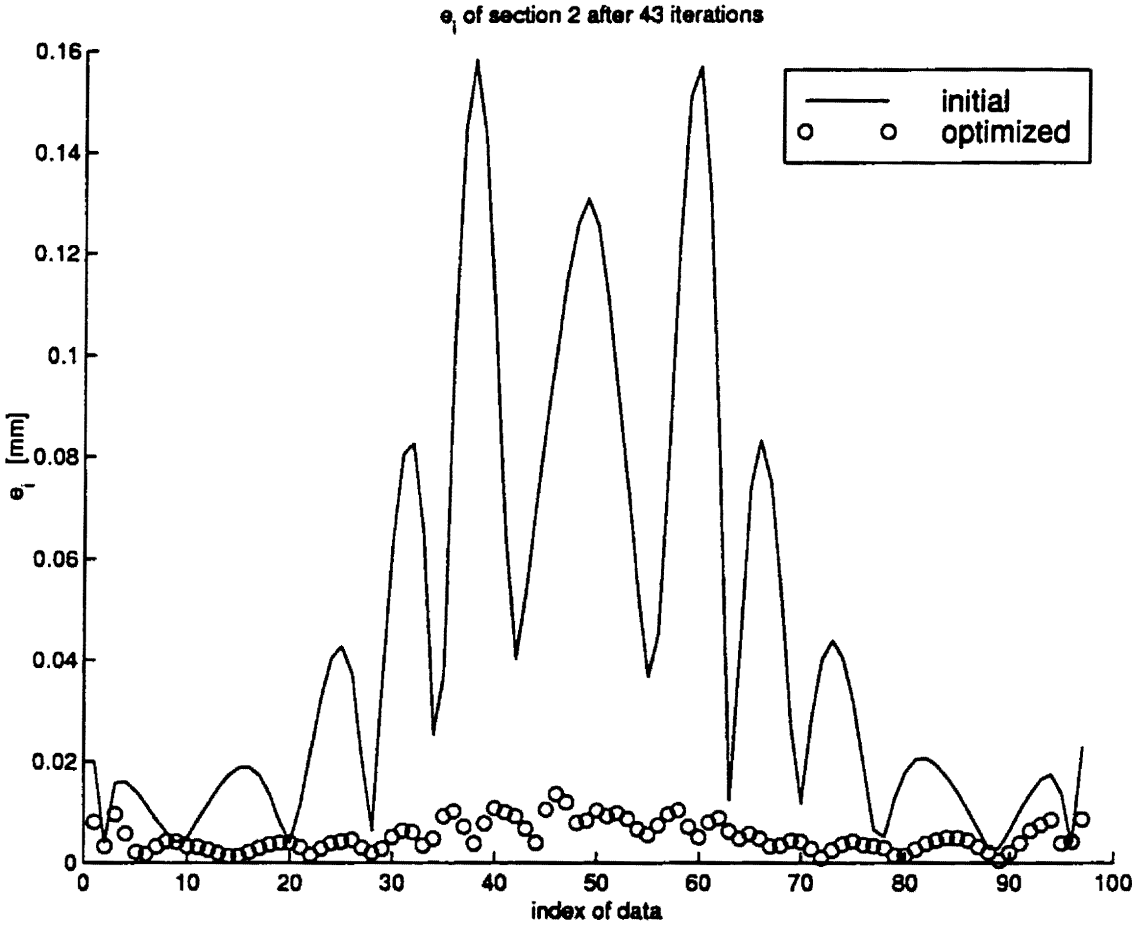


Figure 5.17: Error distribution of WTEA of two-curve wing airfoils obtained from optimizing the combination of knots and weights.

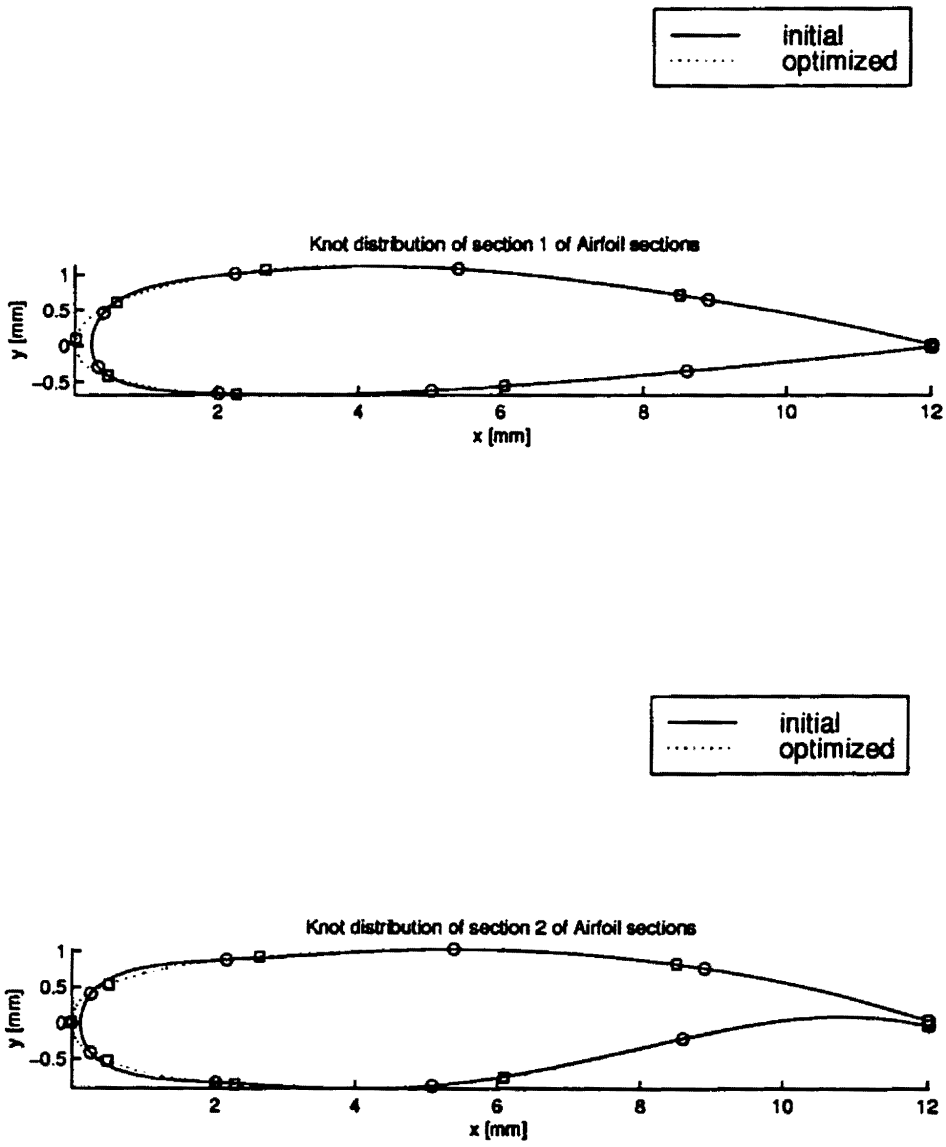


Figure 5.18: Shape of NACA 2415 (top) and WTEA (bottom) before and after optimization of combination of knots and weights.

5.2.5 Optimization of Knots, Weights, and Parameterization

The sum of least-square error decreased from 0.771 [mm^2] to 0.199 [mm^2] in 5 iteration as shown in Figure 5.19. The largest distances between the data and the approximation curves decreased from 0.244 millimeters to 0.143 millimeters for NACA 2415 and from 0.158 millimeters to 0.073 millimeters for the WTEA as shown in Figure 5.20 and 5.21. The shape of the NACA 2415 and the WTEA, along with their junctions, are shown in Figure 5.22.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after five iteration.

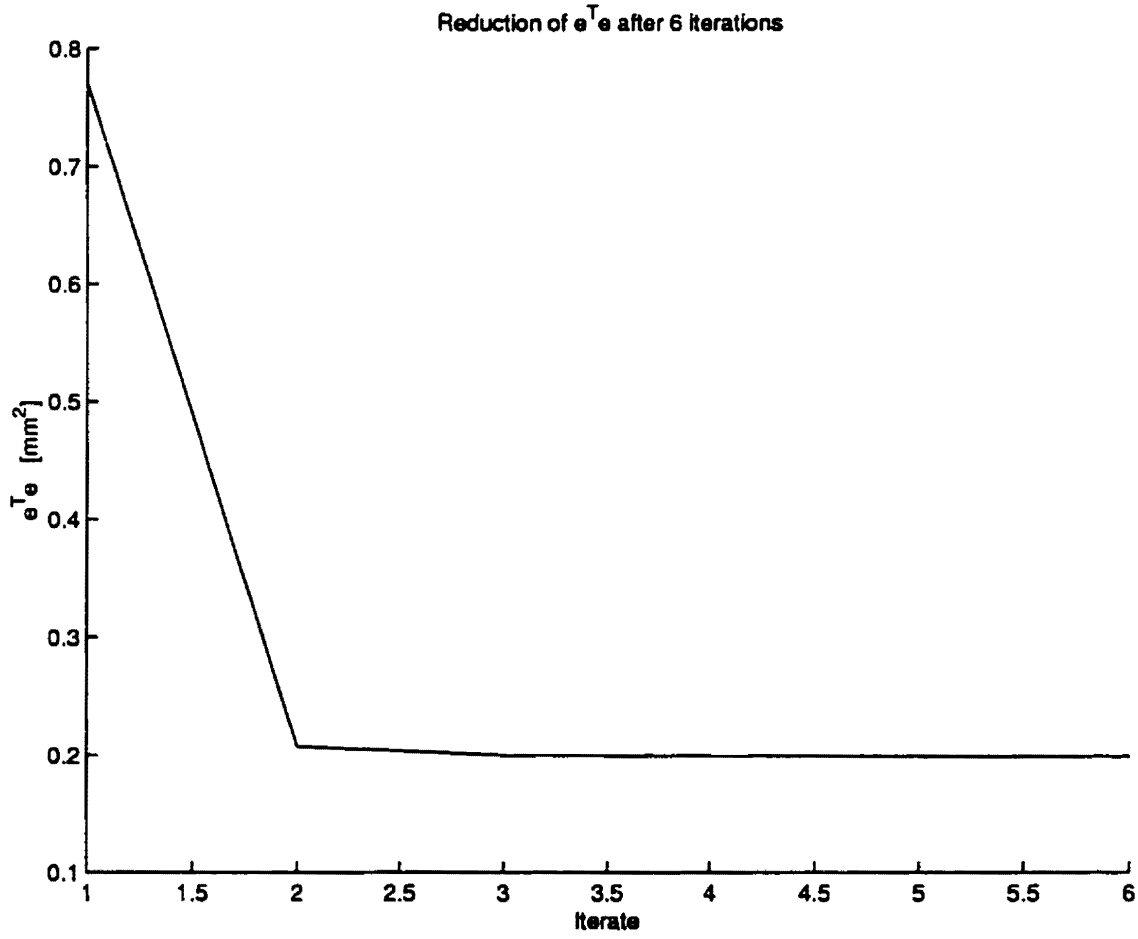


Figure 5.19: Reduction of error vs. number of iteration for two-curve wing airfoils obtained from optimizing the combination of knots, weights, and parameterization for data

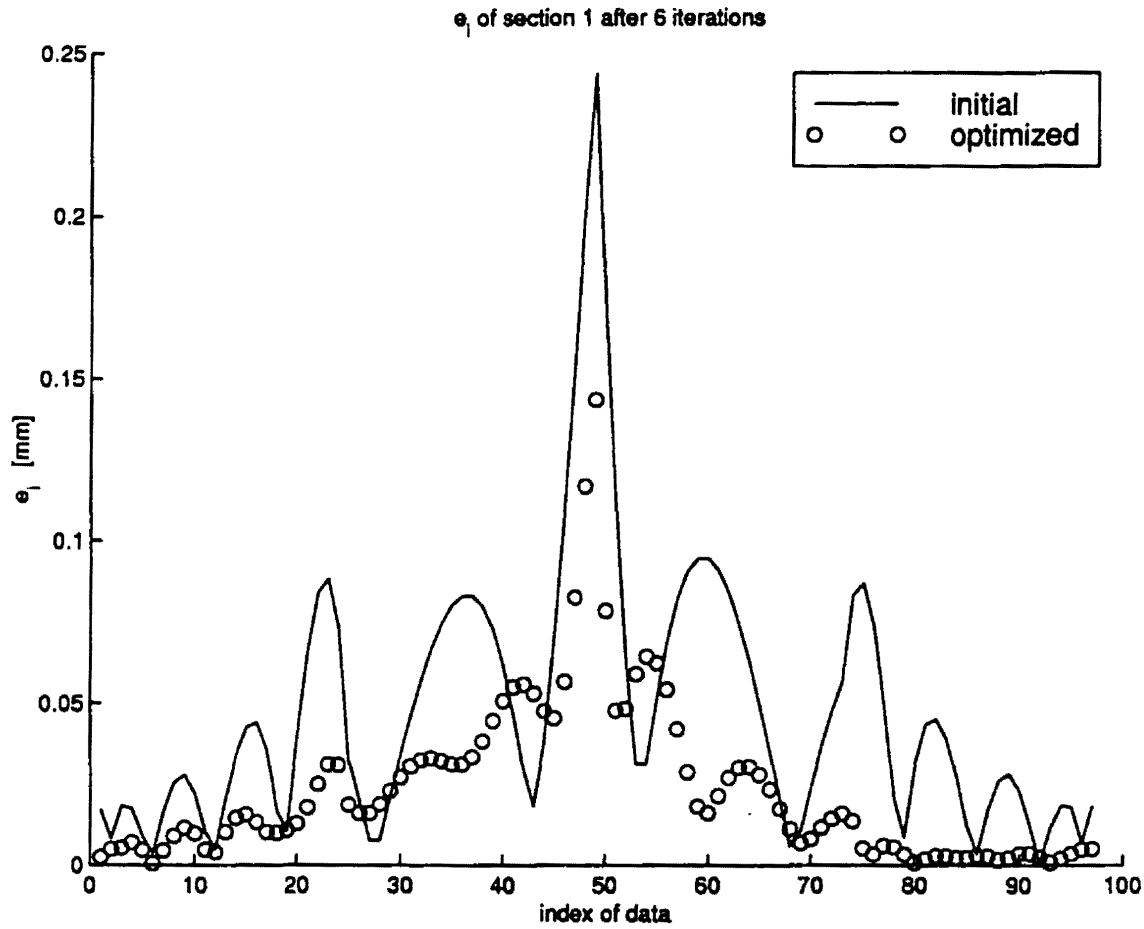


Figure 5.20: Error distribution of NACA airfoil of two-curve wing airfoils obtained from optimizing the combination of knots, weights, and parameterization for data.

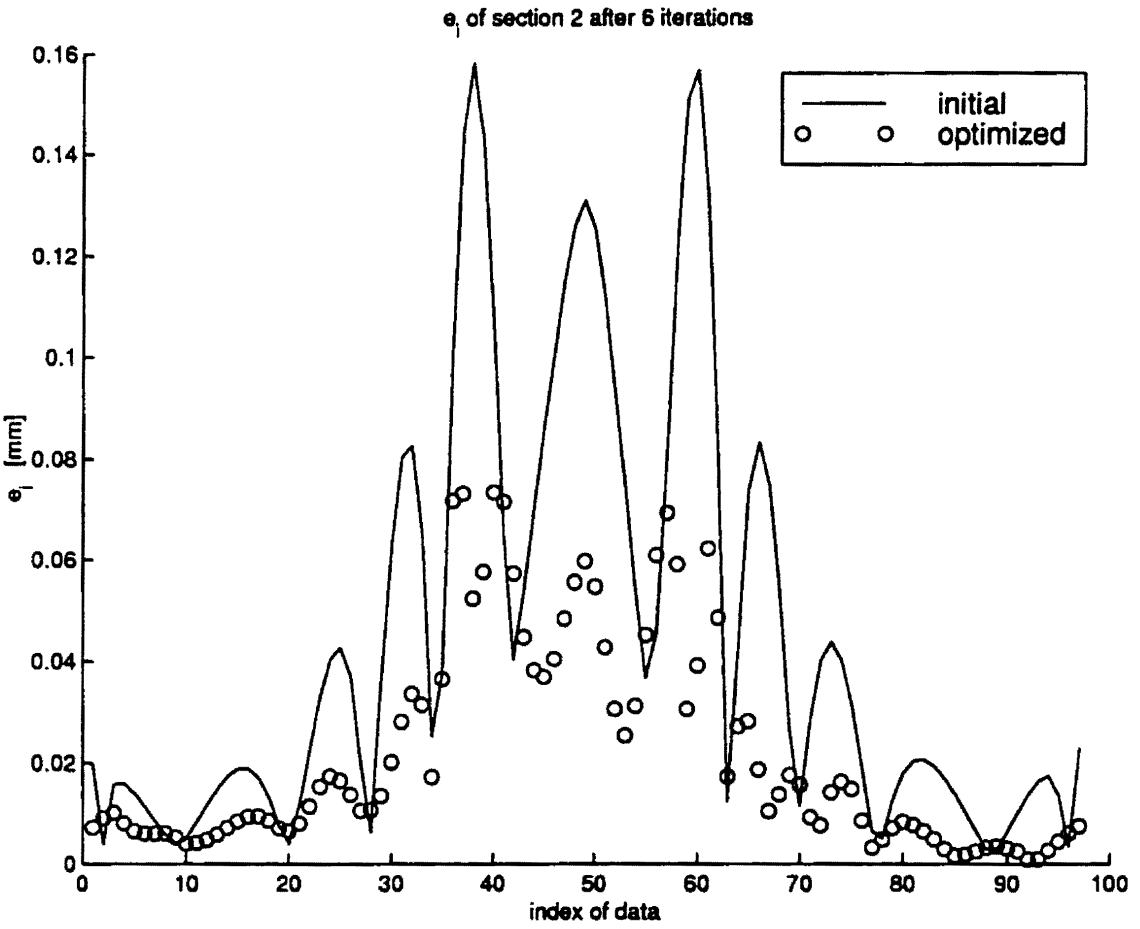


Figure 5.21: Error distribution of WTEA of two-curve wing airfoils obtained from optimizing the combination of knots, weights, and parameterization for data.

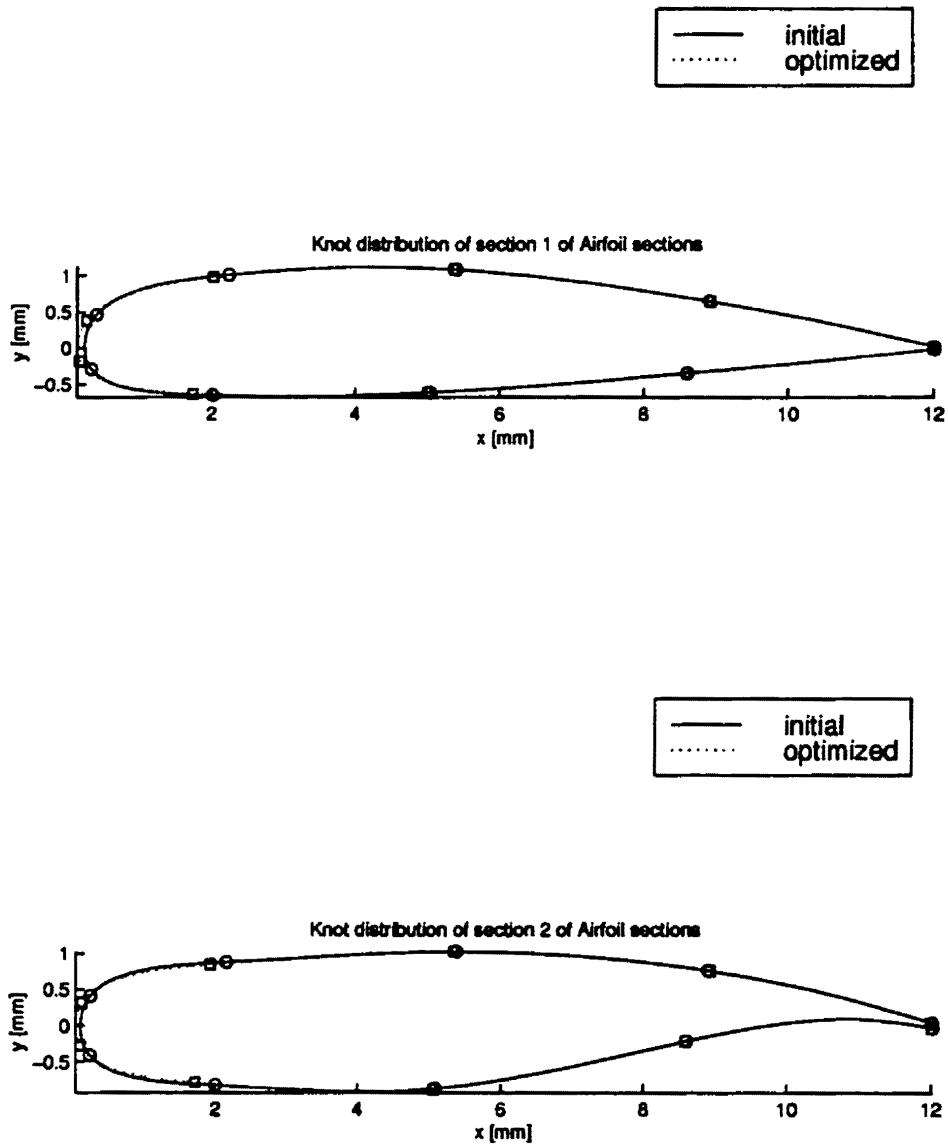


Figure 5.22: Shape of NACA 2415 (top) and WTEA (bottom) before and after optimization of combination of knots, weights, and parameterization for data.

5.3 Compressor Blades

Figure 5.23 shows three airfoils defining the blade of an axial compressor. The blade is a proprietary of the Concepts ETI, Inc. The airfoils are shown in four views, which are the XZ-plane at the upper left box, the isometric view at the upper right box, the XY-plane at the lower left box, and the YX-plane at the lower right box. The axis of rotation of the compressor is the Z-axis. The relative position of the airfoils along the radial, i.e. the X-axis, direction is shown in the XZ-plane view and the XY-plane view. In these views, the left most airfoil is at the hub of the shaft of the compressor and the right most airfoil is at the tip of the blade. The location of the airfoil in the middle is at $r = (1 - 0.605)\, r_{\text{hub}} + 0.605\, r_{\text{tip}}$.

5.3.1 Optimization of Knots

The sum of least-square error decreased from 27.5 [mm^2] to 1.53 [mm^2] in 51 iteration as shown in Figure 5.24. The largest distances between the data and the approximation curves decreased from 1.16 millimeters to 0.359 millimeters for hub airfoil, from 1.25 millimeters to 0.229 millimeters for mid section airfoil, and from 1.19 millimeters to 0.268 millimeters for tip airfoil as shown in Figure 5.25, 5.26, and 5.27. The shape of the airfoils, along with the distribution of knots, are shown in Figure 5.28, 5.29, and 5.30. All constraints are inactive.

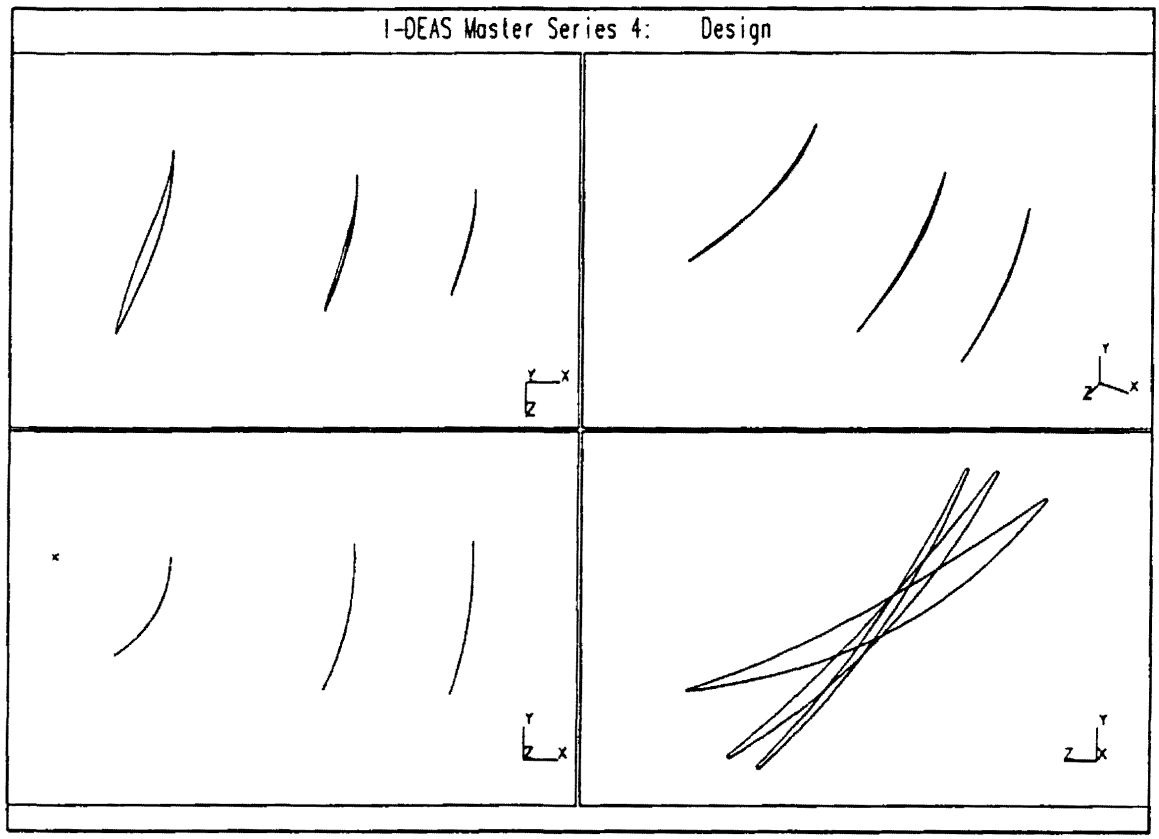


Figure 5.23: Data of Compressor

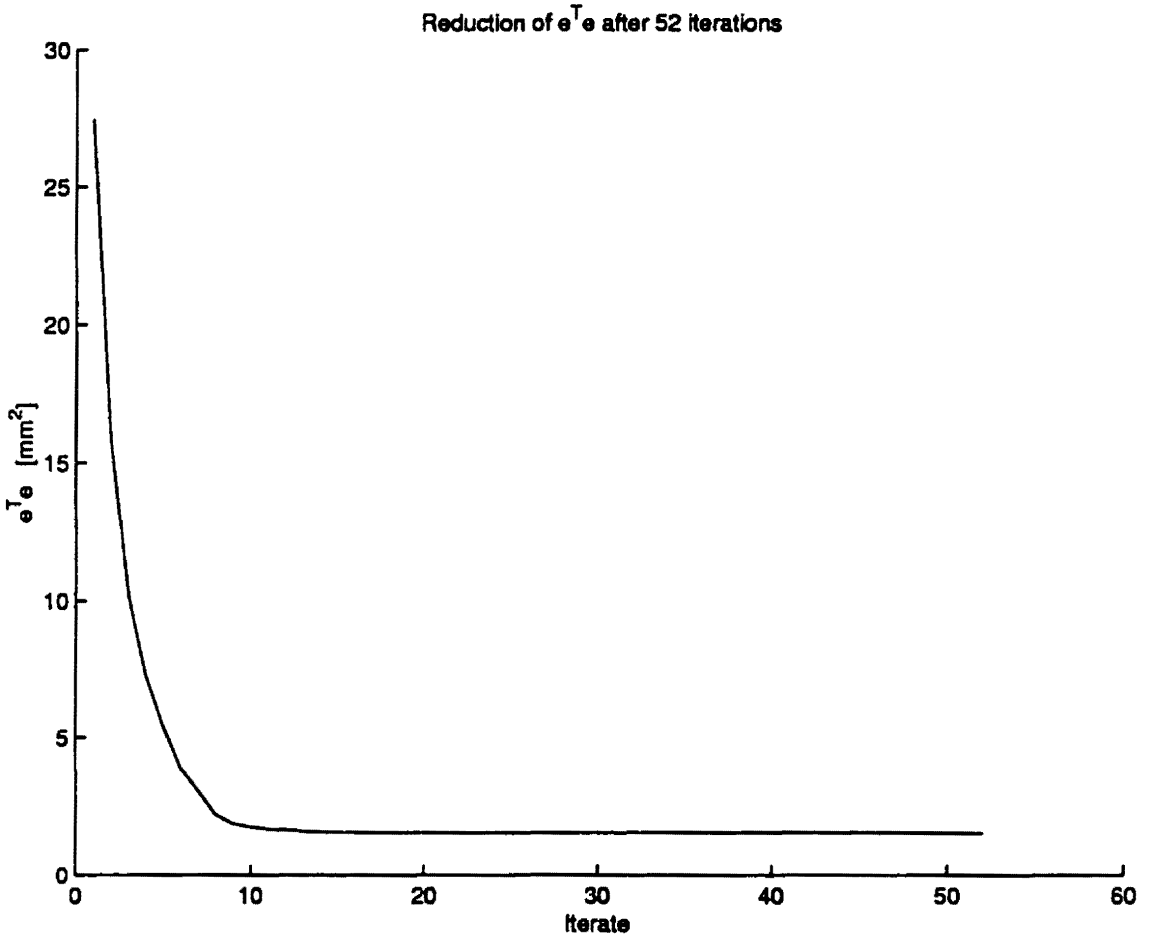


Figure 5.24: Reduction of error vs. number of iteration for three-curve compressor blade's airfoils obtained from optimizing the knots

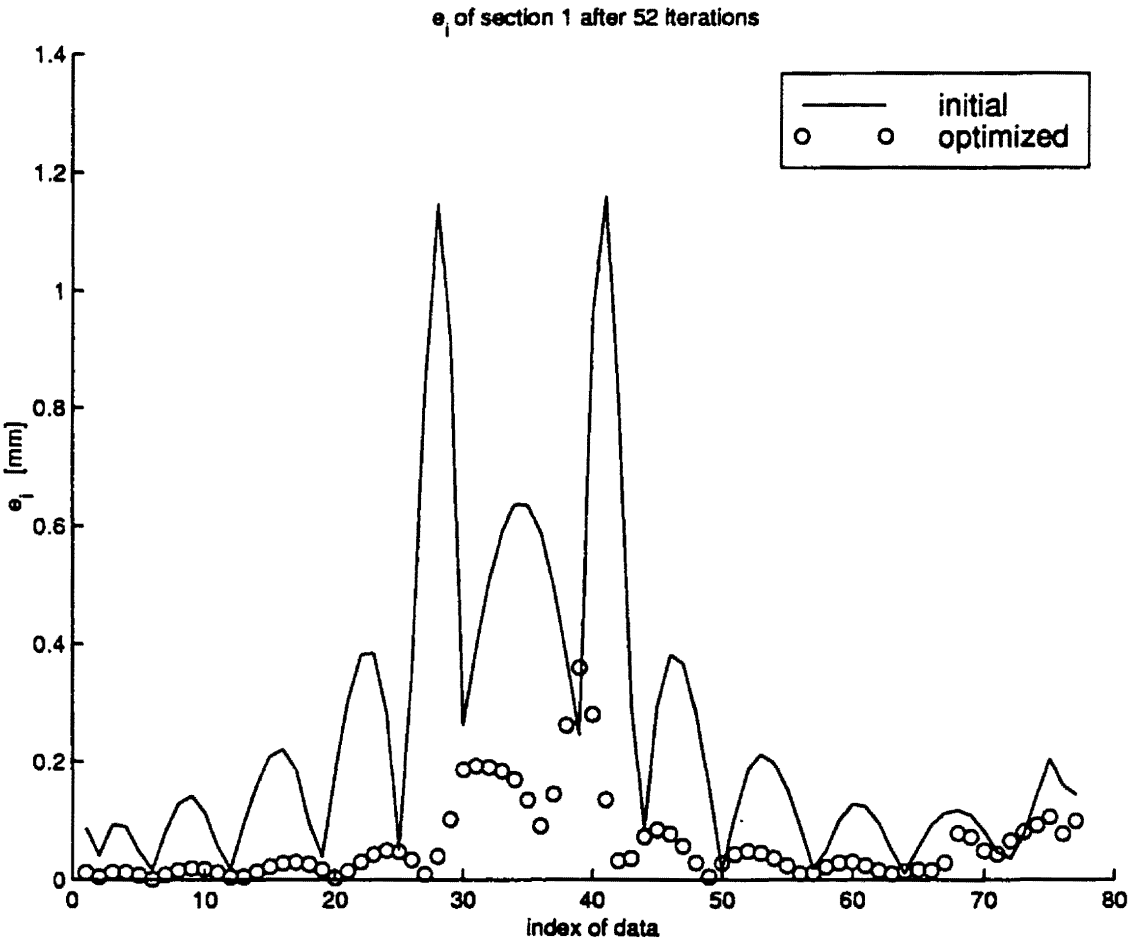


Figure 5.25: Error distribution of hub airfoil of three-curve compressor blade obtained from optimizing the knots.

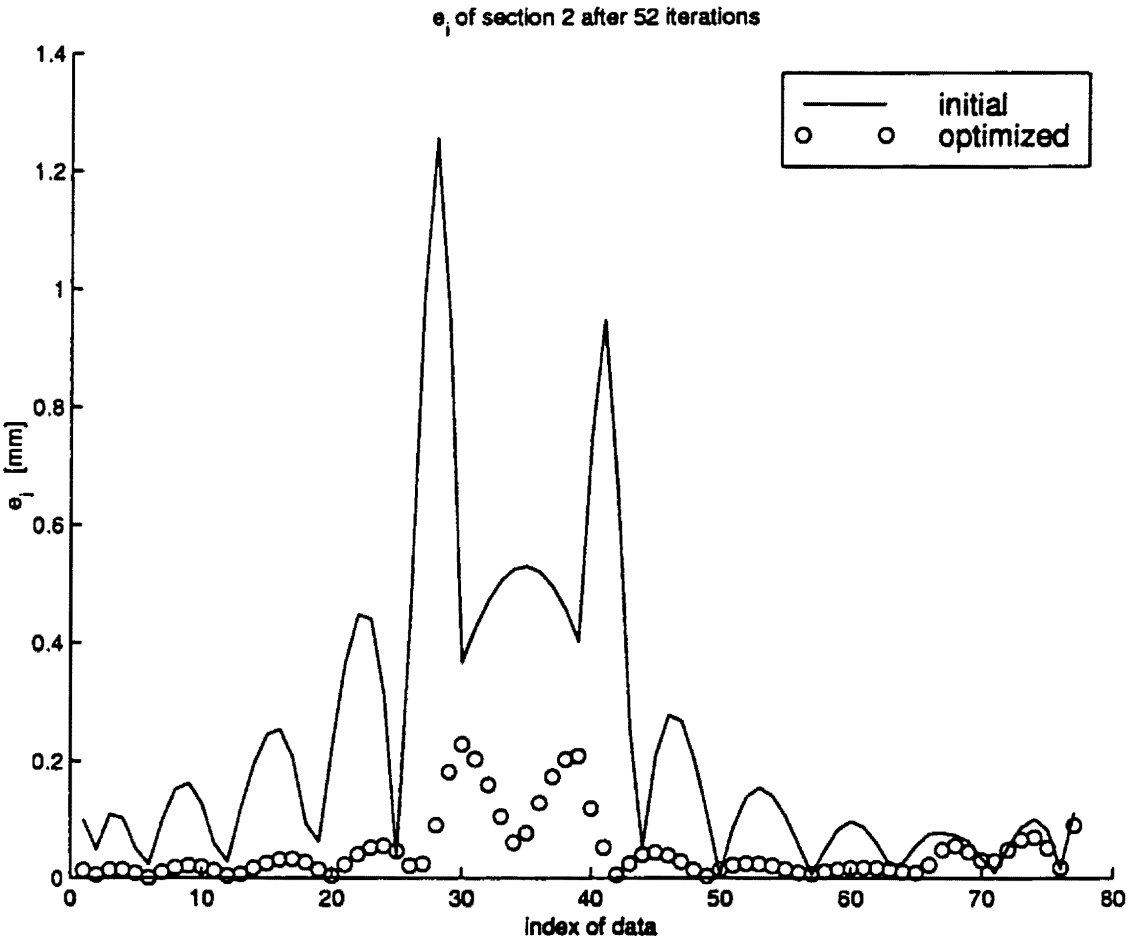


Figure 5.26: Error distribution of midsection airfoil of three-curve compressor blade obtained from optimizing the knots.

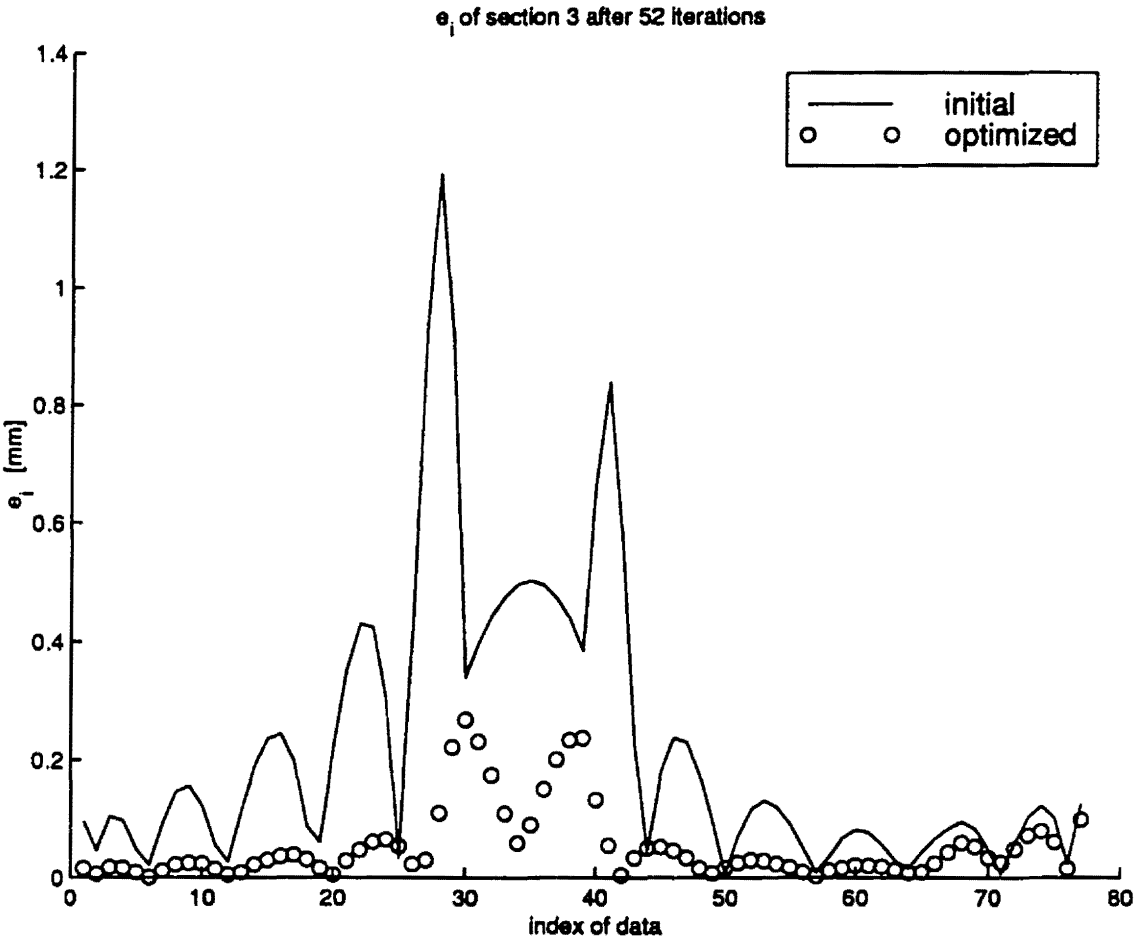


Figure 5.27: Error distribution of tip airfoil of three-curve compressor blade obtained from optimizing the knots.

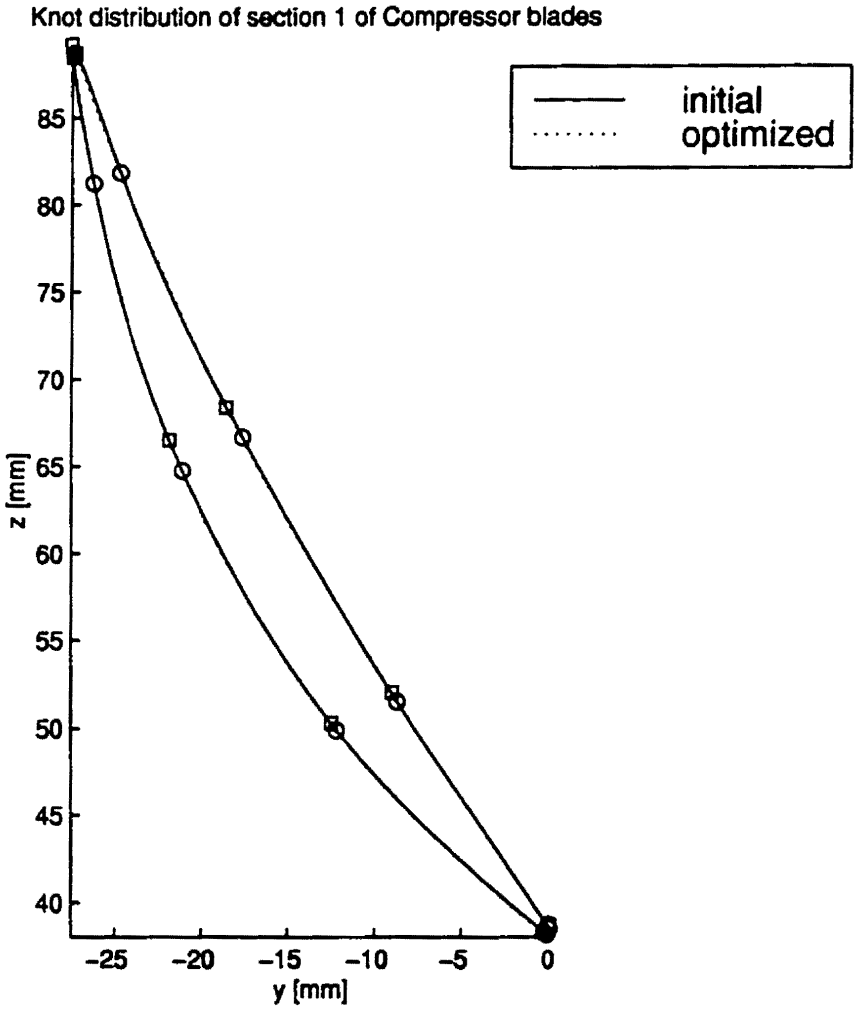


Figure 5.28: Fitted curves of hub airfoil of compressor blade obtained from optimizing knots.

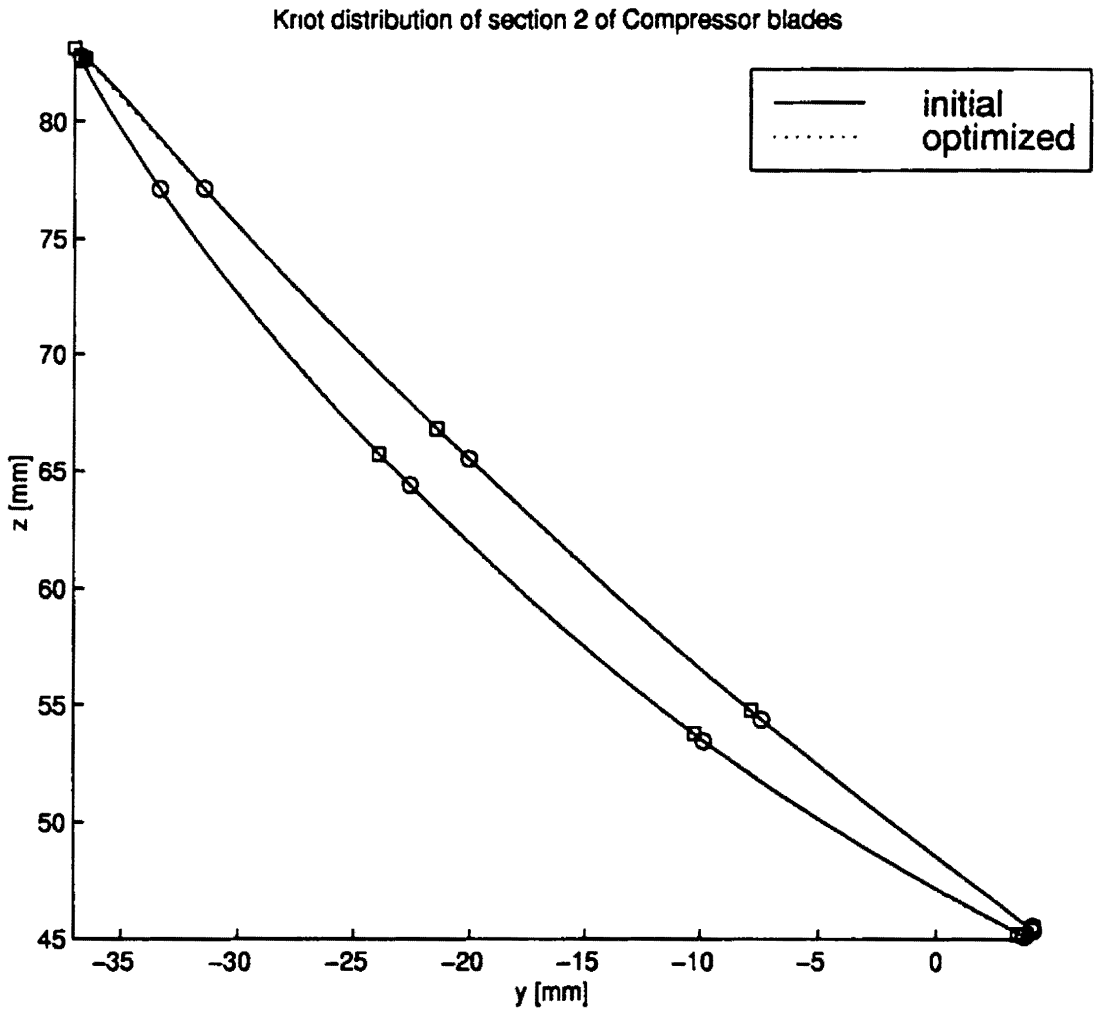


Figure 5.29: Fitted curves of mid section airfoil of compressor blade obtained from optimizing knots.

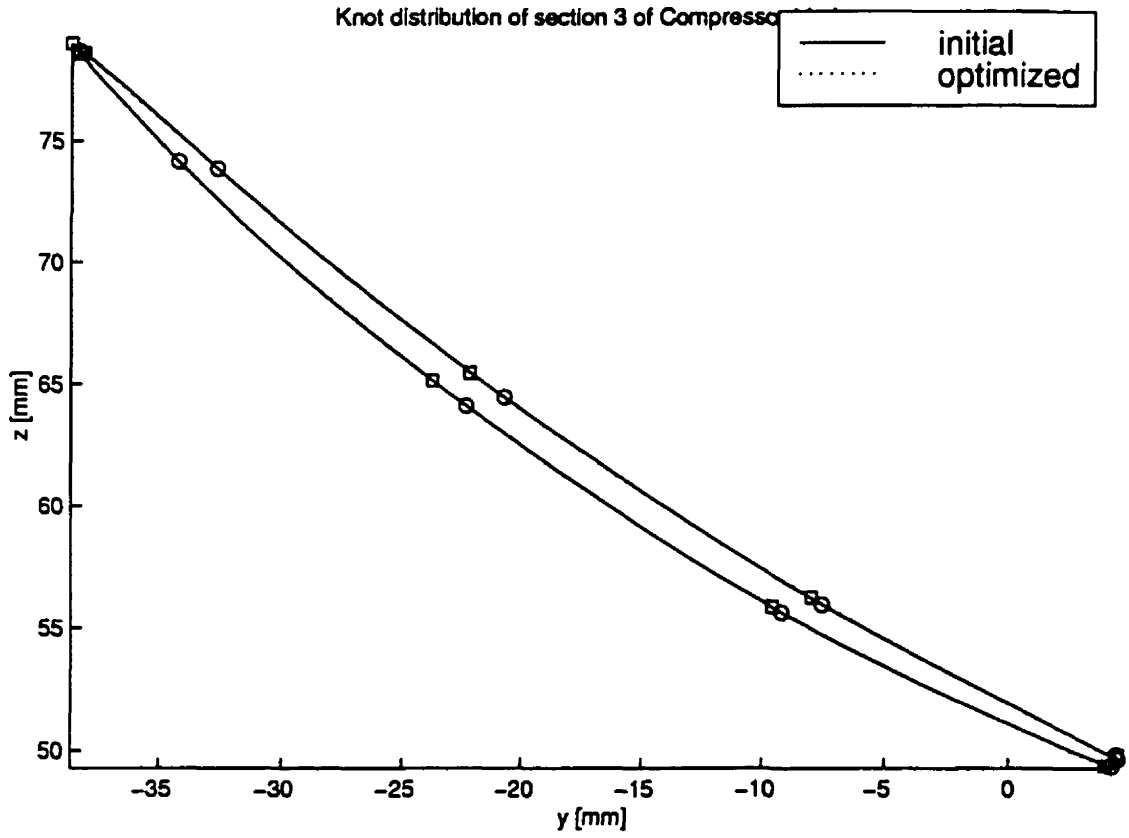


Figure 5.30: Fitted curves of tip airfoil of compressor blade obtained from optimizing knots.

5.3.2 Optimization of Weights

The sum of least-square error decreased from 27.5 [mm^2] to 3.45 [mm^2] in 51 iteration as shown in Figure 5.31. The largest distances between the data and the approximation curves decreased from 1.16 millimeters to 0.485 millimeters for hub airfoil, from 1.25 millimeters to 0.424 millimeters for mid section airfoil, and from 1.19 millimeters to 0.430 millimeters for tip airfoil as shown in Figures 5.32, 5.33, and 5.34, respectively. The shape of the sections, along with their junctions, are shown in Figures 5.35, 5.36, and 5.37. All constraints are inactive.

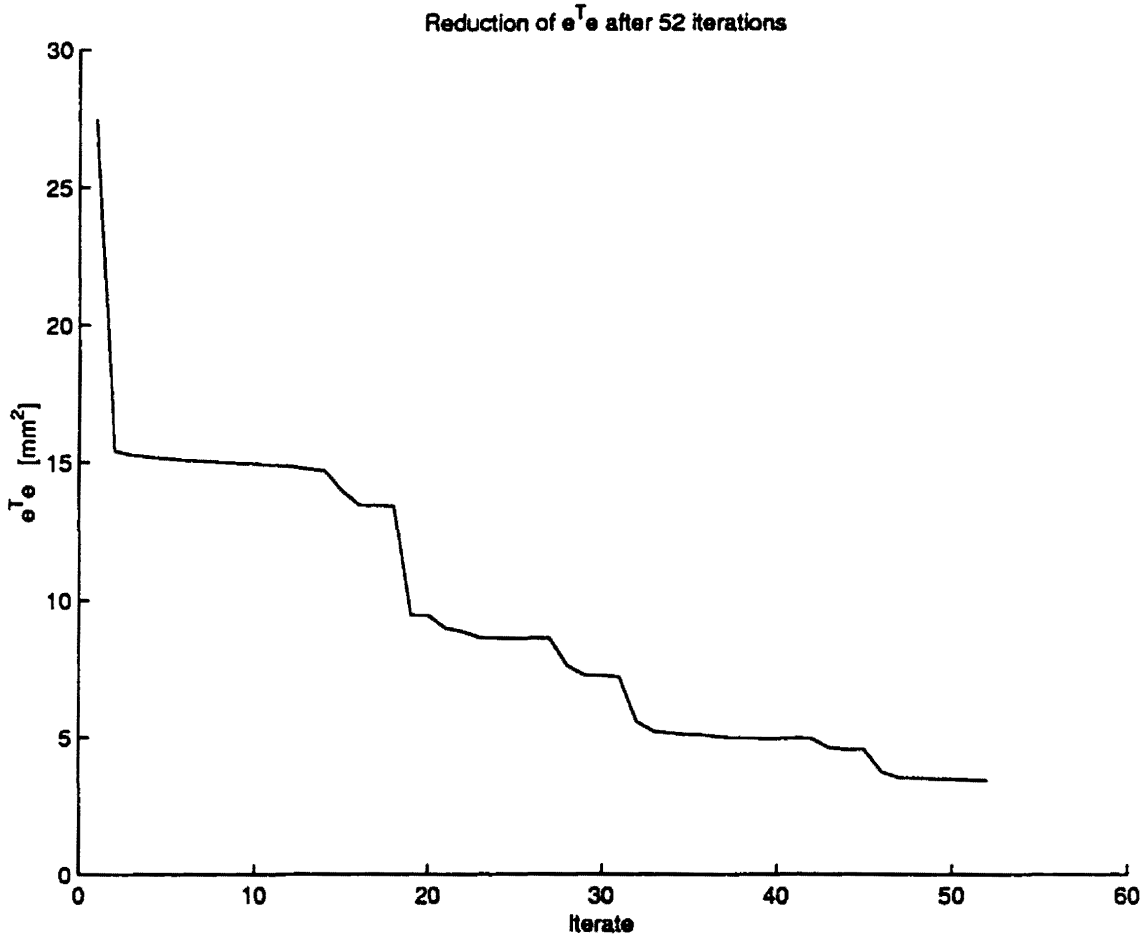


Figure 5.31: Reduction of error vs. number of iteration for three-curve compressor blade's airfoils obtained from optimizing the weights

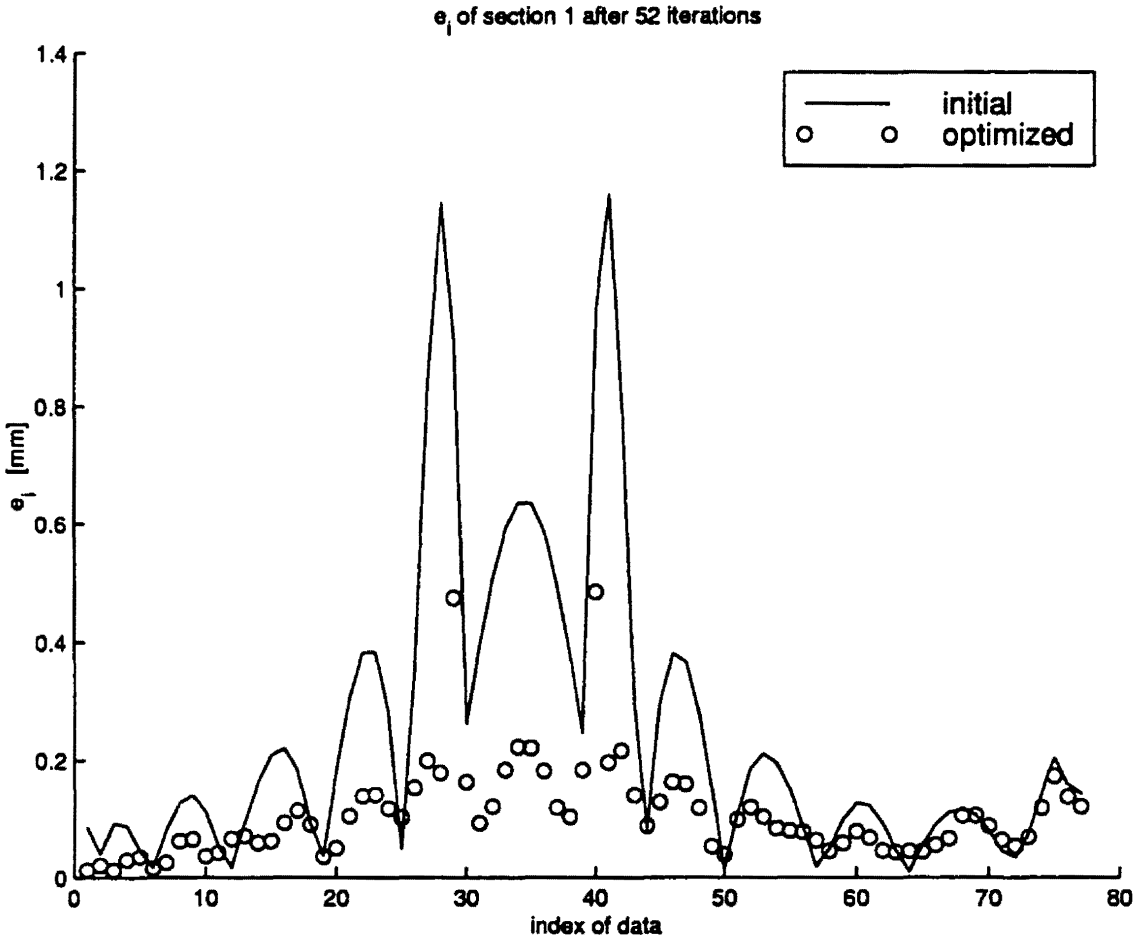


Figure 5.32: Error distribution of hub airfoil of three-curve compressor blade obtained from optimizing the weights.

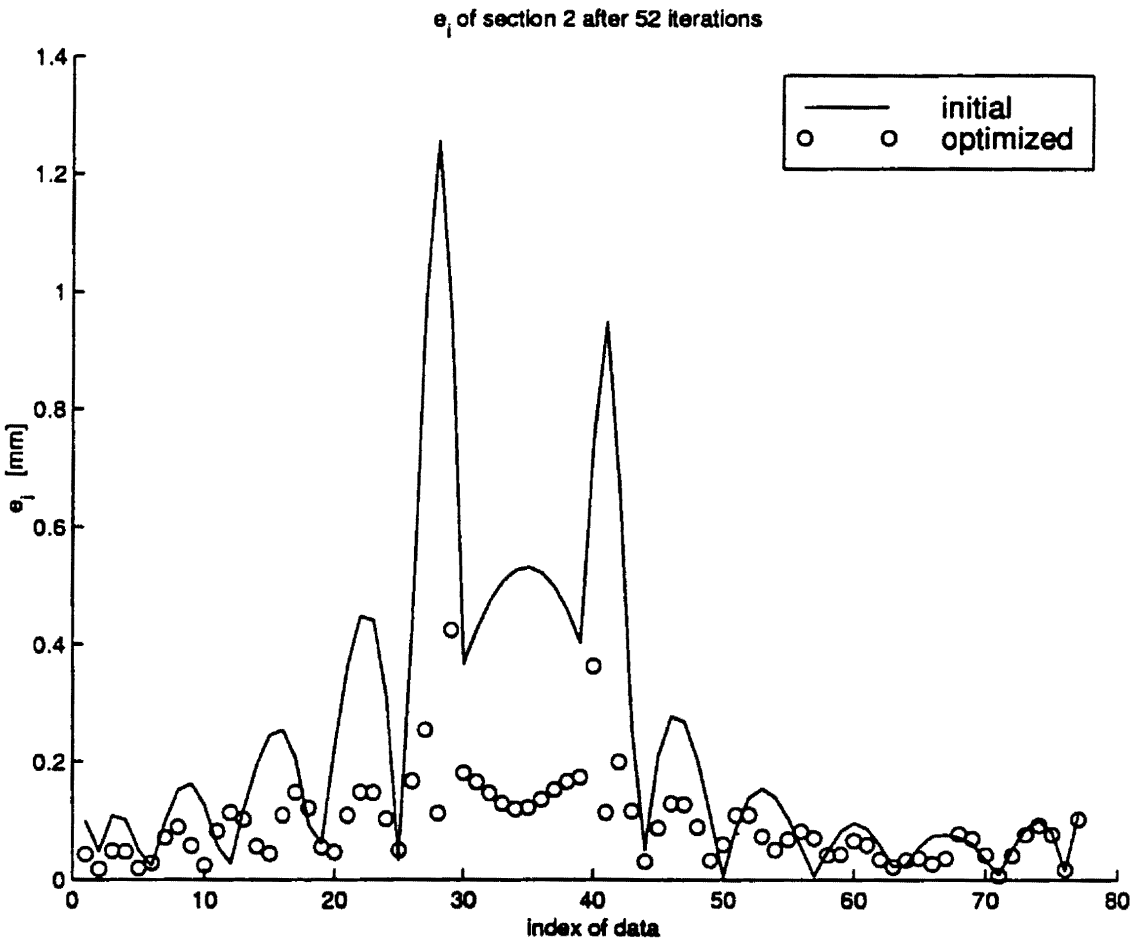


Figure 5.33: Error distribution of midsection airfoil of three-curve compressor blade obtained from optimizing the weights.

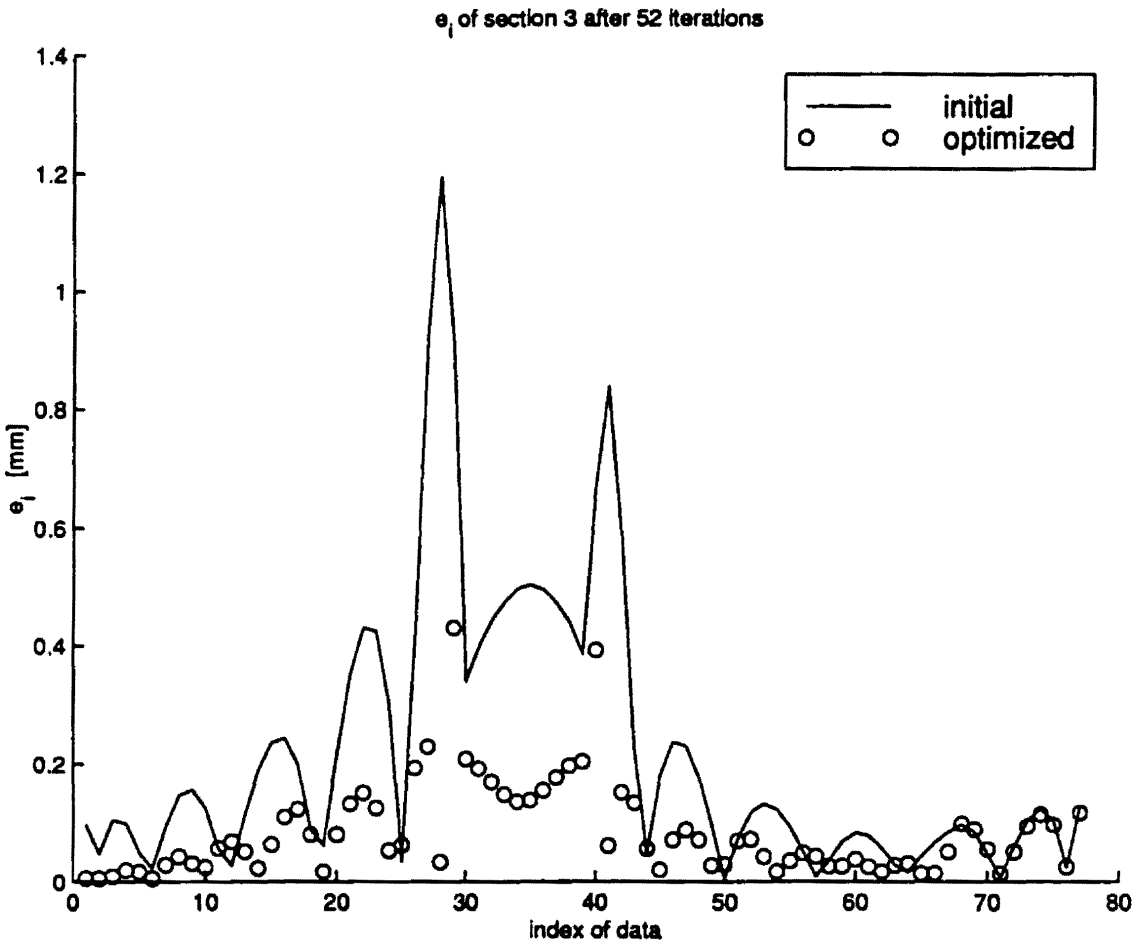


Figure 5.34: Error distribution of tip airfoil of three-curve compressor blade obtained from optimizing the weights.

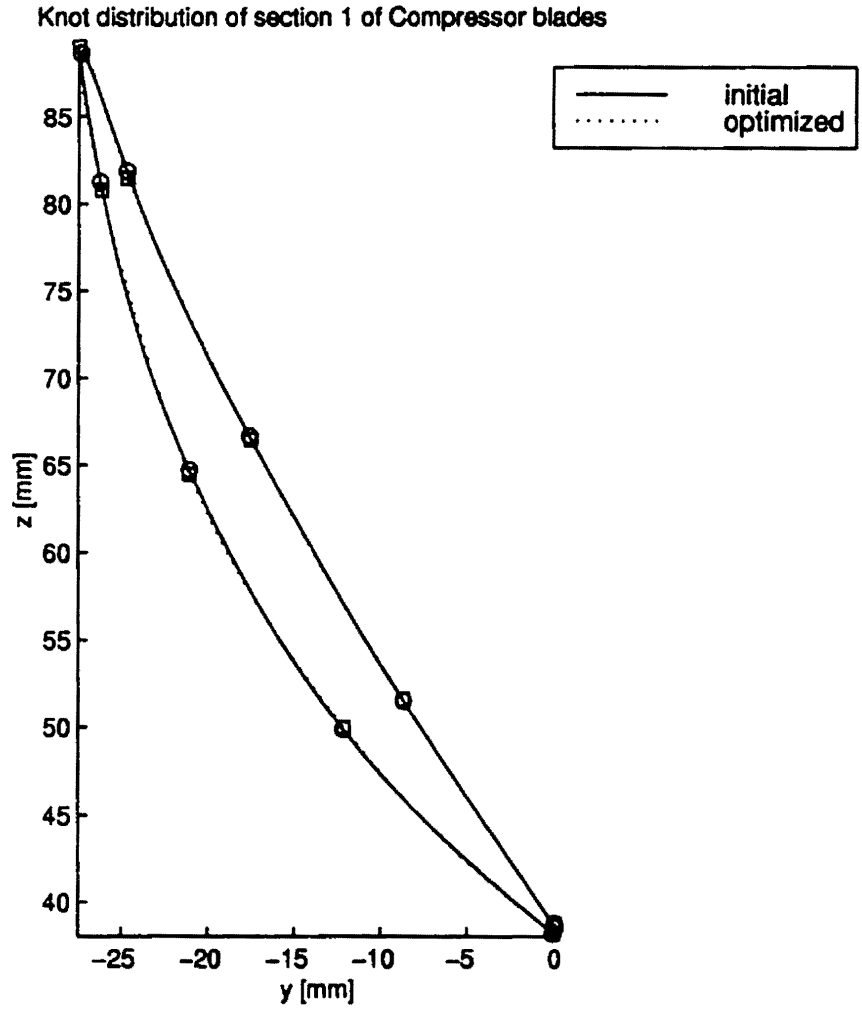


Figure 5.35: Fitted curves of hub airfoil of compressor blade obtained from optimizing weights.

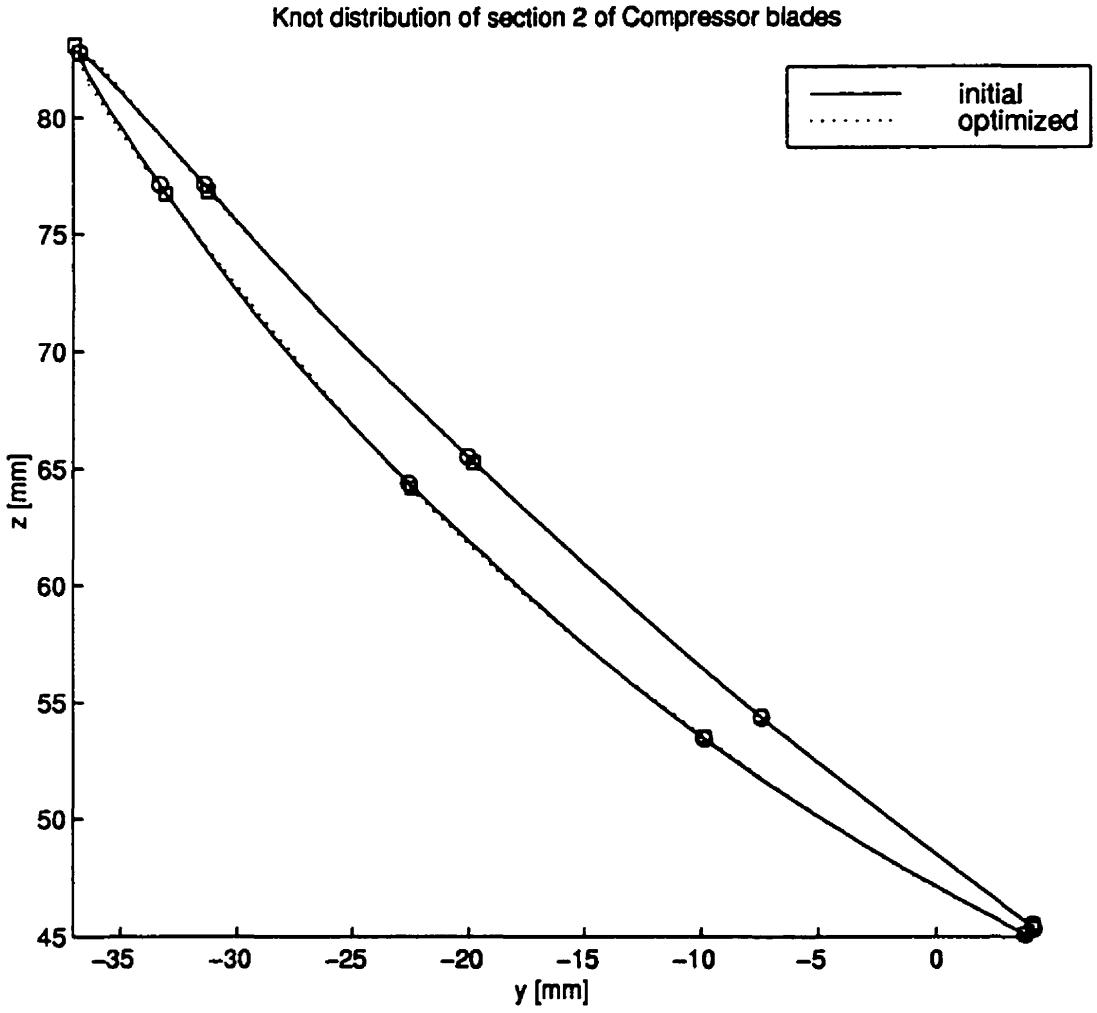


Figure 5.36: Fitted curves of mid section airfoil of compressor blade obtained from optimizing weights.

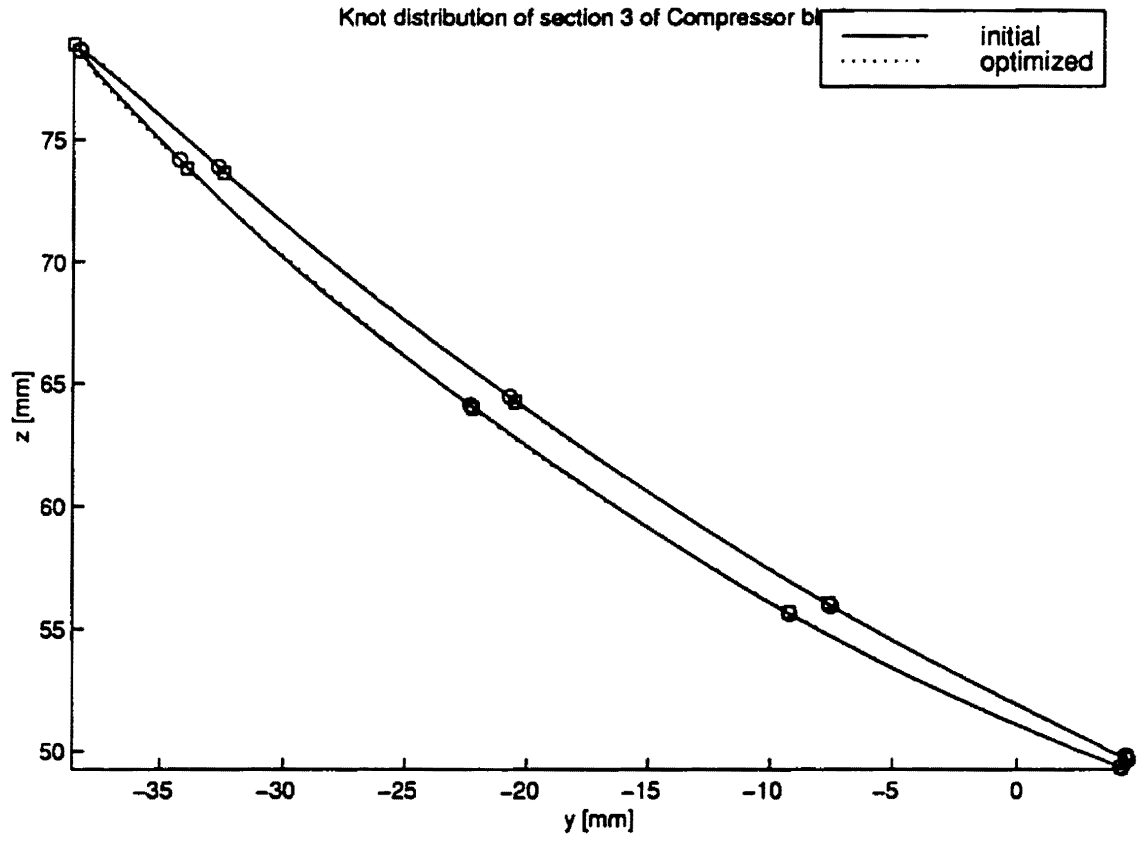


Figure 5.37: Fitted curves of tip airfoil of compressor blade obtained from optimizing weights.

5.3.3 Optimization of Parameterization

The sum of least-square error decreased from 27.5 [mm^2] to 6.56 [mm^2] in 2 iteration as shown in Figure 5.38. The largest distances between the data and the approximation curves decreased from 1.16 millimeters to 0.661 millimeters for hub airfoil, from 1.25 millimeters to 0.774 millimeters for mid section airfoil, and from 1.19 millimeters to 0.776 millimeters for tip airfoil as shown in Figurea 5.39, 5.40, and 5.41. The shape of the sections, along with their junctions, are shown in Figures 5.42, 5.43, and 5.44. All constraints are inactive.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after two iteration.

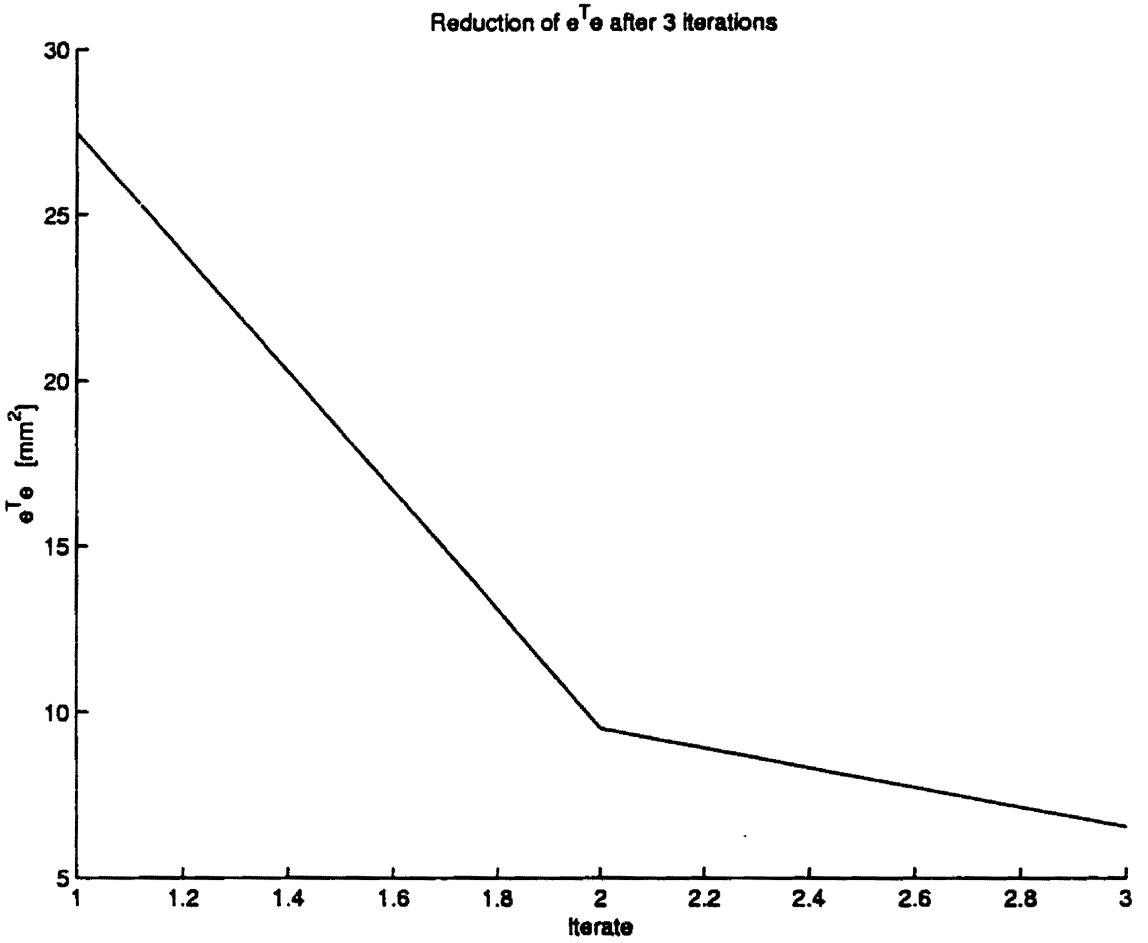


Figure 5.38: Reduction of error vs. number of iteration for three-curve compressor blade's airfoils obtained from optimizing the parameterization for data

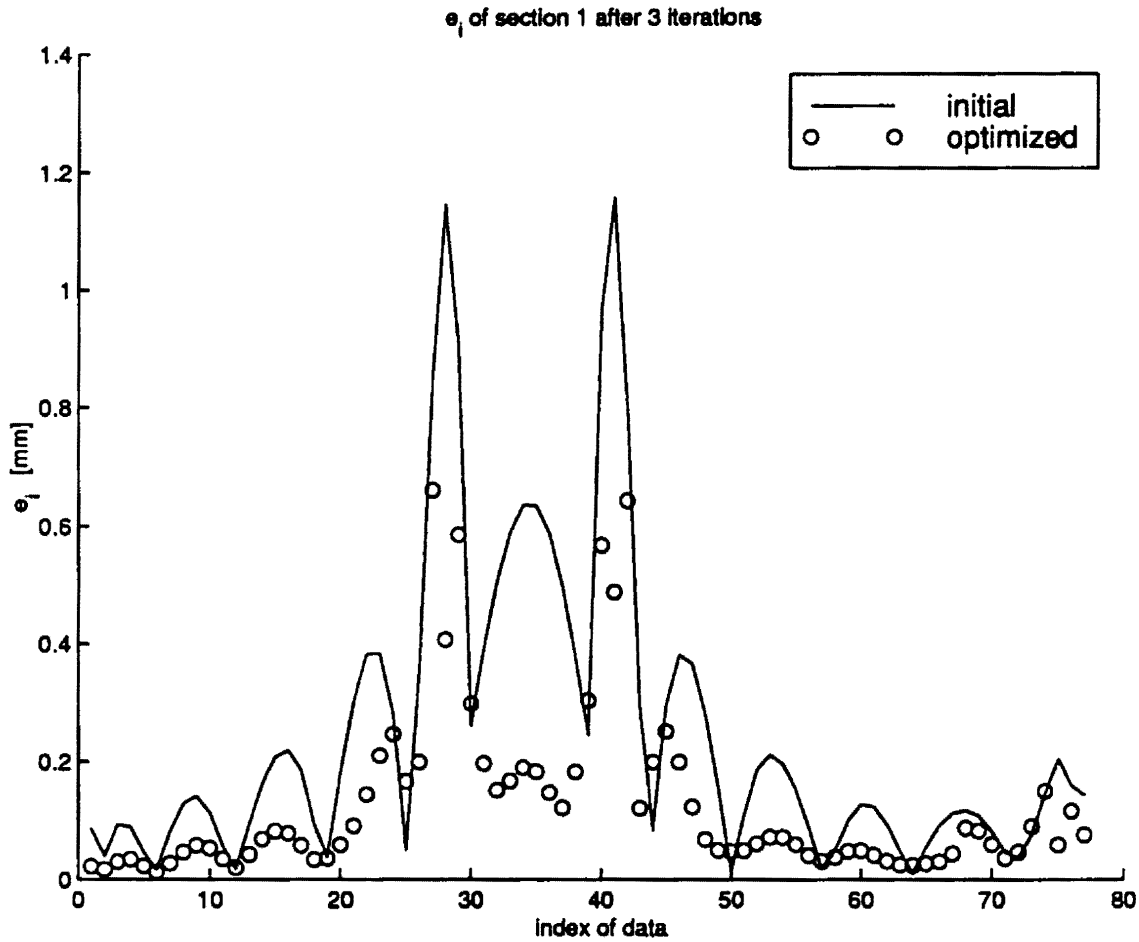


Figure 5.39: Error distribution of hub airfoil of three-curve compressor blade obtained from optimizing the parameterization for data.

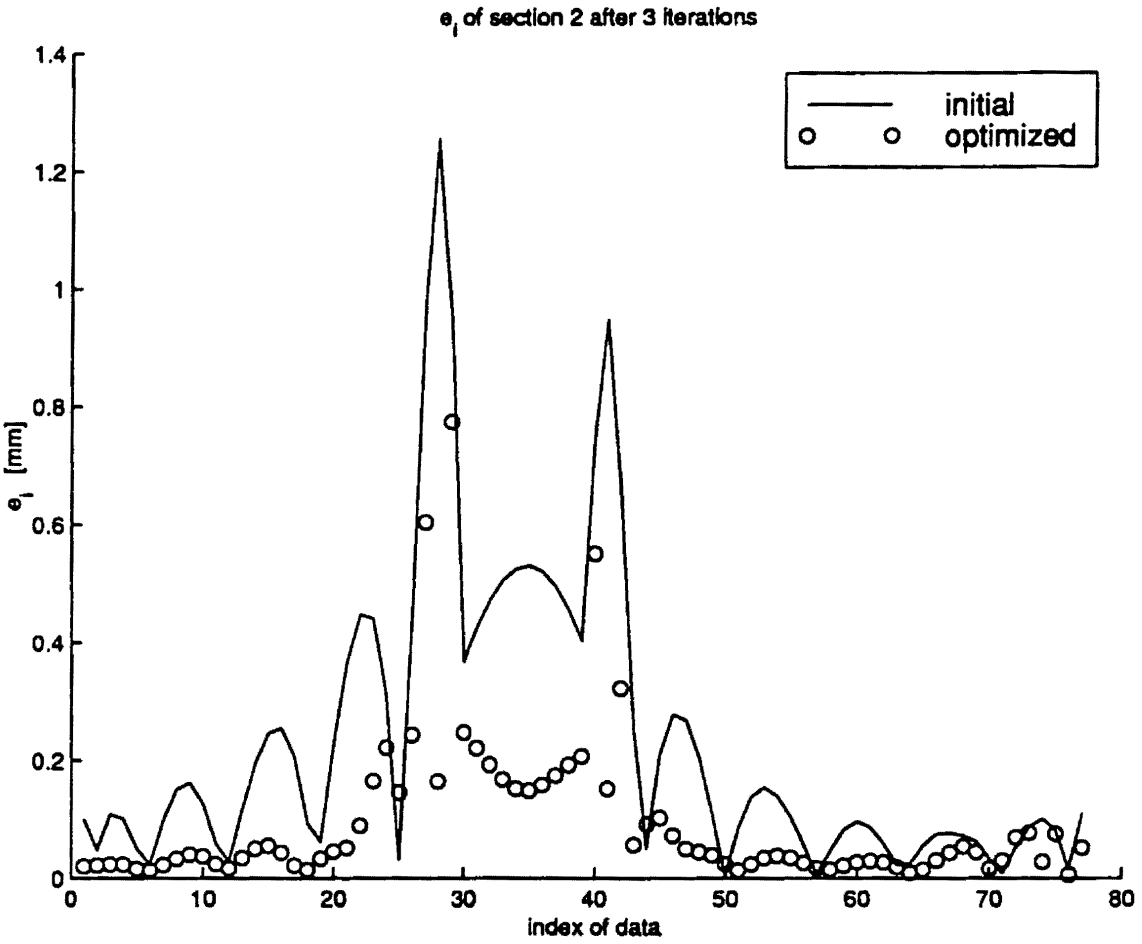


Figure 5.40: Error distribution of midsection airfoil of three-curve compressor blade obtained from optimizing the parameterization for data.

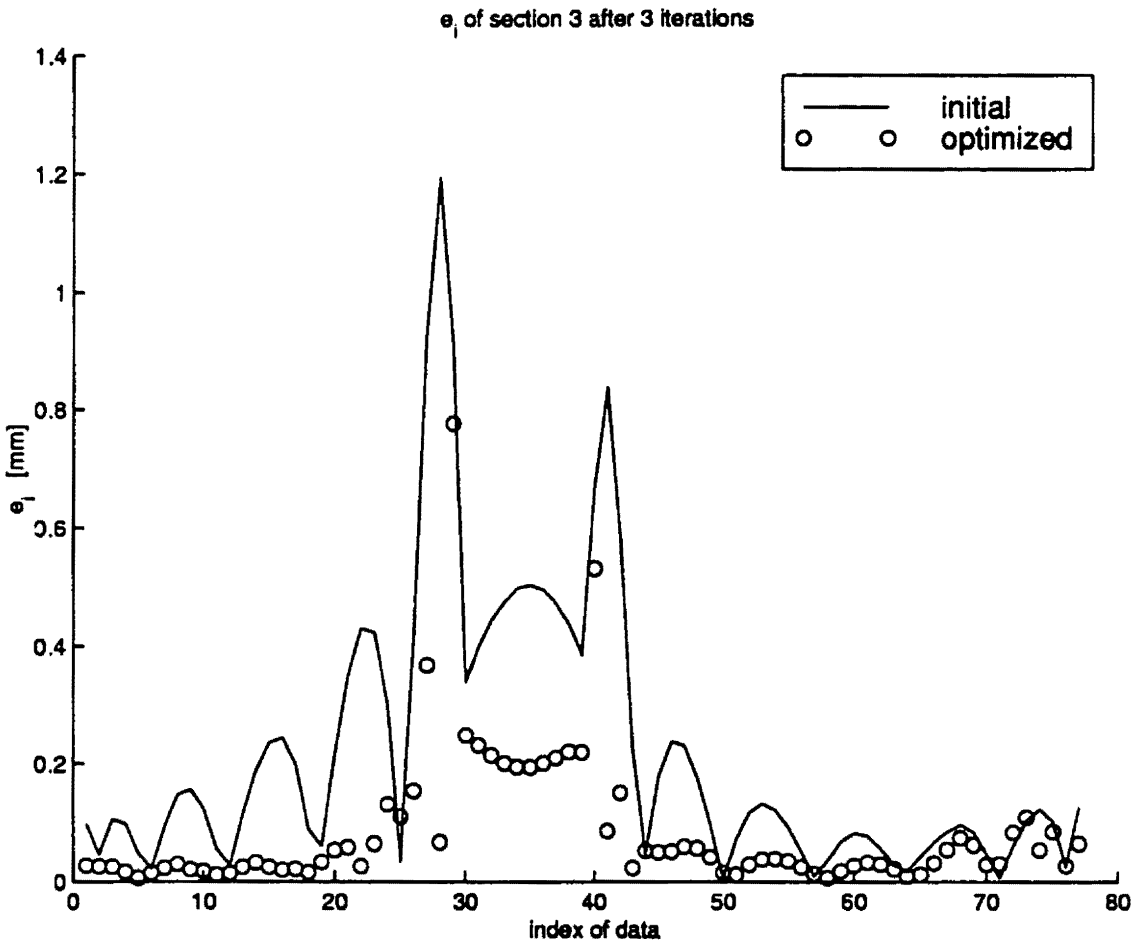


Figure 5.41: Error distribution of tip airfoil of three-curve compressor blade obtained from optimizing the parameterization for data.

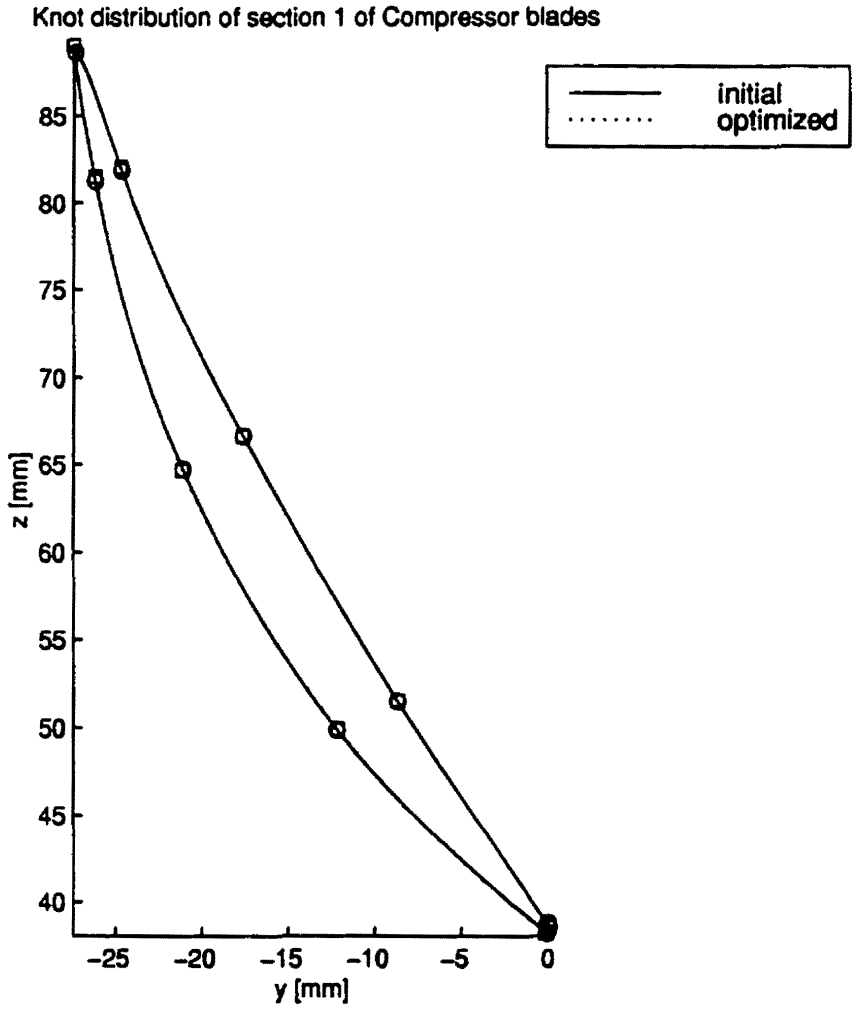


Figure 5.42: Fitted curves of hub airfoil of compressor blade obtained from optimizing parameterization for data.

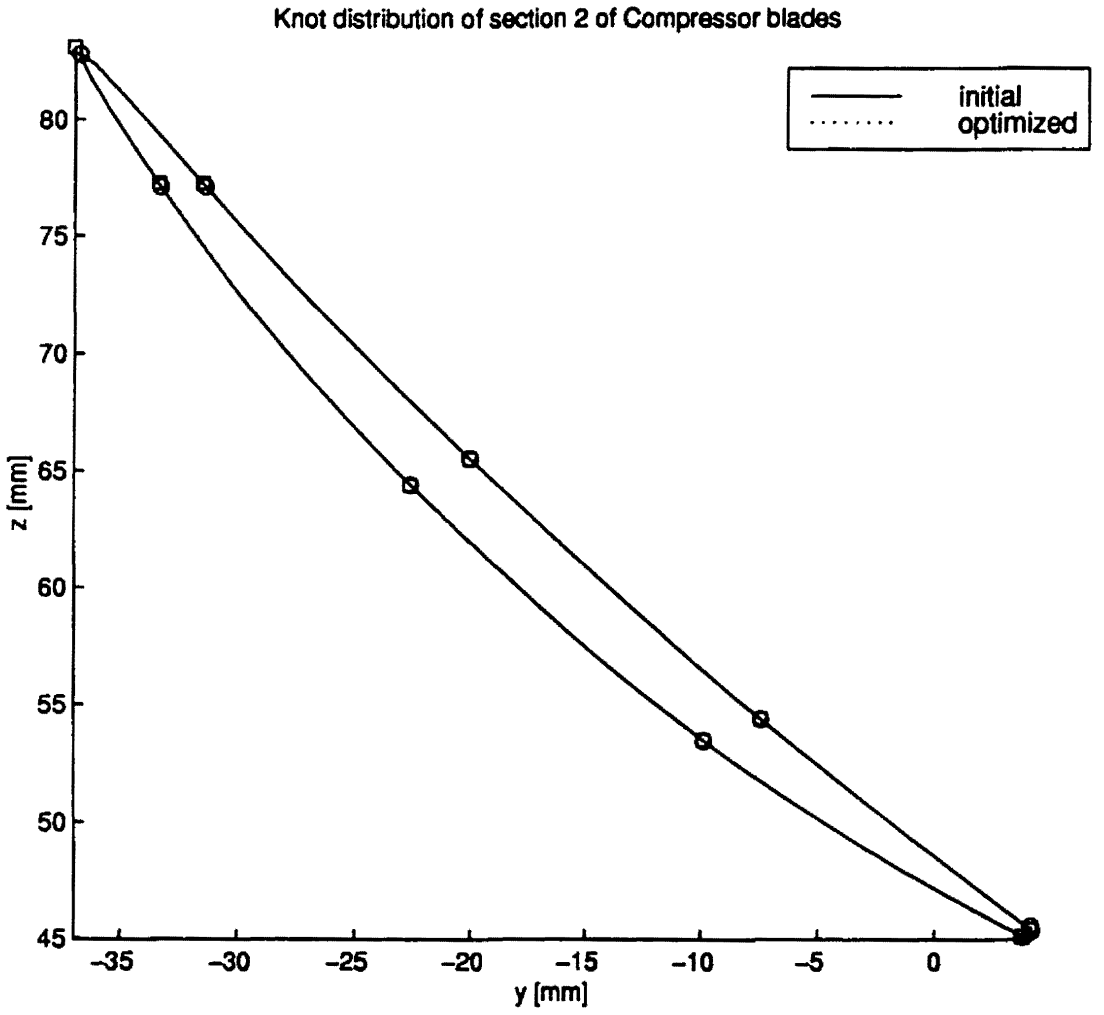


Figure 5.43: Fitted curves of mid section airfoil of compressor blade obtained from optimizing parameterization for data.

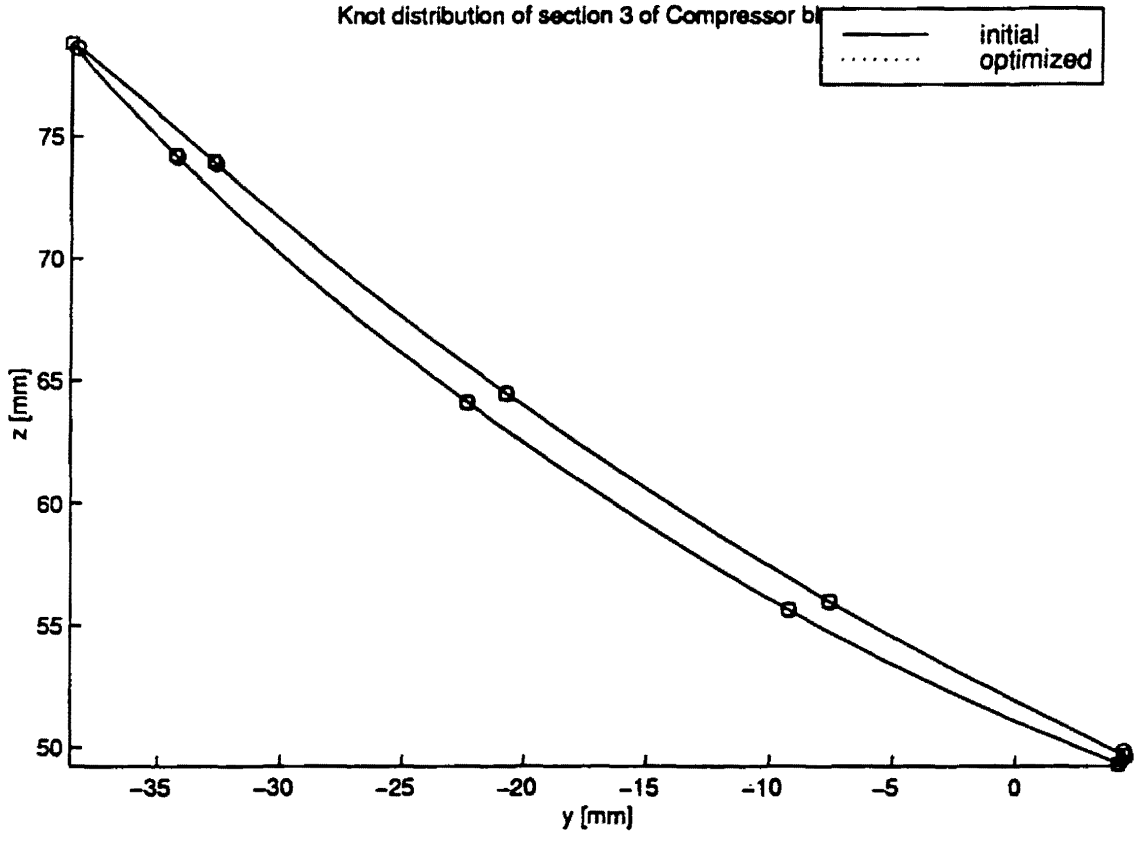


Figure 5.44: Fitted curves of tip airfoil of compressor blade obtained from optimizing parameterization for data.

5.3.4 Optimization of Knots and Weights

The sum of least-square error decreased from 27.5 [mm^2] to 1.52 [mm^2] in 19 iteration as shown in Figure 5.45. The largest distances between the data and the approximation curves decreased from 1.16 millimeters to 0.344 millimeters for hub airfoil, from 1.25 millimeters to 0.235 millimeters for mid section airfoil, and from 1.19 millimeters to 0.276 millimeters for tip airfoil, as shown in Figures 5.46, 5.47, and 5.48. The shape of the sections, along with their junctions, are shown in Figures 5.49, 5.47, and 5.48. All constraints are inactive. The backtrack algorithm failed due to excessively small step length. Change of weights are less than 0.5% for the three sections.

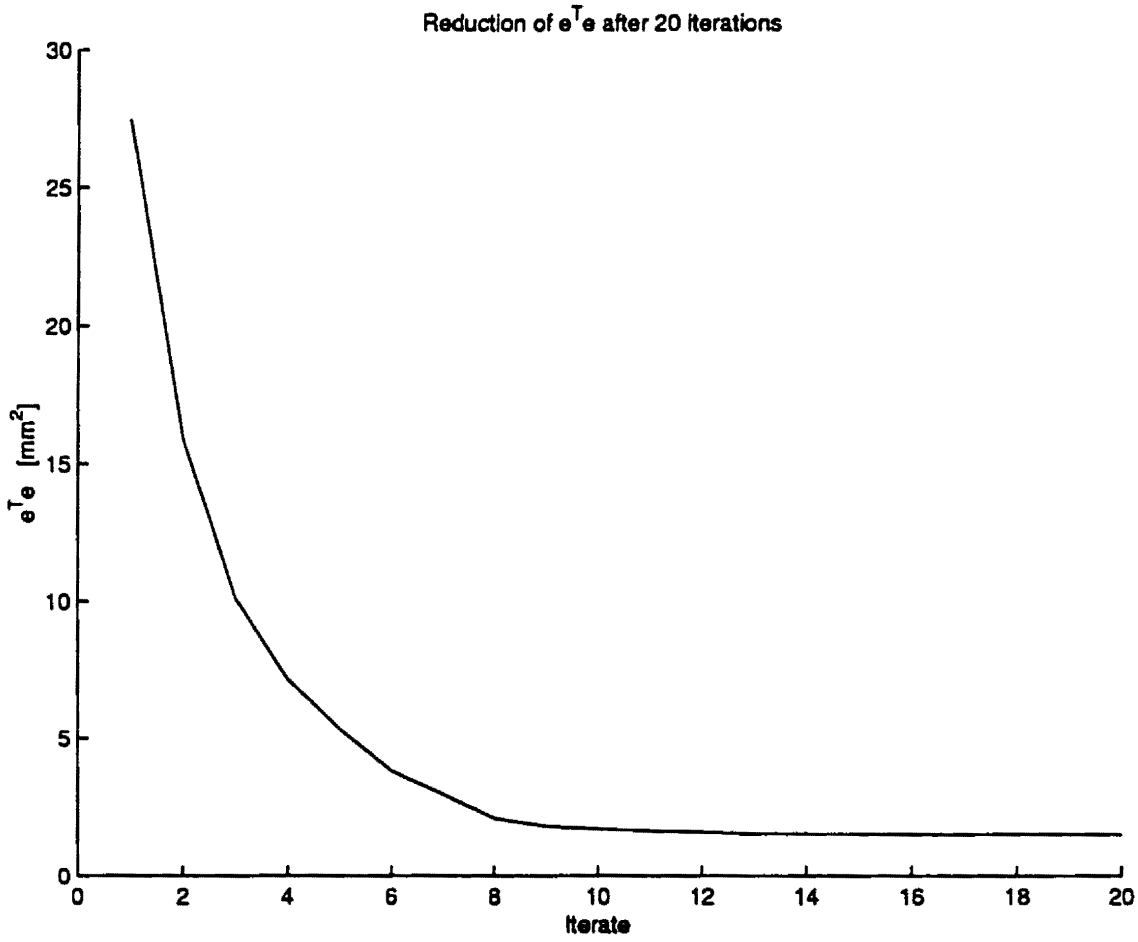


Figure 5.45: Reduction of error vs. number of iteration for three-curve compressor blade's airfoils obtained from optimizing the combination of knots and weights

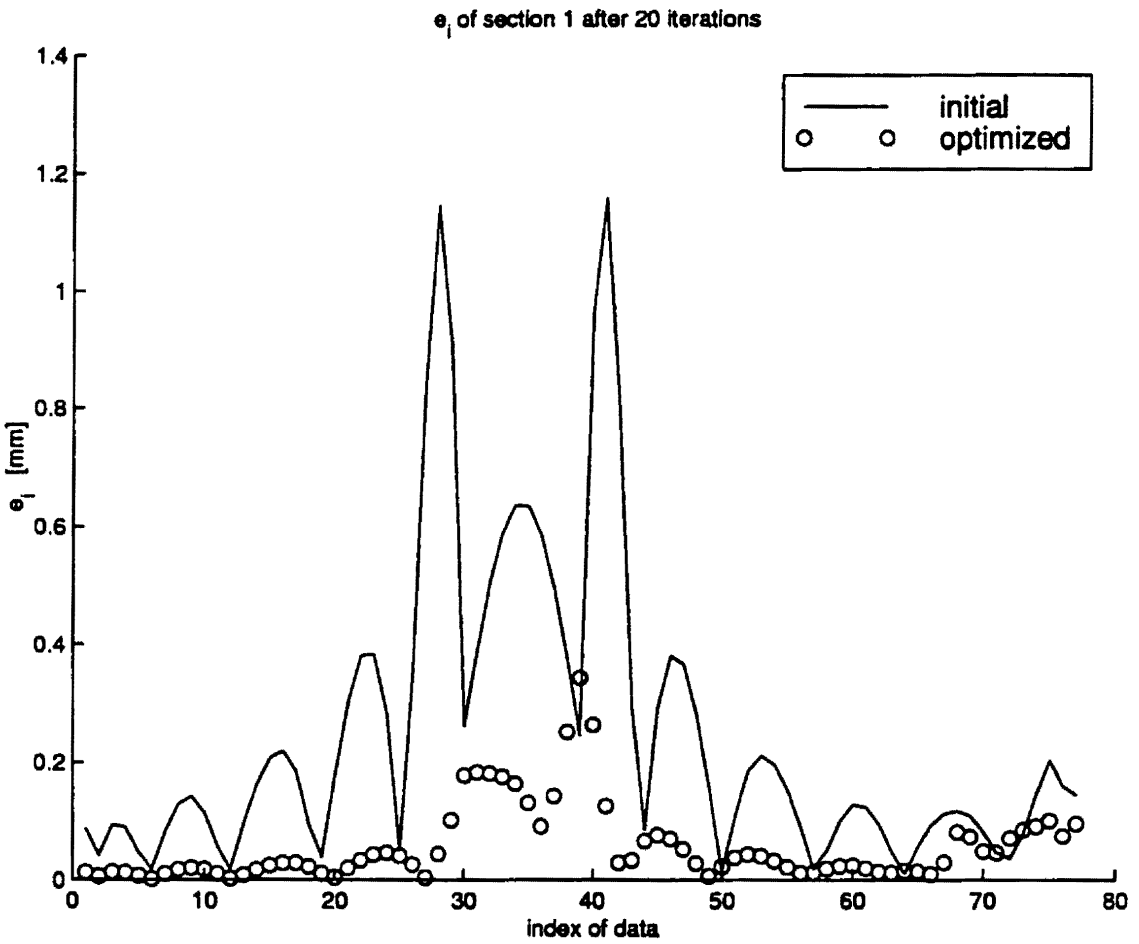


Figure 5.46: Error distribution of hub airfoil of three-curve compressor blade obtained from optimizing the combination of knots and weights.

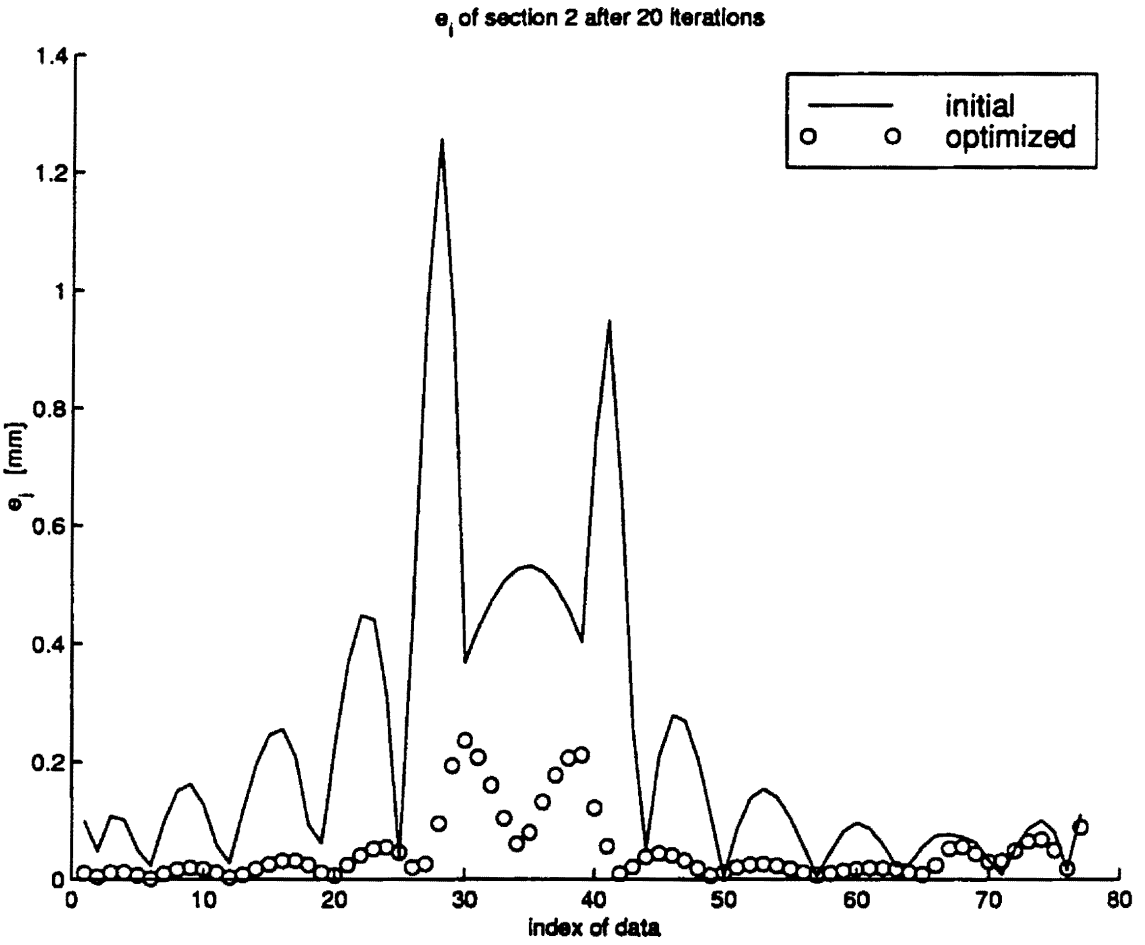


Figure 5.47: Error distribution of midsection airfoil of three-curve compressor blade obtained from optimizing the combination of knots and weights .

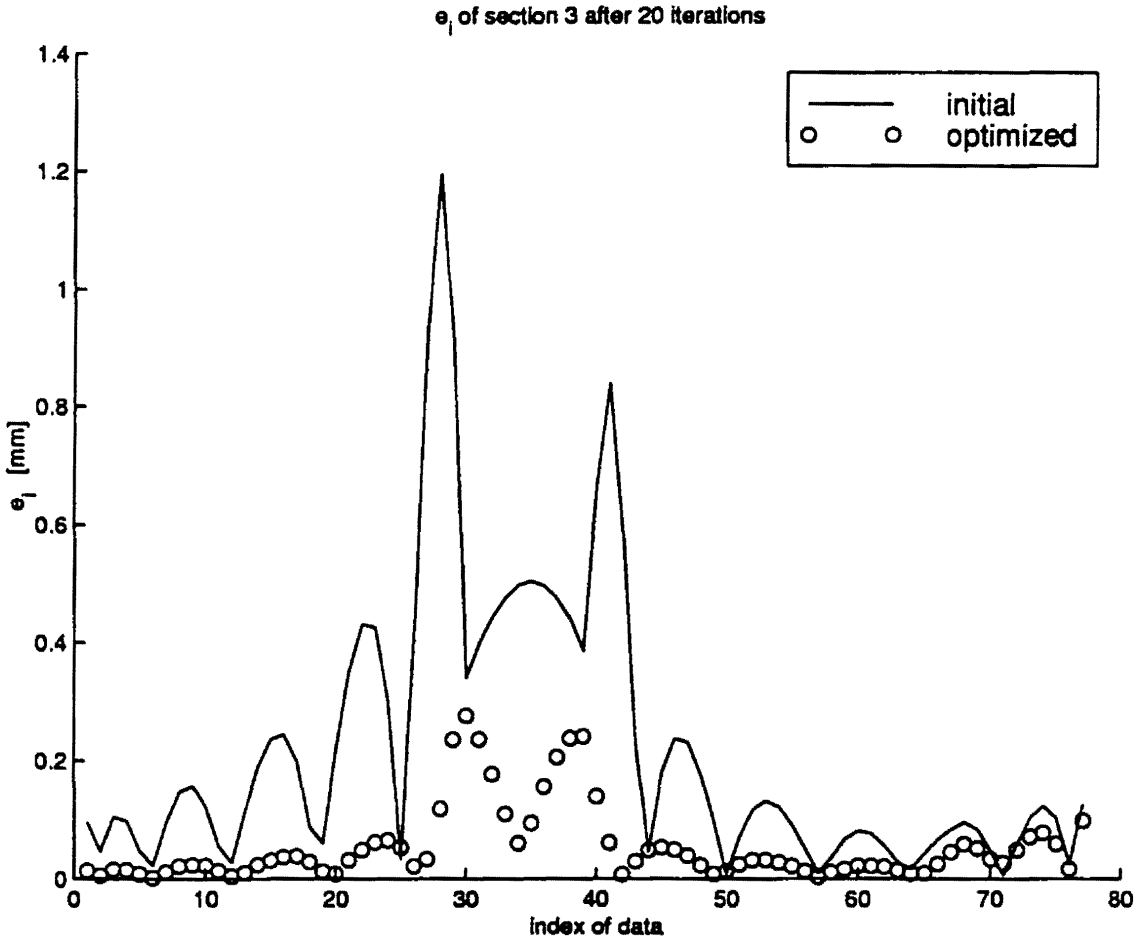


Figure 5.48: Error distribution of tip airfoil of three-curve compressor blade obtained from optimizing the combination of knots and weights.

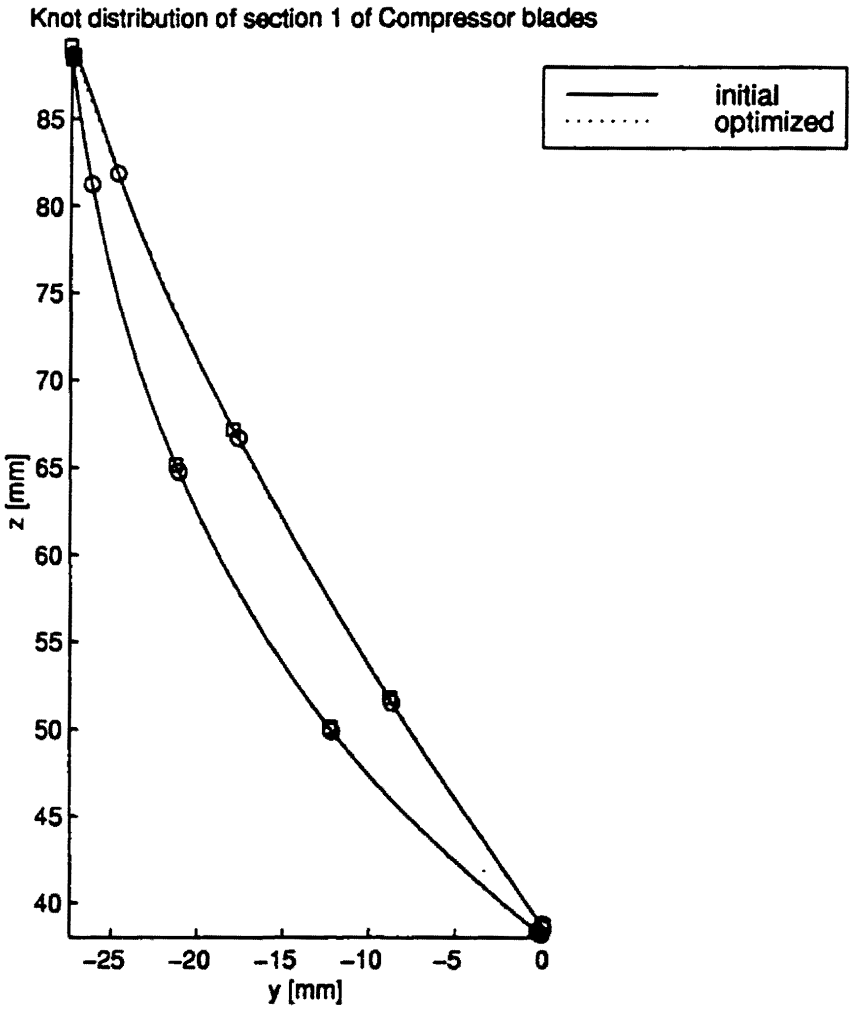


Figure 5.49: Fitted curves of hub airfoil of compressor blade obtained from optimizing combination of knots and weights.

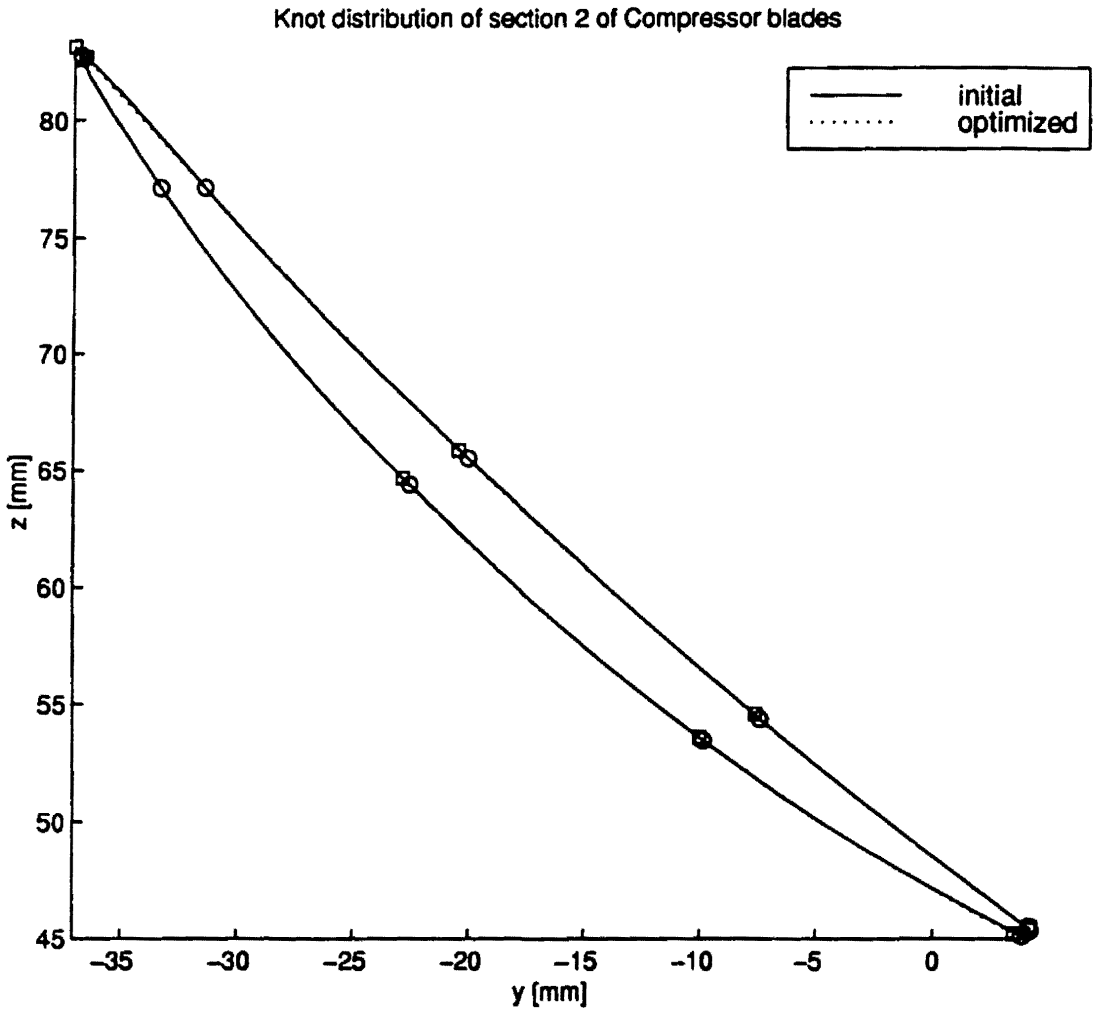


Figure 5.50: Fitted curves of mid section airfoil of compressor blade obtained from optimizing combination of knots and weights.

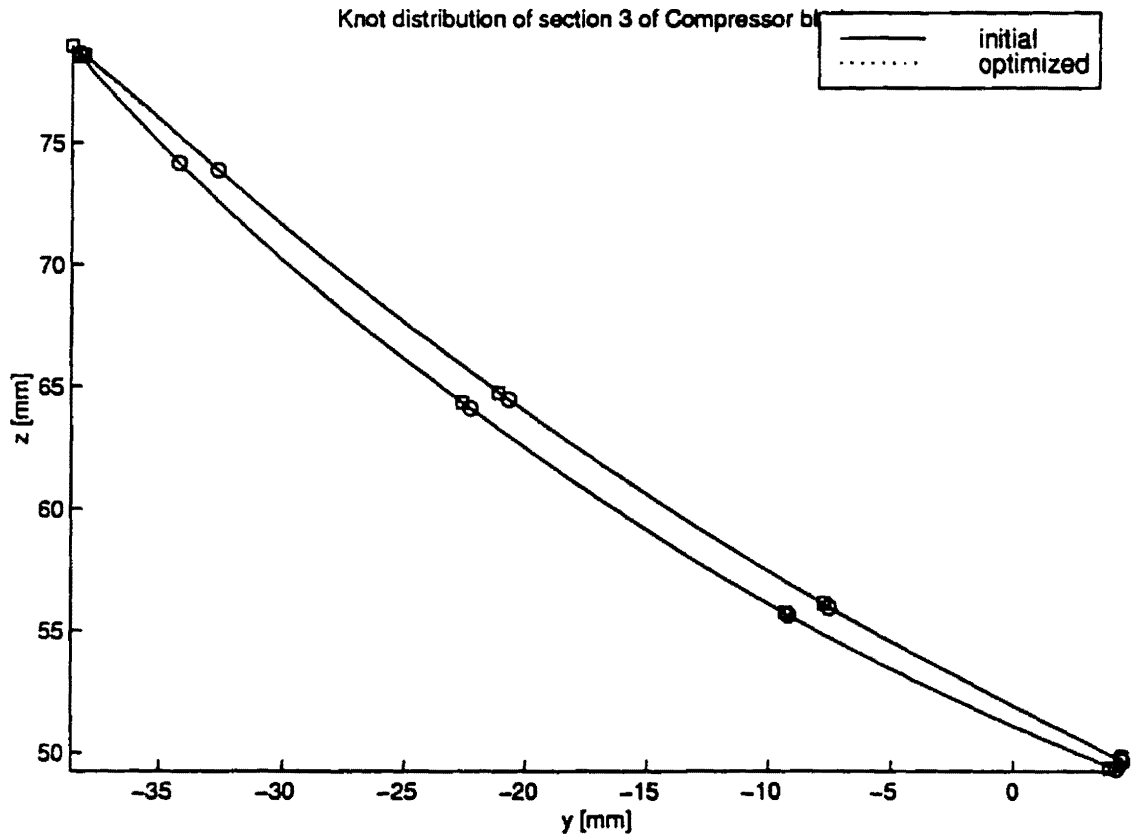


Figure 5.51: Fitted curves of tip airfoil of compressor blade obtained from optimizing combination of knots and weights.

5.3.5 Optimization of Knots, Weights, and Parameterization

The sum of least-square error decreased from 27.5 [mm^2] to 7.34 [mm^2] in 51 iteration as shown in Figure 5.52. The largest distances between the data and the approximation curves decreased from 1.16 millimeters to 0.871 millimeters for hub airfoil, from 1.25 millimeters to 0.871 millimeters for mid section airfoil, and from 1.19 millimeters to 0.795 millimeters for tip airfoil as shown in Figures 5.53, 5.53, and 5.53, respectively. The shapes of the sections, along with their junctions, are shown in Figures 5.56, 5.57, and 5.58.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after two iteration.

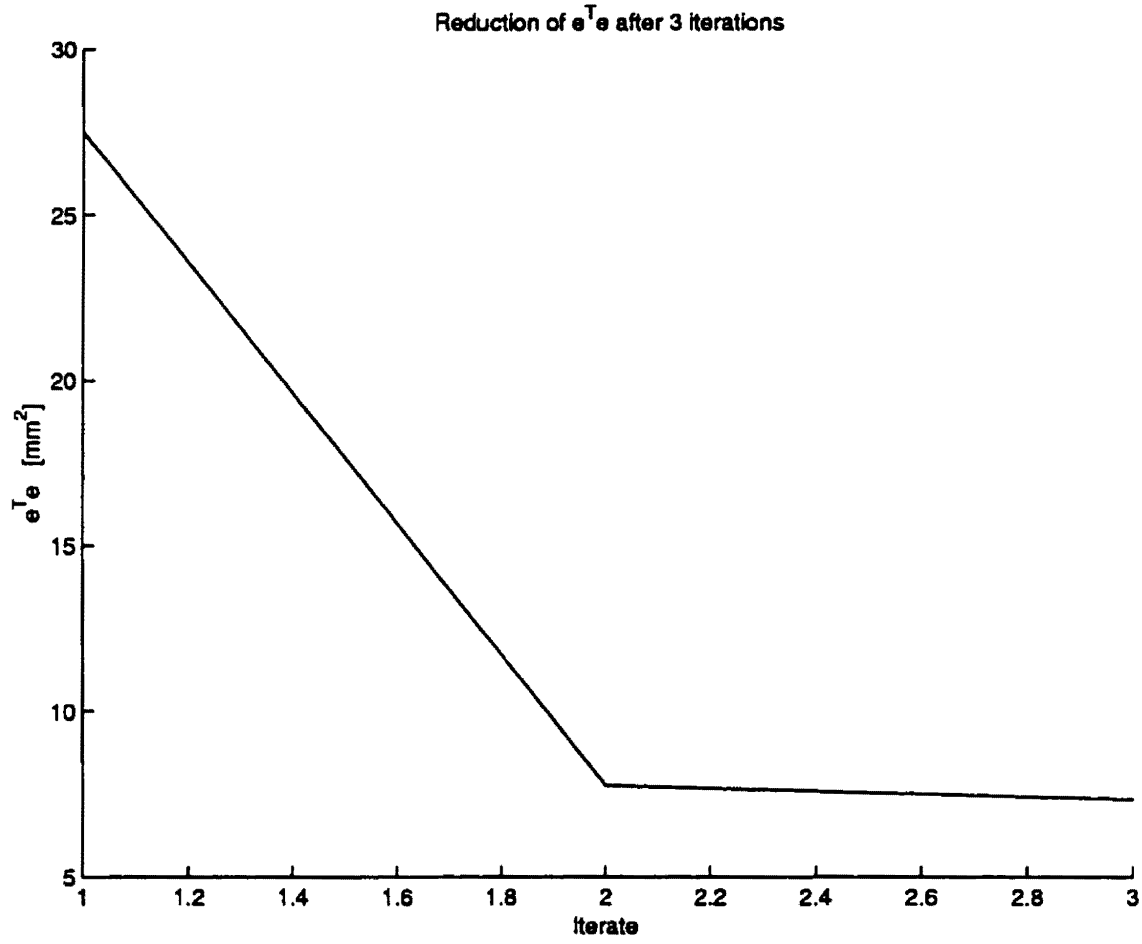


Figure 5.52: Reduction of error vs. number of iteration for three-curve compressor blade's airfoils obtained from optimizing the combination of knots, weights, and parameterization for data

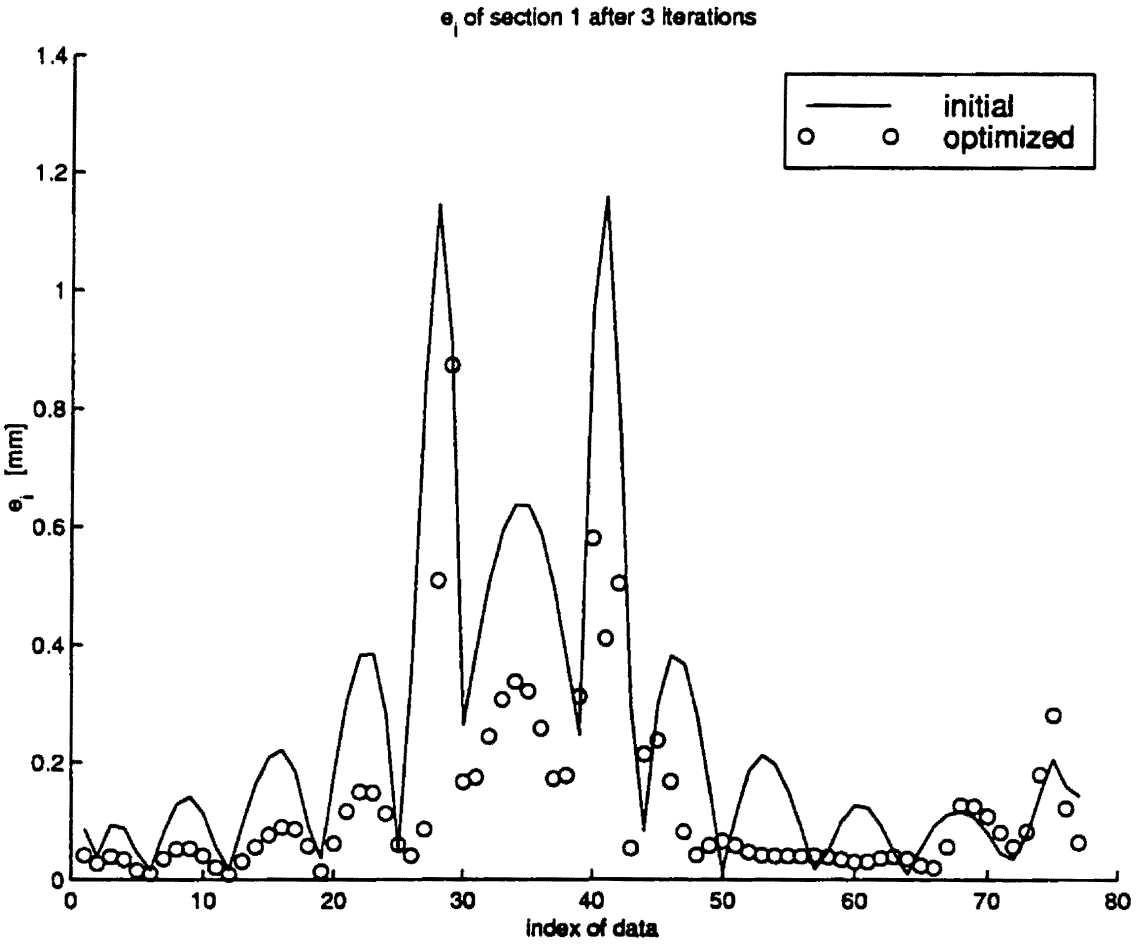


Figure 5.53: Error distribution of hub airfoil of three-curve compressor blade obtained from optimizing the combination of knots, weights, and parameterization for data.

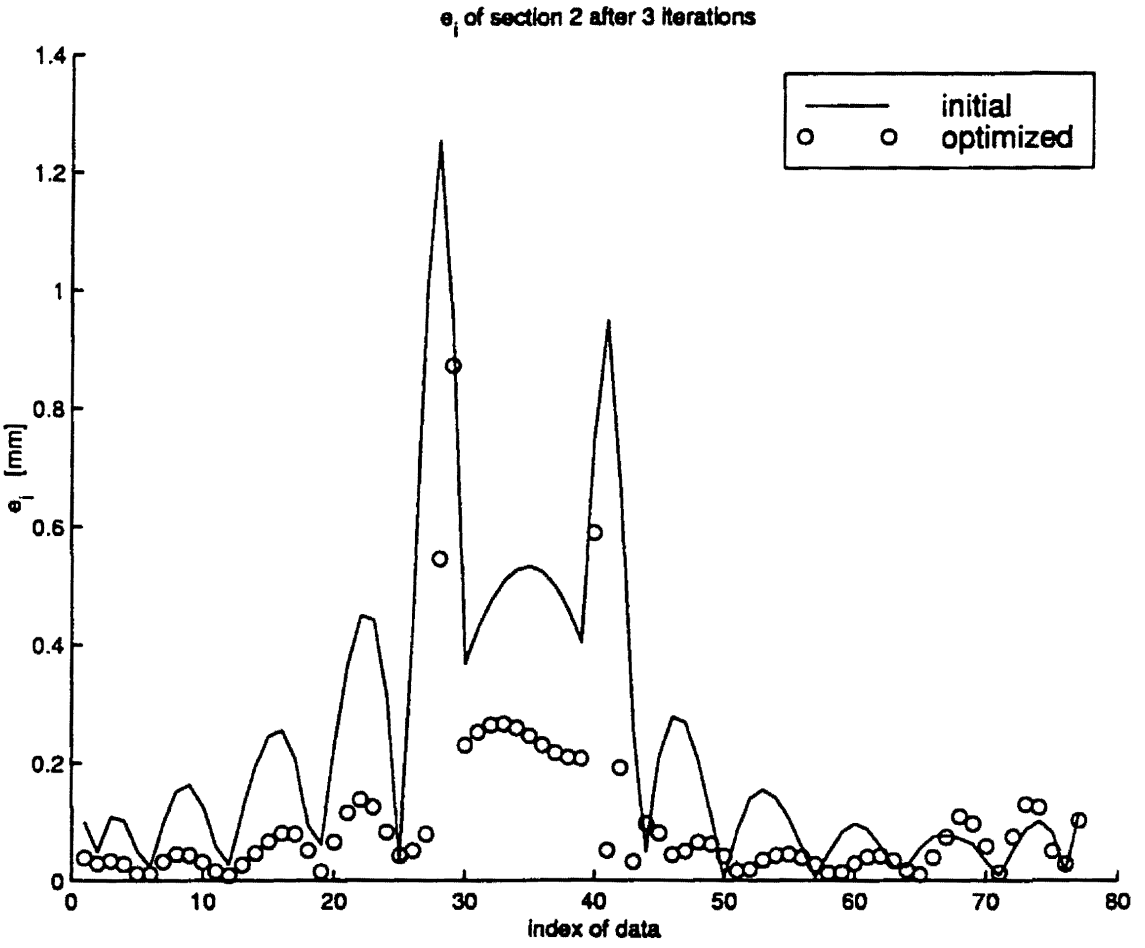


Figure 5.54: Error distribution of midsection airfoil of three-curve compressor blade obtained from optimizing the combination of knots, weights, and parameterization for data.

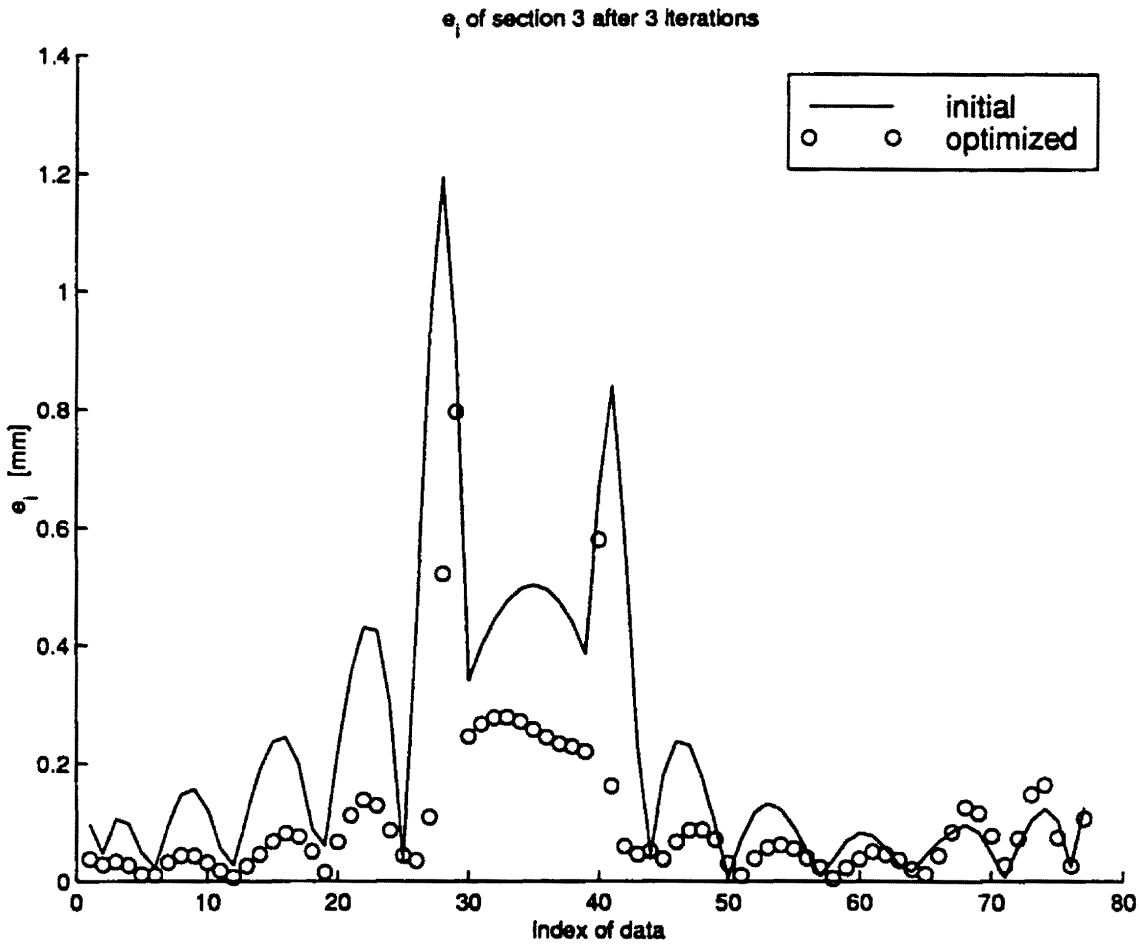


Figure 5.55: Error distribution of tip airfoil of three-curve compressor blade obtained from optimizing the combination of knots, weights, and parameterization for data.

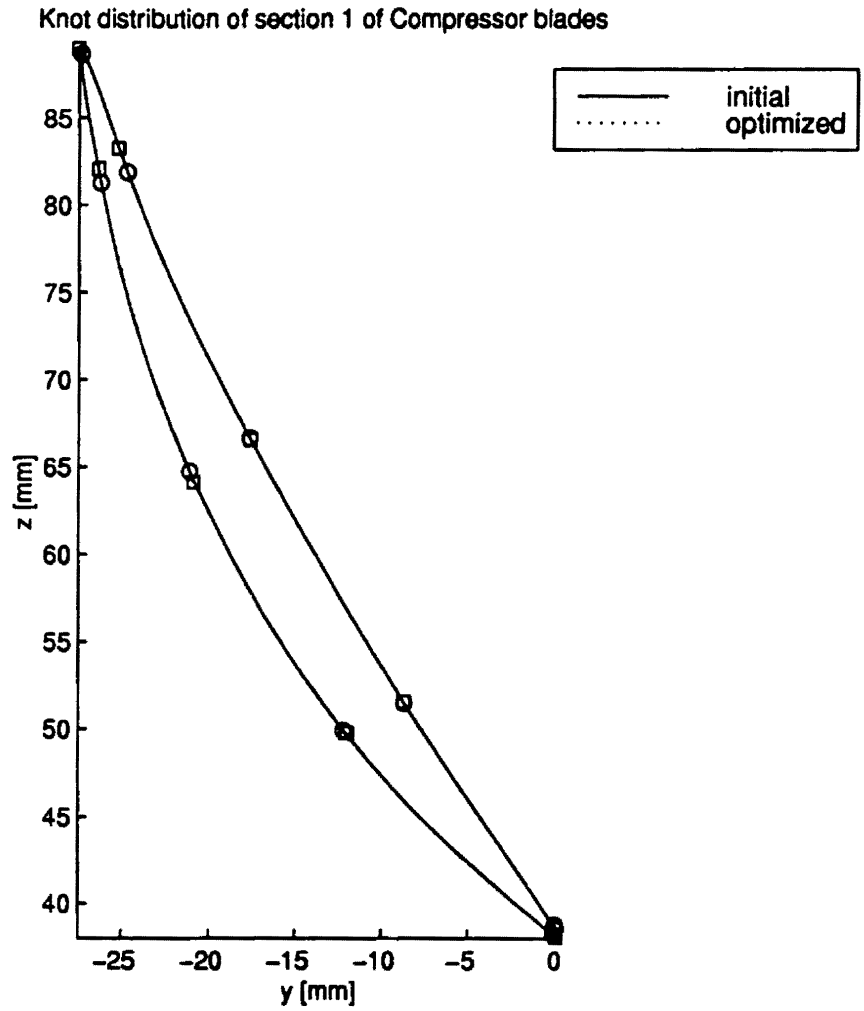


Figure 5.56: Fitted curves of hub airfoil of compressor blade obtained from optimizing combination of knots, weights, and parameterization for data.

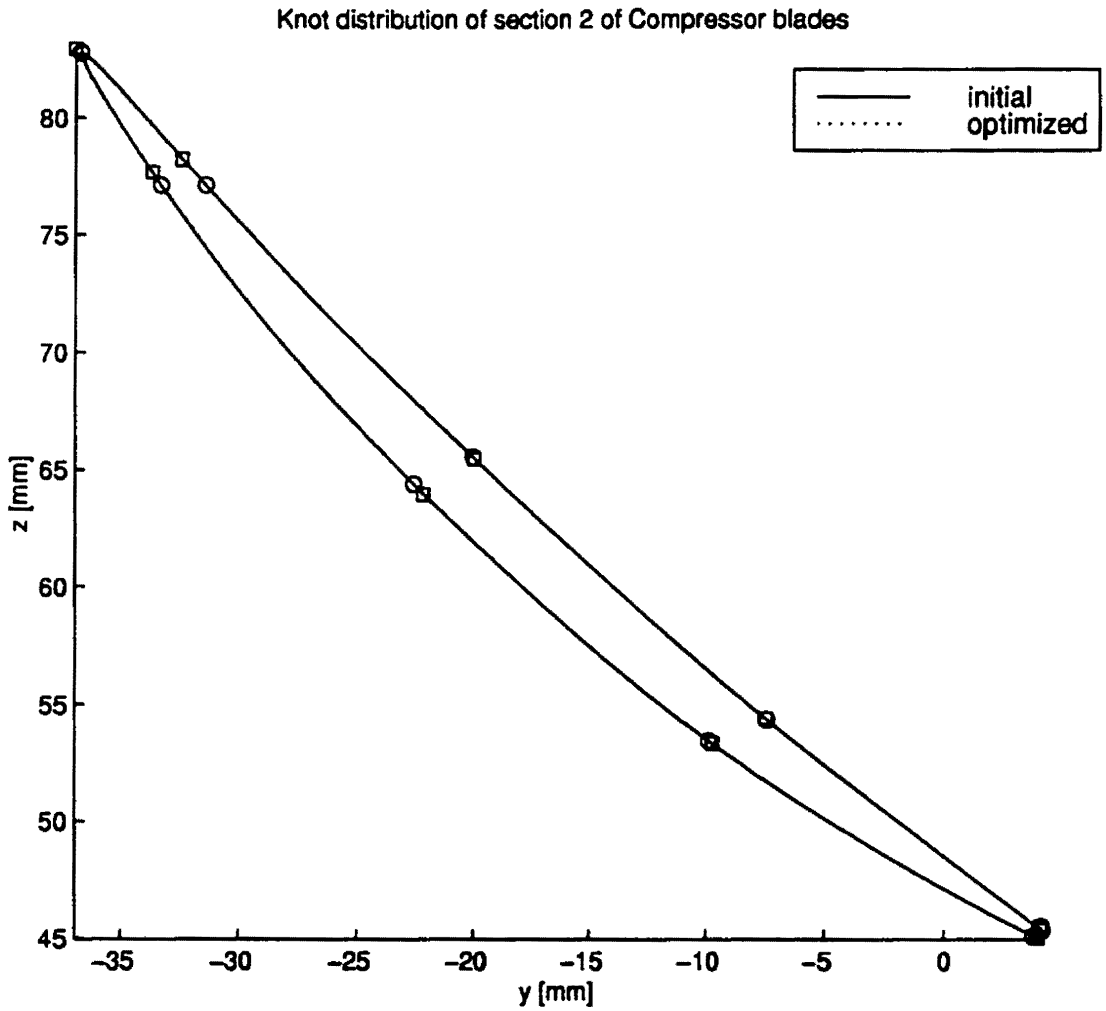


Figure 5.57: Fitted curves of mid section airfoil of compressor blade obtained from optimizing combination of knots, weights, and parameterization for data.

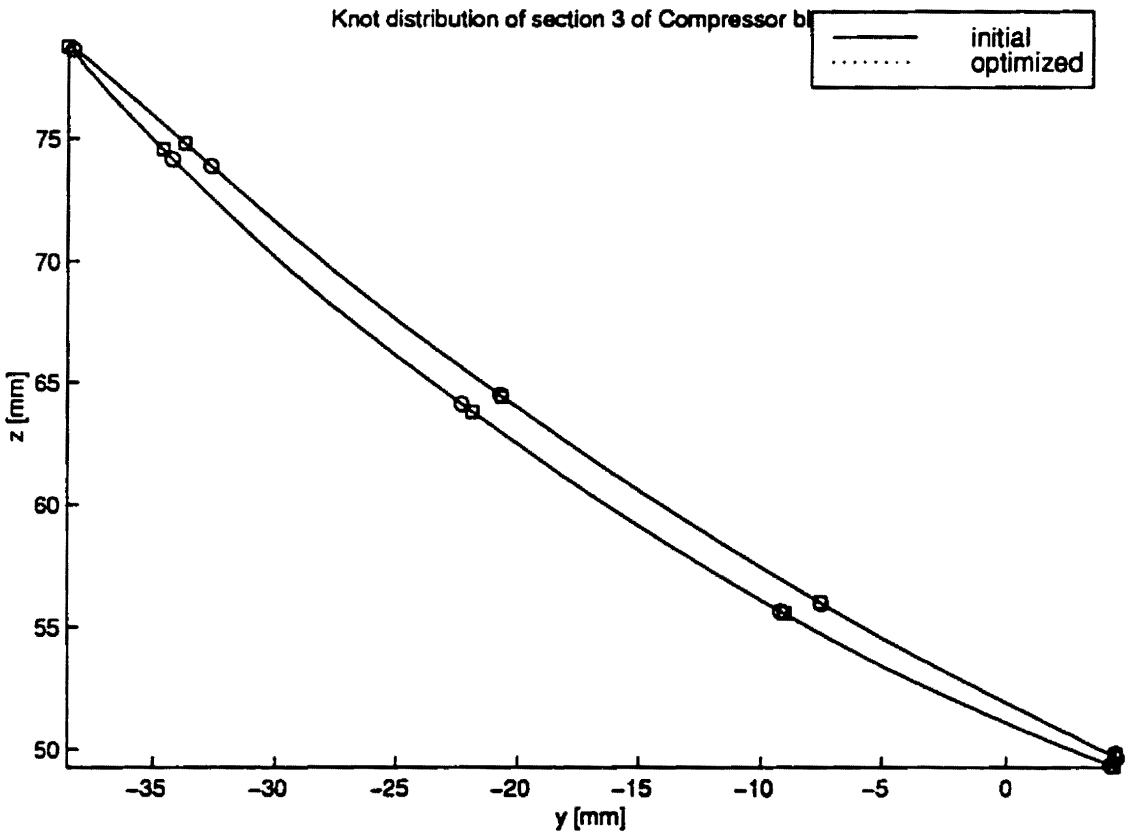


Figure 5.58: Fitted curves of tip airfoil of compressor blade obtained from optimizing combination of knots, weights, and parameterization for data.

5.4 Turbine Blades

Figure 5.59 shows three airfoils defining the blade of an axial turbine. The blade is a proprietary of the Concepts ETI, Inc. The airfoils are shown in an isometric view. The axis of rotation of the turbine is the Z-axis. The left most airfoil is at the hub of the shaft of the turbine and the right most airfoil is at the tip of the blade. The location of the airfoil in the middle is at $r = (1 - 0.529)\, r_{\text{hub}} + 0.529\, r_{\text{tip}}$.

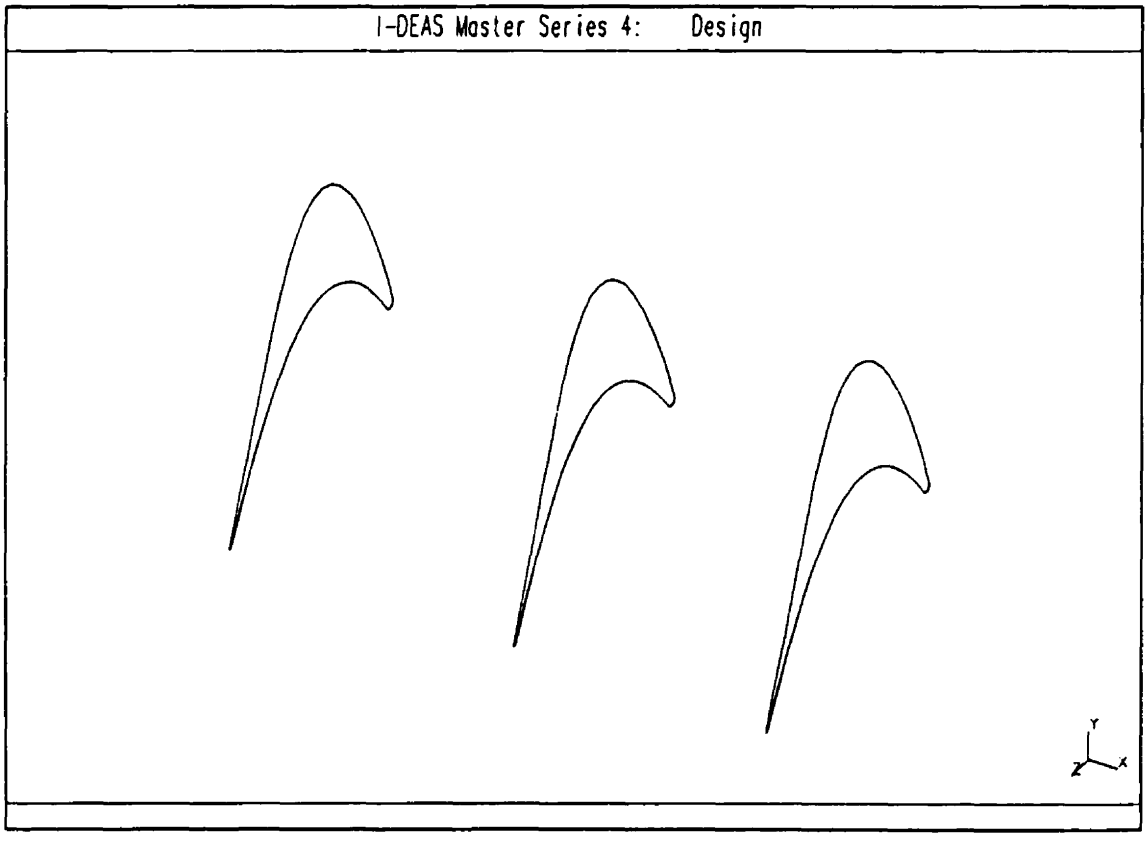


Figure 5.59: Data of Turbine Blades: Left Blade is Section 1, Middle Blade is Section 2, and Right Blade is Section 3

5.4.1 Optimization of Knots

The sum of least-square error decreased from 16.8 [mm^2] to 5.0 [mm^2] in 50 iteration as shown in Figure 5.60. The largest distances between the data and the approximation curves decreased from 0.983 millimeters to 0.438 millimeters for hub airfoils, from 0.881 millimeters to 0.476 millimeters for mid section airfoil, and from 0.785 millimeters to 0.521 millimeters for tip airfoil, as shown in Figures 5.61, 5.62, and 5.63, respectively. The shapes of the sections, along with their junctions, are shown in Figures 5.64, 5.65, and 5.66. All constraints are inactive.

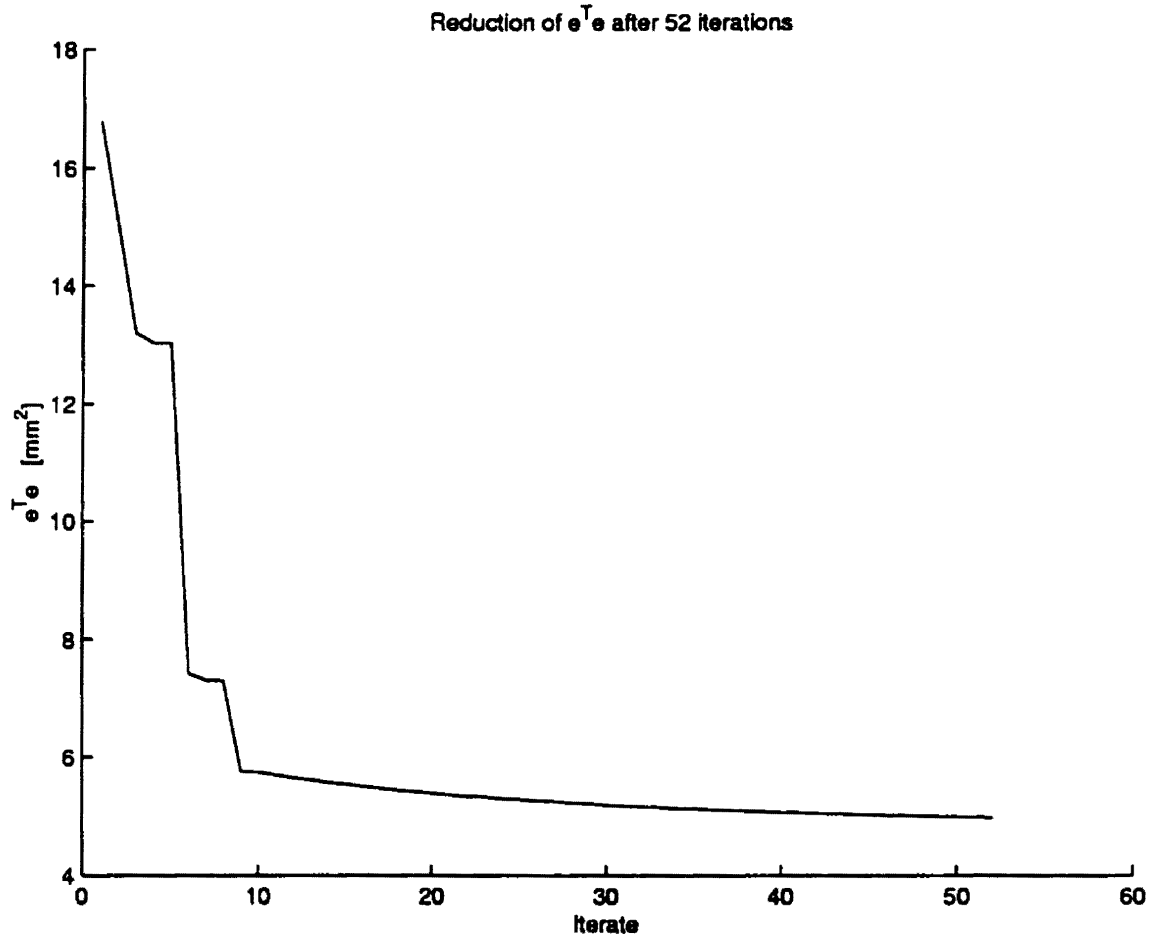


Figure 5.60: Reduction of error vs. number of iteration for three-curve turbine blade's airfoils obtained from optimizing the knots

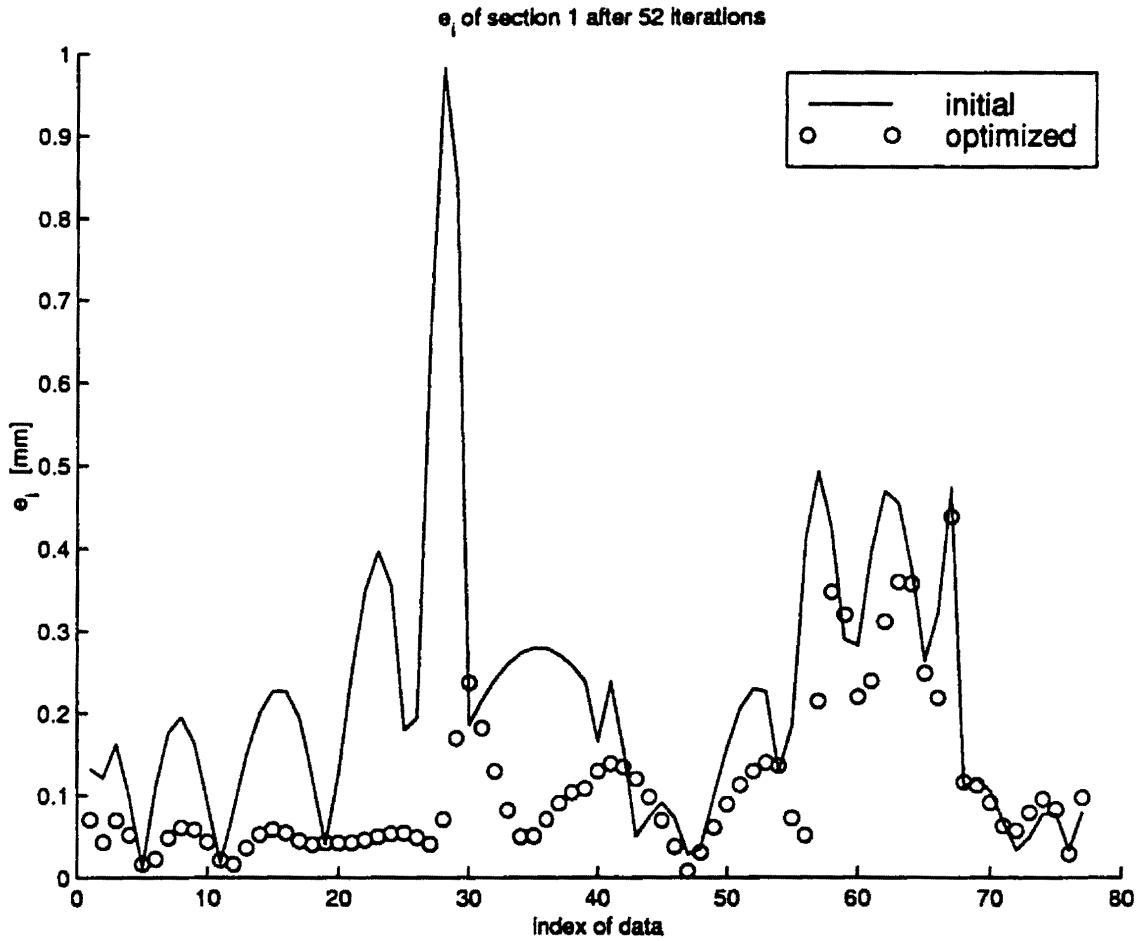


Figure 5.61: Error distribution of hub airfoil of three-curve turbine blade obtained from optimizing the knots.

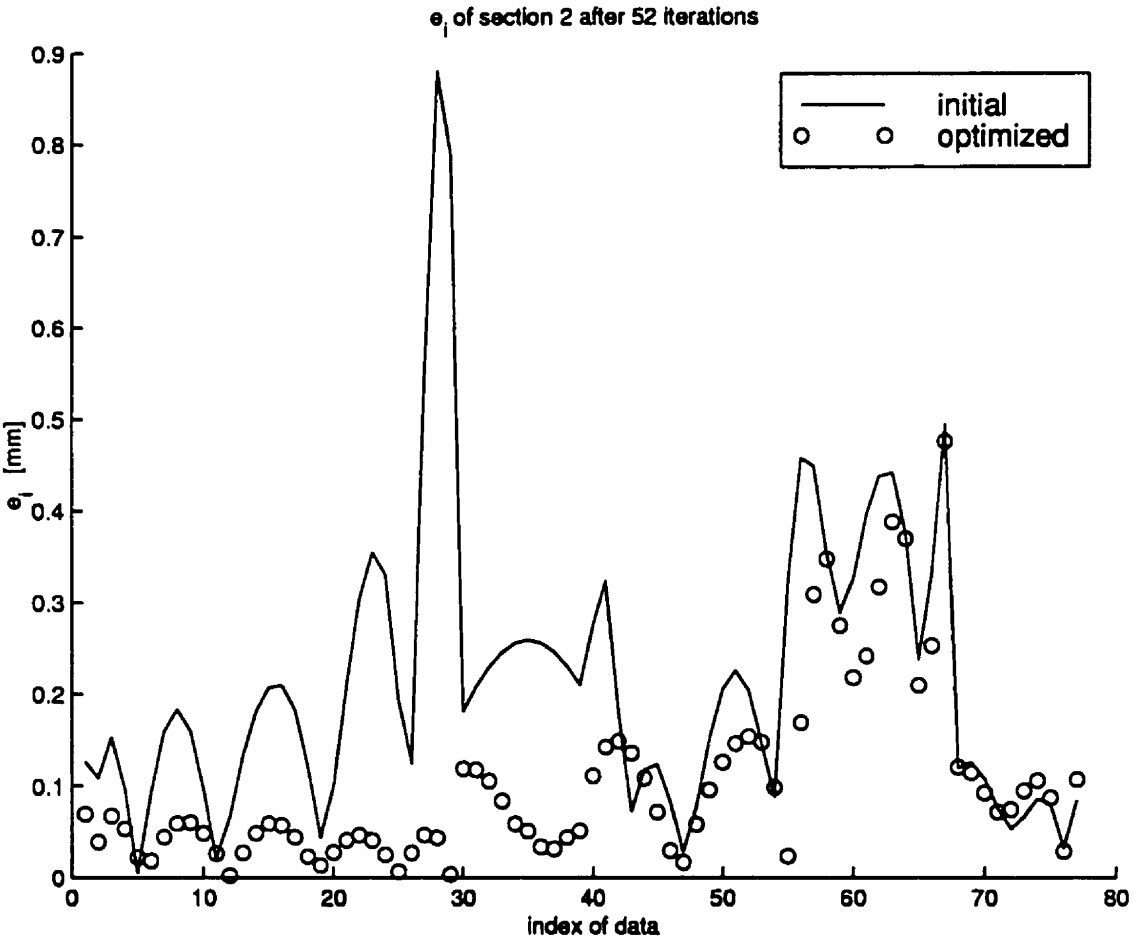


Figure 5.62: Error distribution of midsection airfoil of three-curve turbine blade obtained from optimizing the knots.

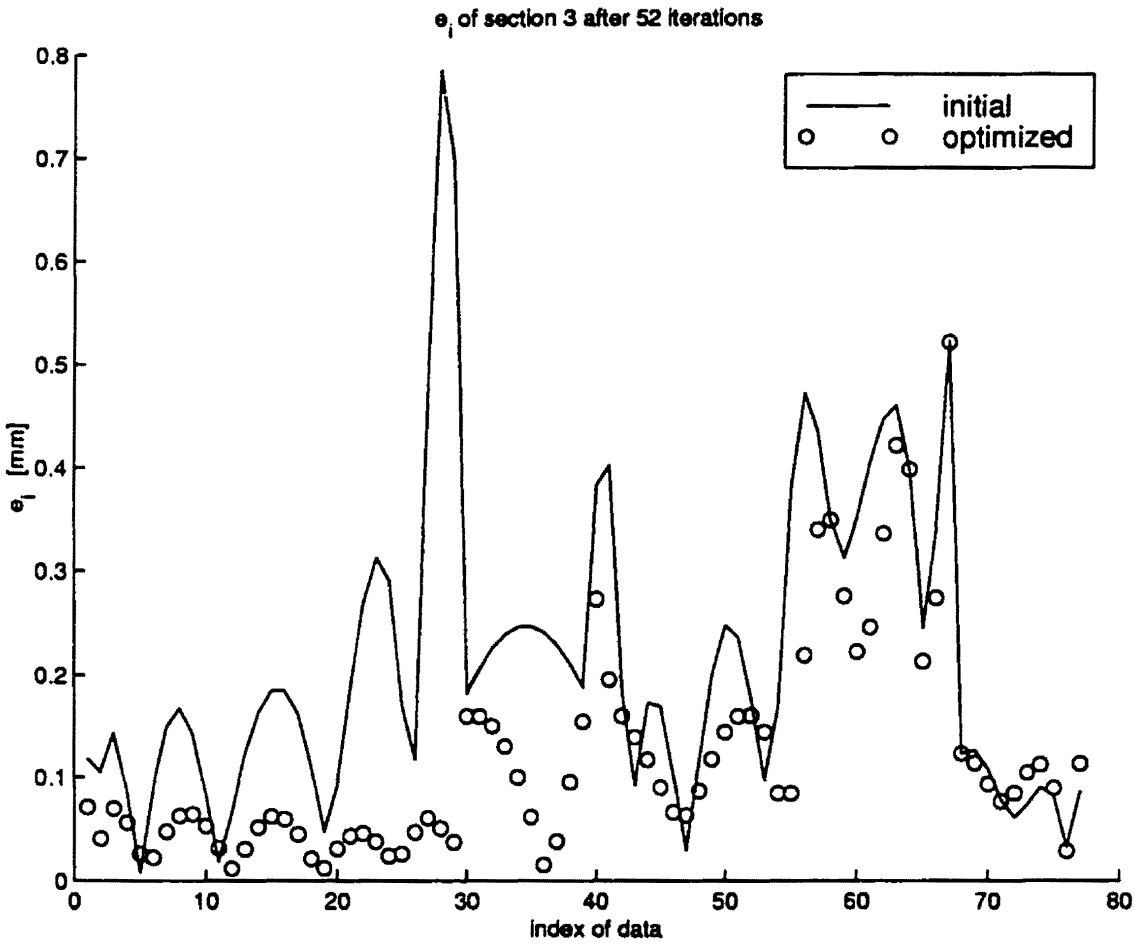


Figure 5.63: Error distribution of tip airfoil of three-curve turbine blade obtained from optimizing the knots.

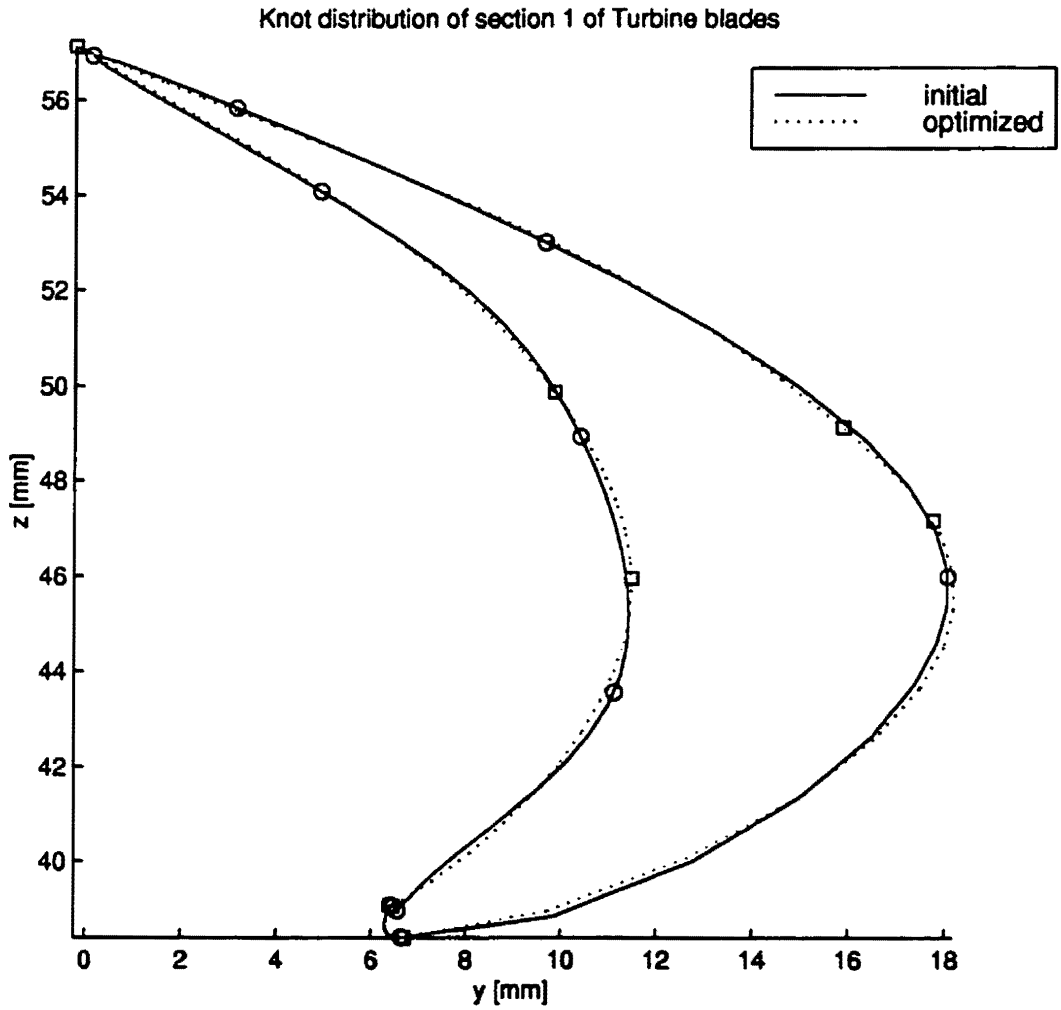


Figure 5.64: Fitted curves of hub airfoil of turbine blade obtained from optimizing knots.

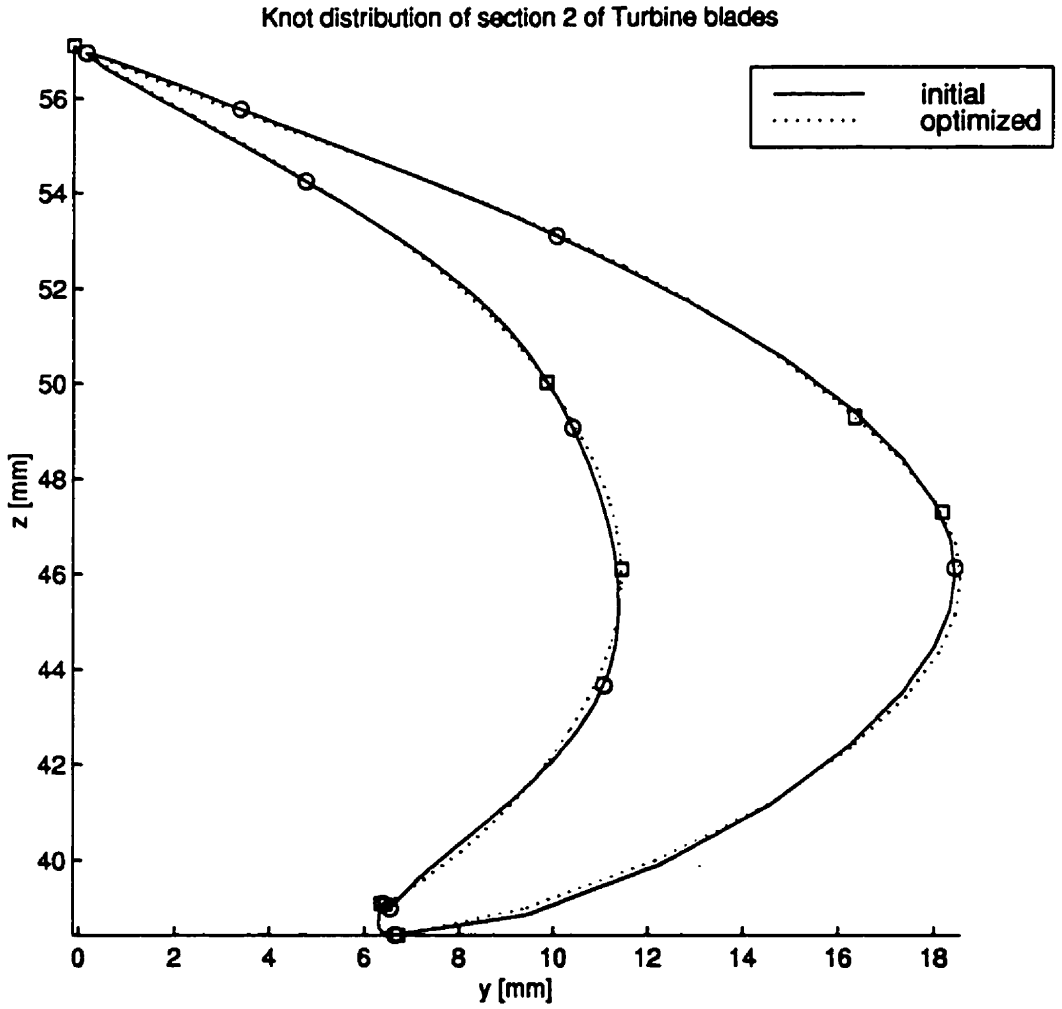


Figure 5.65: Fitted curves of mid section airfoil of turbine blade obtained from optimizing knots.

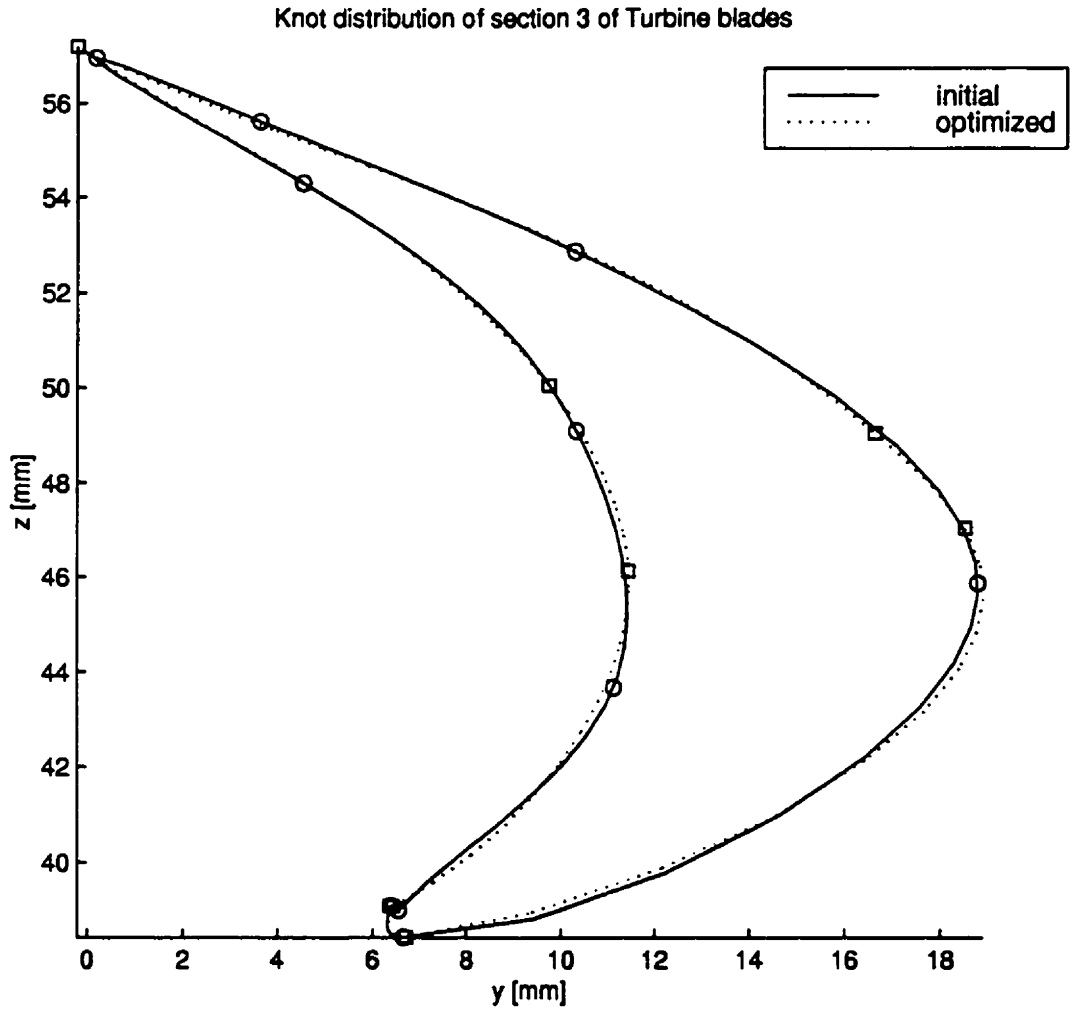


Figure 5.66: Fitted curves of tip airfoil of turbine blade obtained from optimizing knots.

5.4.2 Optimization of Weights

The sum of least-square error decreased from 16.8 [mm^2] to 2.1 [mm^2] in 25 iteration as shown in Figure 5.67. The largest distances between the data and the approximation curves decreased from 0.983 millimeters to 0.416 millimeters for hub airfoil, from 0.881 millimeters to 0.413 millimeters for mid section airfoil, and from 0.785 millimeters to 0.301 millimeters for tip airfoil, as shown in Figures 5.68, 5.69, and 5.70, respectively. The shapes of the sections, along with their junctions, are shown in Figures 5.71, 5.72, and 5.73. w_5 of section 3 is active. The backtrack algorithm encountered round off error.

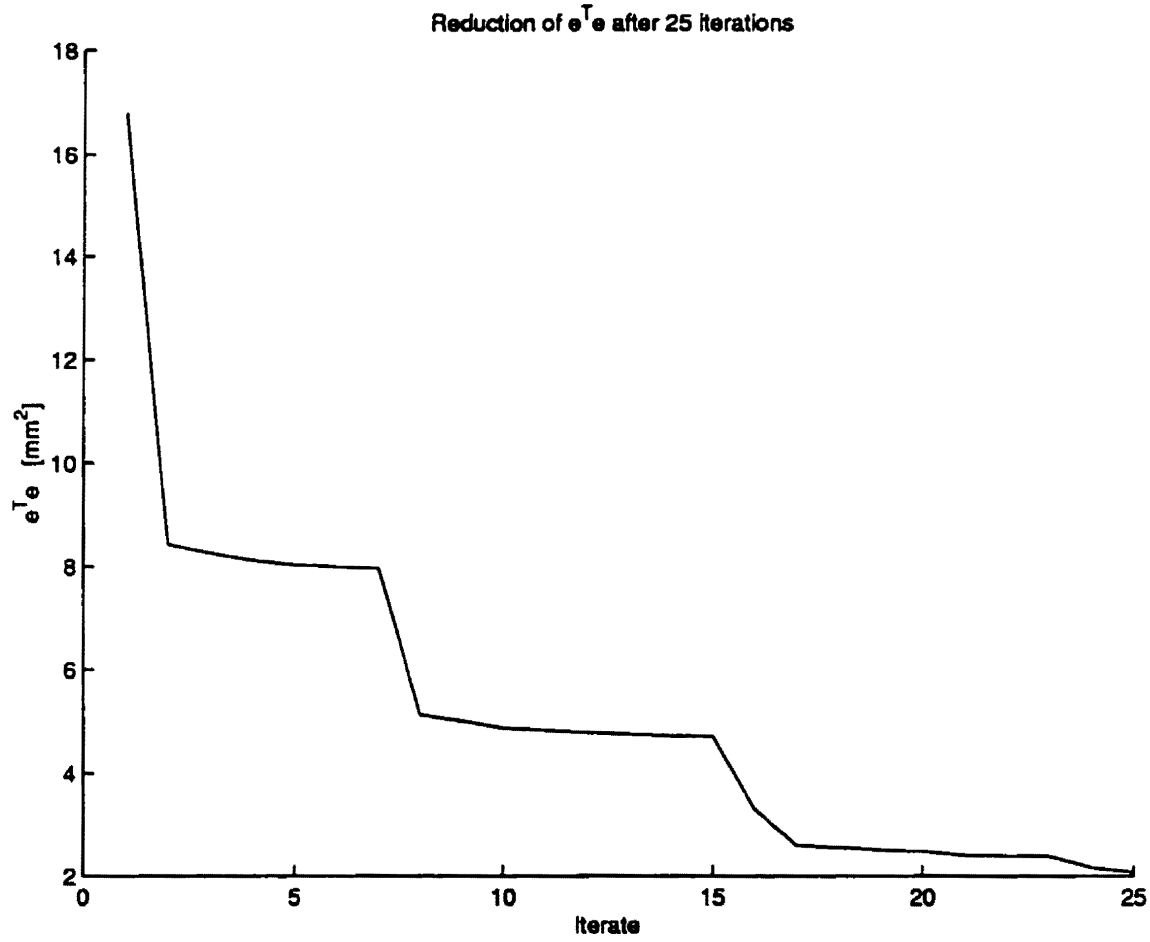


Figure 5.67: Reduction of error vs. number of iteration for three-curve turbine blade's airfoils obtained from optimizing the weights

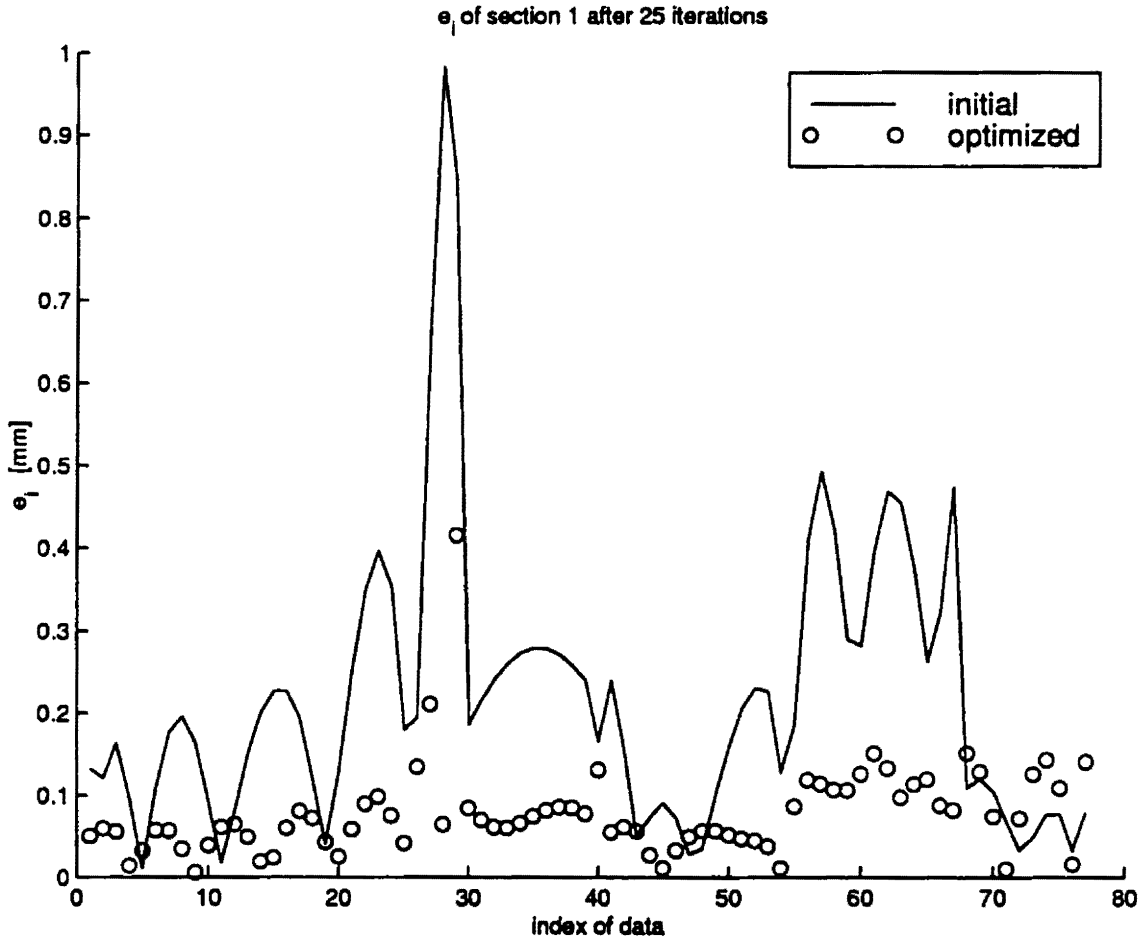


Figure 5.68: Error distribution of hub airfoil of three-curve turbine blade obtained from optimizing the weights.

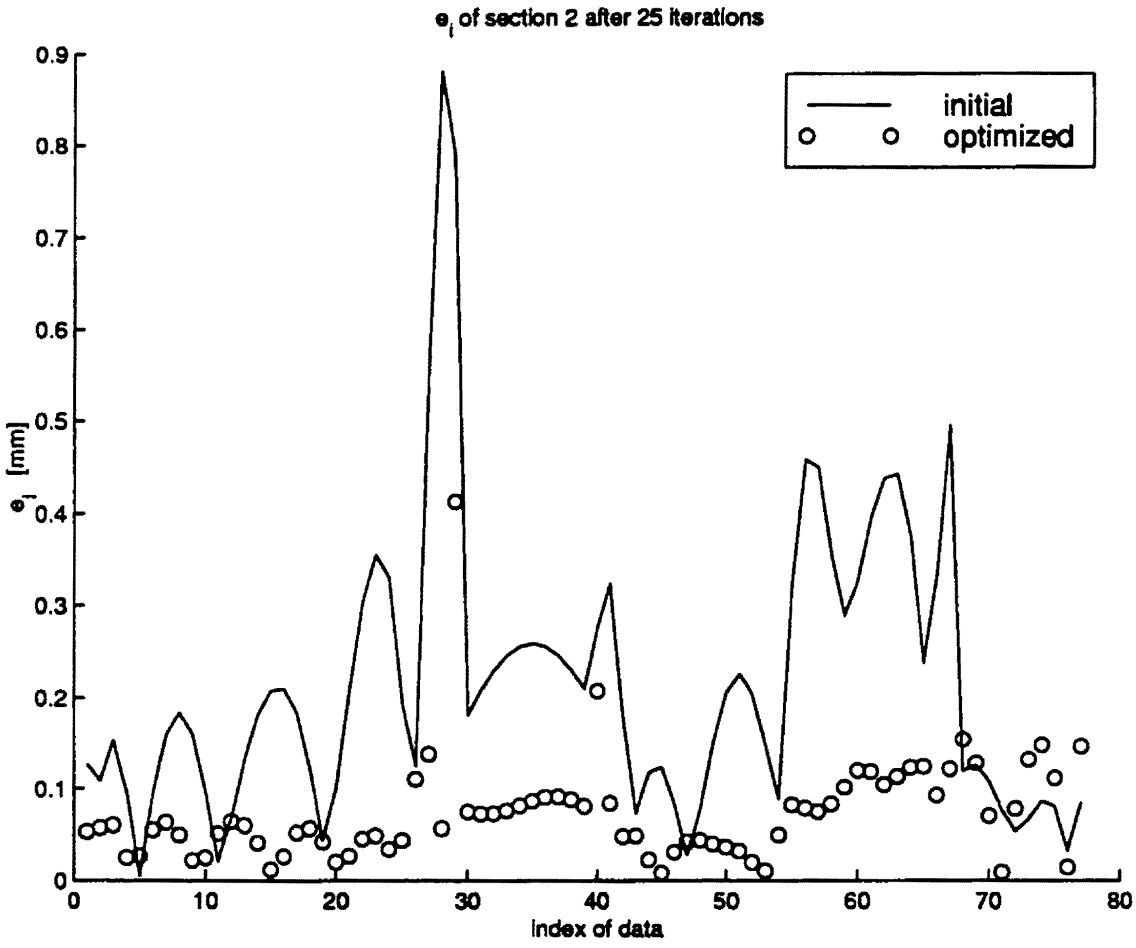


Figure 5.69: Error distribution of midsection airfoil of three-curve turbine blade obtained from optimizing the weights.

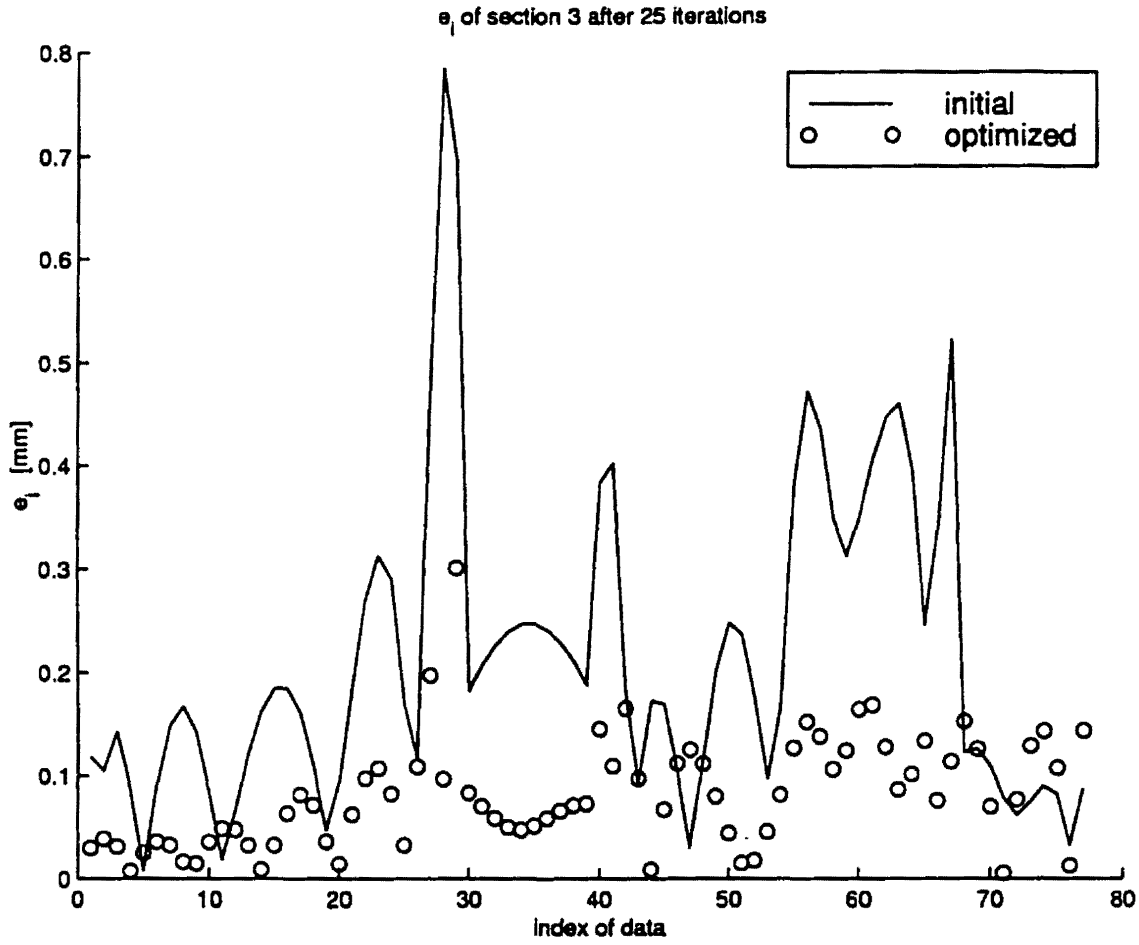


Figure 5.70: Error distribution of tip airfoil of three-curve turbine blade obtained from optimizing the weights.

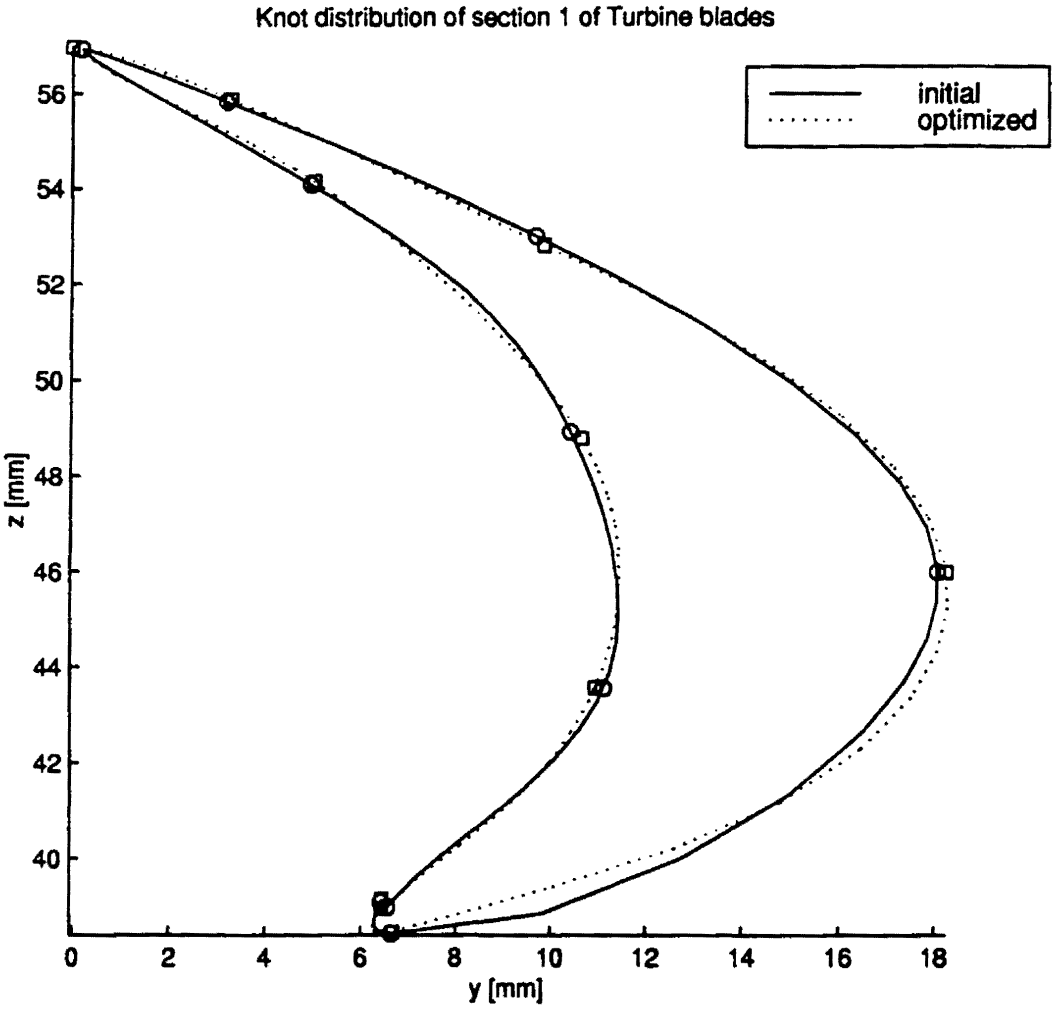


Figure 5.71: Fitted curves of hub airfoil of turbine blade obtained from optimizing weights.

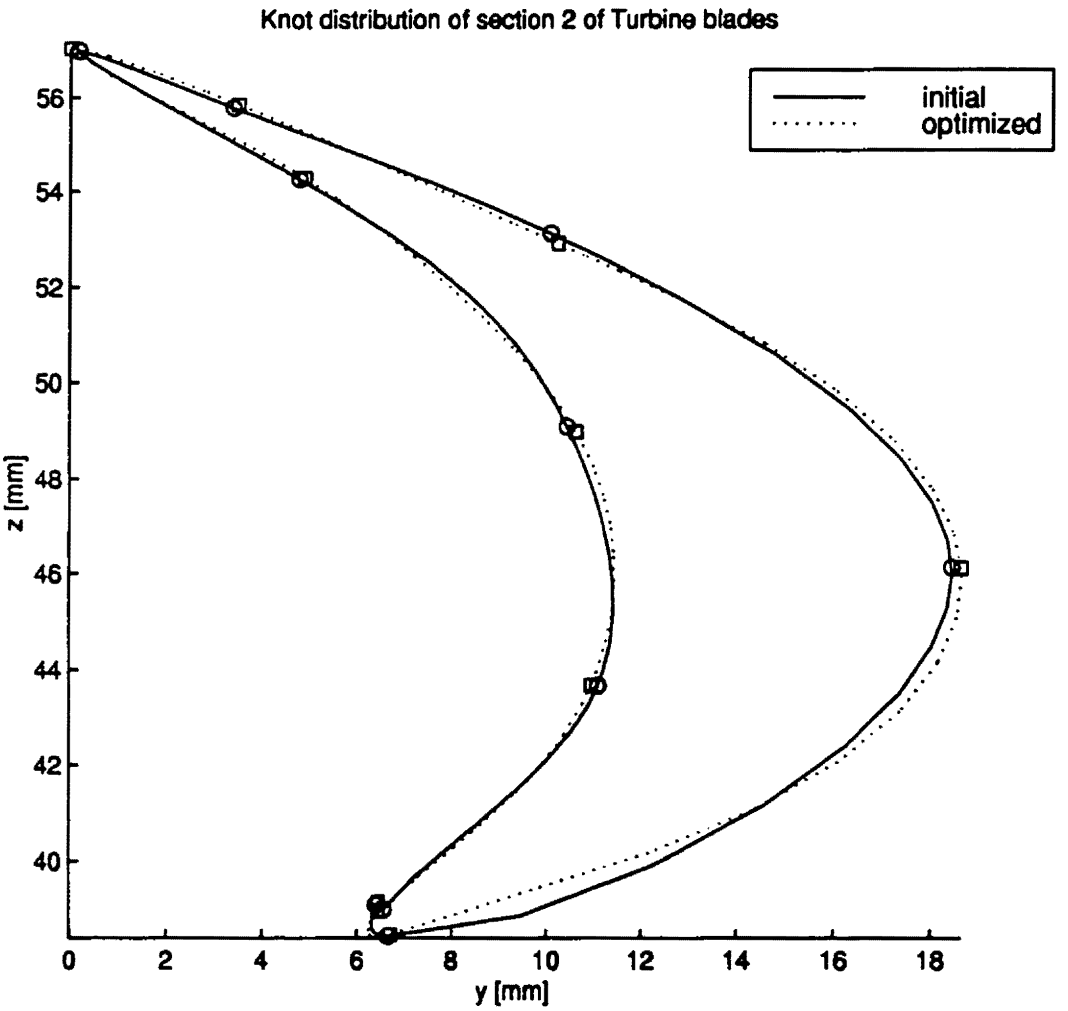


Figure 5.72: Fitted curves of mid section airfoil of turbine blade obtained from optimizing weights.

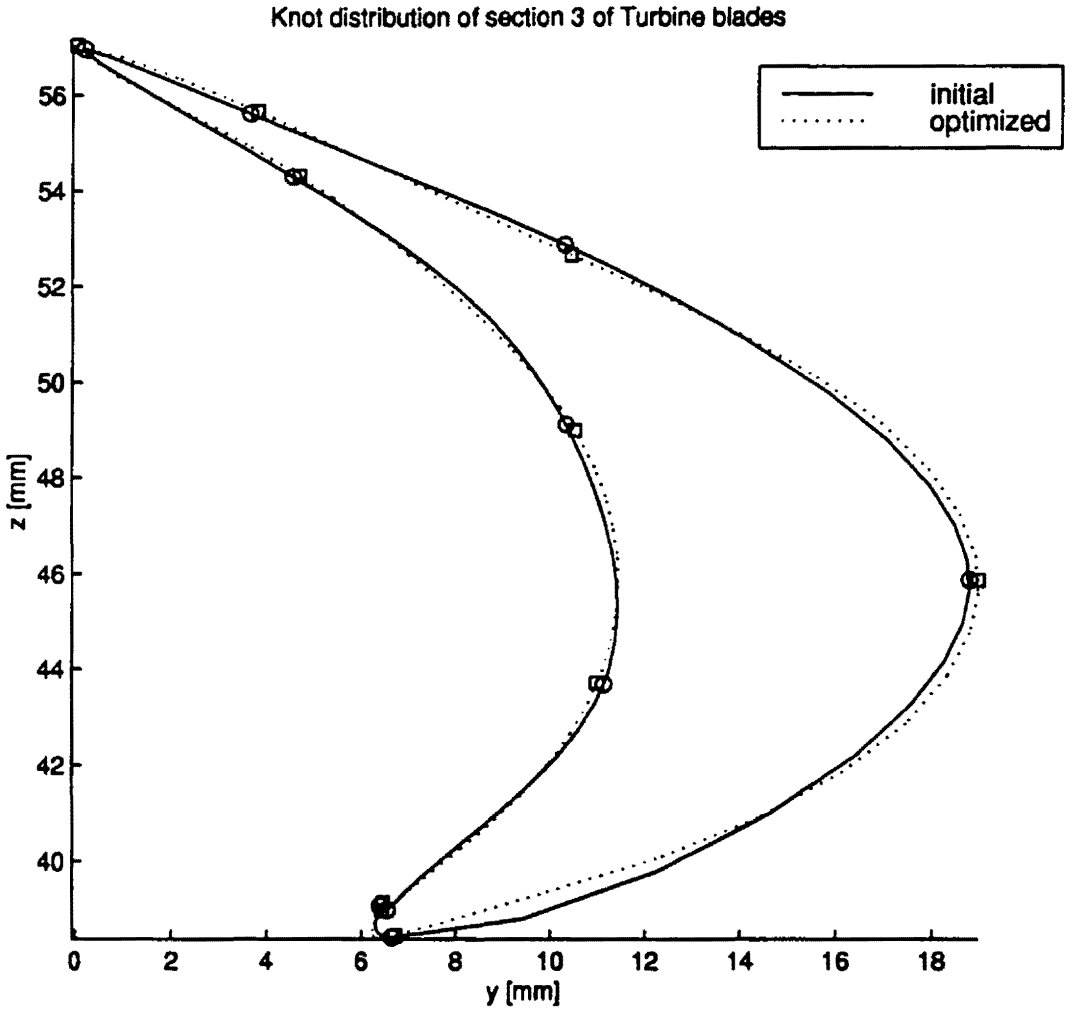


Figure 5.73: Fitted curves of tip airfoil of turbine blade obtained from optimizing weights.

5.4.3 Optimization of Parameterization

The sum of least-square error decreased from 16.8 [mm^2] to 15.6 [mm^2] in one iteration as shown in Figure 5.74. The largest distances between the data and the approximation curves **increased (!)** from 0.983 millimeters to 1.31 millimeters for hub airfoil, from 0.881 millimeters to 0.1.24 millimeters for mid section airfoil, and from 0.785 millimeters to 1.06 millimeters for tip airfoil, as shown in Figures 5.75, 5.76, and 5.77. The shapes of the sections, along with their junctions, are shown in Figures 5.78, 5.79, and 5.80.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after two iteration.

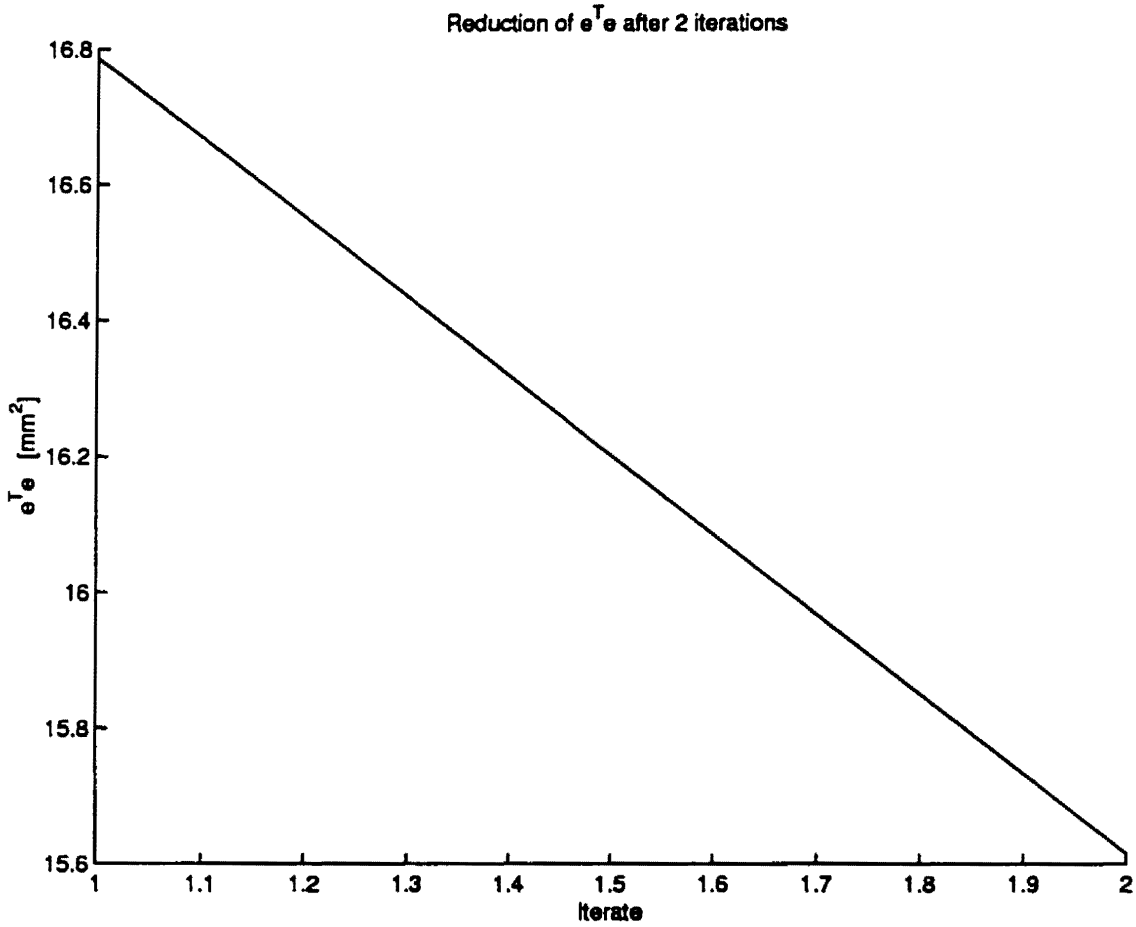


Figure 5.74: Reduction of error vs. number of iteration for three-curve turbine blade's airfoils obtained from optimizing the parameterization for data

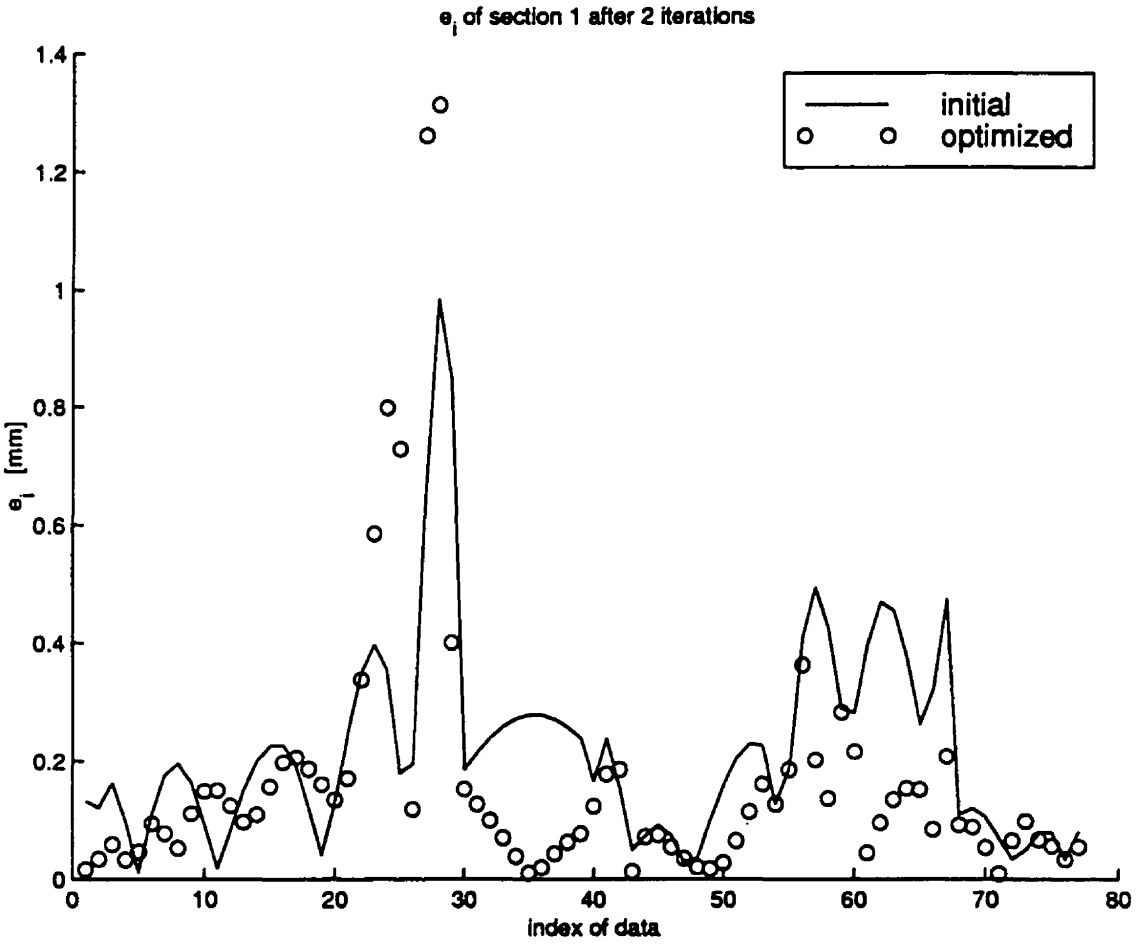


Figure 5.75: Error distribution of hub airfoil of three-curve turbine blade obtained from optimizing the parameterization for data.

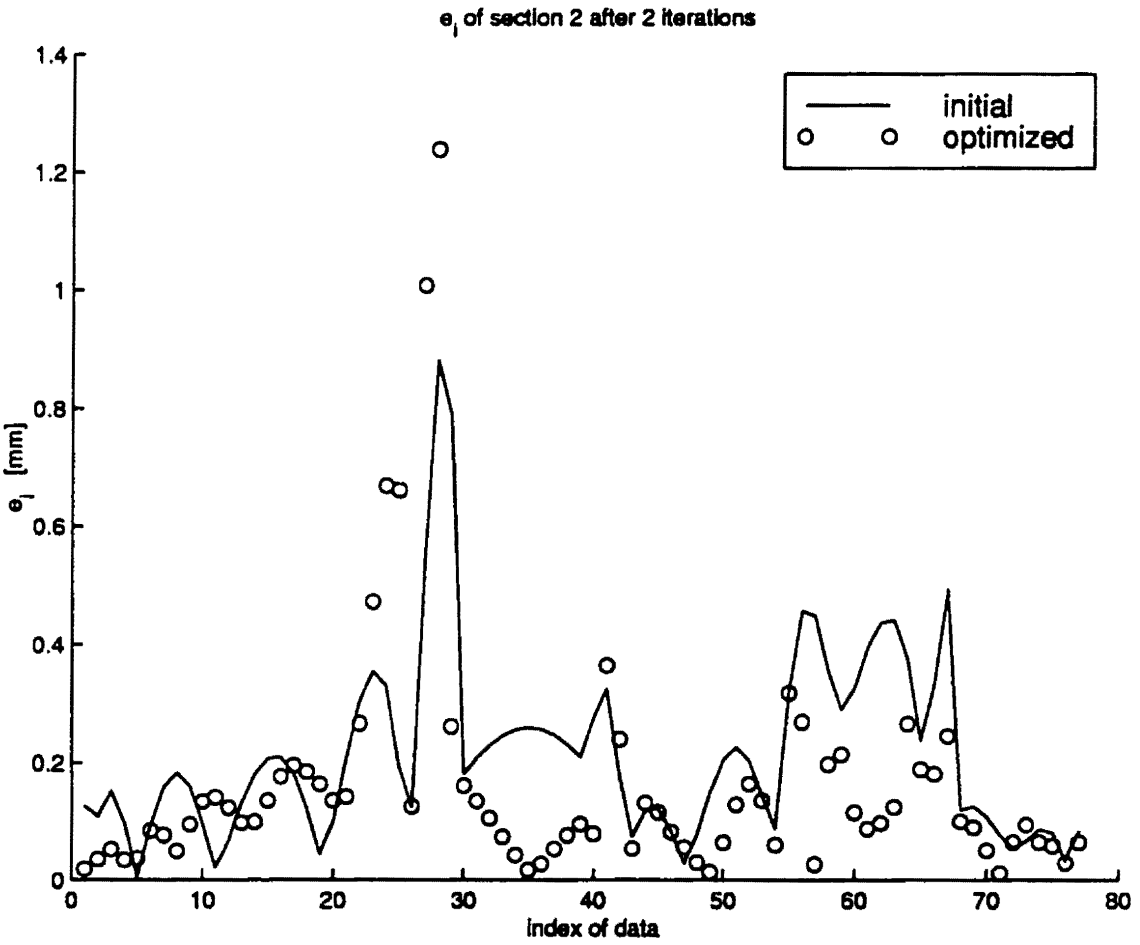


Figure 5.76: Error distribution of midsection airfoil of three-curve turbine blade obtained from optimizing the parameterization for data.

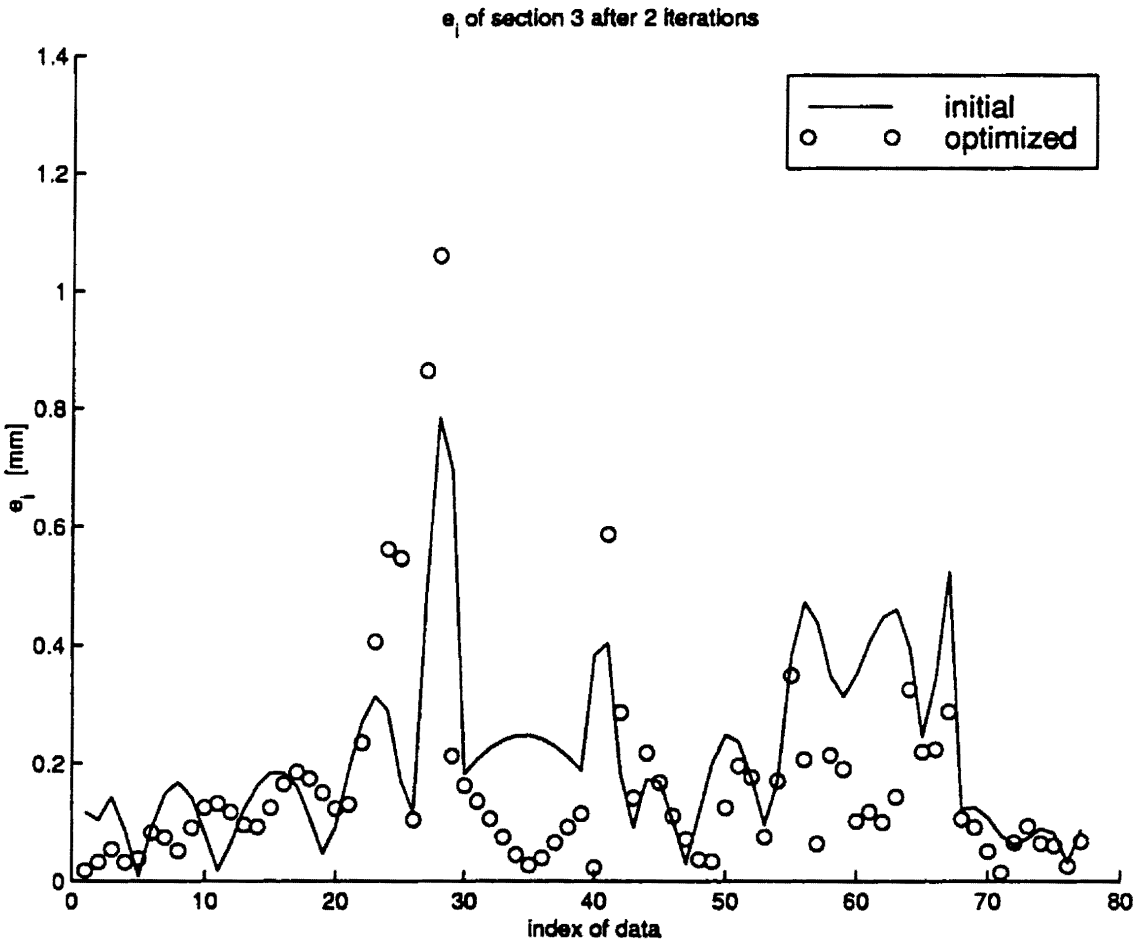


Figure 5.77: Error distribution of tip airfoil of three-curve turbine blade obtained from optimizing the parameterization for data.

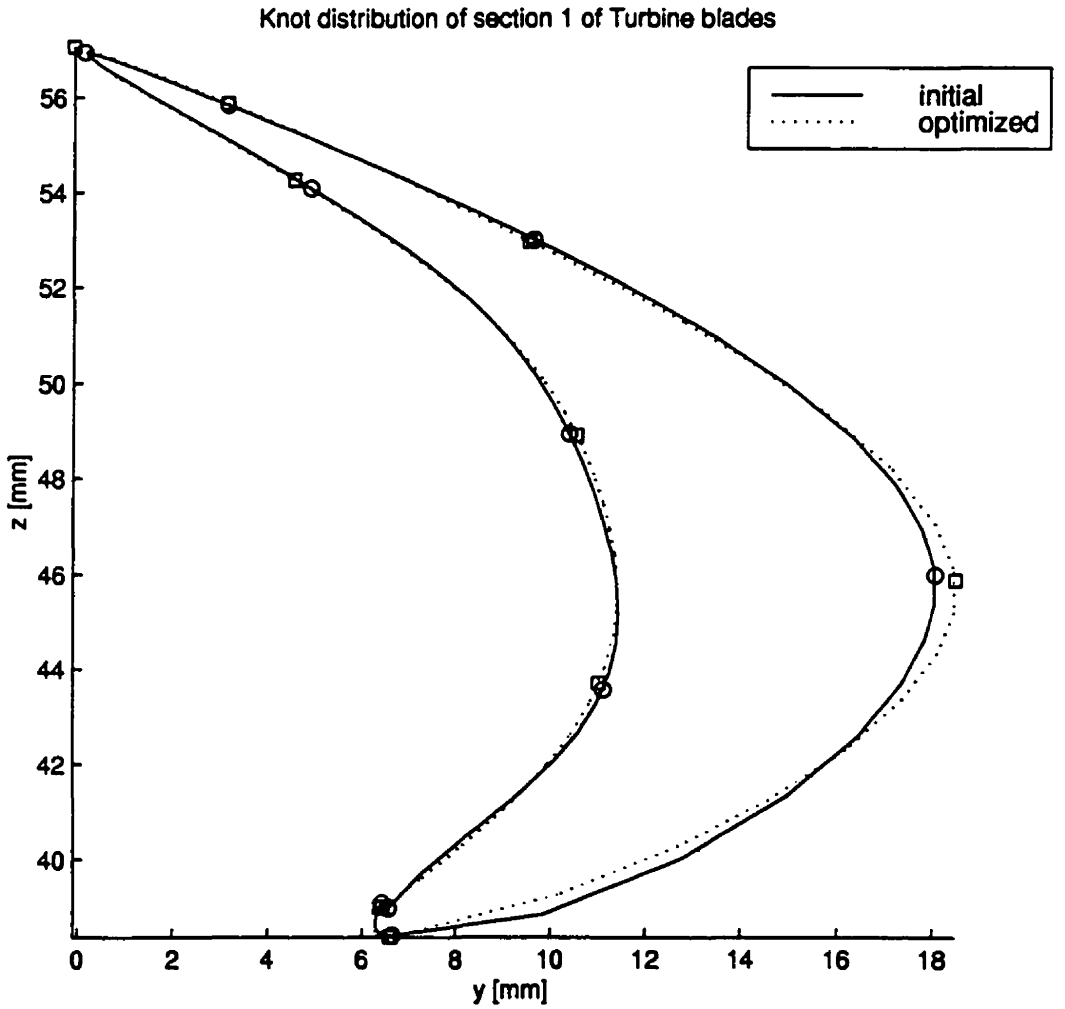


Figure 5.78: Fitted curves of hub airfoil of turbine blade obtained from optimizing parameterization for data.

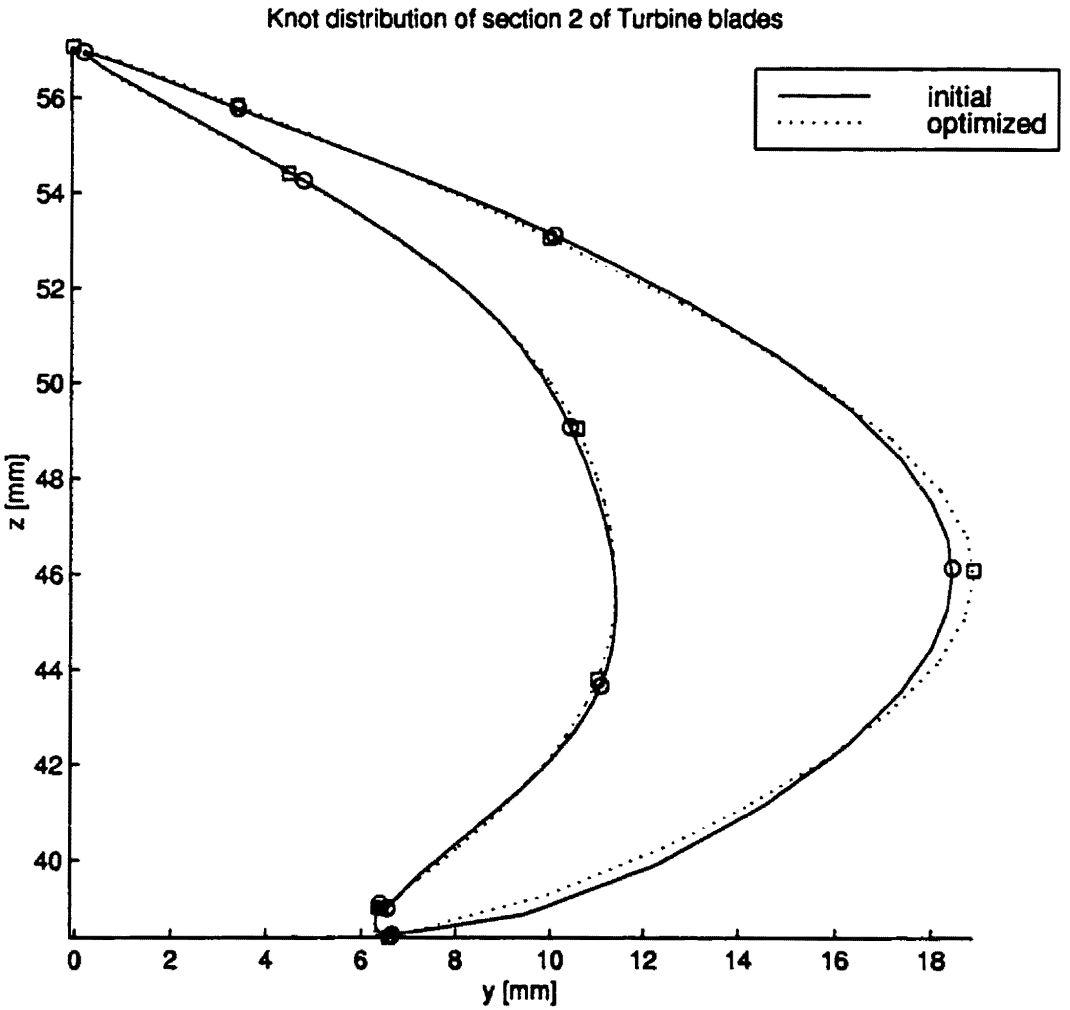


Figure 5.79: Fitted curves of mid section airfoil of turbine blade obtained from optimizing parameterization for data.

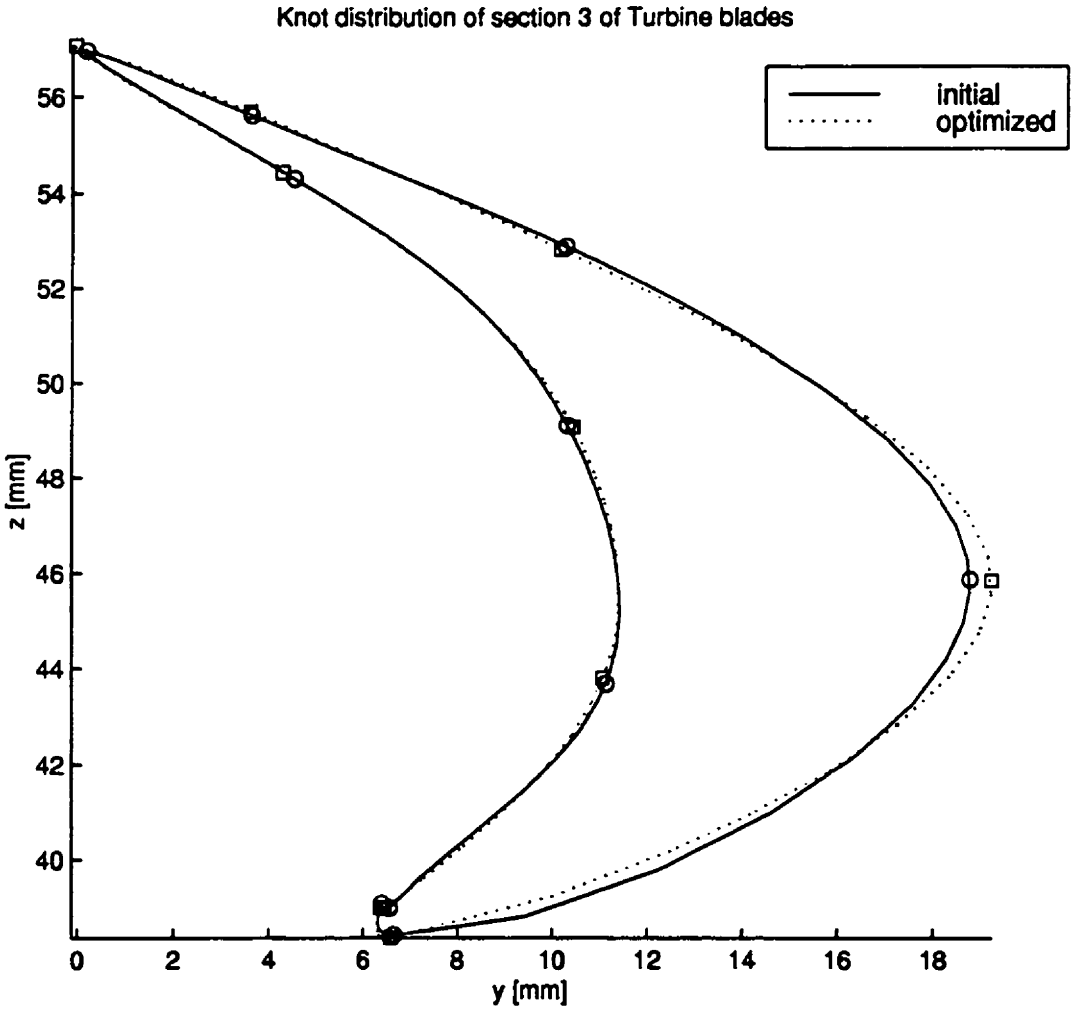


Figure 5.80: Fitted curves of tip airfoil of turbine blade obtained from optimizing parameterization for data.

5.4.4 Optimization of Knots and Weights

The sum of least-square error decreased from 16.8 [mm^2] to 4.470 [mm^2] in 51 iteration as shown in Figure 5.81. The largest distances between the data and the approximation curves decreased from 0.983 millimeters to 0.423 millimeters for hub airfoil, from 0.881 millimeters to 0.453 millimeters for mid section airfoil, and from 0.785 millimeters to 0.495 millimeters for tip airfoil, as shown in Figures 5.82, 5.83, and 5.84, respectively. The shapes of the sections, along with their junctions, are shown in Figures 5.85, 5.86, and 5.87. Active knot constraints are $u_7 = u_8 = u_9$. Change of weights are less than 3 percent for the three section curves.

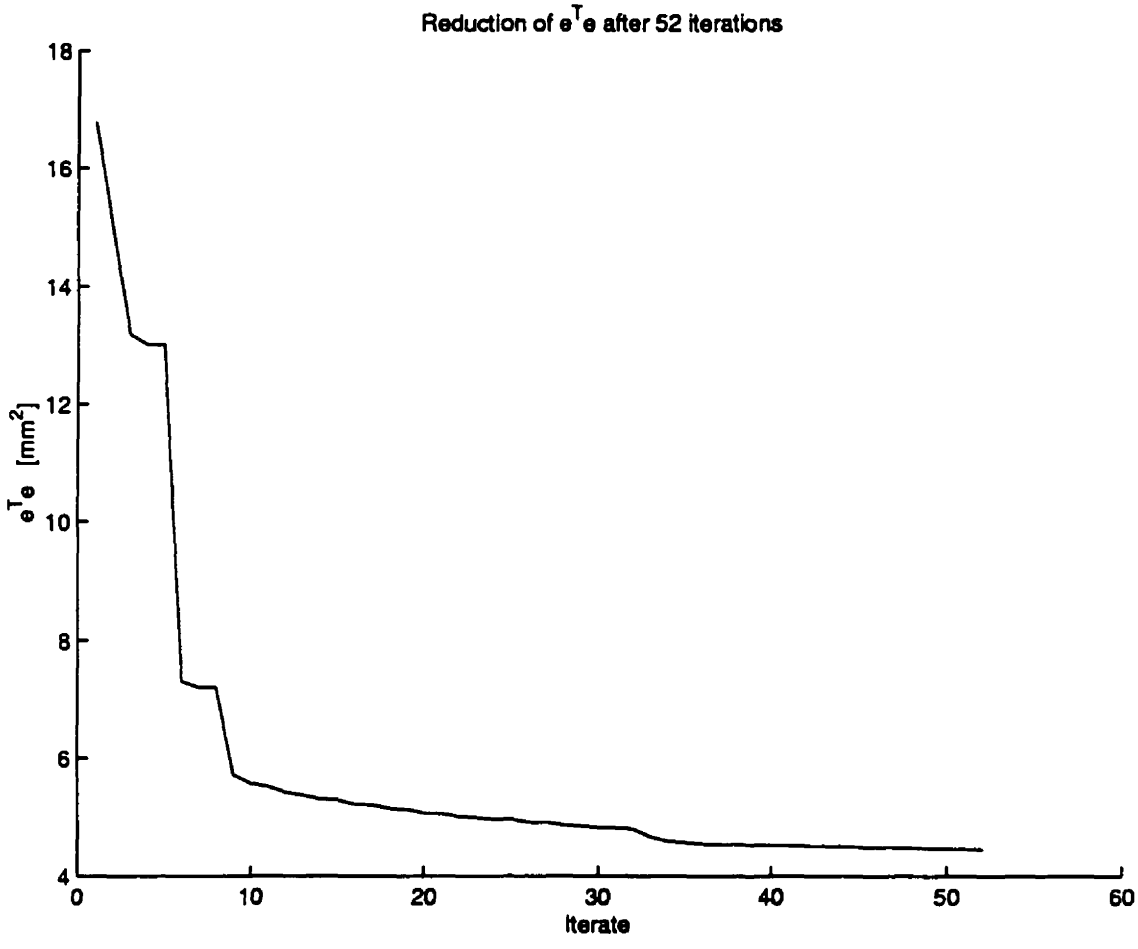


Figure 5.81: Reduction of error vs. number of iteration for three-curve turbine blade's airfoils obtained from optimizing the combination of knots and weights

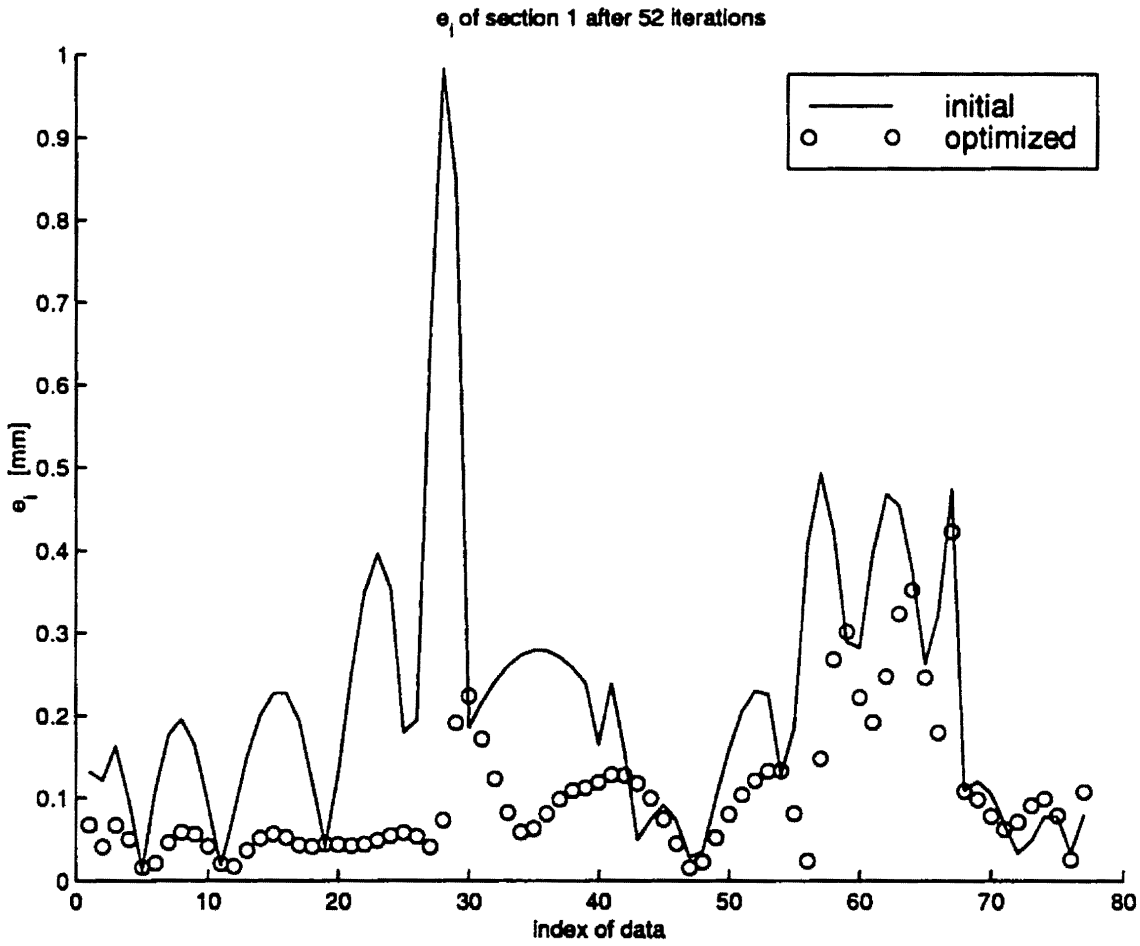


Figure 5.82: Error distribution of hub airfoil of three-curve turbine blade obtained from optimizing the combination of knots and weights.

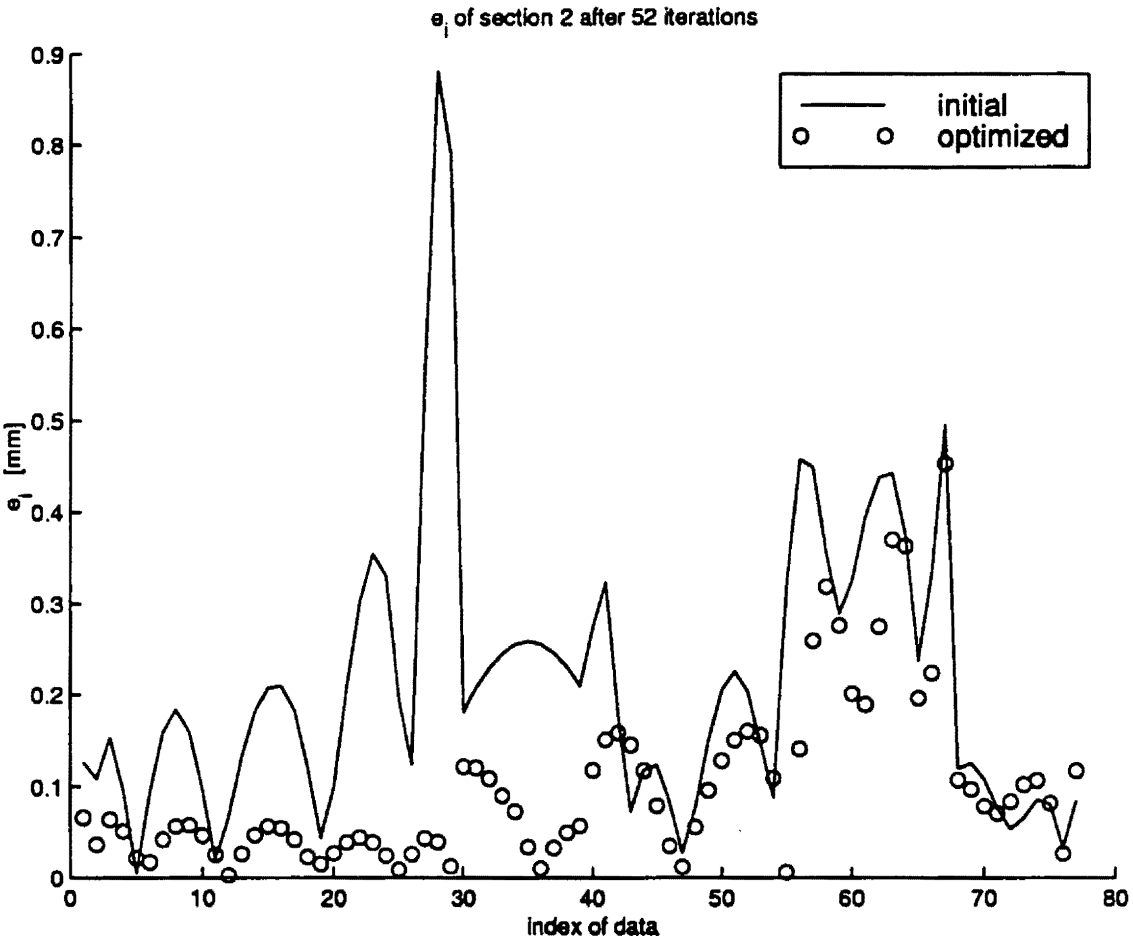


Figure 5.83: Error distribution of midsection airfoil of three-curve turbine blade obtained from optimizing the combination of knots and weights .

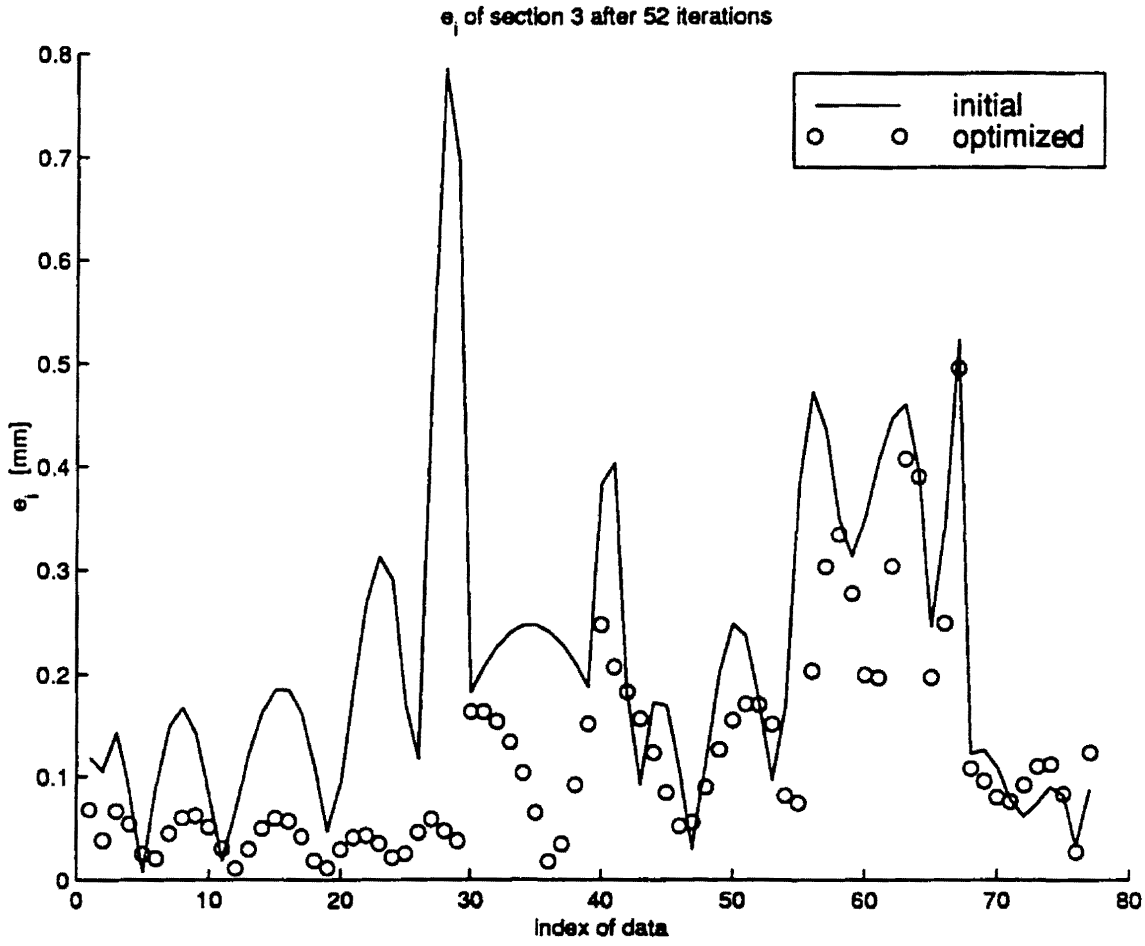


Figure 5.84: Error distribution of tip airfoil of three-curve turbine blade obtained from optimizing the combination of knots and weights.

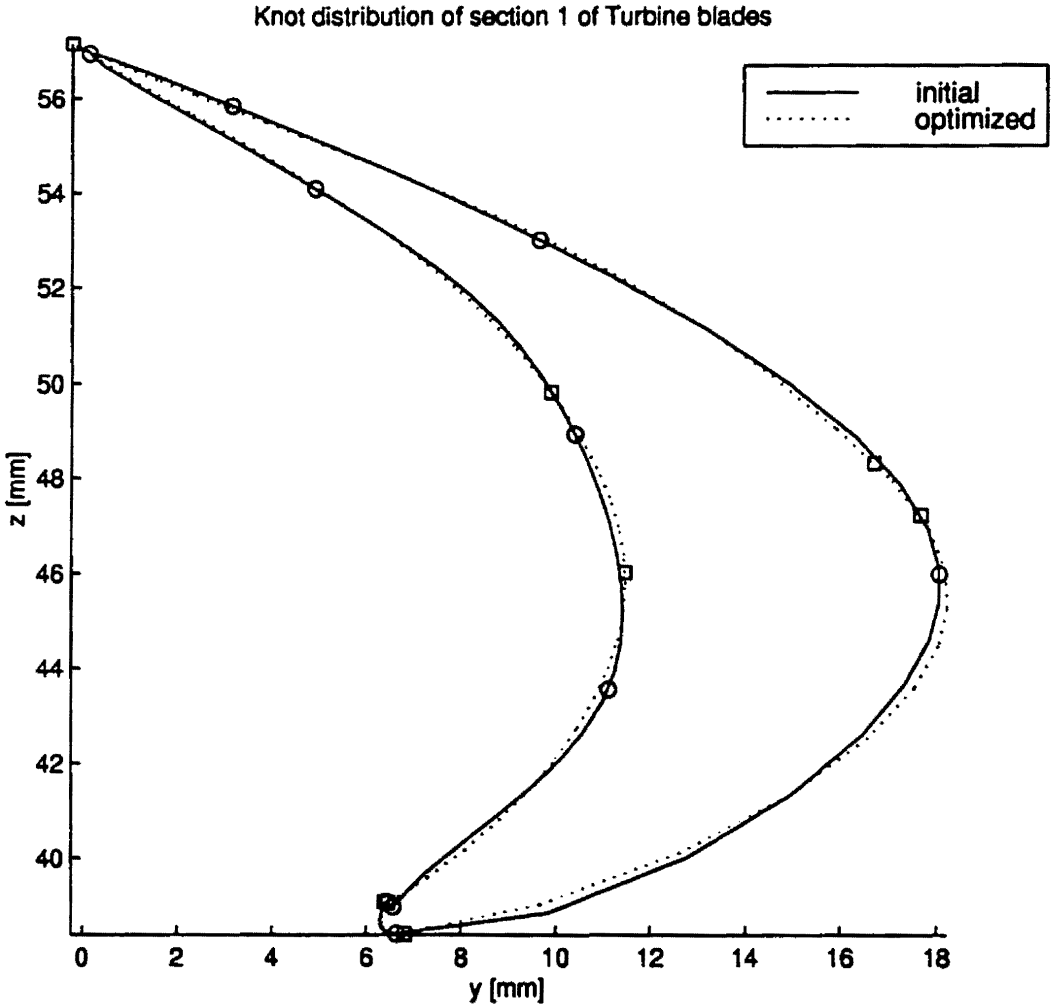


Figure 5.85: Fitted curves of hub airfoil of turbine blade obtained from optimizing combination of knots and weights.

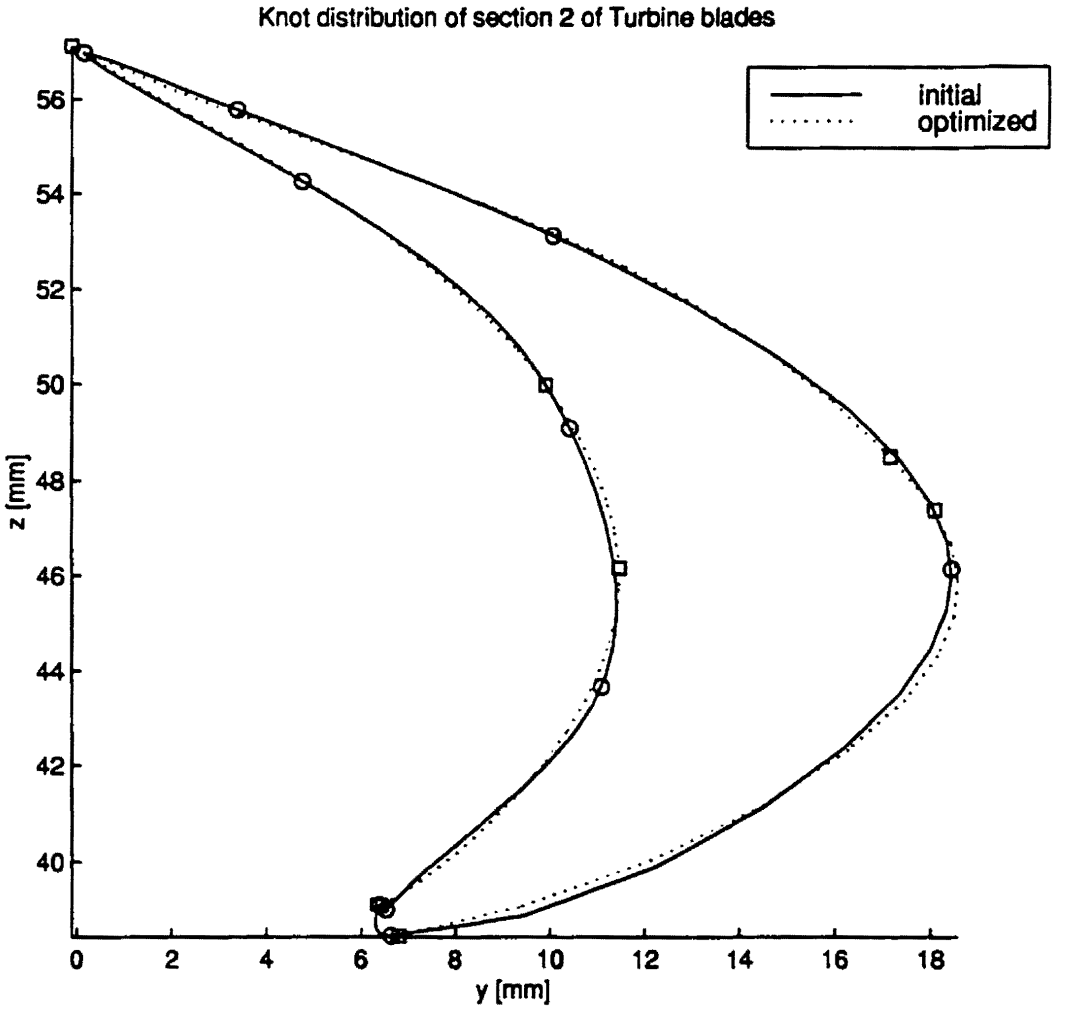


Figure 5.86: Fitted curves of mid section airfoil of turbine blade obtained from optimizing combination of knots and weights.

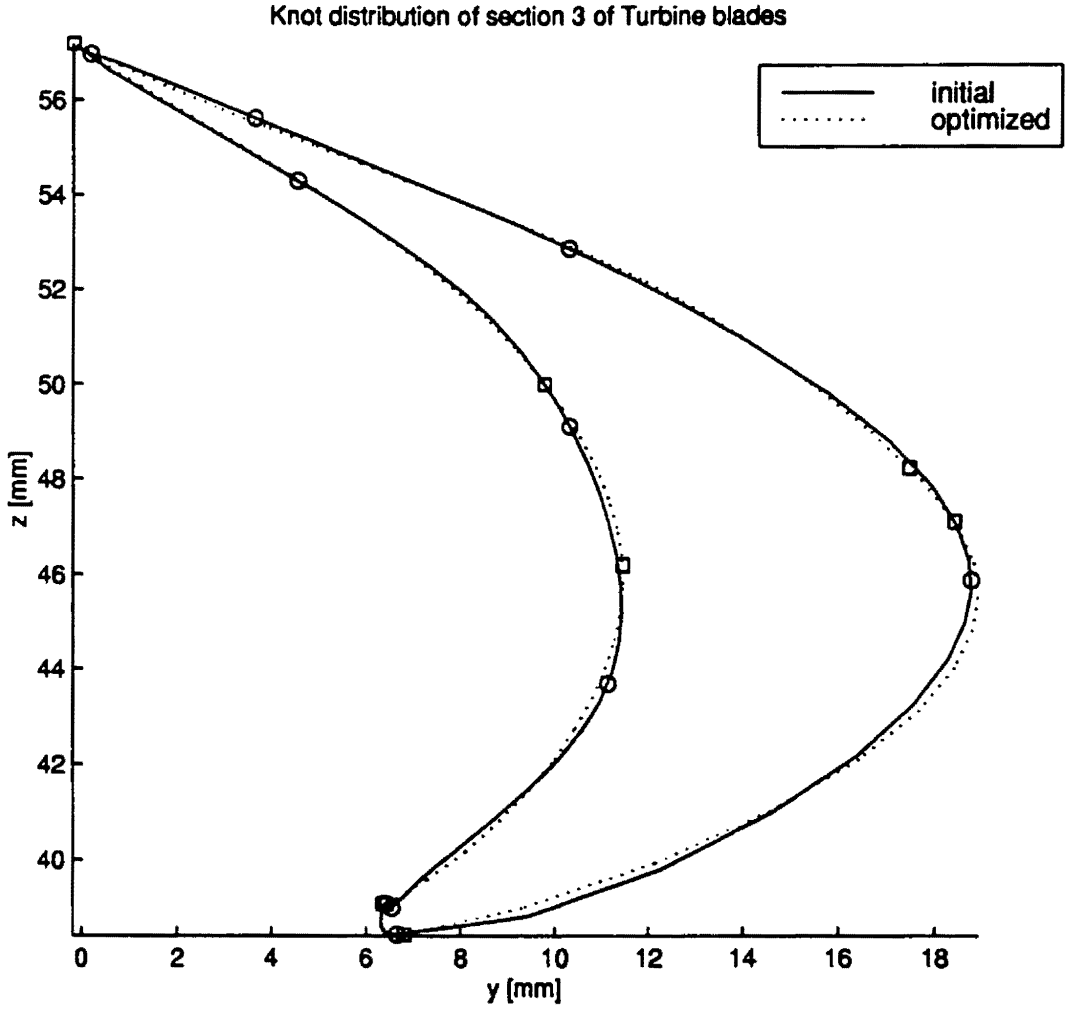


Figure 5.87: Fitted curves of tip airfoil of turbine blade obtained from optimizing combination of knots and weights.

5.4.5 Optimization of Knots, Weights, and Parameterization

The sum of least-square error decreased from 16.8 [mm^2] to 5.52 [mm^2] in 1 iteration as shown in Figure 5.88. The largest distances between the data and the approximation curves decreased from 0.983 millimeters to 0.502 millimeters for hub airfoil, from 0.881 millimeters to 0.556 millimeters for mid section airfoil, and from 0.785 millimeters to 0.615 millimeters for tip airfoil, as shown in Figures 5.89, 5.90, and 5.91. The shapes of the sections, along with their junctions, are shown in Figures 5.92, 5.93, and 5.94.

Optimization of parameters exhibited a phenomena of ill-conditioned Hessian. This phenomena forced our algorithm to reset the approximate Hessian at every iteration, leading to the use of the steepest-descent direction instead of the BFGS one. Backtrack algorithm failed to obtain adequate step length after two iteration.

5.5 Discussion

Results of the experiments are tabulated in Tables 5.1, 5.2, and 5.3, each of which lists the least-squares error, the maximum distance between data and the approximating curves, and the number of iteration for parameter adjustment respectively.

These results must be analyzed based on three criteria:

- performance of the approximation,
- the sensitivity of various combinations of optimized parameters, and

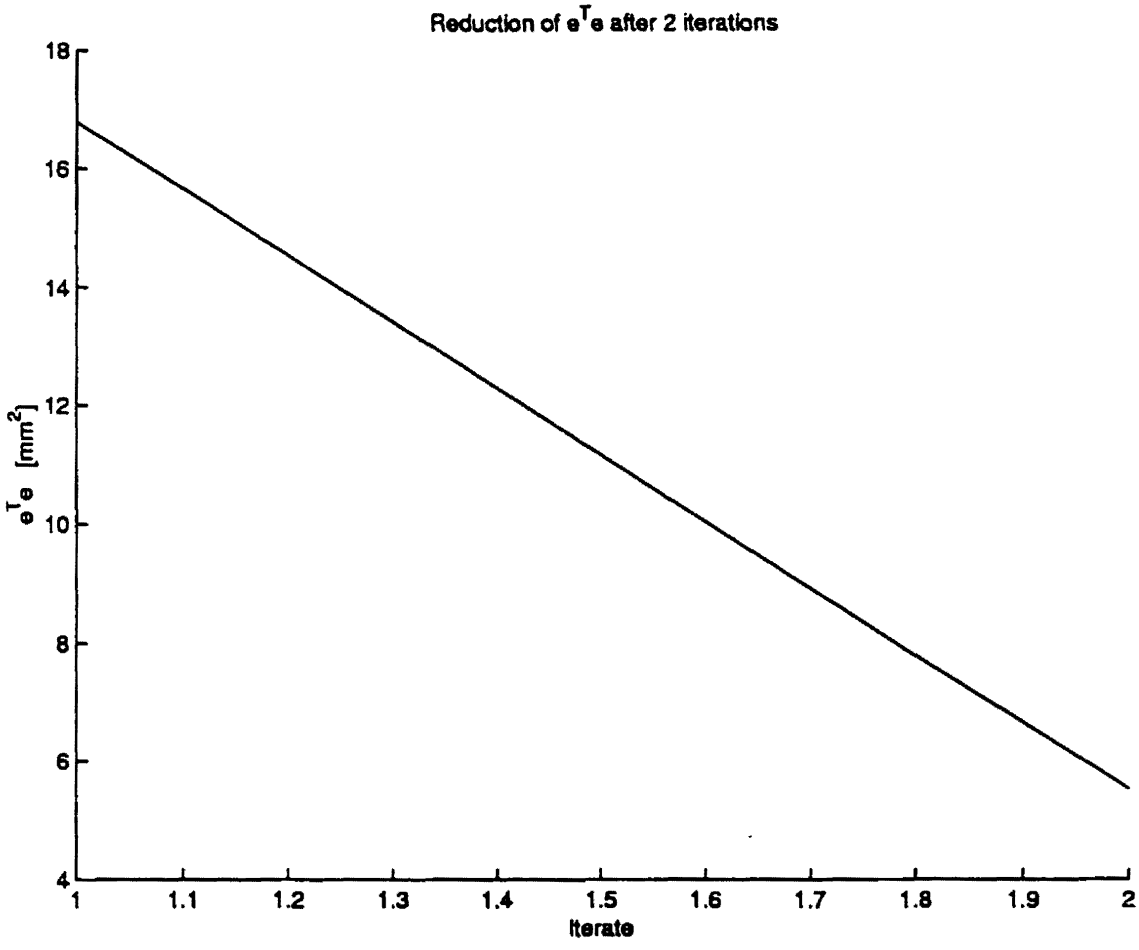


Figure 5.88: Reduction of error vs. number of iteration for three-curve turbine blade's airfoils obtained from optimizing the combination of knots, weights, and parameterization for data

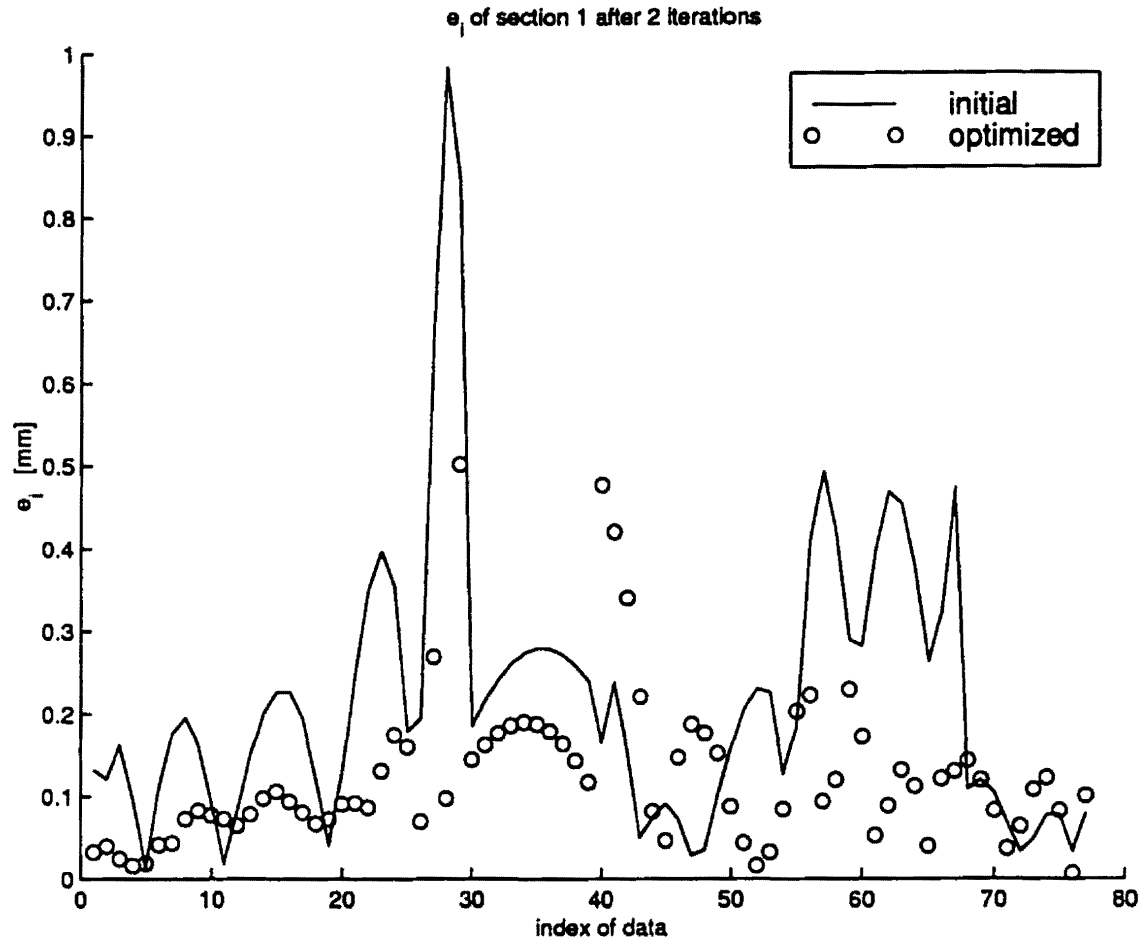


Figure 5.89: Error distribution of hub airfoil of three-curve turbine blade obtained from optimizing the combination of knots, weights, and parameterization for data.

Table 5.1: Least Square Error of Three Skeleton and Five Combinations of Adjustable Parameters

	Residual $\sum_{i=1}^{n} \left(\mathbf{e}_x^T\mathbf{e}_x + \mathbf{e}_y^T\mathbf{e}_y + \mathbf{e}_z^T\mathbf{e}_z\right)_i$ [mm^2]					
		Adjusted parameters				
Data	Initial	**u**	**w**	**t**	**(u, w)**	**(u, w, t)**
Airfoils	0.771	0.005	0.073	0.240	0.006	0.199
Turbine blades	16.80	4.98	2.07	15.6	4.47	5.52
Compressor blades	27.5	1.53	3.45	6.56	1.52	7.34

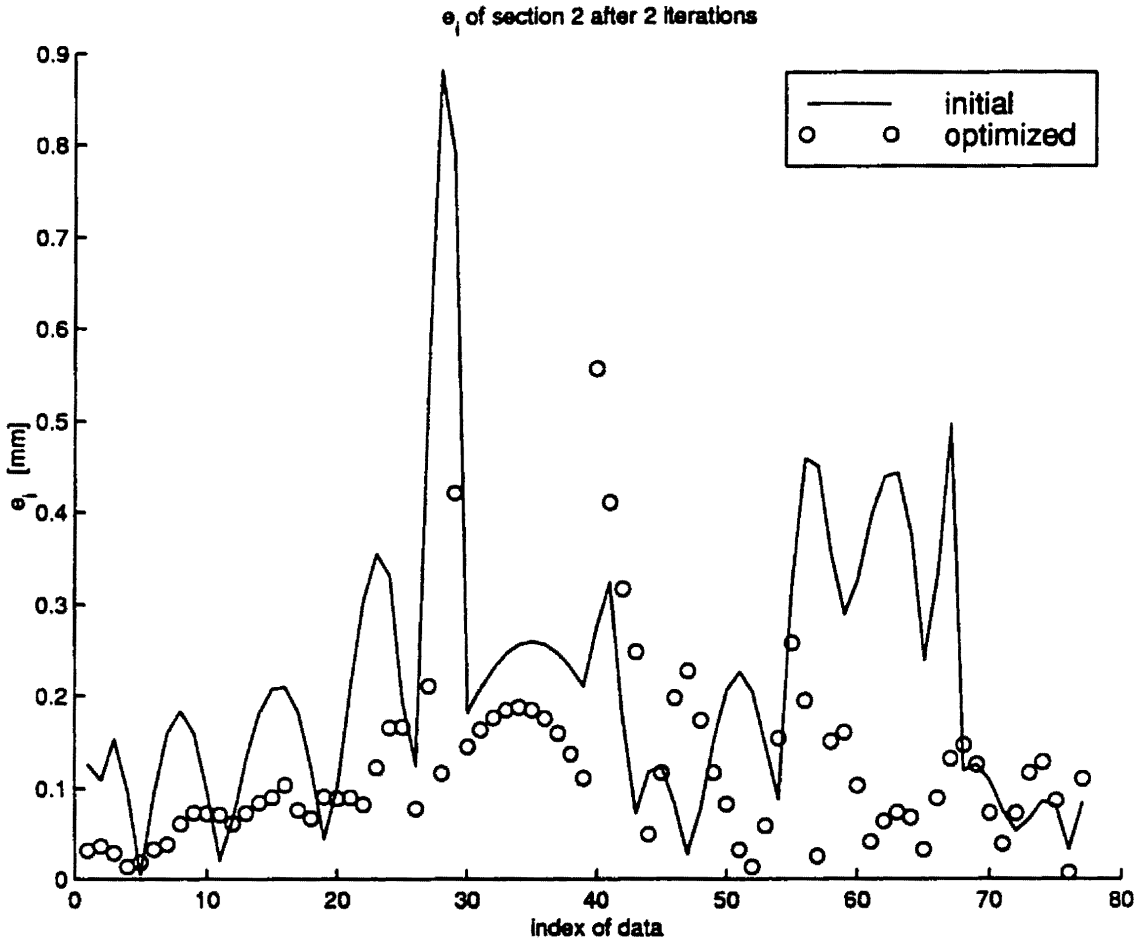


Figure 5.90: Error distribution of midsection airfoil of three-curve turbine blade obtained from optimizing the combination of knots, weights, and parameterization for data.

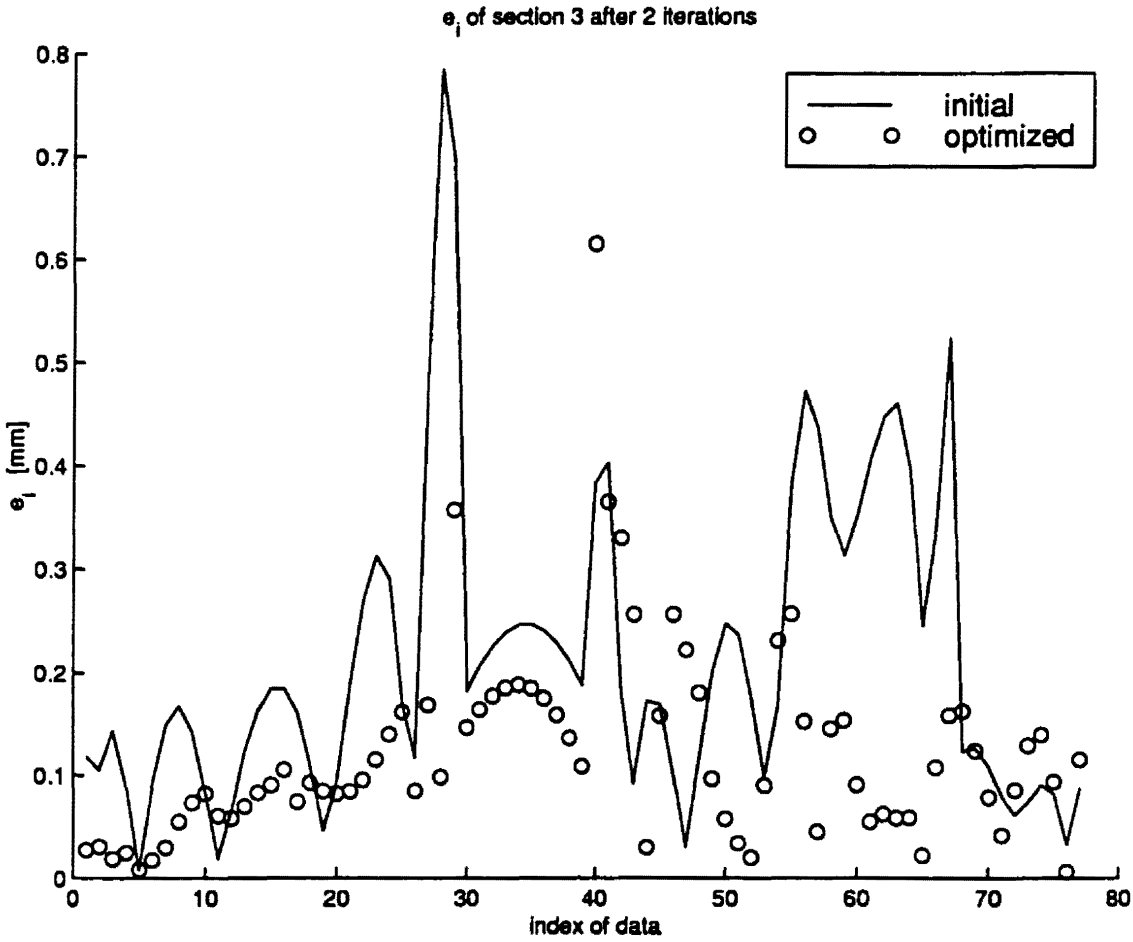


Figure 5.91: Error distribution of tip airfoil of three-curve turbine blade obtained from optimizing the combination of knots, weights, and parameterization for data.

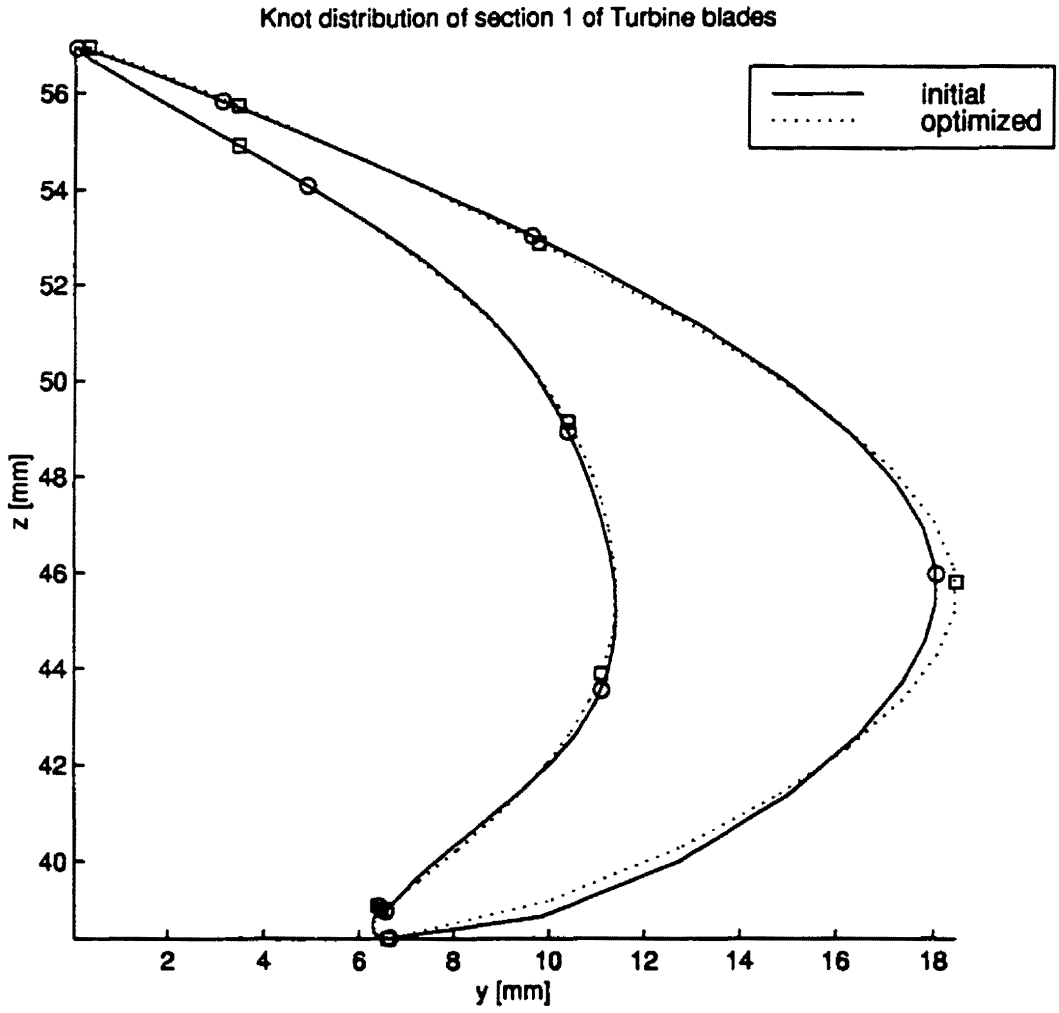


Figure 5.92: Fitted curves of hub airfoil of turbine blade obtained from optimizing combination of knots, weights, and parameterization for data.

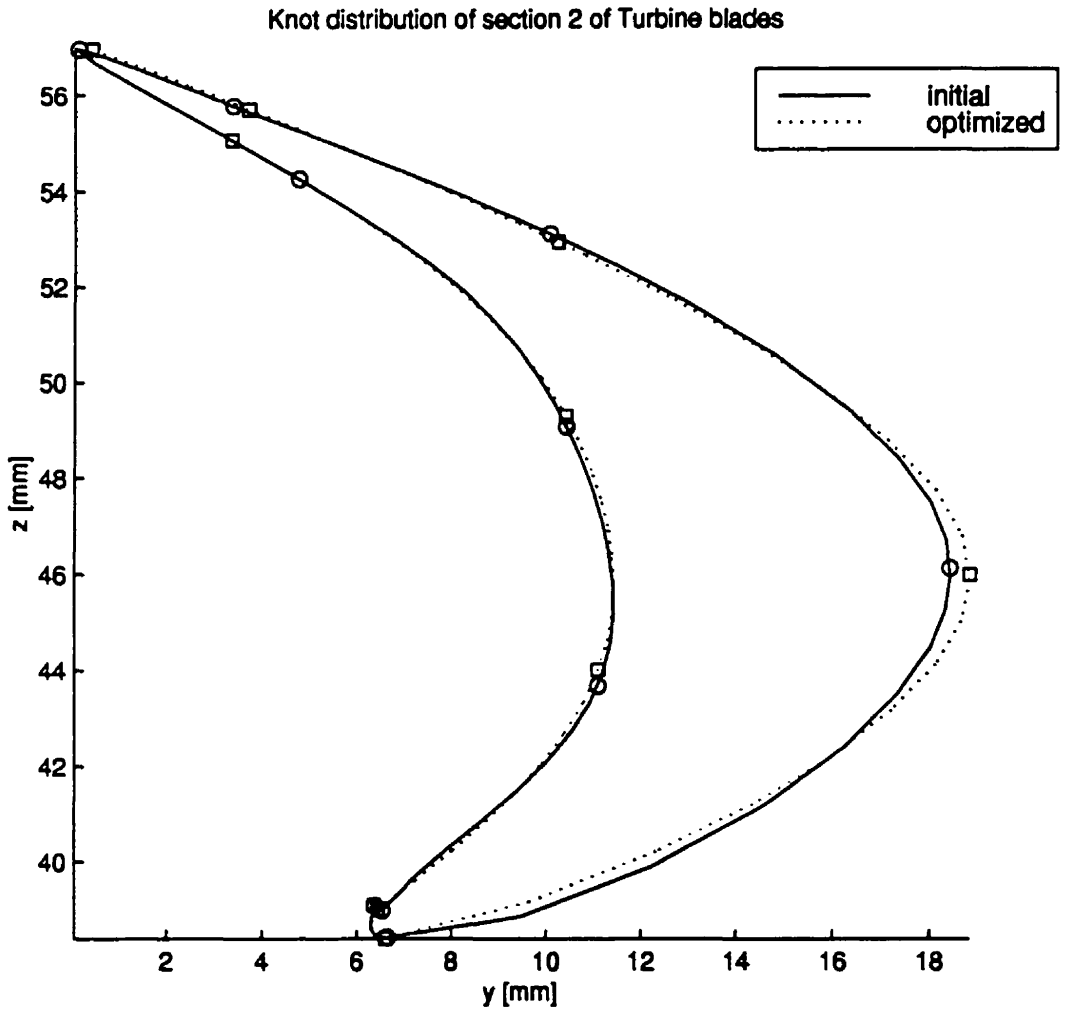


Figure 5.93: Fitted curves of mid section airfoil of turbine blade obtained from optimizing combination of knots, weights, and parameterization for data.

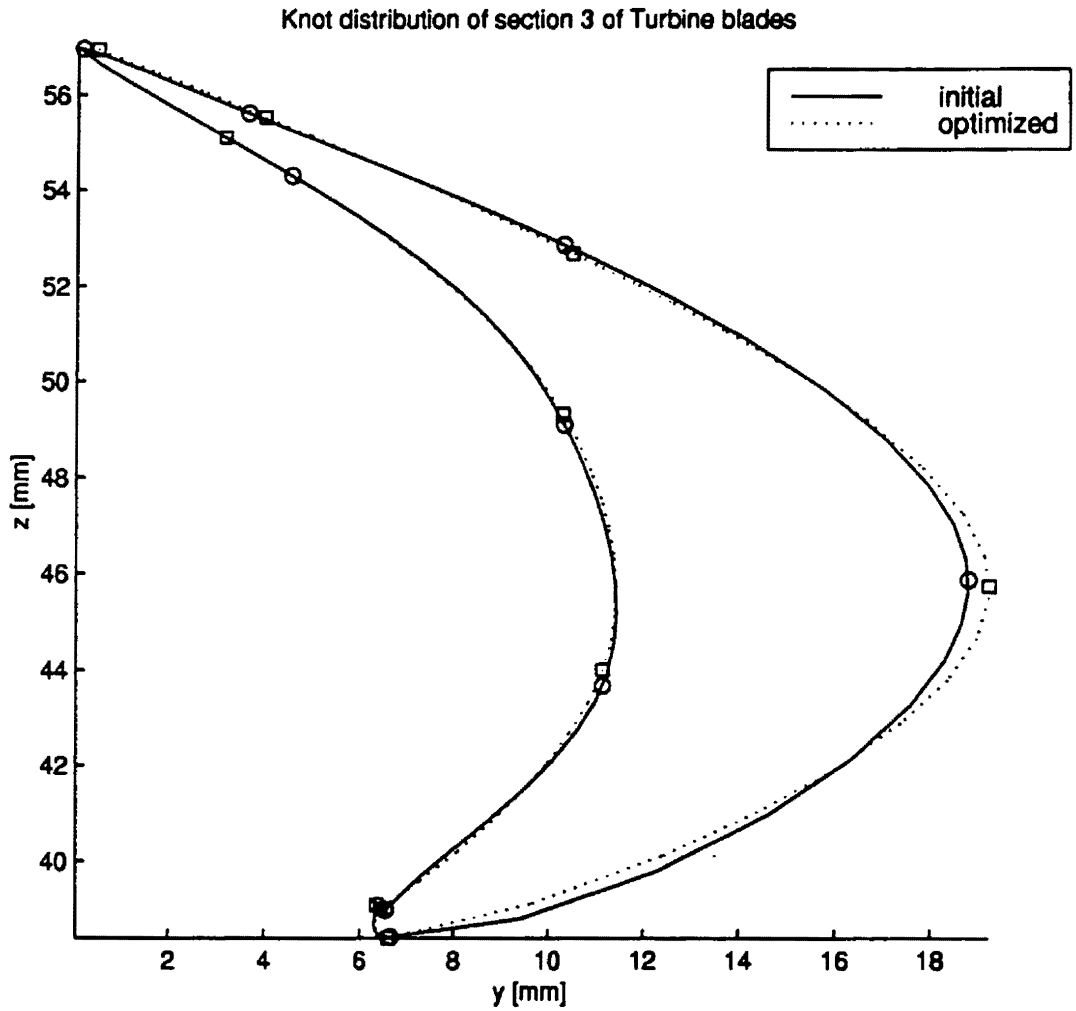


Figure 5.94: Fitted curves of tip airfoil of turbine blade obtained from optimizing combination of knots, weights, and parameterization for data.

Table 5.2: Maximum Distances Between Three Skeletons and The Approximating Curves for Five Combinations of Adjustable Parameters

	Maximum distance between data and curve [mm]					
		Adjusted parameters				
Data	Initial	u	w	t	(u, w)	(u, w, t)
Airfoils:						
NACA	0.244	0.015	0.063	0.158	0.027	0.143
WTEA	0.158	0.011	0.038	0.080	0.014	0.073
Turbine Blades:						
Section 1	0.983	0.438	0.416	1.310	0.423	0.502
Section 2	0.881	0.476	0.413	1.240	0.453	0.556
Section 3	0.785	0.521	0.301	1.060	0.495	0.615
Compressor Blades:						
Section 1	1.16	0.359	0.485	0.661	0.344	0.871
Section 2	1.25	0.220	0.424	0.774	0.235	0.871
Section 3	1.19	0.268	0.430	0.776	0.276	0.795

Table 5.3: Number of Iteration for Three Skeletal Curves and Five Combinations of Adjustable Parameters

	Adjusted parameters				
Data	u	w	t	(u, w)	(u, w, t)
Airfoils	51	51	2	42	5
Turbine blades	51	24	1	51	1
Compressor blades	51	51	2	19	2

- the effect of constraining the condition number of matrix **R**.

The performance of the approximation showed that the least squares error is reduced by a factor of 154 for the wing skeleton. This factor is obtained by dividing the initial error by the best final error from various combinations of adjustable parameters. The factor of error reduction is 3.8 and 18 for the turbine and the axial compressor blades respectively.

Table 5.4 presents the factors of reduction of maximum distance between the curve and the data. The largest factors of reduction consistently occur under column **u**, i.e. when optimization is performed only on knots. For the two sections of the wing, the factors of reduction are 16.3 and 14.4 respectively. For the three sections of the turbine blade, these factors are 2.2, 1.9, and 1.5, respectively. For the three sections of the compressor blade, these factors are 3.2, 5.7, and 4.4.

Table 5.1, 5.2, and 5.4, suggest that the best results (smallest residuals and smallest maximum error) are obtained from knot adjustment and knot-weight adjustment as shown in columns **u** and $(\mathbf{u}, \mathbf{w})$. However, these columns also show that knot-weight adjustment produce only marginal improvement of accuracy over knot adjustment. This phenomena is consistent for the three different sets of data. Sections 5.2.4, 5.3.4, and 5.4.4, show that weights are practically unchanged on the knot-weight adjustments.

This behavior can be used to justify the exclusion of weights from the adjustable parameters whenever knots are optimized. The effect of this exclusion for low order curves, whose orders are significantly smaller than their number of basis, is that the adjustable parameters will be halved. This will result in faster computation.

Table 5.4: Reduction Maximum Distances Between Three Sets of Data and Their Approximation for Five Sets of Adjusted Parameters

	Maximum distance in mm				
	Adjusted parameters				
Data	**u**	**w**	**t**	**(u, w)**	**(u, w, t)**
Airfoils:					
NACA	16.3	3.9	1.5	9.0	1.7
WTEA	14.4	4.2	2.0	11.3	2.2
Turbine:					
Section 1	2.2	2.4	0.75	2.3	2.0
Section 2	1.9	2.1	0.7	1.9	1.6
Section 3	1.5	2.6	0.7	1.6	1.3
Compressor:					
Section 1	3.2	2.4	1.8	3.4	1.3
Section 2	5.7	2.9	1.6	5.3	1.4
Section 3	4.4	2.8	1.5	4.3	1.5

Table 5.3 lists the number of iteration before our algorithm terminated the adjustment process. The table clearly shows that premature termination occured when adjustable parameters contain parameterization of data; this phenomena is shown in columns **t** and **(u, w, t)**.

Optimizing the parameterization for data, both individually and in combination with other parameter, caused premature termination of optimization as shown in Table 4.3. This premature termination consistently occured in the backtracking part of the optimization. Standard texts of optimization suggest that this early termination is due to nonlinearity of the parameterization for the data[31], and cannot be handled easily. This finding can be used to justify dropping the parameterization for data from the adjustable parameters of NURBS least squares problem. The most significant benefit from this elimination is a significant reduction of number

of adjustable parameters, which yields significantly lower computation time.

Columns **u**, **w**, and $(\mathbf{u}, \mathbf{w})$ in Tables 5.1 and 5.2 show the reduction of error where parameterization for the data were not adjusted. These columns clearly indicate that adjustment of knots produces significantly greater reduction of error than that of unadjusted knots, i.e. columns **w**. This finding can be used to justify that knots should be adjusted. The favorable reduction of error in this research's knot adjustment can be attributed to the proposed method of elimination of active knot constraints as described in Section 3.3.

Focusing on columns **u** and $(\mathbf{u}, \mathbf{w})$ in Table 4.1 and 4.2 shows that adjusting knots and weights simultaneously yields only marginal improvement in the reduction of least squares error over the error obtained from adjusting only the knots. This finding can be used to justify, along with those findings in previous paragraphs, the elimination of weights and parameterization for the data from the set of adjustable parameters. This significant elimination of parameters only results in insignificant loss of reduction of error. Moreover, this elimination reduces the number of adjustable parameters, which leads to significant reduction of computation time.

In conclusion, this chapter has demonstrated that the proposed methods to satisfy the linear constraint and to overcome the lethargic property are effective in reducing the least squares error. However, the most important feature of this proposed approach of multicurve fitting is the compatibility of the fitted curves. The results in this chapter have clearly indicated that optimization of the mutual knot vector improves the error distribution along the fitted curves. To the author's

knowledge, no literature is available for this compatible multicurve approximation. The advantages of this approach will be showed in the next chapter, in which these compatible curves are subjected to repetitive knots insertion to satisfy the prespecified tolerance.

Chapter 6

Tolerance Based Knot Insertion

In the previous chapter a skeleton of curves was created over a mutual knot vector and a mutual degree. These curves approximate a discrete data set of points. The accuracy of this approximation, measured by the largest deviation between discrete point and the closest point on the curve, must be controlled in many applications such as airplane wing design. The methods presented so far, fit the best approximation without changing the numbers of control points of the NURBS curves. These methods do not take into account the prespecified tolerance and may result in curves that violate this requirement.

This chapter presents a method which extends the multicurve approximation method to be able to approximate the set of skeletal curves to within a prespecified tolerance. The method is based on knot insertion.

The purpose of knot insertion is to increase the number of basis functions, and thereby the flexibility of the curve. The compatibility among skeletal curves must be maintained throughout the knot insertion process. The final results are

compared with Piegl's knot removal based method.

This chapter starts with a literature review of existing curve fitting methods that are designed to satisfy a prespecified tolerance. It is followed by a brief exposition of the proposed strategy of knot insertion and the results of applying the proposed method to a skeleton of curves described in the previous chapter.

6.1 Literature Survey

Curve approximation to achieve a prespecified tolerance must have a mechanism to adjust the number of basis supporting the approximation curves. For B-spline curves, the number of basis is equal to the number of knots less the order of the basis. Adjustment of number of basis can only be performed by knot insertion and knot removal. Forsey and Bartels developed the hierarchical fitting method based on knot insertion[28]. Tiller developed a knot-removal based curve approximation[82, 70]. Brief description of these methods are presented in the following sections.

6.1.1 Hierarchical Fitting

The Hierarchical Surface Fitting technique, proposed by Forsey and Bartels[28], uses knot insertion to satisfy a prespecified accuracy. This method consists of repetition of the following set of three sequential tasks:

1. identification and isolation of out-of-tolerance regions,

2. refinement of the basis and the control points supporting the out-of-tolerance regions by knot insertion, and

3. solving control points supporting the out-of-tolerance region from a linearly constrained least-square equation.

In Figure 6.1 the shaded areas are an example of out-of-tolerance regions that have been identified. Bounding boxes of constant u and v covering these regions are introduced to isolate the out-of-tolerance regions. The use of constant u and v is intended to simplify the definition of linear constraints and the subsequent refinement.

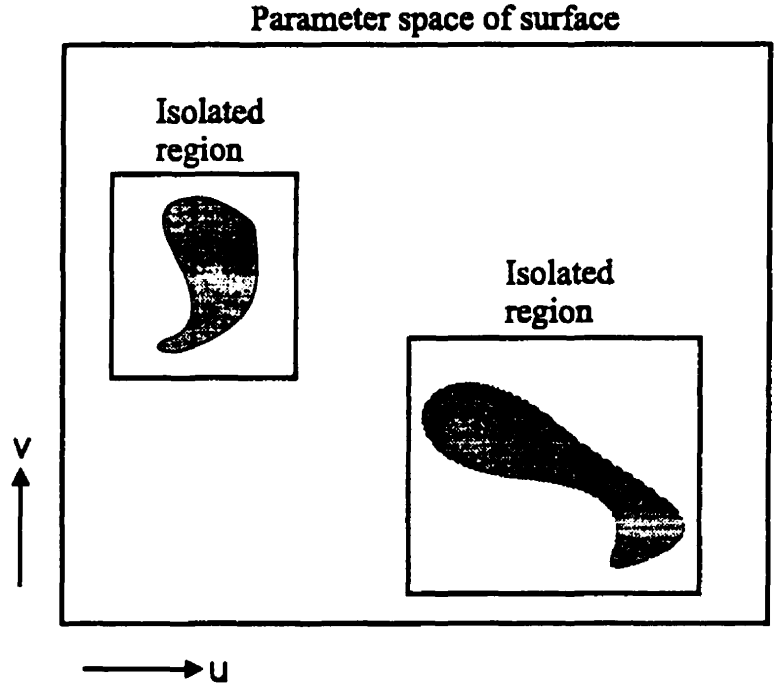


Figure 6.1: The Shaded Regions are Out-Of-Tolerance Region

The purpose of refinement is to increase the degree of freedom of the least-square problem in order to achieve the required tolerance. The refinement is performed by inserting knots. To minimize the computation, refinement is performed locally

in the out-of-tolerance region. The refinement task produces refined control points forming the control net supporting the out-of-tolerance region. The perimeter of this net will be fixed during the third task in order to force the perimeter of the out-of-tolerance region to interpolate perimeter curves defined by the constant parameter lines of the isolating box. This also localizes the influence of the solution of the least-square equation to within the out-of-tolerance region. Equation (6.1) illustrates the refined control points that must be fixed for surface of degree p and q in u and v respectively.

$$\mathbf{V}_{\text{refined}} = \begin{bmatrix} [\mathbf{V}_{\text{fixed}}]_{p\times q} & [\mathbf{V}_{\text{fixed}}]_{p\times n} & [\mathbf{V}_{\text{fixed}}]_{p\times q} \\ [\mathbf{V}_{\text{fixed}}]_{m\times q} & \boxed{[\mathbf{V}_{\text{free}}]_{m\times n}} & [\mathbf{V}_{\text{fixed}}]_{m\times q} \\ [\mathbf{V}_{\text{fixed}}]_{p\times q} & [\mathbf{V}_{\text{fixed}}]_{p\times n} & [\mathbf{V}_{\text{fixed}}]_{p\times q} \end{bmatrix} \tag{6.1}$$

$\mathbf{V}_{\text{fixed}}$ refers to fixed control points whereas $\mathbf{V}_{\text{free}}$ refers to the control points that will be computed from the least-square equation. Recalling that the expression of this equation is $[\mathbf{B}_u(u)][\mathbf{V}][\mathbf{B}_v^T(v)] = [\mathbf{P}]$, the expression of the linearly constrained least-square equation of the free control points will be

$$[\mathbf{B}_u(u)][\mathbf{V}_{\text{free}}][\mathbf{B}_v^T(v)] = [\mathbf{P}] - [\mathbf{B}_u(u)][\mathbf{V}_{\text{fixed}}][\mathbf{B}_v^T(v)] \tag{6.2}$$

Retaining the control points on the perimeter of the refined $\mathbf{V}$ allows trimming p left-most, and right-most, columns of $\mathbf{B}_u$; for $\mathbf{B}_v$, trimming applies to the q left-most, and right-most, columns. This trimming reduces the computation of solving

the least-square equation. Details of theoretical background and implementation can be found in Forsey's article[28].

The advantageous features of the Hierarchical Fitting are: (1) efficient computation due to localized computation of the refined control points, and (2) economical storage throughout the many levels of the fitting process. Absent from Forsey's article[28] is any discussion about the degeneration of the least squares equation (6.2) when it becomes ill-conditioned or rank deficient. This degeneration may occur in tight tolerance situation where the required number of basis approaches the number of data such that there is one or more B-spline basis not supported by the data.

The application of the Hierarchical fitting to curve approximation is straightforward; in this case, the out-of-tolerance region takes the form of a line segment instead of a box. Suppose that the span of out of tolerance segment is $t \in [t_i, t_j)$ and that the B-spline basis supported in this segment are N_i for $i = k : m$. The refinement can be performed by repetitive knot insertion at t_i and t_j such that the augmented B-spline basis N_i^* for $i = k^* : m^*$ are all zero for $t \notin [t_i, t_j]$. In general, the number of knots that must be inserted at t_i equals the order of the B-spline basis; the same applies to t_j. Thus, for B-spline curve of order k, each out-of-tolerance segment requires not more than $2k$ new knots. The result of this insertion is an increase of the number of basis which contradicts the objective of minimizing them.

Application of the Hierarchical fitting to a skeleton results in distribution of out-of-tolerance region such as the one shown in Figure 6.2. The simplest way to apply

the Hierarchical fitting is to consider each out-of-tolerance segment as an individual region and to perform local refinement in this region. However, compatibility will be destroyed because knot insertion is performed only on one curve. The violation of compatibility is not tolerable in our application. Therefore this simple scheme is not a viable way to eliminate the out-of-tolerance regions.

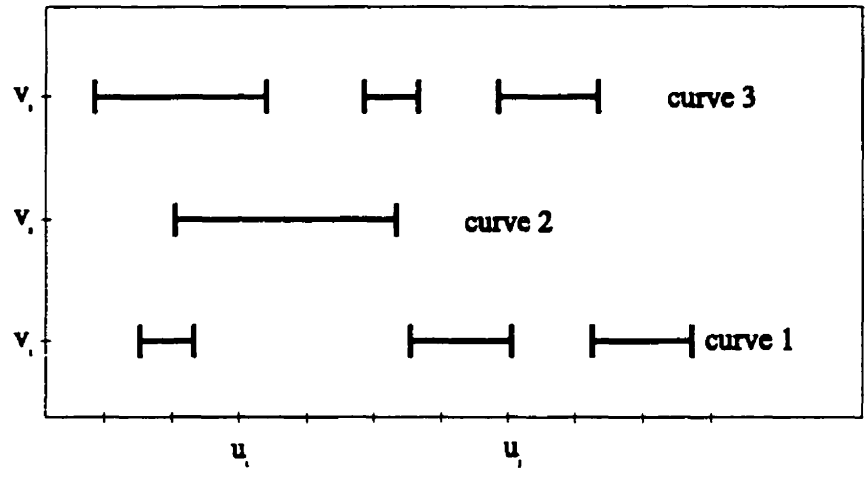


Figure 6.2: Line Segments of Out-Of-Tolerance Region

An alternative scheme to maintain the compatibility of the skeletal curves is to merge all the out-of-tolerance lines. Figure 6.3 illustrates the out-of-tolerance segment, generated by merging the out-of-tolerance regions shown in Figure 6.2. It is clear that merging has a tendency to destroy the locality of the out-of-tolerance regions. In this situation, the number of out-of-tolerance data increase sharply such that computationally-efficient feature of the Hierarchical fitting method is gone. Thus, the Hierarchical fitting is not readily applicable to this problem.

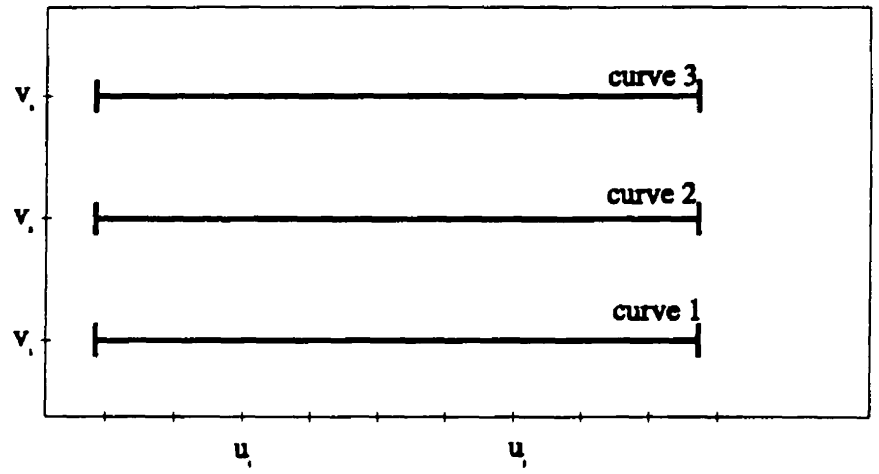


Figure 6.3: Merging of Out-Of-Tolerance Region, Resulting in The Out of Tolerance Region Shown Here for Each Curve

6.1.2 Method Based on Knot Removal

To provide background on this method, a brief discussion on the theory of knot removal is presented below. Consider the removal of knot $u_r = \{u_r \mid u_r \neq u_{r+1}\}$ of multiplicity $s = \{s \mid 1 \leq s \leq p\}$ from the B-spline of degree p. The removal of u_r is performed by computing new control points $\mathbf{c}^*$ using the following equations[70, 82]:

$$\mathbf{c}_i^* = \frac{1}{\alpha_i}\left(\mathbf{c}_i - (1-\alpha_i)\,\mathbf{c}_{i-1}^*\right) \qquad \text{for } i = (r-p) : (r-p+j) \tag{6.3}$$

$$\mathbf{c}_i^* = \frac{1}{1-\alpha_i}\left(\mathbf{c}_i - \alpha_i \mathbf{c}_{i+1}^*\right) \qquad \text{for } i = (r-s) : -1 : (r-p+j+1) \tag{6.4}$$

where :

$$j = \begin{cases} \frac{1}{2}(p-s-1) & \text{if } (p-s+1) \text{ is even} \\ \frac{1}{2}(p-s) & \text{if } (p-s+1) \text{ is odd} \end{cases}$$

$$\alpha_i = \frac{u_r - u_i}{u_{i+p+1} - u_i}$$

$$\mathbf{c}_{r-p-1}^* = \mathbf{c}_{r-p-1}$$

$$\mathbf{c}_{r-s}^* = \mathbf{c}_{r-s+1}$$

The knot u_r is said to be *mathematically removable* if the following equation is satisfied:

$$\mathbf{c}_{r-p+j}^* = \frac{1}{1-\alpha_{r-p+j}}\left(\mathbf{c}_{r-p+j} - \alpha_{r-p+j}\mathbf{c}_{r-p+j+1}^*\right) \tag{6.5}$$

Removal of knots that satisfy the above equation will produce the original curve. On the contrary, removal of knots which do not satisfy the above equation will produce a different curve whose maximum deviation from the original curve is not greater than ε, where ε is given by:

$$\varepsilon = \left\| \mathbf{c}^*_{r-p+j} - \frac{\left(\mathbf{c}_{r-p+j} - \alpha_{r-p+j}\mathbf{c}^*_{r-p+j+1}\right)}{1-\alpha_{r-p+j}} \right\|_2 \tag{6.6}$$

By setting ε equal to a given tolerance, knot removal can be performed to all possible knots which result in new curve whose deviation from the original curve is less than the tolerance. The complete exposition of the knot removal can be found in the works by Lyche, et. al.[58, 59, 60].

Based on Equation 6.6, Tiller[70, 82] developed a curve approximation method to satisfy a prespecified tolerance. The method follows the following steps:

1. **Interpolation of data with linear B-spline curves.** This is the step where curve compatibility must be enforced. Successful enforcement requires that the number of data be uniform and that the curves share a mutual parameter vector t. The latter has a potential to yield unsatisfactory parameterization when the data are not well distributed.

2. **Removal of all possible knots without violating the tolerance.** For a set of compatible curves, a knot can only be removed if it satisfies Equation 6.6 for every curve in the set of the compatible curves. Therefore, the probability that a knot of a set of compatible curves can be removed while satisfying Equation 6.6 for all curves in the set of compatible curves is lower than the probability to remove the same knot in a single curve problem. The reduction of probability is linearly proportional to the number of curves in the set of compatible curves.

3. **Do the following until prespecified tolerance is achieved:**

   (a) **Increase multiplicity of all domain knots by one in preparation of degree elevation.** This step sharply increases the number of basis (control points), leading to contradictions with our aim of minimum number of control points.

   (b) **Solve for the control points.** Use the least squares equation constructed from the new knots and the elevated degree to obtain the control points. This step is the most questionable link due to lack of theoretical foundation that warrants the preservation of the previously obtained satisfaction of prespecified tolerance.

   (c) **Removal of all possible knots without violating the tolerance.** In compatible curves for a knot to be removed it must be removeable in each of the skeletal curves. This reduces the likelihood of removing a knot significantly.

The final result is a curve of the prespecified degree whose distance from the data is less than the prespecifed tolerance. Tiller[70, 82] claims that the advantages of starting with a linear interpolation of a curve and working up to least square fitting of the prespecified degree are three-fold:

1. Cusp and discontinuities in curvature inherent in the data tend to be captured at the appropriate stage.

2. The evolving curve tends to "settle" into a natural parameterization.

3. Wiggles in the final curve tend to be minimized.

Tiller's approximation method[70, 82] requires that the number of $(p+1)$ points within every interval of domain knots is required in order to prevent the matrix $\mathbf{R}$ of the least square equation $\mathbf{Rc} = \mathbf{p}$ from becoming ill-conditioned or rank deficient. This requirement is satisfied by providing a user-supplied routine to generate the additional data points. In approximation problems, where the tolerance is tight, it is normal to expect that the number of control points of the approximation curve exceeds the number of the original data.

These methods by Forsey and Bartels[26, 28] and Tiller[70, 82] in general are robust and efficient. However, as already explained, they are not suitable for our cases where compatibility and minimum number of degree of freedom (control points) are needed simultaneously. Therefore, we resort to a more conservative and traditional scheme of adjustment of degree of freedom.

6.2 Tolerance Based Knot Insertion

This section presents a method of increasing the degrees of freedom of least square NURBS fitting in order to satisfy a prespecified tolerance. The inputs to the method are a set of data, prespecified tolerance, and a set of compatible NURBS curves that approximate the data but do not, as yet, satisfy the tolerance.

The tolerance based knot insertion method begins by identifying an out-of-tolerance region which is a knot interval containing maximum number of points whose distance from the fitted curve is larger than prespecified tolerance.

The out-of-tolerance region is refined by inserting a knot at a location in this region for all the skeletal curves. The proposed method uses mid-insertion, i.e. the location of inserted knot is at the mid of the interval of interest.

To achieve minimum number of additional control point, the proposed method does not raises the multiplicity of u_i or u_{i+1} (recall that the Hierarchical fitting does this to preserve the shape and continuity of the neighboring intervals and that raising multiplicity will introduce $2k$ new control points). Thus, the refinement is not local, i.e. the segment outside of $[u_i, u_{i+1}]$ is affected by the refinement and change of status from within-tolerance to out-of-tolerance may occur. Nevertheless, when the degree of freedom becomes sufficiently high, all intervals will eventually become within-tolerance intervals. The benefit of eliminating knot insertion at u_i and u_{i+1} is to ensure that increasing the number of control points is only performed when necessary, thereby avoiding unnecessary addition of the control points. To achieve the prespecified tolerance, knot insertion is performed repeatedly.

The strategy of knot insertion is laid out as follows:

1. Start with compatible NURBS curves which contain out-of-tolerance points.
2. Select a knot interval in which knot insertion will be performed. The selection is done using the following steps:
   (a) Rank the intervals based on the number of out-of-tolerance points in those intervals $[u_i, u_{i+1})$ for all i. The higher this number the higher the rank of the interval. If no intervals contains out-of-tolerance points, the curve has satisfied the tolerance. Terminate this algorithm of knot insertion.

(b) If two or more intervals have identical number of out-of-tolerance points, the ranks for these intervals is assigned based on the the least squares error for each of these intervals. Higher error results in higher rank. This step ranks the interval such that the worst interval occupies the highest rank.

(c) Starting from the highest rank, and walking down the rank, find the first interval in which the knot insertion produces matrix $\tilde{\mathbf{R}}$ whose condition number is below the permissible maximum. This interval is one in which knot insertion will be performed. This step is added to prevent ill-conditioning of $\tilde{\mathbf{R}}$. It is possible that all the intervals are ruled out. In this case, additional points (resampling) are introduced in the data, and this step is repeated.

3. Insert a knot at the middle of the selected interval. The insertion will yield a new matrix $\tilde{\mathbf{R}}$. Due to the method of selection described above, this interval is the worst interval which still results in well- conditioned $\tilde{\mathbf{R}}$. The new control points are then solved from the least square equation $\tilde{\mathbf{R}}\tilde{\mathbf{c}} = \mathbf{p}$. Repeat the overall process by looping back to step 2.

This proposed algorithm can achieve the goal of satisfying the tolerance with minimum number of control points (degree of freedom). The additional control points allows the curve to be more accurate in approximating the data. Defining a knot interval as the smallest unit of out-of-tolerance region maintains the compatibility. Controlling the condition number of matrix $\mathbf{R}$ ensures the accuracy of the new control points. Inserting the knot only when it is necessary improves the like-

lihood of obtaining the fewest number of control points. However, this algorithm is not without disadvantages: computing the condition number is expensive and forbidding local refinement slows down the convergence. The next section shows the results obtained from the implementation of this proposed algorithm.

6.3 Experiment

The proposed technique was tested on three skeletons, namely airplane wing, compressor blade, and turbine blade. Furthermore, each of these skeleton was fitted with integral and rational B-spline curves. This resulted in fitting of 6 skeletons which are summarized in Table 6.1. The results of the six cases are given in the following subsections.

Table 6.1: Sources of skeleton

Data	Tolerance [mm]	Type of B-Spline	
		Integral	Rational
Airfoils of wing	0.002	Section 5.2.1	Section 5.2.4
Sections of compressor	0.010	Section 5.3.1	Section 5.3.4
Sections of turbine	0.040	Section 5.4.1	Section 5.4.4

6.3.1 Wing Skeletal Curves

In this test case, the prespecified tolerance is set to 2 microns, i.e. 0.016% of chord (which is a common value used in airfoil design). The skeleton of compatible curves were obtained from sections 5.2.1 and 5.2.4. Repeated knot insertions were performed until the curves satisfy the prespecified accuracy. Figures 6.4 and 6.5 show the error distribution after repeated knot insertion; the former is the case where optimization is performed on the mutual knots whereas in the latter case optimization is performed on both knots and weights. Figures 6.6 and 6.7 show the control polygon for the final compatible integral and rational curves, respectively,

that satisfy the prespecified tolerance. These figures demonstrates that requiring the condition number of **R** to be below a certain prespecified maximum results in well behaved control points.

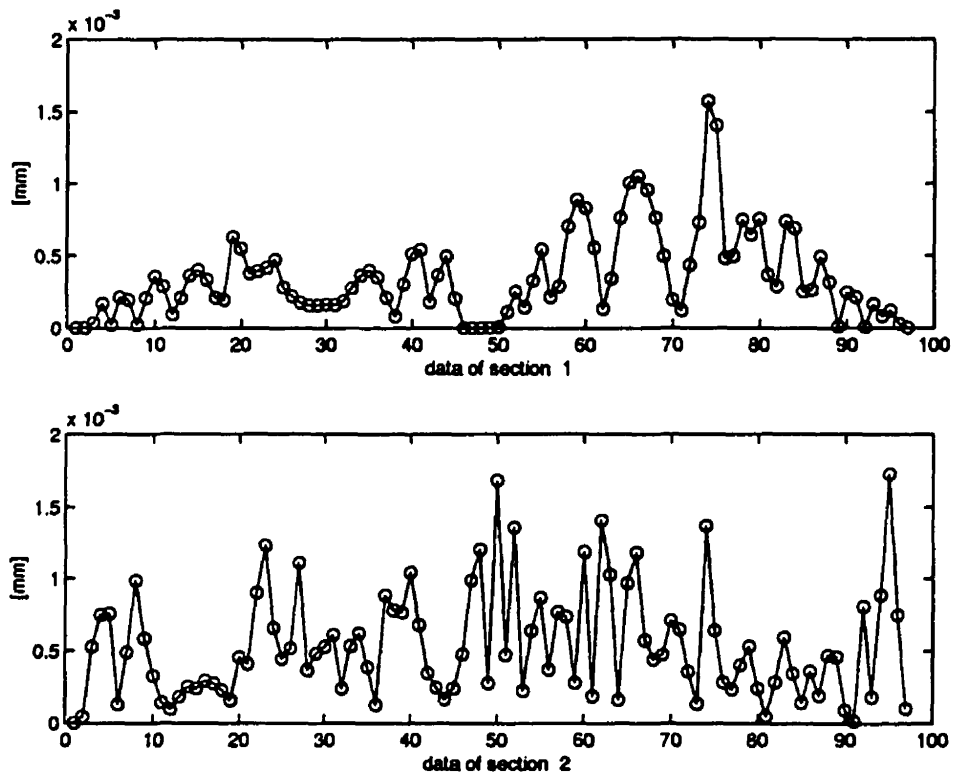


Figure 6.4: Error Distribution along Integral B-spline Wing Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 2 microns

6.3.2 Compressor Skeletal Curves

In this setup, the accuracy is set to 10 microns as requested by the designer. The already compatible curves are obtained from sections 5.3.1 and 5.3.4. Repeated knot insertion were performed until the curves satisfy the prespecified accuracy.

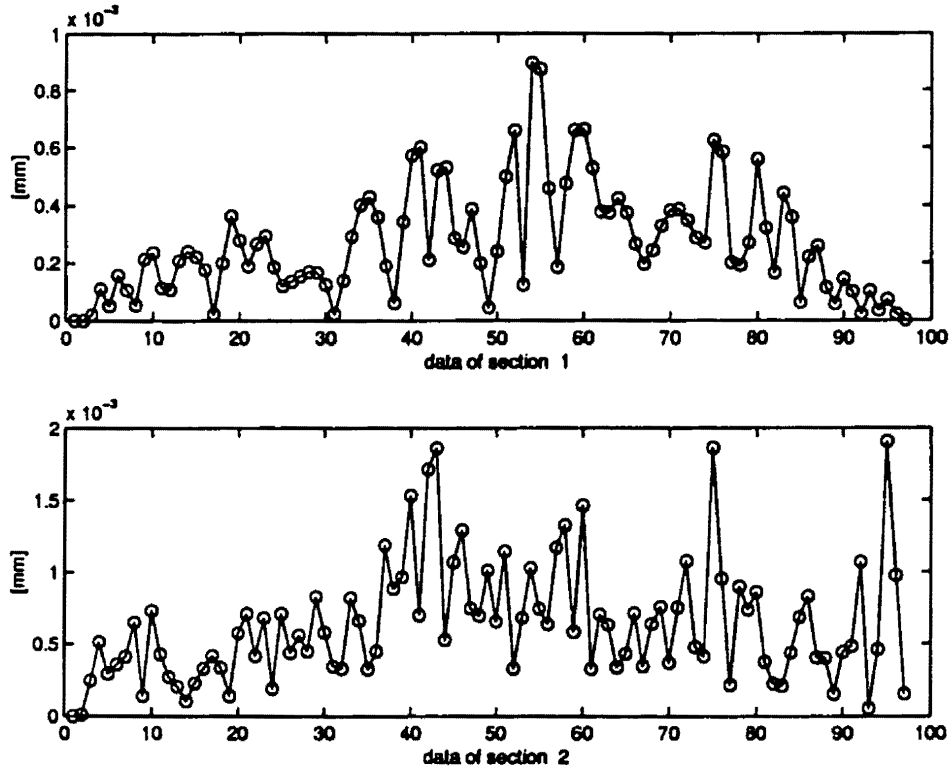


Figure 6.5: Error Distribution along Rational B-spline Wing Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 2 microns

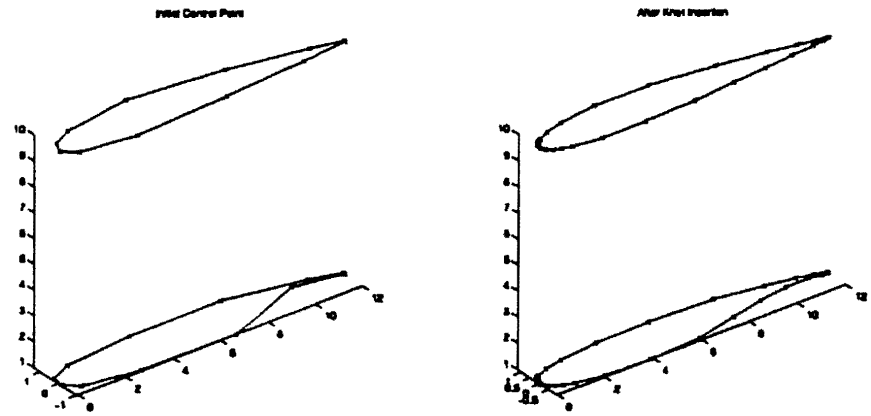


Figure 6.6: Control Points of Integral B-spline Wing Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

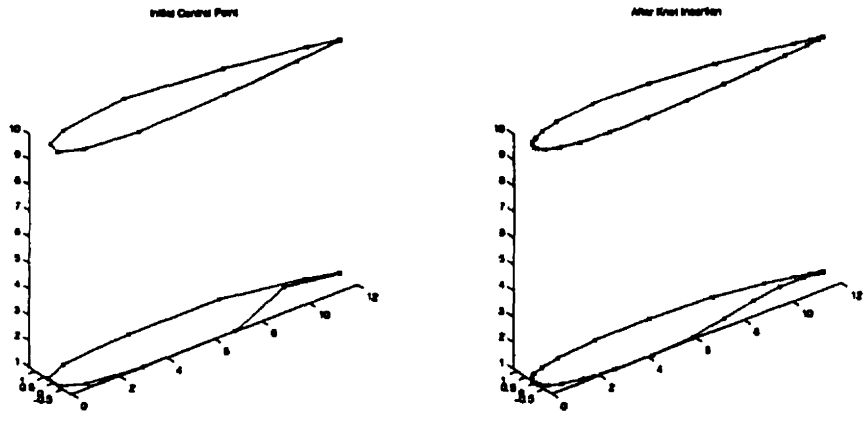


Figure 6.7: Control Points of Rational B-spline Wing Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

Figures 6.8 and 6.9 show the error distribution after repeated knot insertion; the former is the case where optimization is performed on the mutual knots whereas the latter case optimization is performed on both knots and weights. Figures 6.10 and 6.11 show the control polygon for the final compatible curves satisfying the prespecified tolerance; notice the finite components of the control points.

6.3.3 Turbine Skeletal Curves

In this setup, the accuracy is set to 40 microns. The already compatible curves are obtained from Sections 5.4.1 and 5.4.4. Repeated knot insertion were performed until the curves satisfy the prespecified accuracy. Figures 6.12 and 6.13 show the error distribution after repeated knot insertion; the former is the case where optimization is performed on the mutual knots whereas in the latter case optimization is performed on both knots and weights. Figures 6.14 and 6.15 show the control polygon for the final compatible curves satisfying the prespecified tolerance; notice

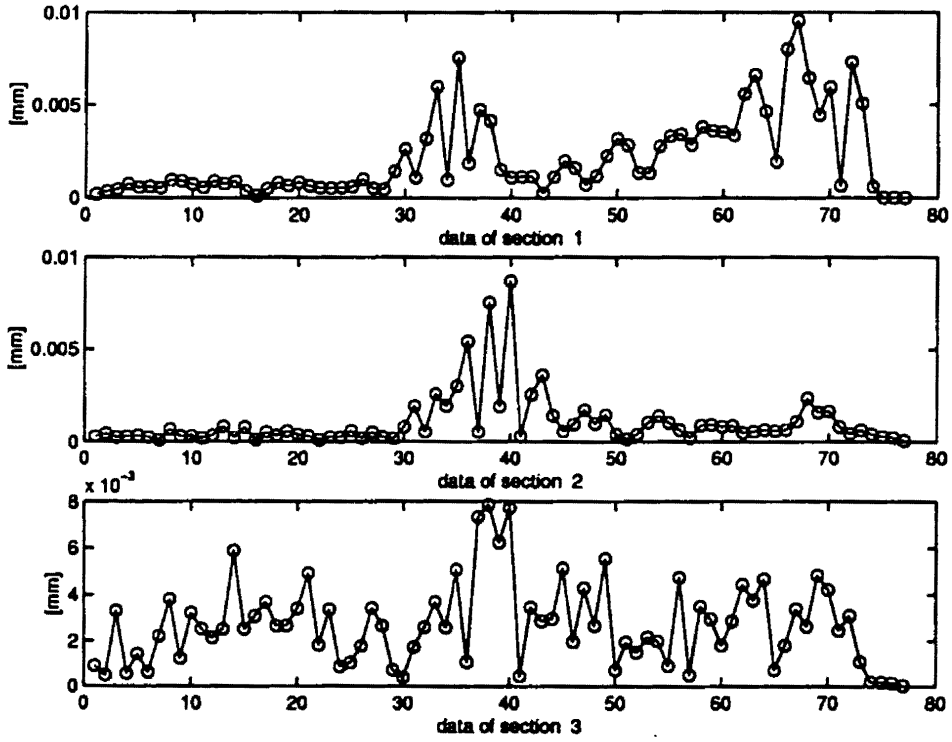


Figure 6.8: Error Distribution along Integral B-spline Compressor Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 10 microns

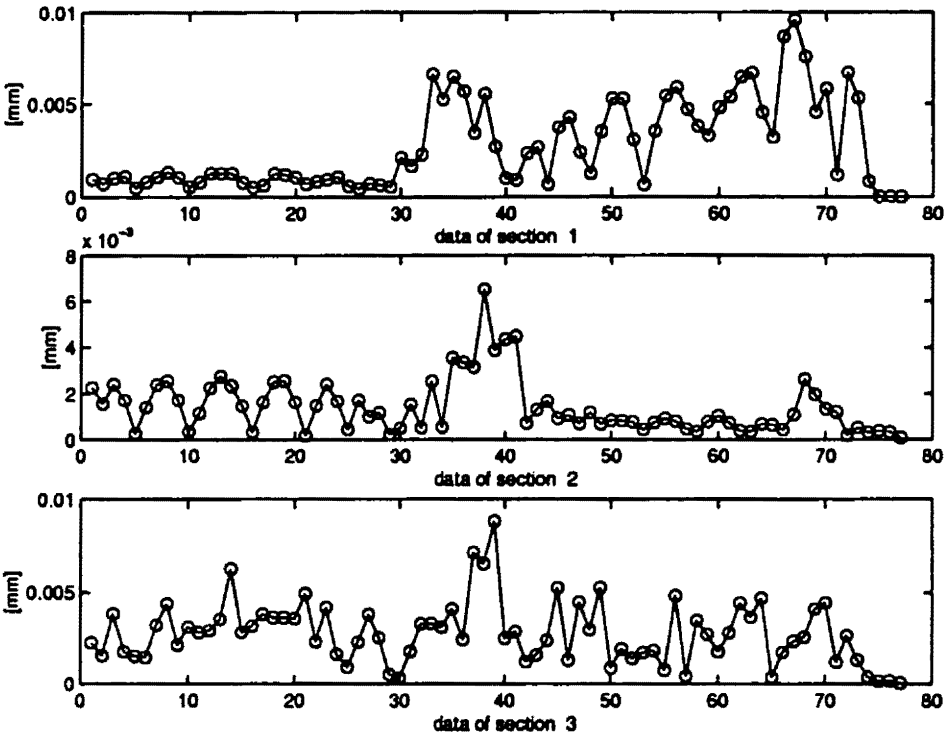


Figure 6.9: Error Distribution along Rational B-spline Compressor Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 10 microns

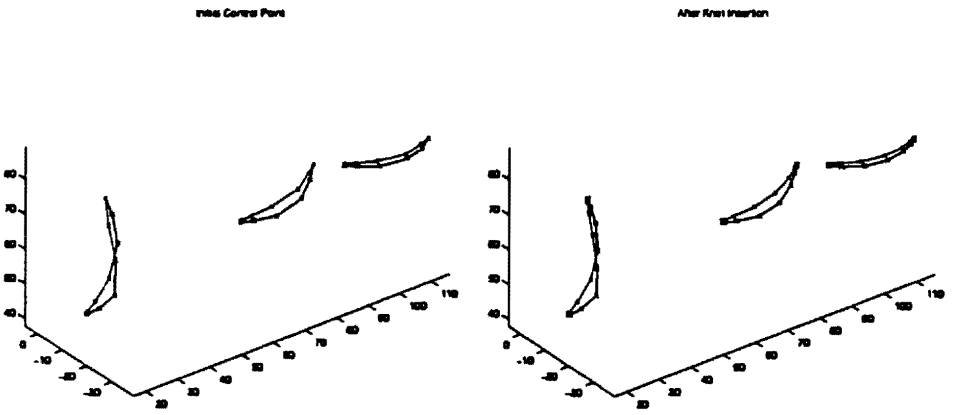


Figure 6.10: Control Points of Integral B-spline Compressor Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

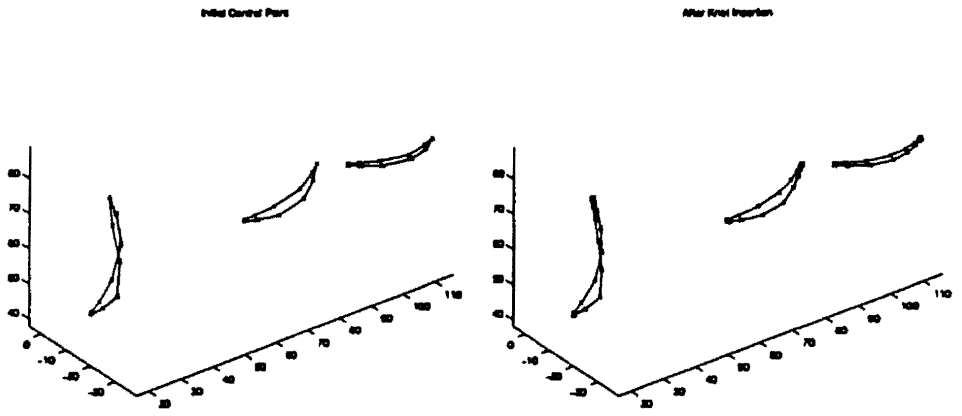


Figure 6.11: Control Points of Rational B-spline Compressor Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

the finite components of the control points.

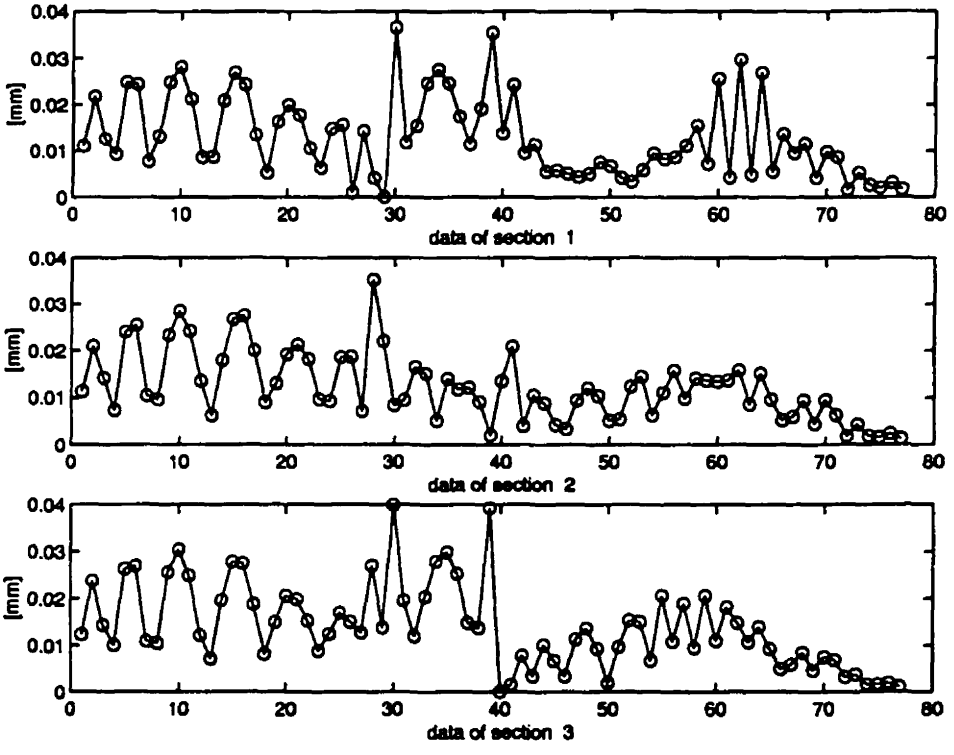


Figure 6.12: Error Distribution along Integral B-spline Turbine Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 40 microns

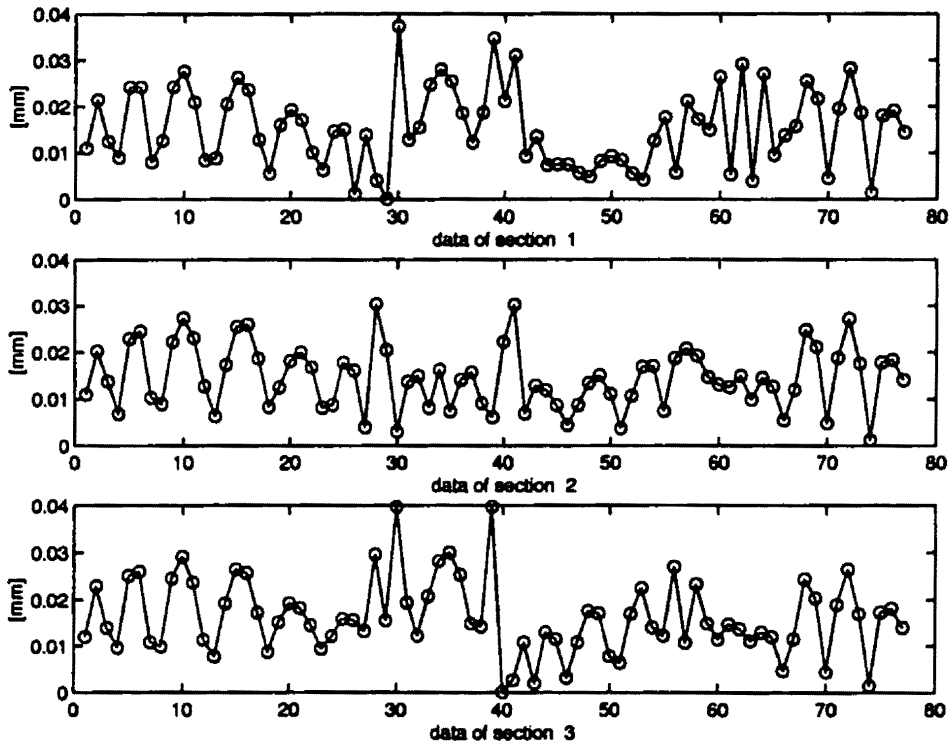


Figure 6.13: Error Distribution along Rational B-spline Turbine Skeleton after Applying Tolerance Based Knot Insertion Method with Prespecified Tolerance of 40 microns

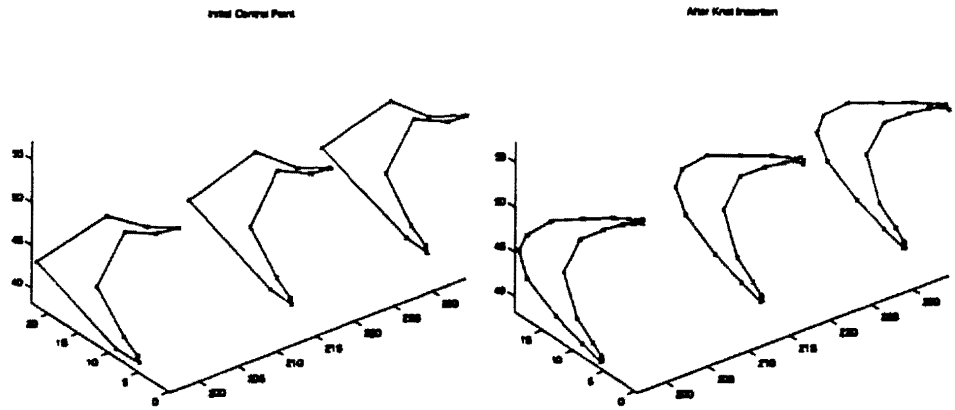


Figure 6.14: Control Points of Integral B-spline Turbine Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

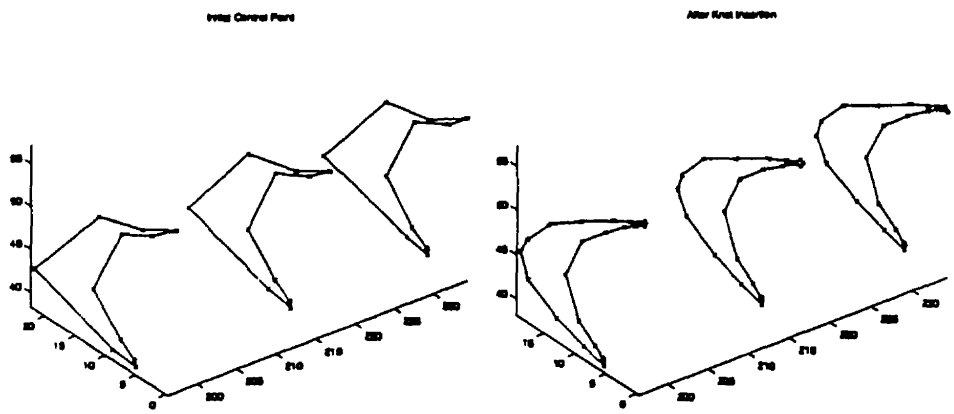


Figure 6.15: Control Points of Rational B-spline Turbine Skeletal Curves Before and After Application of Tolerance Based Knot Insertion Method

6.4 Discussion

Results of experiments presented in previous section clearly show that the tolerance based knot insertion method is viable to achieve a prespecified tolerance. Table 6.2 lists the number of control points of compatible curves obtained by the application of the proposed method, and compares the number of control points with those from knot-removal+knot-merging method due to Tiller[70, 82]. The table shows that the proposed method is better than the knot-removal/merging approach in term of minimizing the number of control points.

Table 6.2: Comparison between Proposed Method and Piegl's Method

	Methods	
	Knot Removal	Proposed
Airfoils of wing	78	30
Sections of compressor	124	24
Sections of turbine	97	25

Table 6.3 lists the resulting numbers of control points; for comparison, the number of control points from tolerance based knot insertion method in Table 6.2 are also listed. The numbers of control points under the knot-removal column are obtained by applying knot-removal method to every curve individually.

Table 6.3: Results of Individual Curve Fitting

	Number of Basis to Satisfy Tolerance	
Section curves	Knot Removal	Tolerance Based Method from Table 6.2
NACA 2415	31	30
WTEA	51	30
Compressor (hub)	41	24
Compressor (mid)	36	24
Compressor (tip)	55	24
Turbine (hub)	36	25
Turbine (mid)	35	25
Turbine (tip)	34	25

A larger number of control points is required in the the knot-removal method because it starts with linear interpolation and repetitively raises the degree until the desired degree is obtained. For B-spline curves of order k and with q control points, raising the degree by one produces $q - k$ additional control points. Therefore, if degree is raised m times, the total number of additional control points is $m\,(q - k)$; in this experiment, cubic curve is desired so that the total number of additional

control points is $2(q-k)$. The degree raising step is responsible for the higher number of control points obtained from the knot-removal method. The tolerance based knot insertion method avoids unnecessary addition of control points.

The curves obtained from the knot-removal method are not compatible yet whereas those curves from the tolerance based knot insertion method are already compatible. Compatibility of the latter is achieved by merging the knots of the curves. The net effect of merging is summing up of the number of control points. Comparison of knot-removal columns in Table 6.2 and 6.3 shows that those numbers are related as follows:

$$\begin{array}{lcrcl} \text{Wing} & : & 78 & \approx & 31+51 \\ \text{Compressor} & : & 124 & \approx & 41+36+55 \\ \text{Turbine} & : & 97 & \approx & 36+35+34 \end{array}$$

Thus the knot-removal/merging approach suffers from the degree raising and knot merging steps such that the number of control points required to satisfy tolerance and compatibility is much higher than that obtained from the tolerance based knot insertion approach.

However, the proposed approach is not without drawback. It requires much higher computation than the knot-removal/merging approach. The most expensive computation part in the proposed method is the decomposition of matrix $\mathbf{R}$; this decomposition is performed in the computation of condition number of $\mathbf{R}$ and the solution of the least square equation $\mathbf{Rc} = \mathbf{p}$. However, for the proposed application of blading and airfoil design this is not a major concern as the computational time is a small fraction of time spent in manual modification. In the proposed

approach, the decomposition is performed very frequently, in particular during the optimization of parameters. For m number of iteration and n number of curves, the decomposition is performed mn times. On the other hand, the knot-removal approach performs the decomposition in the degree raising step only; for n curves of order k, the decomposition is performed $n(k-1)$ times. If $m \gg k$, which is very common in practice, the proposed approach requires significantly higher number of decomposition than that of the knot-removal approach.

The results of experiments in this chapter have highlighted the feature of the proposed approach in minimizing the number of required control points to satisfy tolerance and compatibility. This feature can be attributed to the lack of need to merge the knots because curve compatibility is being maintained throughout the approximation process.

The following sections will present attempt to smooth the curves and visualization of the skinned surface.

6.5 Attempt of Smoothing

This section presents an attempt to smooth the curves obtained in the previous sections. Brief description of curve smoothing can be found in AppendixB. Smoothing is performed by moving control points to eliminate inflection points. Unfortunately, the amount of displacement required is larger than the tolerance. A compromise is made by displacing the curve by no more than the tolerance. In all cases the tolerance is too tight to allow improvement in the smoothness of the curves. For this reason only the smoothing of the NACA airfoil is presented.

The displacement needed to remove inflection points in the NACA 2415 airfoil reached 50ϵ where ϵ is the prespecified tolerance set during the previous least square approximation. To illustrate our attempts, a distribution of curvature is plotted in Figure 6.16.

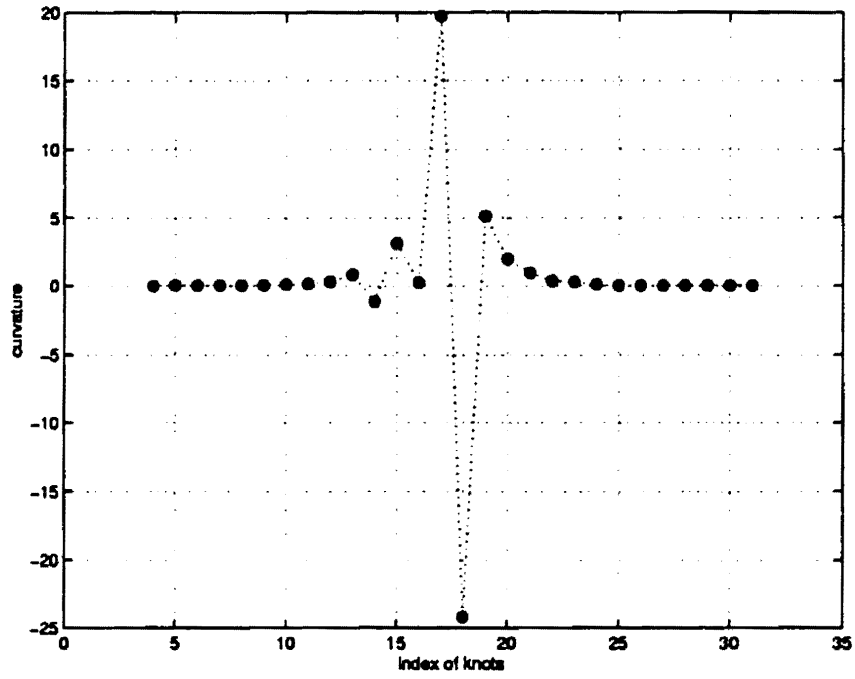


Figure 6.16: Curvature Distribution at Knots of NACA 2415. Changes of Signs at u_{14} and u_{18} causing Inflection Points

It is very clear that knots u_{14} and u_{18} are offending since the curvature at these knots change sign. The smoothing begins with removal of inflection points at u_{18}. In the first attempt, the prespecified tolerance of 2 microns is ignored, i.e. the curve is allowed to displace infinitely. Figure 6.17 shows improvement of curvature around the vicinity of u_{18} and the curves before and after removal of inflection points are shown as dotted and solid lines respectively. The figure clearly shows

that the inflection at u_{18} is completely removed. The displacement of the curve is about 98 microns, approximately 50 times the tolerance.

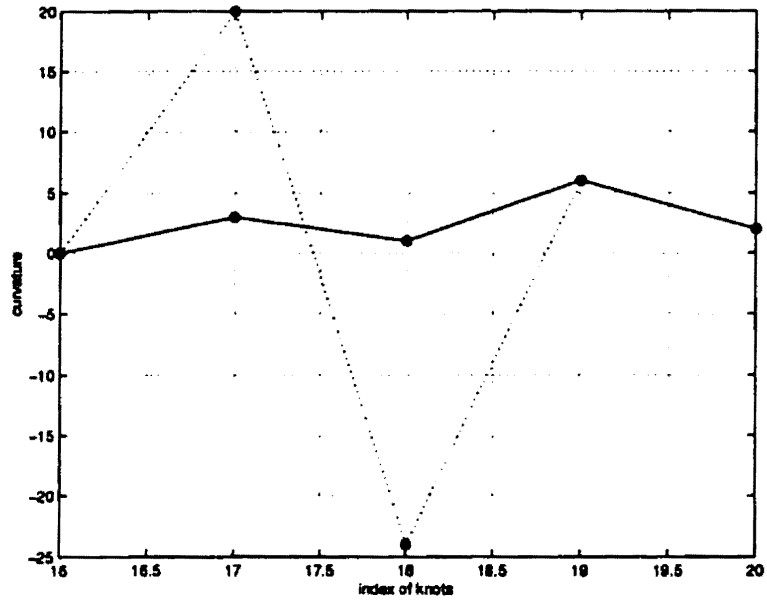
In the second attempt, the displacement is limited not to exceed 2 microns tolerance. Figure 6.18 shows improvement of curvature around the vicinity of u_{18} and the curves before and after removal of inflection points are shown as dotted and solid lines respectively. The inflection at u_{18} still exists and jump of curvature barely improves. The change in the curve is not visually observable. Unfortunately, this phenomena also occurred during attempts to smooth the rest of the curves obtained from Chapter 6.

6.6 Skinning

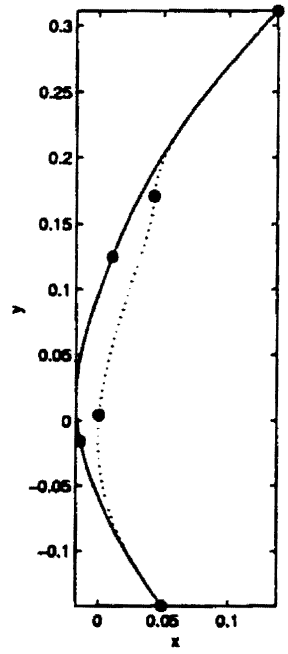
This section presents the visualization of surfaces obtained by skinning the curves from the previous section. No preparation, e.g. knot merging, is required prior to skinning since the curves are already compatible. The skinning is performed on two types of curves: the integral B-spline curves and the NURBS curves. The following two subsections shows the results of skinning for each case.

6.6.1 Integral B-spline Case

Figure 6.19, 6.20, and 6.21, show the skinned surfaces for wing, compressor blade, and turbine blade, respectively. The figures also show the control net of the skinned surface. The purpose of presenting the control net is to show that parameterization of data and averaging the knots yields well conditioned matrix $\mathbf{R}$

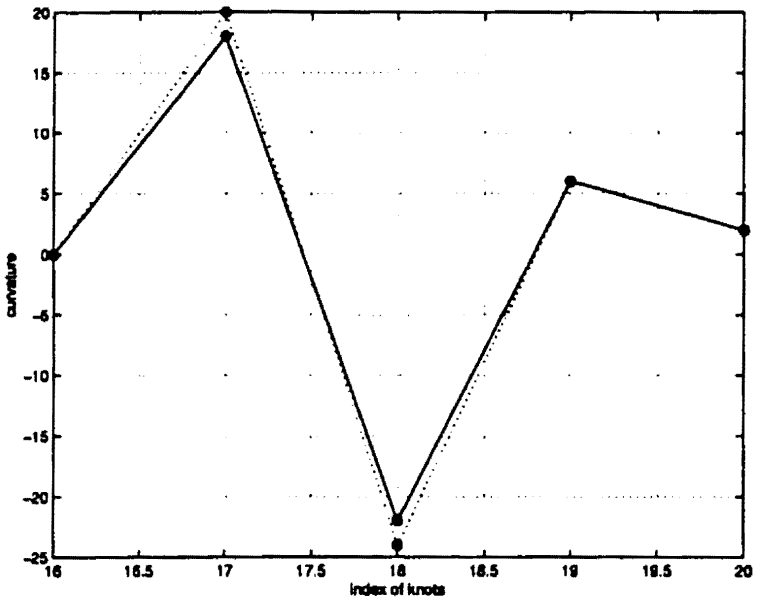


(a)

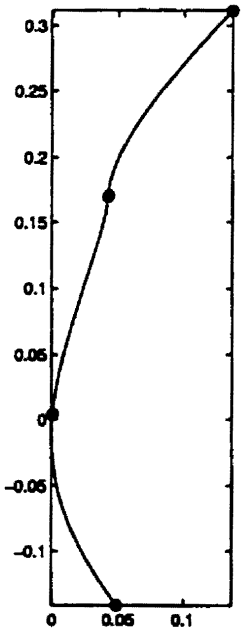


(b)

Figure 6.17: Smoothed NACA 2415 Curve: (a) Curvature at u_{18} Change Sign to Positive and (b) Inflection Point is Eliminated. The Smoothed Curve Displaced by No More Than 98 microns. Dotted and solid lines represent before and after smoothing.

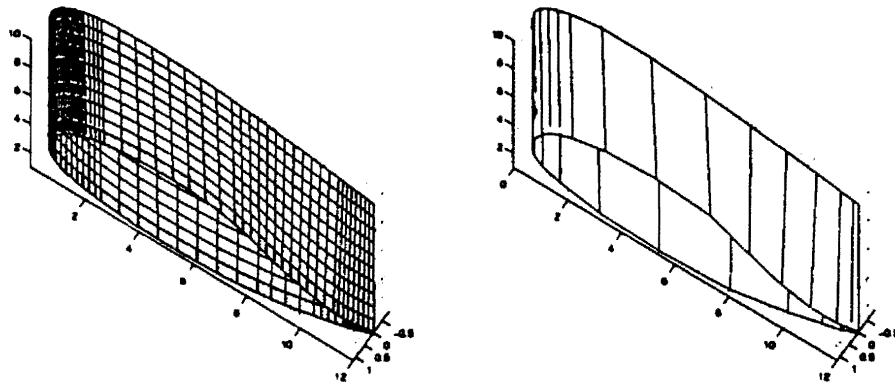


a



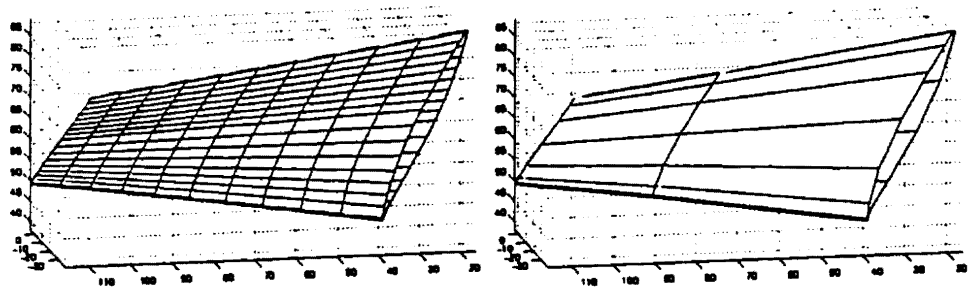
b

Figure 6.18: Partially Smoothed NACA 2415 Curve: (a) Curvature at u_{18} Remains Negative and (b) Inflection Point still Exist. The Smoothed Curve Displaced by No More Than 2 microns. Dotted and solid lines represent before and after smoothing.



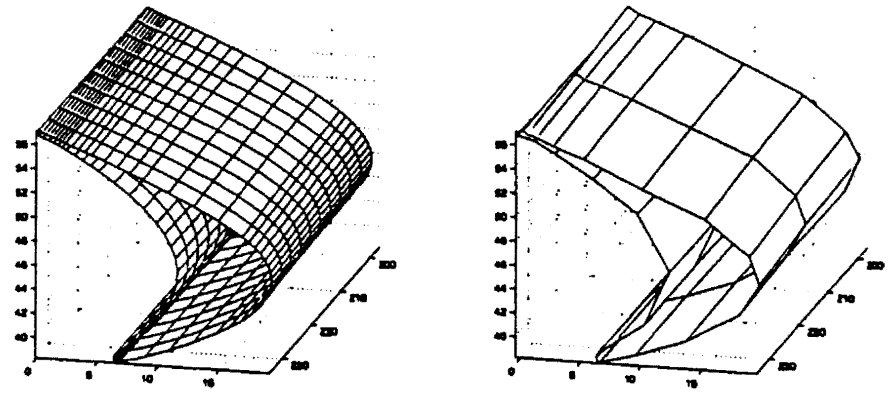
(a) (b)

Figure 6.19: Integral B-spline Skinned Surface of Wing: (a) Surface and (b) Control Net



(a) (b)

Figure 6.20: Integral B-spline Skinned Surface of Compressor blade: (a) Surface and (b) Control Net

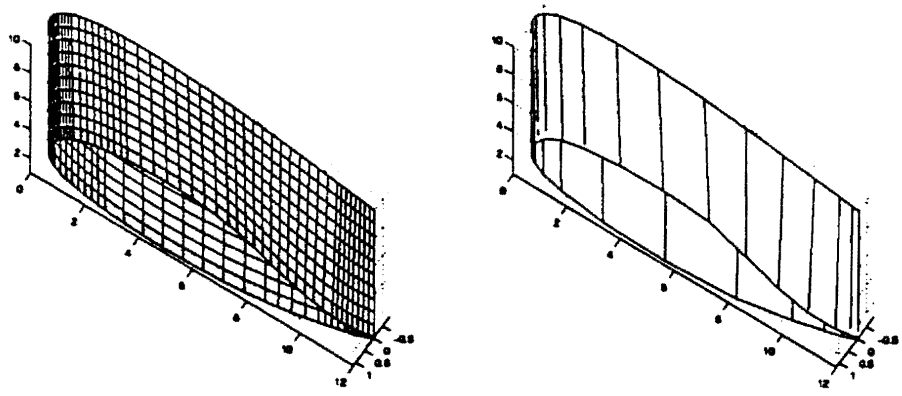


(a) (b)

Figure 6.21: Integral B-spline Skinned Surface of Turbine blade: (a) Surface and (b) Control Net

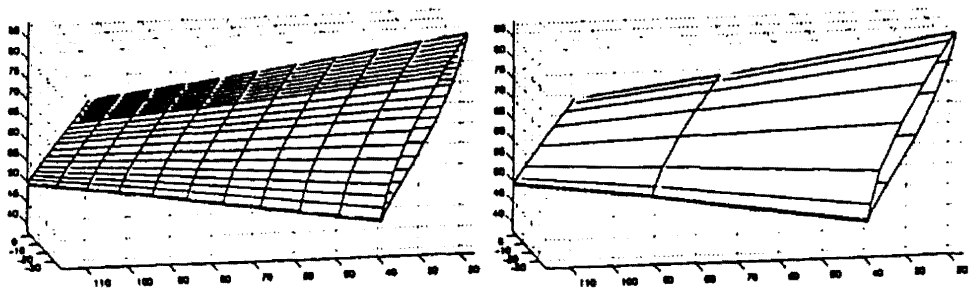
6.6.2 Rational B-spline Case

Figure 6.22, 6.23, and 6.24, show the skinned surfaces for wing, compressor blade, and turbine blade, respectively. The figures also show the control net of the skinned surface. The purpose of presenting the control net is to show that parameterization of data and averaging the knots yields well conditioned matrix **R**



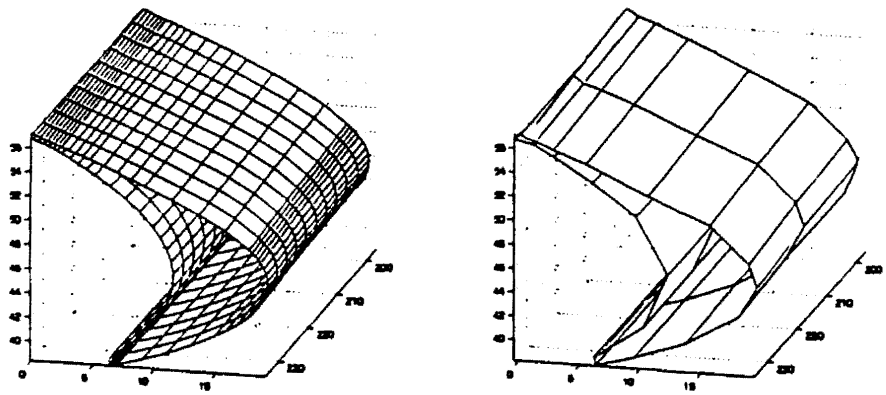
(a) (b)

Figure 6.22: Rational B-spline Skinned Surface of Wing: (a) Surface and (b) Control Net



(a) (b)

Figure 6.23: Rational B-spline Skinned Surface of Compressor blade: (a) Surface and (b) Control Net



(a) (b)

Figure 6.24: Rational B-spline Skinned Surface of Turbine blade: (a) Surface and (b) Control Net

Chapter 7

Conclusions and Future Work

This thesis has presented a new method to construct a skeleton made of NURBS curves that are ready for the skinning with no additional requirement. Figure 7.1 shows the existing and the proposed approaches for comparison purposes.

The highlight of the figure is the lack of knot merging prior to skinning in the proposed approach. This is achieved by enforcing compatibility during the simultaneous least squares fitting of the curves. The results have shown that the proposed approach requires significantly fewer number of control points than the existing approaches and produces compatible curves.

The compatibility enforcement requires that the curves share a mutual knot vector and a mutual degree. This results in a poor distribution of parameterization for the data and a poor overdetermined matrix **R**, which in turn results in relatively big least squares error. Optimization of parameters, i.e. knots, weights, and parameterization for the data, is performed to improve the matrix in order to reduce the resulting least squares error. This optimization is characterized by

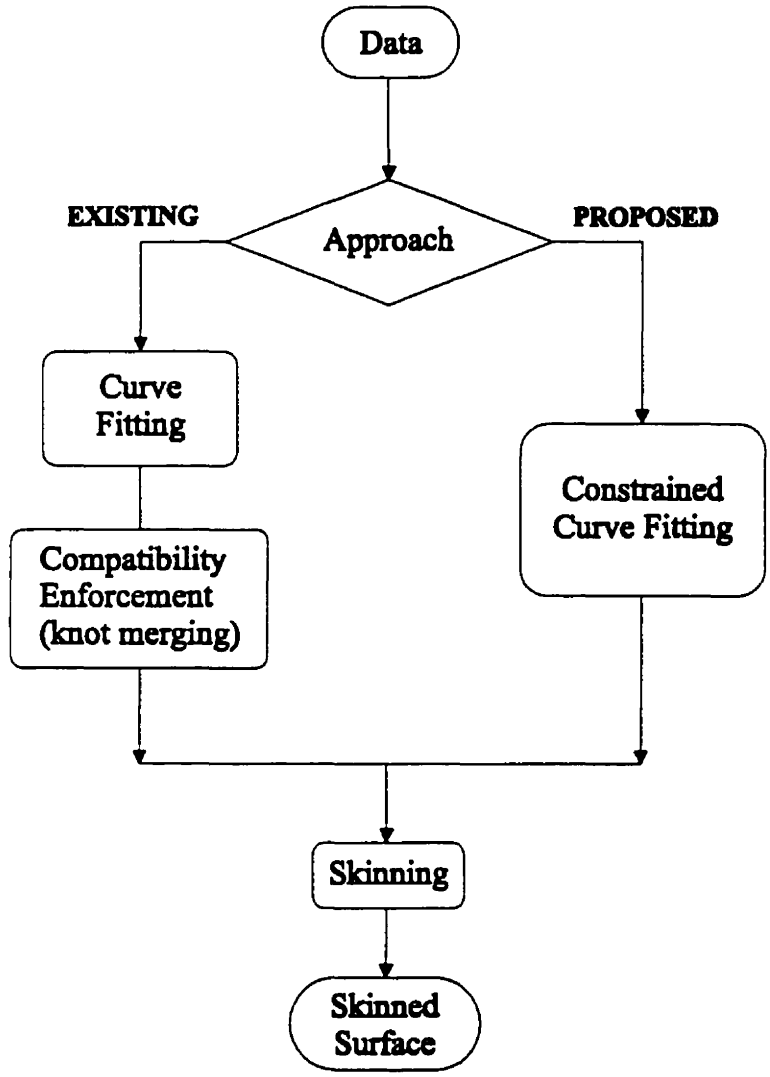


Figure 7.1: Comparison of Existing and Proposed Approaches

two critically unfavorable phenomena: lethargic behavior of gradient with respect to knots and poor control points due to ill- conditioning of matrix **R**. The former is overcome by deletion of linear constraints from the active set and by the use of BFGS descent direction to obtain a descent direction that has a nonzero component in the range space of the active set. The latter is overcome by incorporating the condition number of **R** into a nonlinear constraint.

Careful optimization of parameters has resulted in reduction of least squares error while maintaining the quality of control points obtained from $\mathbf{Rc} = \mathbf{p}$ and enforcing the compatibility of the curves.Once the optimization is completed, knot insertion is performed to the curves to meet the prespecified tolerance. Again, compatibility constraint is enforced during this insertion. The final results are compatible curves that satisfy the tolerance. These curves are the ultimate results that this research was seeking. These curves can then be skinned immediately without the need to merge the knots.

7.1 Achievements

The achievements that have been presented in this thesis are:

1. **A new method to construct a NURBS skeleton using a constrained least-square approximation.** The new method simultaneously fits NURBS curves to different sets of data points while sharing the same knot vector and degree. This is done using constrained least squares approximation. The constraint forces the curves to share a mutual knot and a mutual degree, and results in the following least squares equations:

$$\begin{aligned} \varepsilon_i(\mathbf{u}, \mathbf{w}_i, \mathbf{t}_i) &= \left(\mathbf{I} - \mathbf{R}_i\mathbf{R}_i^{+}\mathbf{p}_i\right)^T \left(\mathbf{I} - \mathbf{R}_i\mathbf{R}_i^{+}\mathbf{p}_i\right) \\ \text{where} \quad & \mathbf{R}_i = \mathbf{R}_i(\mathbf{u}, \mathbf{w}_i, \mathbf{t}_i) \text{ and } \mathbf{R}_i^T = \mathbf{R}_i^T(\mathbf{R}_i) \end{aligned} \tag{7.1}$$

This also forced the objective functions $\varepsilon_i(\mathbf{u}, \cdots)$ and the overdetermined matrices $\mathbf{R}_i(\mathbf{u}, \cdots)$ to be functions of a mutual knot $\mathbf{u}$ and a mutual degree resulting in compatible skeletal curves. The least squares error ε_i's were then combined into a single objective function expressed as $\varepsilon = \sum_i \varepsilon_i$. The summation operator was selected to provide simple expression for the gradient of this objective function and to force the continuity of this objective function to be equal to the lowest continuity of those of the ε_i's. Of course, this objective function requires that the measurement error (variances) of ε_i are more or less uniform with respect to i. It is safe to assume that the requirement is satisfied because the data are obtained from a single measuring equipment. The compatibility produced by this constraint eliminates the need to merge the knots prior to skinning such that the number of control points of the skinned surface are kept reasonably lower than that obtained from the existing method.

2. **Dealing with lethargic properties of knot adjustment.** Chapter 2 has presented the lethargic property of least square NURBS approximation with adjustable knots. This property arises from the exact similarity of expressions of the derivatives of NURBS basis with respect to coincident knots. This property will cause the first derivatives of R_i with respect to coincident knots

to be identical. When this is combined with the active linear constraints on the coincident knots, i.e. $u_j = u_k$, the first derivative of ϵ_i is located in the null space of the active linear constraints. A new method was presented to deal with the lethargy problem. By keeping the approximate Hessian (besides the projected one) during the optimization and using the BFGS descent direction, components into the feasible region can be obtained. By moving along this direction the constraint can be removed from the active set and allow the knots to separate in the subsequent iteration. Because of this method, coincident knots need not prevent knot optimization. Results of experiments have shown that knot optimization contributed to the largest decrease in least squares error.

3. **Insignificant reduction of least squares error obtained from the adjustment of parameterization for the data.** Numerical comparisons of the reduction of least squares error with respect to variation of its parameters, i.e. knots u, weights w, and parameterization for data t have showed that parameterization for the data has relatively negligible contribution in the reduction of least squares error. This phenomenon can be used as a justification for eliminating data parameterization from the list of adjustable parameters of the least squares problem. Since typical least squares problems are overdetermined with number of rows being much greater than the number of columns, the parameterization for the data constitutes the largest portion of the parameters. Therefore, its elimination greatly reduces the number of parameters of the least squares problem, and hence reduces the computations

required within the least squares problem.

4. **Less reduction of least squares error obtained from the adjustment of weights compared with the reduction resulted from knots adjustment.** This thesis has presented the phenomenon that the weights contributes significantly less to the reduction of least squares error than the knots. This phenomenon can be used as a justification to eliminate the weights from the parameters of the problem. Benefits of eliminating the weights are twofold: decreasing the number of parameters and simplifying the NURBS curves to become integral B-spline curves. The latter benefit is especially advantageous: singularity no longer exist, convex hull property is guaranteed, and portability of the curves becomes greater; the last one is due to the fact that not all NURBS geometric package are designed to receive negative weights.

5. **Introduction of nonlinear constraint on the condition number of $\mathbf{R}_i$.** Adjustment of parameters is constrained to keep the condition number below a prespecified permissible value, which is set to 500 in this work. Limiting the permissible value on the condition number maintains the quality of the control points $\mathbf{c}$, i.e. there is a unique solution of $\mathbf{Rc} = \mathbf{p}$ and none of the elements has a wildly large value.

6. **New method to satisfy the nonlinear constraint, i.e. condition number of matrix R.** This thesis uses a new mechanism to satisfy this nonlinear constraints, based on its unique characteristic instead of the standard methods. The matrix $\mathbf{R}$ degenerates to become ill-conditioned if one or more of

the following occur: two or more knots come close together; two or more parameterization for the data come close together; and one or more weights approaches zero. These three conditions will activate linear constraints. Therefore, activation of linear constraints can be used as a warning signal that violation of nonlinear constraint may occur. Since activation of linear constraints are verified in the determination of permissible step length before line search, the definition of the permissible step length is modified to satisfy the nonlinear constraint along with the linear ones. This new method has two benefits: elimination of the need of an explicit expression for the nonlinear constraints, which is highly complicated if not impossible at all, and elimination of the standard mechanism to satisfy the nonlinear constraint. These benefits reduces the complexity of the implementation.

7.2 Future Work

Hourmouziadis[46] points out that, **when the degrees of freedom for the geometry variation in a turbine are considered as adjustable parameters**, no feasible true 3D flow optimization can be carried out because it may take one or more years to define the turbine. The hidden information is the assumption that the geometry of the blade is represented as a set of points instead of some mathematical representation and that the lack of a geometric modeler within the fluid flow analysis package.

The technique proposed in this thesis, namely the use of NURBS representation instead of point representation coupled with the minimum number of parameters,

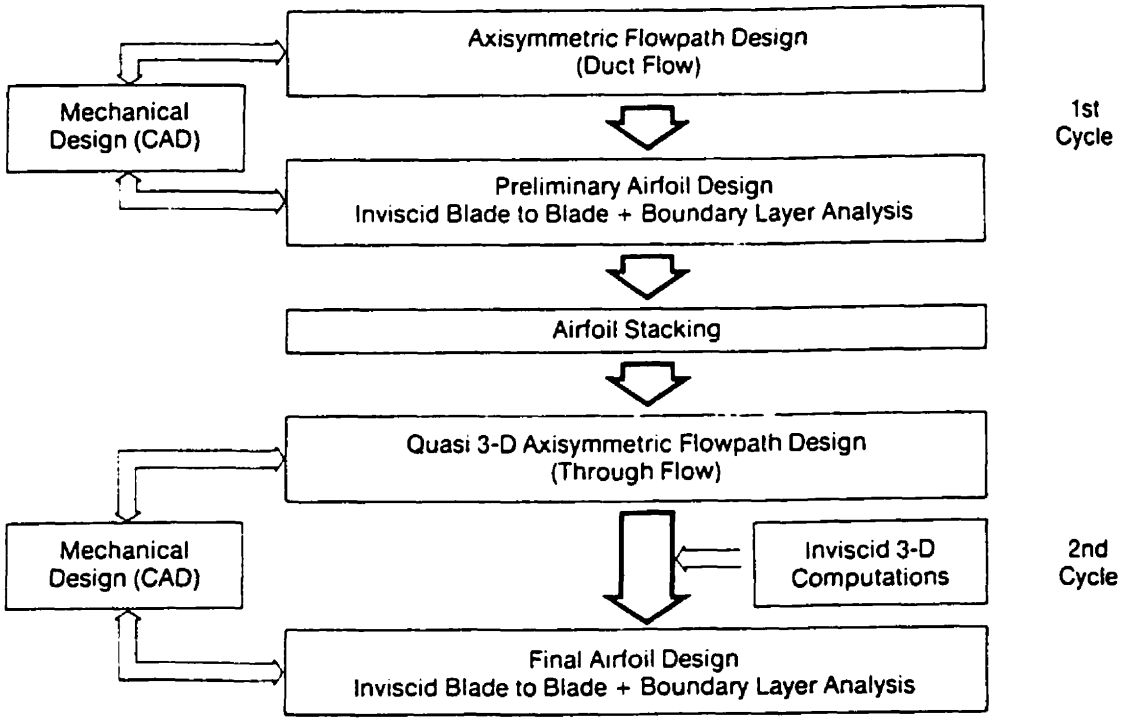


Figure 7.2: Turbine Design Procedure proposed by Hourmouziadis[46]

can overcome the above obstacles by reducing the degree of freedom by a factor of 5 at least. This evaluation is based on the skeleton discussed in this thesis. By allowing true 3D flow optimization to be conducted on a geometry this work has opened a vast field of research. The research effort could be focused in: integrating NURBS representation with flow analysis; and developing a heuristic/artificial intelligence/optimization based system for closing the automation loop as discussed in Chapter 1.

Bibliography

[1] I.H. Abbott and A.E. Von Doenhoff. *Theory of Wing Sections.* Dover Publications, Inc., 1959.

[2] L. Alt. Parameterization for data approximation. In P.J. Laurent, A. Le Mehaute, and L.L. Schumaker, editors, *Curves and Surfaces*, pages 1–4. 1991.

[3] C. Barghiel, R.H. Bartels, and D. Forsey. Pasting spline surfaces. In M. Dahlen, T. Lyche, and L.L. Schumaker, editors, *Mathematical Methods ic CAGD III*, pages 1–10. 1995.

[4] R.H. Bartels. Object oriented spline software. In P.J. Laurent, A. Le Mehaute, and L.L. Schumaker, editors, *Curves and Surfaces II.* AK Peters, Wellesley, Massachusetts, 1991.

[5] R.H. Bartels. Constraint based curve manipulation. Technical report, University of Waterloo, cari tahunnya.

[6] R.H. Bartels and J.C. Beatty. A technique for the direct manipulation of spline curves. In *Graphics Interface 89*, pages 33–39, 1989.

[7] R.H. Bartels and I. Hardtke. Speed adjustment for key-frame interpolation. In *Graphics Interface 89*, pages 14–19, 1989.

[8] R.H. Bartels and D.R. Warn. Experiment with curvature-continuous patch-boundary fitting. *IEEE Computer Graphics & Applications*, cari volume(dan number):64–73, September 1994.

[9] G. Birkhoff. *Aesthtic Measure*. University Press, 1933.

[10] W. Boehm. Bezier presentation of airfoils. *Computer Aided Geometric Design*, 4:17–22, 1987.

[11] J.J. Chou and L.A. Piegl. Data reduction using cubic rational B-splines. *IEEE Computer Graphics & Applications*, 12(3):60–68, May 1992.

[12] E. Cohen and C.L. O'Dell. A data dependent parametrization for spline approximation. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design*, pages 155–166. Academic Press, San Diego, California, 1989.

[13] C. de Boor. *A practical guide to splines*, volume 27. Applied Mathematical Sciences, 1978.

[14] W.L.F. Degen. Best approximation of parametric curves by splines. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in Computer-Aided Geometric Design II*, pages 171–183. Academic Press, San Diego, California, 1992.

[15] J.E. Dennis. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* SIAM, 1996.

[16] R.R. Dickinson, R.H. Bartels, and A.H. Vermeulen. The interactive editing and contouring of empirical fields. *IEEE Computer Graphics & Applications*, cari volume(and cari number):34–43, May 1989.

[17] P. Dierckx. *Curve and Surface Fitting with Splines.* Oxford University Press, 1993.

[18] J. Dill. An application of color graphics to the display of surface curvature. *Computer Graphics*, 15:153–161, 1981.

[19] M. Eck and J. Hadenfeld. Knot removal for B-spline curves. *Computer Aided Geometric Design*, 12:259–282, 1995.

[20] R. Eppler. *Airfoil Design and Data.* Springer-Verlag, 1990.

[21] M.P. Epstein. On the influence of parameterization in parametric interpolation. *SIAM Journal of Numerical Analysis*, 13:261–268, 1976.

[22] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design.* Academic Press, 1993.

[23] G. Farin and et.al. Fairing cubic B-spline curves. *Computer Aided Geometric Design*, 4:91–103, 1987.

[24] D.R. Ferguson. Construction of curves and surfaces using numerical optimization techniques. *Computer Aided Design*, 18(1):15–21, January 1986.

[25] T.A. Foley and G.M. Nielson. Knot selection for parametric spline interpolation. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design*, pages 261–272. 1988.

[26] D.R. Forsey and R.H. Bartels. Hierarchical B-spline refinement. *Computer Graphics*, 22(4):205–212, August 1988.

[27] D.R. Forsey and R.H. Bartels. Local refinement editing of B-spline surfaces. In *Graphics Interface 88*, pages 125–126, 1988.

[28] D.R. Forsey and R.H. Bartels. Surface fitting with hierarchical splines. *ACM Transactions on Graphics*, 14(2):134–161, April 1995.

[29] B. Fowler and R.H. Bartels. Constraint-based curve manipulation. *IEEE Computer Graphics & Applications*, cari volume(and number):43–49, September 1993.

[30] P.L. Gengoux and M. Mekhilef. Optimization of a NURBS representation. *Computer Aided Design*, 25(11):699–710, November 1993.

[31] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization.* Academic Press, San Diego, California, 1981.

[32] G. Golub. Differentiation of pseudoinverses, separable nonlinear least squares problems and other tales. In M.Z. Nashed, editor, *Generalized Inverses and Applications : Proceedings of an Advanced Seminar*, pages 303–324. 1973.

[33] G.H. Golub and V. Pereyra. The differentiation of pseudo-inverse and nonlinear

least square problems whose variables separate. *SIAM Journal of Numerical Analysis*, 10(2):413–432, April 1973.

[34] G.H. Golub and C. van Loan. *Matrix Computations*. John Hopkins University Press, 1996.

[35] H. Hagen. Geometric spline curves. *Computer Aided Geometric Design*, find the volume, 1985.

[36] P.J. Hartley and C.J. Judd. Parametrization and shape of B-spline curves for CAD. *Computer Aided Design*, 12(5):235–238, September 1980.

[37] J.G. Hayes and J. Halliday. The least-squares fitting of cubic spline surfaces to general data sets. *Journal of The Institute of Mathematics and Its Applications*, 14:89–103, 1974.

[38] W. Heidrich. An OpenGL driver for MAPLE. Master's thesis, University of Waterloo, 1995.

[39] M.E. Hohmeyer and B.A. Barsky. Skinning rational B-spline curves to construct an interpolatory surface. *CVGIP: Graphical Models and Image Processing*, 53(6):511–521, November 1991.

[40] M. Hosaka. Theory of curves and surface synthesis and their smooth fitting. *Information Processing in Japan*, 9:60–68, 1969.

[41] J Hoschek. Offset curves in the plane. *Computer Aided Design*, 17(2):77–82, 1987.

[42] J Hoschek. Intrinsic parametrization for approximation. *Computer Aided Geometric Design*, 5:27–31, 1988.

[43] J Hoschek. Spline approximation of offset curves. *Computer Aided Geometric Design*, 5:33–40, 1988.

[44] J. Hoschek and D. Lasser. *Fundamentals of Computer Aided Geometric Design.* AK Peters, Wellesley, Massachusetts, 1993.

[45] J Hoschek, F.J. Schneider, and P. Wassum. Optimal approximate conversion of spline surfaces. *Computer Aided Geometric Design*, 6:293–306, 1989.

[46] J. Hourmouziadis. Aerodynamic design of low pressure turbine. In *Advisory Group for Aerospace Research & Devlopement Lecture Series No. 167: Blading Design for Axial Turbomachines*, pages 8.1–8.40. 1989.

[47] D. Japikse and N.C. Barnes. *Introduction to Turbomachinery.* 1994.

[48] A.K. Jones. Shape control of curves and surfaces through constrained optimization. In G.E. Farin, editor, *Geometric Modeling: Algorithms and New Trends, SIAM*, pages 265–278. 1987.

[49] D.L.B. Jupp. Curve fitting by splines as an example of unconstrained optimization. In R.S. Anderssen, L.S. Jennings, and D.M. Ryan, editors, *Optimization*, pages 49–59. University of Queensland Press, 1972.

[50] D.L.B. Jupp. The lethargy theorem - a property of approximation by γ-polynomials. *Journal of Approximation Theory*, 14:204–217, 1975.

[51] D.L.B. Jupp. Approximation to data by splines with free knots. *SIAM Journal of Numerical Analysis*, 15(2):328–343, April 1978.

[52] J.A.P. Kjellander. Smoothing of bicubic parametric surfaces. *Computer Aided Design.*

[53] D. Kuchemann. *The Aerodynamic Design of Aircraft.* Pergamon Press, 1978.

[54] E.T.Y. Lee. Choosing nodes in parametric curve interpolation. *Computer Aided Design*, 21:363–370, 1989.

[55] E.T.Y. Lee. On a class of data parametrization: variations on a theme of Epstein, II. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design II*, pages 381–390. 1991.

[56] E.T.Y. Lee. Corners, cusps, and parameterizations: variations on a theorem of Epstein. *SIAM Journal of Numerical Analysis*, 29, 1992.

[57] N.J. Lott and D.I. Pullin. Method for fairing B-spline surfaces. *Computer Aided Design*, 20(10):597–604, December 1988.

[58] T. Lyche, E. Cohen, and K. Morken. Knot line refinement algorithms for tensor product splines. *Computer Aided Geometric Design*, 2(1-3):133–139, 1985.

[59] T. Lyche and K. Morken. Knot removal for parametric B-spline curves and surfaces. *Computer Aided Geometric Design*, 4:217–230, 1987.

[60] T. Lyche and K. Morken. A data reduction strategy for splines with applica-

tions to the approximation of functions and data. *IMA Journal of Numerical Analysis*, 8:185–208, 1988.

[61] W. Ma and J.P. Kruth. Parameterization of randomly measured points for the least squares fitting of B-spline curves and surfaces. *Computer Aided Design.*

[62] W. Ma and J.P. Kruth. NURBS curve and surface fitting and interpolation. In M. Daehlen, T. Lyche, and L.L. Schumaker, editors, *Mathematical Methods for Curves and Surfaces*, pages 315–322. 1995.

[63] S.P. Marin. An approach to data parametrization in parametric cubic spline interpolation problems. *Journal of Approximation Theory*, 41:64–86, 1984.

[64] B.W. McCormick. *Aerodynamics, Aeronautics, and Flight Mechanics.* John Wiley and Sons, Inc., 1995.

[65] M.J. Milroy, C. Bradley, G.W. Vickers, and D.J. Weir. G-1 continuity of B-spline surface patches in reverse engineering. *Computer Aided Design*, 27(6):471–478, 1995.

[66] S. Mukherjee and B. Gurumoorthy. Surface approximation in reverse engineering. In *DE-Vol.69-1, Advances in Design Automation*, volume 1. ASME, 1994.

[67] B. Noble. Methods for computing the Moorse-Penrose generalized inverse, and related matters. In M.Z. Nashed, editor, *Generalized Inverses and Applications : Proceedings of an Advanced Seminar*, pages 245–302. 1973.

[68] M.R. Osborne. Some special nonlinear least square problems. *SIAM Journal of Numerical Analysis*, 12(4):571–592, September 1975.

[69] L. Piegl and W. Tiller. Algorithm for approximate NURBS skinning. *Computer Aided Design*, 28(9):699–706, 1996.

[70] L. Piegl and W. Tiller. *The NURBS Book*. Springer Verlag, 1997.

[71] L. Piegl and W. Tiller. Symbolic operators for NURBS. *Computer Aided Design*, 29(5):361–368, 1997.

[72] H. Pottmann. Smooth curves under tension. *Computer Aided Design*, 22(4):241–245, May 1990.

[73] W.H Press and et. al. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[74] W. Renz. Interactive smoothing of digitized point data. *Computer Aided Design*, 14(5):267–269, September 1982.

[75] R.F. Riesenfeld. Modeling with NURBS curves and surfaces. In *Fundamental Development of Computer-Aided Geometric Modeling*, pages 77–97. 1993.

[76] D.F. Rogers and N.G. Fog. Constrained B-spline curve and surface fitting. *Computer Aided Design*, 21(10):641–648, December 1989.

[77] N. Sapidis and G. Farin. Automatic fairing algorithm for B-spline curves. *Computer Aided Design*, 22(2):121–129, March 1990.

[78] B. Sarkar and C.H. Menq. Parameter optimization in approximating curves and surfaces to measurement data. *Computer Aided Geometric Design*, 8:267–290, 1991.

[79] B. Sarkar and C.H. Menq. Smooth-surface approximation and reverse engineering. *Computer Aided Design*, 23(9):623–628, November 1991.

[80] B.Q. Su and D.Y. Liu. *Computational Geometry*. Academic Press, 1989.

[81] W. Tiller. Rational B-splines for curve and surface representation. *IEEE Computer Graphics & Applications*, 4(6):61–69, 1983.

[82] W. Tiller. Knot-removal algorithms for NURBS curves and surfaces. *Computer Aided Design*, 24(8):445–453, August 1992.

[83] X. Wang, F. Cheng, and B.A. Barsky. Energy and B-spline interproximation. *Computer Aided Design*, 29(7):485–496, 1997.

[84] W. Welch and A. Witkin. Variational surface modeling. *Computer Graphics*, 26(2):157–166, July 1992.

[85] C. Woodward. Cross-sectional design of B-spline surfaces. *Computer and Graphics*, 11(2):193–201, 1987.

[86] C. Woodward. Skinning techniques for interactive B-spline surfaces. *Computer Aided Design*, 20(8):441–451, 1988.

[87] X. Ye, T.R. Jackson, and M. Patrikalakis. Geometric design of functional surfaces. *Computer Aided Design*, 28(9):741–752, 1996.

[88] I. Zeid. *CAD/CAM Theory and Practice.* 1991.

Appendix A

Non Uniform Rational B-Spline Basics

This chapter briefly discusses the basics of nonuniform rational B-splines, abbreviated as NURBS. The discussion is limited to material which is important to this thesis; complete and detailed discussion on NURBS can be found in textbooks[22, 44, 70]. The first section discusses the definition, properties, and derivatives, of B-spline basis functions. The second section discusses the definition, properties, and derivatives, of rational B-spline basis functions, and the definition and properties of NURBS curves and surfaces. The last section lists existing methods to construct NURBS curves and surfaces.

A.1 B-spline Basis Functions

A.1.1 Definition of B-spline Basis

There are a number of ways to define the B-spline basis functions and to prove their important properties, e.g., by divided differences of truncated power functions, by blossoming, and by a recurrence formula. Here the recurrence formula is used to illustrate the definition of the B-spline basis functions. This formula leads to efficient computer implementation.

A B-spline basis functions is defined along a knot vector, a simple sequence of nondecreasing real numbers. Knot vectors are denoted by $\mathbf{u}$ and their definitions is

$$\mathbf{u} = \{\mathbf{u} \in \Re^q : u_{i+1} - u_i \geq 0 \text{ for } i = 1 : (q-1)\} \tag{A.1}$$

A knot u_i is said to have a multiplicity of $m \geq 1$ if $u_{i-p} = u_{i-p+1} = \cdots = u_{i-1} = u_i = u_{i+1} = \cdots = u_{i+p-1} = u_{i+m-p-1}$ for $0 \leq p < m$.

B-spline basis functions are specified by two integers: the order of the basis function and the index of the basis function. The notation $N_i^k(\mathbf{u}; t)$, or $N_i^k(t)$ for its short notation, refers to the i-th B-spline basis function of order k defined on knot vector $\mathbf{u}$. In most literature, the knot vector is usually eliminated from the notation of the B-spline basis; the short notation is $N_i^k(t)$. The order of B-spline basis function must satisfy inequality $1 \leq k \leq (q-1)$, and the index of B-spline basis function must satisfy inequality $1 \leq i \leq (q-k)$.

B-spline basis functions are evaluated at a value t in the parameter space, and

the complete definition of B-spline basis function at t is

- for $k = 1$:

$$N_i^1(t) = \begin{cases} 1 & \text{if } u_i \leq t < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{A.2}$$

- for $k > 1$:

$$N_i^k(t) = (t - u_i)\frac{N_i^{k-1}(t)}{u_{i+k-1} - u_i} + (u_{i+k} - t)\frac{N_{i+1}^{k-1}(t)}{u_{i+k} - u_{i+1}} \tag{A.3}$$

where

$$\frac{N_j^{k-1}(t)}{u_{j+k-1} - u_j} = 0 \qquad \text{if } u_{j+k-1} - u_j = 0 \text{ for all } j \tag{A.4}$$

A.1.2 Properties of B-spline Basis

The above definition of the basis function results in the following properties for B-spline basis function.

1. **Local support property:** The influence of $N_i^k(t)$ is limited to a portion of the parameter space N_i^k, if t is located outside the support of $N_i^k(t)$, that is $t \notin [u_i, u_{i+k})$.

2. **Nonnegativity:** $N_i^k(t)$ is nonnegative, that is $N_i^k(t) \geq 0$, for all values of t and all permissible values of i and k.

3. **Single Maximum:** Except for $k = 1$, $N_i^k(t)$ attains exactly one maximum value.

4. **Invariance under affine transformation:** when knot vector $\mathbf{u}$ and parameter t are subjected to affine transformation $u_i^* = \alpha u_i + \beta$ for all i and $t^* = \alpha t + \beta$ respectively, $N_i^k(\mathbf{u}^*, t^*) = N_i^k(\mathbf{u}, t)$.

5. **Partition of unity:** this property only applies to the parameter values in the domain knot. For a set of B-spline basis functions that are of order k and that are defined on a knot vector $\mathbf{u}$ with q elements where $q \geq 2k$, the domain knots are defined as

$$\mathbf{u}^d = \left\{\mathbf{u}^d \in \Re^{(q-2k+2)} : u_i^d = u_{i+k-1} \text{ for } i = 1 : (q - 2k + 2)\right\} \tag{A.5}$$

   For all $t \in \mathbf{u}^d$, the sum of all supported $N_i^k(t)$ is equal to one, that is $\sum_{j=i-k+1}^{i} N_j^k(t) = 1$.

In addition to these, B-spline basis functions enjoy other properties which can be found in textbooks[22, 44, 70].

A.1.3 Derivatives of B-spline Basis

The B-spline basis function depends on the parameter and the knot sequence. The derivatives with respect to them are shown below:

Derivatives of B-spline Basis with Respect to Parameter

The derivative of a B-spline basis function with respect to parameter t is defined as

$$\frac{\partial N_i^k(t)}{\partial t} = (k-1)\left(\frac{N_i^{k-1}(t)}{u_{i+k-1}-u_i} - \frac{N_{i+1}^{k-1}(t)}{u_{i+k}-u_{i+1}}\right) \tag{A.6}$$

The derivative of B-spline basis function with respect to parameter exists if and only if t is inside the support of N_i^k, that is $t \in [u_i, u_{i+k})$. In general, the B-spline basis function is $(k-2)$-times continuously differentiable; however, if t assumes the value of one of the supporting knots of N_i^k, the continuity will also depends on the multiplicity of the knot and the B-spline basis functions becomes $(k-1-m)$-times continuously differentiable where m is the multiplicity of the knot. Details of this derivation can be found in standard B-spline textbooks[22, 44, 70].

Derivatives of B-spline Basis with Respect to Knots

The derivative of a B-spline basis function with respect to an element of the knot vector is defined as

$$\frac{\partial N_i^k(\mathbf{u},t)}{\partial u_j} = \frac{\hat{N}_{i+1}^k(\hat{\mathbf{u}},t)}{\hat{u}_{i+k+1}-\hat{u}_{i+1}} - \frac{\hat{N}_i^k(\hat{\mathbf{u}},t)}{\hat{u}_{i+k}-\hat{u}_i} \tag{A.7}$$

where $\hat{\mathbf{u}}$ is a knot vector obtained by raising the multiplicity of u_j by one, and $\hat{N}^k$ is B-spline basis of the same order as N^k's and is define on $\hat{\mathbf{u}}$.

The derivative of a B-spline basis function with respect to a support knot possesses a special characteristic when the multiplicity of the knot is more than one. Suppose the multiplicity of knot u_j is two, that is $u_j = u_{j+1}$ as shown in Figure (A.1). Let $\mathbf{s}$ and $\mathbf{v}$ denote the knot vectors obtained from $\mathbf{u}$ by raising the multiplicity of u_j and u_{j+1} by one respectively, as shown in Figure (A.2), it is clear that $\mathbf{s} = \mathbf{v}$ since $u_j = u_{j+1}$. Thus, the basis of order k defined on $\mathbf{s}$ and $\mathbf{v}$ are identical. When these basis are applied to Equation (A.7), they yield $\partial N/\partial u_j = \partial N/\partial u_{j+1}$. In general, if a knot u_j has multiplicity of m, i.e. $u_j = u_{j+1} = \cdots = u_{j+m-1} < u_{j+m}$ the equalities on the first derivative of B-spline basis with respect to these coinciding knots are

$$\frac{\partial N}{\partial u_j} = \frac{\partial N}{\partial u_{j+1}} = \cdots = \frac{\partial N}{\partial u_{j+m-1}} \tag{A.8}$$

This property of the derivative is the background of what is known as *the lethargy theorem*. This lethargic property is intrinsic to free knot spline problems and affects the stable and effective computation of optimal knots [50, 49, 51, 17]. This theorem were discussed in Chapter 2.

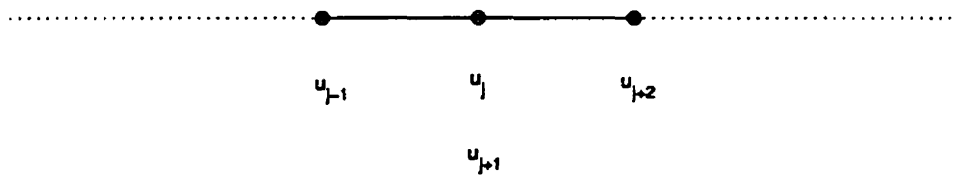


Figure A.1: Double Knots, $u_j = u_{j+1}$

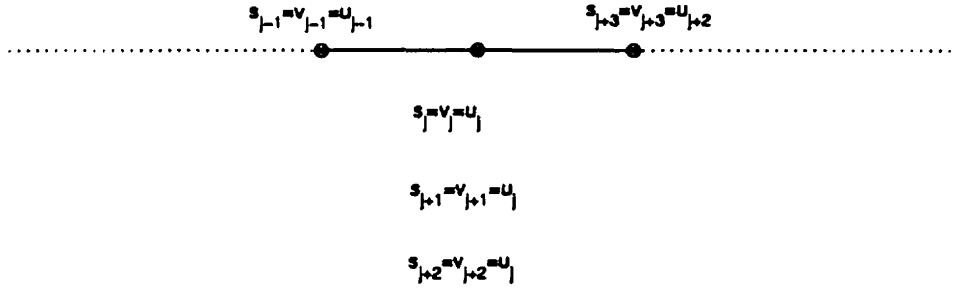


Figure A.2: Triple Knots. s and v are obtained by raising the multiplicity of u_j and u_{j+1} respectively

A.2 NURBS Curves and Surfaces

A.2.1 Rational B-spline Basis Function

The rational B-spline basis function is defined as

$$R_i^k(\mathbf{w}, \mathbf{u}, t) = \frac{w_i N_i^k(u)}{\sum_{i=1}^{n} w_i N_i^k(u)} \tag{A.9}$$

where $\mathbf{w}$ is an array of real numbers, $\mathbf{w} \in \Re^n$ and $n = q - k$; the short notation of the rational B-spline basis function is $R_i^k(t)$. The rational B-spline basis function may become singular whenever the denominator in Eq. (A.9) is zero. Singularities can be avoided by setting all weights to be nonzero and uniform in sign. Depending

on the signs of the weights, rational B-spline basis functions may have positive or negative values. However, most of the implementations of rational B-spline curves and surfaces restrict the weights to be positive nonzero in order to preserve the convex hull property of the curves and surfaces. ***Throughout this thesis, the scope of discussion to rational B-spline basis functions is limited to nonzero and positive weights.*** With this limitation, the rational B-spline basis function enjoys the same properties of the B-spline basis function.

The derivatives of rational B-spline basis function with respect to its variables can be obtained by using the chain-rule of derivatives and the derivatives of B-spline basis in Eqns. (A.6) and (A.7). They are defined as the followings:

Derivative of Rational Basis with respect to Parameter

The derivative of a rational B-spline basis function with respect to parameter t is defined as

$$\frac{\partial R_j}{\partial t} = \frac{w_j}{\sum_{i=1}^{n} w_i N_i^k} \frac{\partial N_j}{\partial t} - \frac{w_j N_j}{\left(\sum_{i=1}^{n} w_i N_i^k\right)^2} \left(\sum_{k=1}^{n} w_k \frac{\partial N_k}{\partial t} \right) \tag{A.10}$$

Derivative of Rational Basis with respect to Knot

The derivative of a rational B-spline basis function with respect to element of an element of knot vector u is defined as

$$\frac{\partial R_{ij}}{\partial u_q} = \frac{w_j}{\sum_{i=1}^{n} w_i N_i^k} \frac{\partial N_j}{\partial u_q} - \frac{w_j N_j}{\left(\sum_{i=1}^{n} w_i N_i^k\right)^2} \left(\sum_{k=1}^{n} w_k \frac{\partial N_k}{\partial u_q} \right) \tag{A.11}$$

Derivative of Rational Basis with respect to Weight

The derivative of a rational B-spline basis function with respect to weight w is defined as

$$\frac{\partial R_j}{\partial w_q} = \frac{N_j}{\sum_{i=1}^{n} w_i N_i^k} \delta_{jq} - \frac{w_j N_j}{\left(\sum_{i=1}^{n} w_i N_i^k\right)^2} N_q \tag{A.12}$$

A.2.2 NURBS Curves

NURBS curves and surfaces are defined by three sets of parameters: (1) the knot vector, (2) the control polygon and (3) the weights; the formulation of NURBS curves is

$$\mathbf{x}(t) = \sum_{i=1}^{n} \mathbf{p}_i R_i^k(t) \tag{A.13}$$

where $\mathbf{p}$'s are the nodes of control polygon and $R_i^k(u)$ is the rational B-spline basis function as defined in Eq. (A.9).

The dimensions of NURBS curves is dictated by the dimension of the Euclidean point $\mathbf{p}$ of the control polygon.

NURBS curves enjoy the following properties:

1. **Affine invariance:** affine invariance transformations can be performed by applying the transformations to the control polygon.

2. **Strong convex hull property:** for $t \in [u_i, u_{i+1})$, $\mathbf{x}(t)$ lies within the convex hull of control points $\mathbf{p}_{i-k+1}, \mathbf{p}_{i-k+2}, \cdots, \mathbf{p}_{i-1}, \mathbf{p}_i$.

3. **Local approximation:** control point $\mathbf{p}_i$ has local influence on the curve, that is only portion of the curve where $t \in [u_i, u_{i+k})$ is influenced by $\mathbf{p}_i$.

4. For $t \in \mathbf{u}^d$, $\mathbf{x}(t)$ is infinitely differentiable except when t assumes a value of a knot; in this case, $\mathbf{x}(t)$ becomes $k - m - 1$ times differentiable where m is the multiplicity of the knot.

A.2.3 NURBS Surfaces

A NURBS surface is defined as

$$\mathbf{x}(t,s) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} \mathbf{p}_{ij} R_j^l(s) \right) R_i^k(t) \tag{A.14}$$

where $\mathbf{p}_{ij}$'s are the nodes of control nets, R_i^k is rational B-spline basis functions defined on a knot $\mathbf{u}$, and R_j^l are rational B-spline basis functions defined on a knot $\mathbf{v}$. NURBS surfaces enjoy the following properties

1. **Affine invariance:** affine invariance transformations can be performed by applying the transformations to the control net.

2. **Strong convex hull properties:** for $(t, s) \in [u_i, u_{i+k}) \times [v_j, v_{j+l})$, the $\mathbf{x}(t, s)$ is in the convex hull of sub-net $\mathbf{p}_{[i-k+1:i],[j-l+1:j]}$

3. **Local modification:** $\mathbf{p}_{ij}$ only influences a part of the surfaces where $u_i \leq t < u_{i+k}$ and $v_j \leq s < v_{j+l}$.

4. For $t \in \mathbf{u}^d$, $\mathbf{x}(t, s)$ is infinitely differentiable in the direction of u except when t assumes a value of one of knots in $\mathbf{u}$; in this case, $\mathbf{x}(t, s)$ becomes $k - m - 1$ times differentiable in the direction of $\mathbf{u}$ where m is the multiplicity of the knot.

   For $s \in \mathbf{v}^d$, $\mathbf{x}(t, s)$ is infinitely differentiable in the direction of v except when s assumes a value of one of knots in $\mathbf{v}$; in this case, $\mathbf{x}(t, s)$ becomes $l - h - 1$ times differentiable in the direction of $\mathbf{v}$ where h is the multiplicity of the knot.

A.3 Construction of Curves and Surfaces

Construction refers to procedures to compute or specify the parameters of NURBS curves and surfaces: control polygon/net, weights, degrees, and knots. Many types of curve and surface construction techniques exist; the selection depends on the type of information passed on to the designers. There are three basic types of curve constructions:

1. Conversion from other types of curves. In this method of construction, a curve defined in a certain representation, e.g. conics or monomials, is given

and the parameters of a NURBS curve are computed based on the parameters of the given curve. However, conversion from one type of curves to NURBS is not always possible; exact conversion can only be performed on few types of given curves.

2. Fitting, refers to the construction of NURBS curves from a given set of geometric data, usually in the form of points and derivative vectors. There are two types of curve fitting: interpolation and approximation. Interpolation produces a NURBS curve that exactly satisfies the data whereas approximation only approximate the given data.

3. Modification of NURBS curves. This type of construction starts with a NURBS curve and a geometric modification information (shape, constraints, etc.) and subsequently the NURBS curve is modified to incorporate the geometric modification information. Many interesting modification techniques exist: warping, flattening, bending, constraint-based curve modification, free-form deformation, to name a few.

There are four types of techniques of surface constructions:

1. Conversion: almost identical to construction of curves except that the input is a surface definition instead of a curve.

2. Modification of NURBS surfaces. As in the curve modification, surface modification techniques also recognize warping, flattening, constrained surface modification, bending, and many more.

3. Fitting NURBS surfaces to a given set of points and derivatives vector; as in the construction of curves, surface fitting techniques are classified into interpolation and approximation.

4. Curve based construction. This type of construction is specific to surfaces; no equivalent type of construction exists in curve construction methods. One or more set of NURBS curves are given, sometimes accompanied by a set of points and derivative vectors, and a NURBS surface is constructed based on the given information. The surfaces produced from this technique have special names that describe the way the surfaces are constructed, e.g. ruled surface, surface of revolution, swung surface, skinned surface, swept surface, to name a few.

Appendix A has given the definition of NURBS curves and surfaces and has described some of the important properties relevant to this work.

Appendix B

Curve Smoothing

This appendix presents a brief summary on the smoothness of a curve and the standard methods to improve it. Curve smoothness, a.k.a. curve fairness, is a measure of geometric quality. Farin[22] defines fair curve as a curve whose curvature plot is continuous and consists of only a few monotone pieces. The number of curvature extrema of a fair curve should be few and the extrema should only occur where explicitly desired by the designer, and nowhere else! This definition of fairness is well accepted[9, 18, 80]. Although this definition is subjective, it has proven to be very practical. Other existing definitions of measures of smoothness can be found in Hoschek's text[44].

Improvement of smoothness consists of two parts: searching for unwanted inflections and subsequently removing them. Methods to search unwanted inflections can be classified into two: manual and automatic. Manual search is performed by manual inspection of the curvature plot, i.e. the plot of curvature versus arc length. For planar curves, the inflection points are characterized by the curvature's

change of sign. On the other hand, the curvature of a space curve is nonnegative by definition; the most common practice to locate the inflection points is to project the curve onto a plane and search for change in sign on the projected curve. Existing methods of automatic search for surfaces consist of three types: isoline method, reflection line method, and mapping method; details of these methods can be obtained in Hoschek's text[44]. This research uses manual search of inflection points.

Once unwanted inflection points are found, their removal is performed. As far as removal of inflection points is concerned, there are two types of inflection points: those at the knot, i.e. $t = u_r$, and those within a knot interval, i.e. $t \neq u_r$. This section focuses on the removal of first type of inflection points. They are caused by the decrease in the differentiability of the B-spline basis at the offending knot; this decrease is one of the properties of the basis. Farin suggests that for an offending knot, u_r, the remedy is to decrease the B-spline basis' differentiability by translating the control points such that two consecutive segments $[u_{r-1}, u_r]$ and $[u_r, u_{r+1}]$ actually becomes a single segment, in effect, eliminating the decrease of number of times differentiability of B-spline basis at u_r. Farin's method to remove an inflection point at u_r minimizes the number of control points that must be translated in order to localize the change experience by the curve. For NURBS curves of degree p and control points $\mathbf{c}_i$, removal of knot u_r of multiplicity s, translation of control points depends on whether $p - s$ is an even or an odd number. The following presents Farin's method of translation of control points.

- Even $p - s$

Let $j = (p-s)/2$, $\mathbf{c}^*_{r-p-1} = \mathbf{c}_{r-p-1}$, and $\mathbf{c}^*_{r-s} = \mathbf{c}_{r-s+1}$.

Compute in the forward direction

$$\mathbf{c}^*_i = \frac{\mathbf{c}_i - (1-\alpha_i)\,\mathbf{c}^*_{i-1}}{\alpha_i} \qquad \text{for } i = (r-p) : (r-p+j-1) \tag{B.1}$$

Compute in the backward direction

$$\mathbf{c}^*_i = \frac{\mathbf{c}_{i+1} - \alpha_{i+1}\mathbf{c}^*_{i+1}}{(1-\alpha_{i+1})} \qquad \text{for } i = (r-s-1) : -1 : (r-p+j) \tag{B.2}$$

Translate $\mathbf{c}_{r-p+j}$ to a new position,

$$\mathbf{c}^*_{r-p+j} = \alpha_{r-p+j}\mathbf{c}^*_{r-p+j} + (1-\alpha_{r-p+j})\,\mathbf{c}^*_{r-p+j-1} \tag{B.3}$$

The original curve will be displaced not larger than ν where

$$\nu = \begin{cases} \left|\left|\mathbf{c}^*_{r-p+j} - \mathbf{c}_{r-p+j}\right|\right|_2 & \text{for integral curve} \\ \left(\frac{1+\max\limits_i \|\mathbf{c}_i\|_2}{\min\limits_i w_i}\right) \left|\left|\mathbf{c}^*_{r-p+j} - \mathbf{c}_{r-p+j}\right|\right|_2 & \text{for rational curve} \end{cases} \tag{B.4}$$

- Odd $p-s$

Let $j = (p - s - 1)/2$, $\mathbf{c}^*_{r-p-1} = \mathbf{c}_{r-p-1}$, and $\mathbf{c}^*_{r-s} = \mathbf{c}_{r-s+1}$.

Compute in the forward direction

$$\mathbf{c}^*_i = \frac{\mathbf{c}_i - (1 - \alpha_i)\,\mathbf{c}^*_{i-1}}{\alpha_i} \qquad \text{for } i = (r-p) : (r-p+j-1) \tag{B.5}$$

Compute in the backward direction

$$\mathbf{c}^*_i = \frac{\mathbf{c}_{i+1} - \alpha_{i+1}\mathbf{c}^*_{i+1}}{(1 - \alpha_{i+1})} \qquad \text{for } i = (r-s-1) : -1 : (r-p+j+1) \tag{B.6}$$

Translate $\mathbf{c}_{r-p+j}$ with a translation vector of

$$\frac{-(1 - \alpha_{r-p+j+1})}{(1 - \alpha_{r-p+j+1})^2\,\alpha^2_{r-p+j}}\left(\mathbf{c}^*_{r-p+j} - \mathbf{c}^*_{r-p+j+1}\right) \tag{B.7}$$

Translate $\mathbf{c}_{r-p+j+1}$ with a translation vector of

$$\frac{\alpha_{r-p+j}}{(1 - \alpha_{r-p+j+1})^2\,\alpha^2_{r-p+j}}\left(\mathbf{c}^*_{r-p+j} - \mathbf{c}^*_{r-p+j+1}\right) \tag{B.8}$$

The original curve will be displaced not larger than ν where

$$\nu = \begin{cases} \|\mathbf{d}\|_2 & \text{for integral curve} \\ \left(\frac{1+\max_i \|\mathbf{c}_i\|_2}{\min_i w_i}\right) \|\mathbf{d}\|_2 & \text{for rational curve} \end{cases} \tag{B.9}$$

where

$$\mathbf{d} = \frac{-(1-\alpha_{r-p+j+1}) + \alpha_{r+p+j}}{(1-\alpha_{r-p+j+1})^2 \alpha_{r-p+j}^2} \left(\mathbf{c}^*_{r-p+j} - \mathbf{c}^*_{r-p+j+1}\right) \tag{B.10}$$

B.1 Partial Smoothing

Partial smoothing refers to translation of control points in the direction specified in the previous sections but the amount of translation is smaller than those suggested in Equations (B.3), (B.7)and (B.8). Partial smoothing is used whenever there exists upper limit that ν is prohibited to exceed. Partial smoothing compromises improvement of curvature distribution and limitation on nu.

Appendix C

Basics of Skinning

Skinning is a process of blending a set of section curves together to form a surface[70]. It is a newer term for lofting. The blending direction is in the longitudinal direction. The blending may interpolate or approximate the curves, although the former is more common than the latter[69]. This chapter focuses on interpolatory skinning over sets of NURBS section curves, resulting in NURBS skinned surfaces.

NURBS section curves are defined in u direction and skinning is performed in the v direction. Prior to skinning the curves must be made compatible, i.e. they share a mutual degree and a mutual knot vector $\mathbf{u}$. The mutual degree is achieved by raising the degrees of the curves up to the degree of the curve with highest degree. Then, knot merging is performed to produce a mutual knot vector. Knot insertion is performed as necessary to the curves to make their knot vectors identical to the mutual knot vector. The compatibility makes the curves have an identical number of control points. Figure C.1 shows an example of a set of NURBS section curves that already made compatible; the set contains m NURBS

curves, each of which has n control points. The section curves in this figure are in the top-bottom (vertical) direction and the skinning is performed in the left-right (horizontal) direction. $\mathbf{P}_{i,j}$ denotes the i-th control point of the j-th curve. Skinning is performed by constructing n interpolating curves each of which, $\mathbf{X}_k(v)$, interpolates $\mathbf{P}_{k,1:m}$.

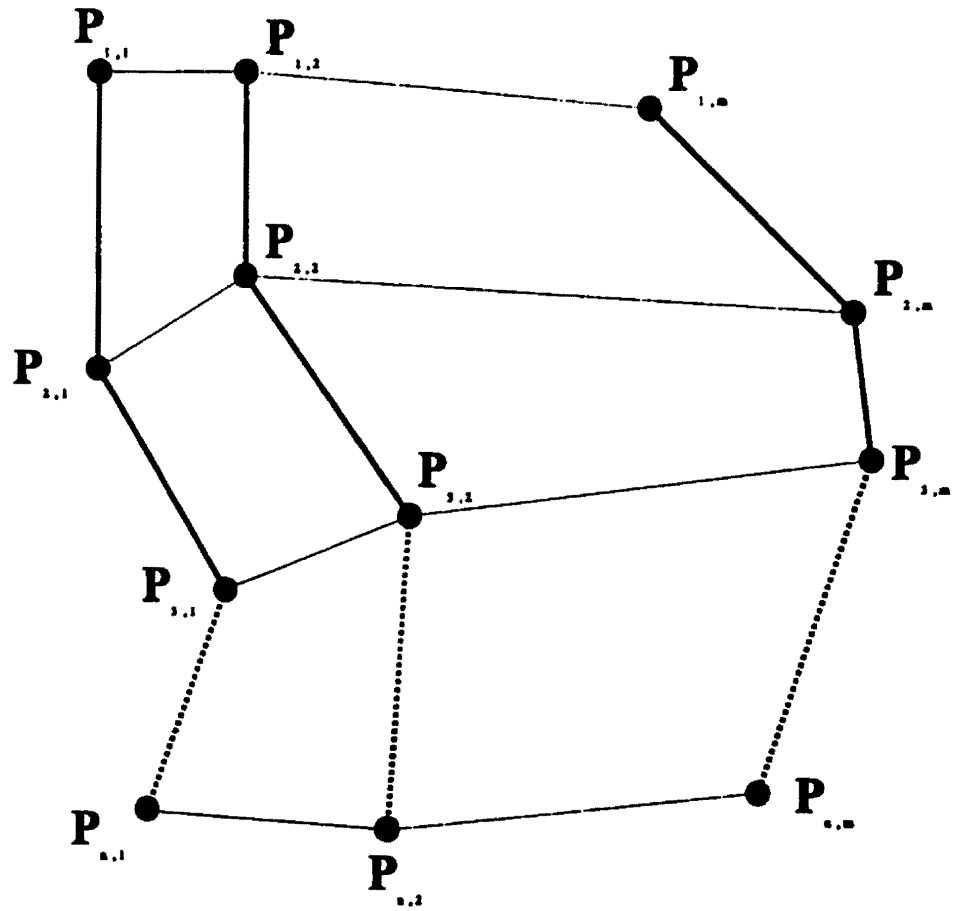


Figure C.1: Control points of m section curves each of which consists of n control points

This appendix focuses on the B-spline interpolation method where the knots need not coincide with the parameter values for the data and where the number

of unknown control points is exactly identical to the number of data[70]. Thereby, end conditions are less significant in this work. This type of interpolation method is described in many texts[44, 70, 22]. For the example of Figure C.1, the interpolation can be expressed as

$$\mathbf{R}\mathbf{d}_i = \mathbf{p}_i \quad \text{for } i = 1 : n \tag{C.1}$$

where

$$\mathbf{p}_i = \begin{Bmatrix} \mathbf{P}_{i,1} \\ \mathbf{P}_{i,2} \\ \vdots \\ \mathbf{P}_{i,m} \end{Bmatrix} \tag{C.2}$$

Equation (C.1) will produce n control polygons each of which consists of m control points. These control polygons form the control net of the NURBS skinned surface. Equation (C.1) requires that a mutual square matrix of rational B-spline basis, $\mathbf{R}$, is shared by all of the interpolating equations. This is needed to conform to the definition of NURBS surface as shown in Equation (A.14).

The need of a mutual $\mathbf{R}$ is satisfied if every row control points $\mathbf{P}_{k,1:j}$ shares a mutual parameter value in the v direction, e.g. s_k. Piegl proposes a method to compute these parameters as follows:

$$
\begin{aligned}
s_k &= \begin{cases} 0 & \text{for } k = 1 \\ s_{k-1} + \phi_k & \text{for } k = 2 : (m-1) \\ 1 & \text{for } k = m \end{cases} \qquad \text{(C.3)} \\
\text{where } \phi_k &= \frac{1}{n} \sum_{i=1}^{n} \left(\frac{\delta_{i,k}}{\sum_{j=2}^{m} \delta_{i,j}} \right) \\
\text{and } \delta_{i,k} &= \delta \left(\mathbf{P}_{i,k} \ , \ \mathbf{P}_{i,k-1} \right)
\end{aligned}
$$

The parameterization function $\delta(\mathbf{P}_{i,k}, \mathbf{P}_{i,k-1})$ can be selected from one of the existing methods of computing the parameterization of given data points, such as uniform parameterization, arc length parameterization, centripetal parameterization, etc; see [12, 14, 38, 54]. However, to the author's knowledge, those parameterization are intended for single curve interpolation. None of them has been tested for parameterization for skinning. Therefore, we take a safe route by selecting the well known arc length parameterization.

The construction of mutual knot vector $\mathbf{v}$ follows after $\mathbf{s}$ has been computed. The method developed by Piegl[70] is selected to calculate the knots along the skinning direction. Interpolation of q points using B-spline basis of order k requires knots $\mathbf{v} \in \Re^{q+k}$ where its values are:

$$v_j = \begin{cases} 0 & \text{for } i = 1:k \\ \frac{1}{k-1}\sum_{i=j:}^{j+k-2} & \text{for } i = (k+1):(q-k) \\ 1 & \text{for } i = (q-k+1):q \end{cases} \tag{C.4}$$

This method of computing the knots $\mathbf{v}$ takes the distribution of parameterization of data $\mathbf{s}$ into account such that in most cases the resulting square matrix $\mathbf{R} \in \Re^{q \times q}$ is full column rank and well-conditioned[70].

Once the parameterization for $\mathbf{P}_{i,j}$ and the knot vector $\mathbf{v}$ is obtained, matrix $\mathbf{R}$ can be computed. Then this matrix is substituted into Equation (C.1) to obtain the control net of the skinned surface.

Because the skinning equation is based on curve interpolations, the skinning inherits unsatisfactory results commonly found in curve interpolation, e.g. self-intersecting, poor resulting shape, unwanted inflection, etc. The most common cause of these unfavorable results is poor distribution of knots and parameterization for the data. Particularly for NURBS cases there are two additional hazards: mixed weights and continuity gap. The former occurs when one or more weights are zero and/or the weights contains both positive and negative elements. Zero weight causes singularity and mixed weights destroys the convex hull property. The latter hazard refers to inequality of continuity between homogeneous curves and Cartesian curves. Hohmmeyer address the nature of this problem of continuity gap along with methods to solve it[39].

Jones' work is particularly attractive in the shape control of interpolating curves

and skinned surface[48]. The shape control refers to constraints to enforce some convexity measures on the curve. Jones defines *polygon convexity constraint* for the shape control function; its definition for a set of n control points is

$$[(\mathbf{p}_{j+1} - \mathbf{p}_j) \times (\mathbf{p}_{k+1} - \mathbf{p}_j)] \cdot \hat{\mathbf{i}} > 0 \quad \text{for } 1 \leq j < k \leq n \tag{C.5}$$

where $\hat{\mathbf{i}}$ is a predetermined unit vector. It is clear that the constraint is a quadratic inequality constraint. Jones states that any set of control points satisfying the polygon convexity constraint will produce a NURBS curve whose projection on the plane perpendicular $\hat{\mathbf{i}}$ is convex everywhere. To obtain such NURBS curve interpolating a set of data, Jones[48] starts with an arbitrary interpolating curve. Then the control points are modified to satisfy the constraint. Denoting the modified control points by $\mathbf{p}^*$, the objective function is a quadratic function defined as

$$\sum_i (\mathbf{p}_i^* - \mathbf{p}_i)^T (\mathbf{p}_i^* - \mathbf{p}_i) \tag{C.6}$$

Then, the problem is formulated to minimize the objective function subject to the polygon convexity constraint. Jones[48] warns that this quadratic problem subject to quadratic inequality constraint may become a severe test for optimization software. The solution will be an interpolating curve satisfying a prespecified convexity. Jones' work[48] also formulates the convexity constraint that must be

satisfied by two neighboring compatible curves if the blending of those curves must produce section curves satisfying the convexity constraint anywhere along the blending direction. However, the formulation of such constraints become complicated as it is required to convert the products of B-splines of order m into sums of B-spline of order $2n-1$. The conversion boils down to careful selection of knots in order to make the conversion possible.

In summary, the author wishes to raise the reader's awareness of potential hazards that may be encountered in the construction of skinned surfaces. Those hazards are either mathematical or engineering in nature. The former includes singularity whereas the latter include control of surface quality. It is beyond the scope of this work to propose a solution to those hazards; neither does this work attempt to implement the aforementioned measure to enforce quality of the skinned surface. The scope of this work is limited to usage of the skinning method as described in Piegl's text[70].