Robust Aerodynamic Design of Mars Exploratory Airplane Wing with a New Optimization Method

A dissertation presented to
The School of Engineering
The University of Tokyo
Japan

In partial fulfillment
of the requirements for the Degree
Doctor of Philosophy
Aeronautics and Astronautics

by

Koji Shimoyama

February, 2006

Abstract

The use of airplanes for Mars exploration is a new and attractive approach because it provides high resolution power and large spatial coverage. However, it is also a challenging approach in engineering viewpoint. Mars airplanes are required to fly in lower Reynolds number and higher subsonic Mach number conditions due to thinner atmosphere and smaller speed of sound on the Mars, compared to typical commercial Earth airplanes. Some studies of Mars exploratory airplanes have been already reported by many researchers. However, these airplanes were designed only by utilizing and modifying existing design approaches for conventional Earth airplanes. Therefore, the use of a design optimization approach is desirable to realize more effective and global search for better design of Mars airplane and establish a new design concept for Mars airplanes. In addition, it is well known that there exist large wind variations on the Mars. Such wind variations may lead to drastic deterioration in performance, and thus failure in expected Mars exploratory mission. Therefore, it is desirable to consider not only the performance at design point but also robustness of performance against wind variations for more realistic and reliable design of Mars airplane. One of solutions to realize such design is use of a robust design optimization approach. However, traditional robust optimization approaches had lack of capability to reveal trade-off information between optimality and robustness which is useful for real-world robust designs.

In this dissertation, a new robust design optimization approach "design for multi-objective six sigma (DFMOSS)" has been developed to solve the drawbacks of a conventional robust optimization approach "design for six sigma (DFSS)" for more efficient and more useful robust design optimizations, and applied to simple robust optimization problems to investigate efficiency and usefulness of the DFMOSS. This study showed that the DFMOSS has some advantages over the DFSS. First, the DFMOSS does not require the advance specification of input parameters such as weighting factors and sigma level. Second, the DFMOSS obtains multiple robust

optimal solutions effectively by only one optimization run and reveals trade-off information between optimality and robustness. Third, satisfied sigma level of each robust optimal solution can be evaluated easily and flexibly in post-processing of the robust optimizations.

Then, aerodynamic design optimizations of Mars exploratory airplane wing considering the effects of wind variations were realized by using the DFMOSS and the computational fluid dynamics (CFD) simulation. Realistic design information about the trade-off relation between the optimality and the robustness of aerodynamic performance of Mars exploratory airplane has been discussed based on the numerical results.

First, three robust aerodynamic design optimizations of airfoil configuration for Mars exploratory airplane considering the effects of wind variations were carried out. In all cases, the robust aerodynamic design optimization using the DFMOSS revealed the trade-off relation between the optimality and the robustness in aerodynamic performance. In the first case considering the robustness of lift to drag ratio against the variation of flight Mach number, it was shown that an airfoil configuration with smaller maximum camber can improve the robustness in lift to drag ratio against the variation of flight Mach number. This is because such airfoil can suppress the growth of shock wave, i.e., realize smaller increment in pressure drag (wave drag) against increment in flight Mach number. In the second case considering the robustness of pitching moment coefficient against the variation of flight Mach number, it was shown that an airfoil configuration with larger curvature in the front part can improve the robustness of pitching moment coefficient against the variation of flight Mach number. This is because such airfoil can suppress the backward movement of shock wave occurred over the upper surface of airfoil against increment in flight Mach number, and eventually it results in smaller change in pitching-down moment produced in the rear part of airfoil against increment in flight Mach number. In the third case considering the robustness of lift to drag ration against the variation of angle of attack, it was shown that an airfoil configuration with blunter leading edge can improve the robustness of lift to drag ratio against the variation of angle of attack. This is because such airfoil configuration can suppress the growth of the separation bubble generated near the leading edge against increment in angle of attack. It leads to gentler change of negative pressure level in the front part near the leading edge, i.e., smaller change of drag against increment in angle of attack, and eventually it results in smaller change of lift to drag ratio against the variation of

angle of attack.

Second, robust aerodynamic design optimization of airfoil and wing planform configurations for Mars exploratory airplane considering effects of wind variations was carried out. The present robust aerodynamic design optimization found the solutions with robust characteristic in aerodynamic performance parameters against wind variations, and revealed qualitative trade-off information between the optimality and the robustness in aerodynamic performance parameters. This result showed that the DFMOSS is an effective approach even in large-scale robust design optimizations with many design variables and objective functions. Among the robust optimal solutions obtained in the present study, the solution with robust characteristic of lift against wind variations had the wing configuration with supercritical-like airfoil and larger twist-up angle locally at the mid-span section. Such wing configuration involved the leading-edge separation, which leaded to small local lift and large local drag, at the twisted-up section at the design point. Such extreme solution with leading-edge separation seems to be undesirable and unrealistic to be used in real-world design. Therefore, more detailed investigation of the obtained trade-off information without such extreme solutions may be useful for more realistic designs of Mars exploratory airplane. In addition, the present CFD simulation approach based on the Favre-averaged Navier-Stokes equations with the Baldwin-Lomax algebraic turbulence model did not have accuracy enough to simulate the flowfields involving large-scale separation phenomena. Therefore, further investigation of the CFD simulation approach for large-scale separation phenomena may lead to more realistic and more reliable future designs of Mars exploratory airplane.

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Professor Kozo Fujii of the Institute of Space and Astronautical Science (ISAS) at Japan Aerospace Exploration Agency (JAXA). He recommended me an attractive and innovative research theme just when I came to his High Speed Aerodynamics Laboratory of ISAS/JAXA, and also has supported my master's and doctoral studies for five years with his precise guidance and encouragement. I also respect his research ability, stance and personality, and aim to be a great researcher like him in the future.

I would also like to express my gratitude to Professor Etsuo Morishita, Professor Kenichi Rinoie, Associate Professor Katsuhiro Nishinari and Lecturer Takeshi Tsuchiya of Department of Aeronautics and Astronautics at the University of Tokyo. They accepted me to become the dissertation committee, and gave me very valuable advice and suggestions.

I am most grateful to Associate Professor Takeshi Imamura of ISAS/JAXA for providing me useful and detailed scientific information about Martian meteorology. My doctoral study could not have been completed without research collaboration with him.

I am also most grateful to Research Associate Hiroaki Kobayashi of the Institute of Aerospace Technology (IAT) at JAXA for his excellent advice and suggestions. It was the starting point of my master's and doctoral studies when I met him and paid attention to his study for the first time.

Special thanks to Research Associate Akira Oyama of ISAS/JAXA for his dedicated guidance and encouragement. His rich knowledge and experience of design optimization helped me complete my doctoral study very much. And, my studies could have been made known to many other researchers owing to his wide acquaintance.

Additional thanks to Associate Professor Nobuyuki Tsuboi and Associate Professor Ryoji Takaki of ISAS/JAXA, Research Scientist Hiroshi Terashima of the

Institute of Physical and Chemical Research (RIKEN) and Post Doctor Soshi Kawai of the University of Tokyo for their enlightening discussions, constructive comments and encouragement.

Furthermore, I want to thank my colleagues, Mr. Nobuyuki Iizuka and Mr. Keiichiro Fujimoto, and all other former and present members in Professor Fujii's High Speed Aerodynamics Laboratory for their friendship and supports. I am very sad to go away from them after this March. I hope and make sure to keep a good relation with them in the future.

Last but not least, I want to thank my parents and sisters for their supports throughout my graduate studies.

February 2006

下山 幸治

Koji Shimoyama

Contents

1 Introduction **1**
1.1 Backgrounds . . . 1
1.2 Past Studies of Mars Exploratory Airplanes . . . 3
1.3 Past Studies of Design Optimization Approaches . . . 12
1.4 Past Studies of Robust Design Optimization Approaches . . . 14
1.5 Research Objectives . . . 17
1.6 Outline of Dissertation . . . 18

2 Numerical Methodology for Robust Aerodynamic Design Optimization **20**
2.1 Introduction . . . 20
2.2 Design Optimization Approaches . . . 21
2.2.1 Traditional Design Optimization Approaches . . . 21
2.2.1.1 General Theory of Traditional Design Optimization 21
2.2.1.2 Gradient-Based Method . . . 23
2.2.1.3 Genetic Algorihm . . . 26
2.2.1.4 Multi-Objective Genetic Algorithm . . . 32
2.2.2 Robust Design Optimization Approaches . . . 39
2.2.2.1 General Theory of Robust Design Optimization . . 39
2.2.2.2 Design for Six Sigma . . . 40
2.2.2.3 Design for Multi-Objective Six Sigma . . . 44
2.2.2.4 Estimation of Statistical Values . . . 48
2.3 CFD Simulation Approach . . . 52
2.3.1 Governing Equations . . . 52
2.3.1.1 Navier-Stokes Equations in Cartesian Coordinate System . . . 52
2.3.1.2 Non-Dimensionalization of Navier-Stokes Equations 54
2.3.1.3 General Transformation of Navier-Stokes Equations 55
2.3.1.4 Thin-Layer Approximation . . . 58
2.3.2 Discretization of Governing Equations . . . 59
2.3.3 Evaluation of Numerical Fluxes . . . 60
2.3.3.1 Inviscid Terms . . . 60
2.3.3.2 Viscous Terms . . . 63
2.3.4 Time Integration . . . 63
2.3.4.1 LU-ADI Factorization Algorithm . . . 64

2.3.4.2 ADI-SGS Factorization Algorithm 68
2.3.4.3 Treatment of Viscous Term 70
2.3.4.4 Space Variable Time Stepping 70
2.3.4.5 Multigrid Method 71
2.3.5 Turbulence Modeling 74
2.3.5.1 Reynolds Averaging 74
2.3.5.2 Favre Averaging 74
2.3.5.3 Baldwin-Lomax Algebraic Turbulence Model 76
2.3.6 Boundary Conditions 77
2.3.6.1 Solid Wall Boundary 77
2.3.6.2 External Boundary 79
2.3.6.3 Symmetric Boundary 79
2.3.6.4 Wake Singularity 80

3 Validation of Design for Multi-Objective Six Sigma Approach 81
3.1 Introduction 81
3.2 Case 1: Robustness in Objective Function Value 81
3.2.1 Case 1.1: Test Function Optimization Problem 81
3.2.1.1 Problem Definition 81
3.2.1.2 Optimization Methods and Conditions 82
3.2.1.3 Numerical Results 83
3.2.2 Case 1.2: Welded Beam Design Problem 85
3.2.2.1 Problem Definition 85
3.2.2.2 Optimization Methods and Conditions 86
3.2.2.3 Numerical Results 87
3.3 Case 2: Robustness in Constraint Function Violation 88
3.3.1 Case 2.1: Test Function Optimization Problem 88
3.3.1.1 Problem Definition 88
3.3.1.2 Optimization Methods and Conditions 89
3.3.1.3 Numerical Results 89
3.3.2 Case 2.2: Welded Beam Design Problem 91
3.3.2.1 Problem Definition 91
3.3.2.2 Optimization Methods and Conditions 92
3.3.2.3 Numerical Results 92
3.4 Summary 93

4 Robust Aerodynamic Design of Airfoil Configuration for Mars Exploratory Airplane 95
4.1 Introduction 95
4.2 Case 1: Robust Design Optimization of Lift to Drag Ratio against the Variation of Flight Mach Number 96
4.2.1 Design Problem Definition 96
4.2.2 Numerical Methods and Conditions 97
4.2.2.1 Optimization 97
4.2.2.2 CFD Simulation 100
4.2.3 Numerical Results 102

4.3 Case 2: Robust Design Optimization of Pitching Moment Coefficient against the Variation of Flight Mach Number . . . 112
4.3.1 Design Problem Definition . . . 112
4.3.2 Numerical Methods and Conditions . . . 113
4.3.3 Numerical Results . . . 114
4.4 Case 3: Robust Design Optimization of Lift to Drag ratio against the Variation of Angle of Attack . . . 122
4.4.1 Design Problem Definition . . . 122
4.4.2 Numerical Methods and Conditions . . . 125
4.4.3 Numerical Results . . . 126
4.5 Summary . . . 138

5 Robust Aerodynamic Design of Airfoil and Wing Planform Configurations of Mars Exploratory Airplane 140
5.1 Introduction . . . 140
5.2 Design Problem Definition . . . 141
5.3 Numerical Methods and Conditions . . . 147
5.3.1 Optimization . . . 147
5.3.2 CFD Simulation . . . 149
5.4 Numerical Results . . . 152
5.5 Summary . . . 175

6 Concluding Remarks 176

Bibliography 179

List of Figures

1.1 Mars orbiting satellites. 2
1.2 NASA's Mars rover "Spirit."[3] 2
1.3 Mars exploratory airplanes with fixed wing. 4
1.4 Entry, descent and deployment sequence of events of NASA's ARES.[12] 5
1.5 Mars exploratory helicopter "Martian Autonomous Rotary-Wing Vehicle (MARV)" designed at the University of Maryland.[16] 6
1.6 NASA's Mars exploratory airplane with flapping wing "Entomopter."[18] 6
1.7 Comparison of polar curves of traditional airfoil, flat plate and curved plate between two cases of low Reynolds number conditions.[21] 9
1.8 Mean meridional cross sections of east-westward gradient wind speeds on the Mars (positive wind speeds represent eastward).[22] 11
1.9 Forced planetary wave.[23] 12
1.10 Example of the effect on aerodynamic performance by the variations of flight conditions 12
1.11 Example of trade-off relation between optimality and robustness. . . 15

2.1 Classification of optimization problems in terms of the number of peaks of objective function. 22
2.2 Classification of optimization problems in terms of the number of objective functions. 24
2.3 Pareto-optimal solutions. 24
2.4 Flowchart of gradient-based method. 25
2.5 Flowchart of genetic algorithm. 27
2.6 Coding and decoding methods. 28
2.7 Selection methods. 29
2.8 Two-point crossover. 30
2.9 Blended crossover. 31
2.10 Bit-wise mutation. 31
2.11 Mutation methods. 32
2.12 Parero-ranking method. 34
2.13 Comparisons between traditional optimization and robust optimization. 40
2.14 Characteristics of sigma level n. 41
2.15 Flowchart of robust optimization using design for six sigma. 43
2.16 Flowchart of robust optimization using design for multi-objective six sigma. 46

2.17 Post-evaluation of sigma level n in design for multi-objective six sigma. 47
2.18 Monte Carlo simulation techniques. 50
2.19 Viscous stress tensor. 53
2.20 General transformation. 56
2.21 Example of grid sequence of multigrid method. 72
2.22 Flowchart of multigrid method with V-cycle strategy. 72

3.1 Test function distribution of Case 1.1. 82
3.2 Robust optimal solution distributions of Case 1.1. 84
3.3 Welded beam structure. 85
3.4 Robust optimal solution distributions of Case 1.2. 88
3.5 Test function contour plots of Case 2.1. 89
3.6 Robust optimal solution distributions of Case 2.1. 90
3.7 Design variable distributions of robust optimal solutions of Case 2.1. 91
3.8 Robust optimal solution distributions of Case 2.2. 94

4.1 Definition of airfoil configuration based on the B-spline curves of Case 1. 98
4.2 Grid distributions. 101
4.3 Robust optimal solution distributions of Case 1. 103
4.4 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 1. 104
4.5 Histories of L/D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1. 104
4.6 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1. 105
4.7 Histories of C_L and C_D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1. 106
4.8 Histories of breakdown of C_D against M_∞ at $\alpha = 2.0$ [deg] of Case 1. 107
4.9 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 1 (black solid lines are sonic lines). 109
4.10 Local Mach number contour plots around the airfoil of robust optimal solution with 6σ robustness quality obtained by using DFMOSS of Case 1 (black solid lines are sonic lines). 110
4.11 Chordwise C_p distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 1. 111
4.12 Robust optimal solution distributions of Case 2. 115
4.13 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 2. 116
4.14 Histories of L/D and C_{M_p} against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2. 117
4.15 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2. 117

4.16 Histories of C_L and C_D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2. 118
4.17 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 2 (black solid lines are sonic lines). 119
4.18 Local Mach number contour plots around the airfoil of robust optimal solution with 8σ robustness quality obtained by using DFMOSS of Case 2 (black solid lines are sonic lines). 120
4.19 Chordwise C_p distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 2. 121
4.20 Chordwise $C_p \times$ moment arm distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 2. 123
4.21 Definition of airfoil configuration based on the B-spline curves of Case 3. 125
4.22 Robust optimal solution distributions of Case 3. 127
4.23 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 3. 128
4.24 Histories of L/D against α at $M_\infty = 0.4735$ of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3. 128
4.25 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3. 129
4.26 Histories of C_L and C_D against α at $M_\infty = 0.4735$ of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3. 130
4.27 Histories of breakdown of C_D against α at $M_\infty = 0.4735$ of Case 3. . 131
4.28 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 3. 133
4.29 Local Mach number contour plots around the airfoil of robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3. 134
4.30 Streamline patterns near the leading edge of the airfoil of one-point optimal solution of Case 3. 135
4.31 Streamline patterns near the leading edge of the airfoil of robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3. 136
4.32 Chordwise C_p distributions over the airfoil surface at various α and $M_\infty = 0.4735$ of Case 3. 137
4.33 Comparison of chordwise C_p distributions over the airfoil surface at various α and $M_\infty = 0.4735$ between one-point optimal solution and robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3. 138

5.1 Definition of airfoil and wing planform configurations. 144
5.2 Geometric constraint on wing foldability. 146
5.3 Geometric constraint on wing storability into aeroshell. 146
5.4 Wing beam structure. 148

5.5 Grid distributions. 151
5.6 Optimal solution distributions obtained by the one-point optimization. 154
5.7 Robust solution distributions obtained by the robust optimization using DFMOSS. 155
5.8 Compared three robust optimal solutions obtained by using the DFMOSS. 156
5.9 Wing planform configurations of three robust optimal solutions obtained by using DFMOSS. 157
5.10 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS at five wing sections. 159
5.11 L histories against dispersive flight condition parameters of three robust optimal solutions obtained by using DFMOSS. 160
5.12 D histories against dispersive flight condition parameters of three robust optimal solutions obtained by using DFMOSS. 161
5.13 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different α of robust optimal solution with larger standard deviation of L (shown for only right half wings). 163
5.14 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different α of robust optimal solution with smaller standard deviation of L (shown for only right half wings). 164
5.15 Streamline patterns over the upper wing surface at three different α of robust optimal solution with larger standard deviation of L (shown for only right half wings). 165
5.16 Streamline patterns over the upper wing surface at three different α of robust optimal solution with smaller standard deviation of L (shown for only right half wings). 166
5.17 Spanwise distributions of local L per unit span length at various α (shown for only right semi-spans). 168
5.18 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different β of robust optimal solution with larger standard deviation of L (shown for full wings). . 169
5.19 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different β of robust optimal solution with smaller standard deviation of L (shown for full wings). 170
5.20 Streamline patterns over the upper wing surface at three different β of robust optimal solution with larger standard deviation of L (shown for full wings). 171
5.21 Streamline patterns over the upper wing surface at three different β of robust optimal solution with smaller standard deviation of L (shown for full wings). 172
5.22 Spanwise distributions of local L per unit span length at various β (shown for full spans). 174

List of Tables

1.1 Physical properties of the Earth's and the Martian atmospheres.[19] . 8

2.1 Safety probability at each sigma level for normal distribution. 41

3.1 Optimization conditions of Case 1.1. 83
3.2 Numerical data of robust optimal solutions of Case 1.1. 84
3.3 Optimization conditions of Case 1.2. 87
3.4 Optimization conditions of Case 2.1. 90
3.5 Optimization conditions of Case 2.2. 92

4.1 Two-dimensional cruising conditions of NASA's AME.[11] 96
4.2 Statistical characteristics of dispersive flight Mach number M_∞ for NASA's AME. 97
4.3 Design variables and corresponding ranges of Case 1. 98
4.4 Optimization conditions of Case 1. 100
4.5 Ratio of aerodynamic performance parameters at $M_\infty = 0.5735$ and $\alpha = 2.0$ [deg] to those at $M_\infty = 0.3735$ and $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 1 [%]. 106
4.6 Optimization conditions of Case 2. 114
4.7 Statistical characteristics of dispersive angle of attack α for NASA's AME [deg]. 125
4.8 Optimization conditions of Case 3. 126

5.1 Three-dimensional cruising conditions of NASA's AME.[11] 142
5.2 Reference values for conversions into the values with real units.[11,19] . 142
5.3 Statistical characteristics of three-dimensional flight conditions for NASA's AME. 143
5.4 Design variables and corresponding ranges. 145
5.5 Optimization conditions. 150

Chapter 1

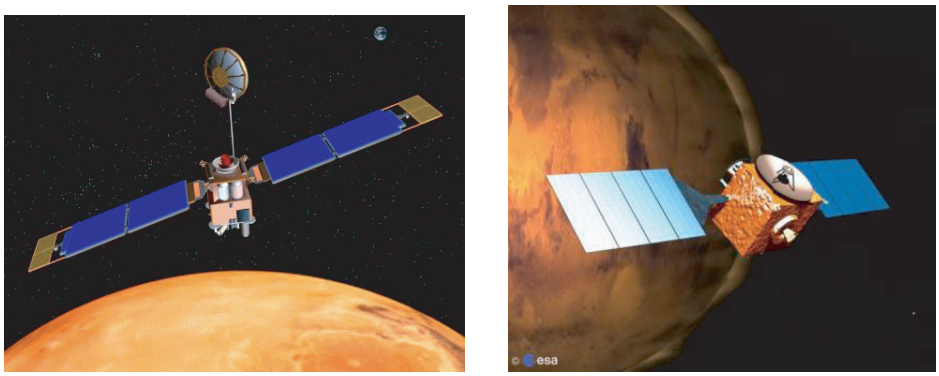
Introduction

1.1 Backgrounds

The exploration of various planets in the solar system has been an academically interesting and attracting research topic in space science field because it may provide an important clue for understanding physical and biological early histories of the solar system evolution including the Earth. Mainly two approaches have been adopted to observe the planet directly; orbiting satellite around the focused planet, and rover moving on the surface of focused planet. Up to the present, each approach have been investigated by many researchers, and actually various orbiting satellites and rovers have been delivered or will be delivered from the Earth to various planets in the solar system by several research institutes in countries all over the world.

Orbiting satellites can provide large spatial coverage over the focused planet. Up to the present, several exploration missions of planets in the solar system using orbiting satellites have been already conducted and many observed data have been provided. National Aeronautics and Space Administration (NASA) delivered the orbiting satellite "Mars Global Suyveyor,"[1] as shown in Fig. 1.1(a), to the Mars and started collecting image data of the Martian surface in 1997. European Space Agency (ESA) also started exploring the Mars by using the orbiting satellite "Mars Express,"[2] as shown in Fig. 1.1(b), in 2003. However, these orbiting satellites still have lack of resolution of observed image data. For example, NASA's Mars Global Surveyor has telephoto lens cameras with 1.4 m resolution power and wide-angle lens cameras with 250 m resolution power on board. Therefore, it is difficult for the orbiting satellites to explore the focused planet in detail.

Rovers, on the other hand, can provide detailed data of the focused planet because the rovers move while touching the surface of the planet. NASA succeeded in putting



(a) NASA's "Mars Global Surveyor."[1] (b) ESA's "Mars Express."[2]

Figure 1.1 Mars orbiting satellites.

the rovers "Spirit" and "Opportunity,"[3] as shown in Fig. 1.2, onto the Martian surface and getting high resolution data in 2003. However, rover's spatial exploratory coverage is limited to a very small area. For example, NASA's Spirit can only move at normal speed of 1 cm/s and maximum speed of 5 cm/s, and actually it was reported that NASA's Spirit had realized only 108 m travel per 45 Martian days and 3,605 m per 225 Martian days (1 Martian day corresponds to 24.6 Earth's hours).

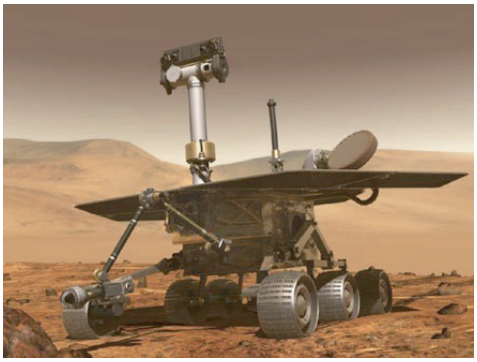


Figure 1.2 NASA's Mars rover "Spirit."[3]

As for in Japan, Japan Aerospace Exploration Agency (JAXA) has been actively exploring the solar system by its unique approach. Currently, the scientific satellite "HAYABUSA"[4] is approaching to the asteroid which was named as "ITOKAWA" to bring back samples of the asteroid's surface to the Earth. Also, the Venus Climate Orbiter project "PLANET-C,"[5] the third Japanese solar physics satellite "SOLAR-B,"[6] the Japanese lunar satellite "SELENE"[7] and penetrator "LUNAR-A"[8] projects, and the joint Mercury exploration project between Japan and Europe

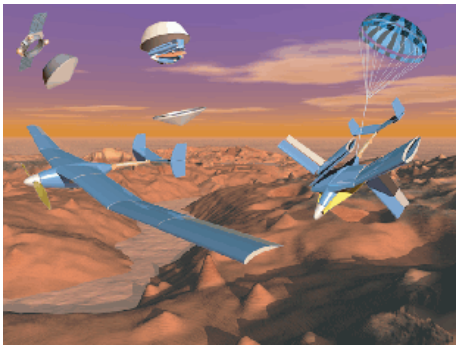
"BepiColombo"[9] are ongoing. As in JAXA long-term visions[10] announced in April 2005, JAXA is aiming at further exploration of the solar system in the next 20 years. Especially in the long-term visions of JAXA, it is notable that Mars exploration using an aircraft is proposed instead of conventional approaches using orbiting satellites or rovers.

The use of aircrafts is expected to be a new and attractive approach to observe the Mars directly and get academically interesting information about the Martian surface and atmosphere because aircrafts can get higher resolution data than orbiting satellites and can provide larger spatial coverage over large region than rovers. However, it is also a challenging approach in engineering viewpoint. There exist several options for Mars exploration using an aircraft; airplane with fixed wing, helicopter, airplane with flapping wing, etc., and each option has already been investigated by many researchers, as described in Sec. 1.2.

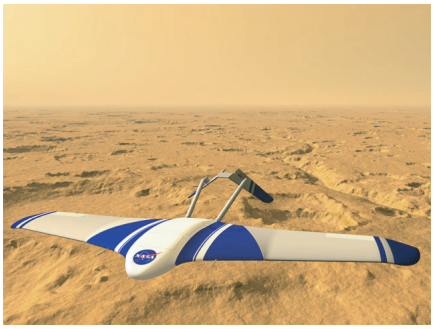
1.2 Past Studies of Mars Exploratory Airplanes

As for airplanes with fixed wing, NASA proposed "Airplane for Mars Exploration (AME),"[11] as shown in Fig. 1.3(a), and "Aerial Regional-scale Environmental Survey of Mars (ARES),"[12–14] as shown in Fig. 1.3(b). AME aims at exploring geological features over a large geographic area including mountains and craters. The span length and area of main wing of AME are 12.44 m and 12.24 m^2, respectively, and the total mass of AME is 203.8 kg. The AME aims to fly at cruising Mach number of 0.4735 and with a range of 3,500 km, using propeller propulsion whose power is supplied by a fuel cell. On the other hand, ARES is one of the proposed Mars Scout missions using an airplane to provide high-value science measurements in the areas of atmospheric chemistry, surface geology and mineralogy, and crustal magnetism. The overall length and wing span length of ARES are 4.4 m and 6.25 m, respectively, and the total mass of ARES is 175 kg. The ARES can fly at cruising Mach number of 0.65 and with a range of 500 km, using rocket propulsion. Both AME and ARES are planned to be delivered to the Mars by using existing launchers such as Viking rockets of the United States, compactly folded into a protective aeroshell. Then, these airplanes will be emerged from its aeroshell and start to fly over the Mars through the entry, descent and deployment sequence of events, as shown in Fig. 1.4. In addition, there exists another interesting case of conceptual design of Mars airplane with fixed wing that master students of the University of Tokyo in

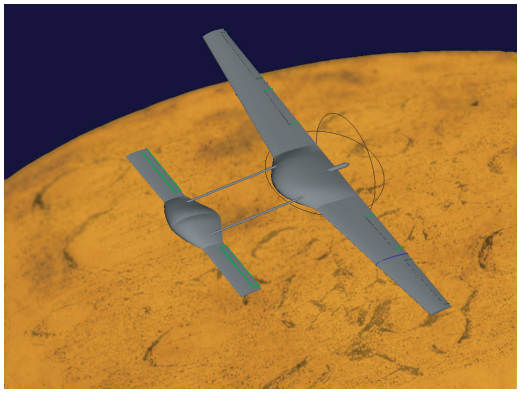
Japan designed a canard-type Mars airplane using a propeller propulsion, as shown in Fig. 1.3(c), in 2005.[15]



(a) NASA's "Airplane for Mars Exploration (AME)."[11]



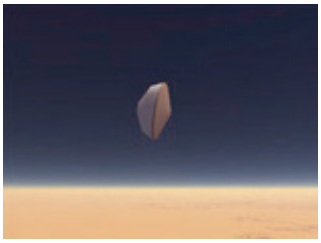
(b) NASA's "Aerial Regional-scale Environmental Survey of Mars (ARES)."[12–14]



(c) Mars airplane designed at the University of Tokyo.[15]

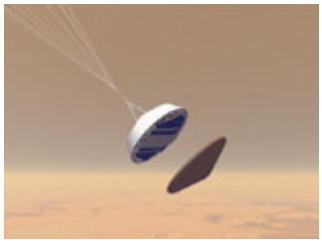
Figure 1.3 Mars exploratory airplanes with fixed wing.

As for the Mars exploratory helicopter, "Martian Autonomous Rotary-Wing Vehicle (MARV),"[16] as shown in Fig. 1.5, was designed at the University of Maryland in the United States in 2000. MARV was designed in response to the request for proposals (RFP) from the American Helicopter Society as part of their 2000 student design competition. The final design of the MARV satisfies all these requirements; the MARV can carry 10.8 kg of payload over a range of 25 km in 39 minutes, make a soft landing, shutdown, re-start, and take off again and fly a short distance. In addition, Tsuzuki[17] conducted the feasibility study and conceptual design of a miniature rotary-wing vehicle for Mars exploration with a rotor radius of 0.10 $\sim$ 0.15 m class



(a) Atmospheric entry.

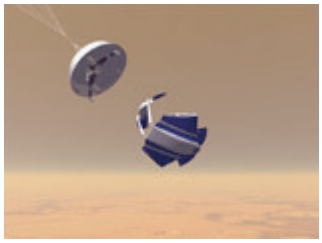
(b) Parachute deployment.



(c) Heatshield release.

(d) Aircraft ejection.

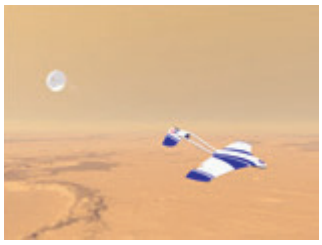
(e) Aircraft release.



(f) Tail deployment start.

(g) Tail deployment complete.

(h) Wings deployment start.



(i) Aircraft pull-up start.

(j) Aircraft pull-up complete.

(k) Mars exploration start.

Figure 1.4 Entry, descent and deployment sequence of events of NASA's ARES.[12]

and a flight endurance of 1/12 hour class in 2005.

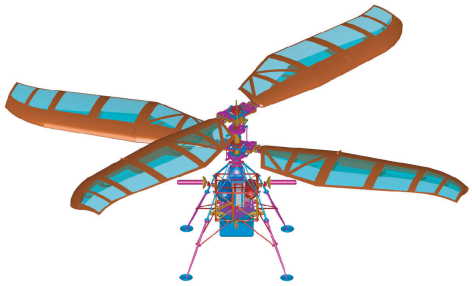


Figure 1.5 Mars exploratory helicopter "Martian Autonomous Rotary-Wing Vehicle (MARV)" designed at the University of Maryland.[16]

Finally, as for the Mars exploratory airplane with flapping wing, NASA proposed "Planetary Exploration Using Biomimetics" which is called "Entomopter,"[18] as shown in Fig. 1.6. In this mission, it is assumed that two Entomopters fly up from a rover, and bring back samples of the Martian surface to the rover. Entomopter can fly by flapping two pairs of wings with a Reciprocating Chemical Muscle (RCM) with a range of 40 km and an endurance of 48 minutes. On the rover, the samples are analyzed and also the fuel of RCM is supplied for the next flight.

Figure 1.6 NASA's Mars exploratory airplane with flapping wing "Entomopter."[18]

Each option for Mars exploration described above has different advantages and disadvantages. The airplane with fixed wing has larger spatial coverage, easier controllability and larger feasibility of automatic flight, but also has difficulties in storage into an aeroshell and structural problems. The helicopter has capability of take-off, landing and fixed-point observation, but has difficulties in automatic flight and stor-

age into an aeroshell and structural problems. Finally, the airplane with flapping wing has advantage in storage into an aeroshell, but its basic technologies such as hovering have not been established yet. In the present study, the airplane with fixed wing is considered as approach for Mars exploration because airplanes with fixed wing are established technologies. Also from a scientific viewpoint, only the airplane with fixed wing can globally and accurately explore the structure of remnant magnetic fields and lower atmosphere, which are important clues for understanding the early history of Mars evolution, because the airplane with fixed wing can fly near the Martian surface with large spatial coverage and then reduce undesirable absorption of signal rays by the atmosphere.[14]

However, it is undesirable to design the Mars airplane just like the Earth airplanes. One of the reasons is that the flight conditions on the Mars are very different from those on the Earth. Table 1.1 shows the physical properties of the Earth's and Martian atmospheres. The density on the Mars is very small and about 1/100 of that on the Earth. In addition, the size of Mars airplane is restricted to be small because the airplane must be stored into an aeroshell delivered by a launcher. Consequently, the Reynolds number based on the chord length of Mars airplane becomes about 10^5, while that of typical commercial airplanes on the Earth is about 10^7. In addition, an airplane must fly with higher velocity on the Mars than that on the Earth because it is required to get lift, i.e., dynamic pressure enough to support the airplane weight in the thin Martian atmosphere. Furthermore, the speed of sound on the Mars is about 2/3 of that on the Earth. It results in the need for an airplane to fly at higher Mach number on the Mars than that on the Earth. For example, the aircraft with the total mass of 1,500 kg, the wing area of 20 m^2 and lift coefficient of 0.785 must fly at cruising Mach number of 0.12 on the Earth to get lift enough to support the aircraft weight. On the other hand, the same aircraft must fly at cruising Mach number of 0.95 on the Mars when it is assumed that lift coefficient is constant on both the Earth and the Mars. Therefore, it is required to design an airplane in the flight conditions with low Reynolds number and high subsonic Mach number for the Mars airplane mission.

However, the Earth airplanes corresponding to similar flight conditions to the Mars airplane are limited to high-altitude airplanes, and currently the design concept of such airplanes has not been established sufficiently yet. Actually, NASA's AME[11] and ARES[12–14] were designed only by utilizing and modifying existing design approaches for conventional Earth airplanes and it may not have found a design

Table 1.1 Physical properties of the Earth's and the Martian atmospheres.[19]

	Earth		Mars (on equator)	
Gas composition (volume ratio [%])	N_2	78.1	CO_2	95.3
	O_2	20.9	N_2	2.7
	Ar	0.9	Ar	1.6
	H_2O	$0 \sim 2$		
Temperature [K]	298		270	
Pressure [Pa]	1.014×10^5		6.0×10^2	
Density [kg/m^3]	1.17		1.18×10^{-2}	
Speed of sound [m/s]	345		220	
Viscosity coefficient [Pa·s]	1.86×10^{-5}		1.36×10^{-5}	
Acceleration of gravity [m/s^2]	9.8		3.78	
Ratio of specific heats	1.4		1.34	

with the best performance on the Mars exactly in these studies. Actually, it is known that the aerodynamic performance of an airfoil changes drastically at low Reynolds number condition. Figure 1.7 compares the polar curves of three configurations (traditional airfoil, flat plate and curved plate) between two cases of low Reynolds number conditions (40,000 and 120,000). In each case, the freestream velocity is about 0.67 m/s. In the case of Reynolds number as 120,000, the traditional airfoil has the best aerodynamic performance. In the case of Reynolds number as 40,000, on the other hand, curved plate and flat plate have the better aerodynamic performance than the traditional airfoil. In addition, Liebeck[20] reported that an thin airfoil like a curved plate has good aerodynamic performance at low Reynolds number condition from his study using an inverse design approach. Therefore, it is required to search the best design more globally in such low Reynolds number condition corresponding to the flight condition of Mars airplane. Furthermore, not only aerodynamic performance but also other performances such as structure, propulsion, control, etc. must be considered simultaneously in the design of Mars airplane. Therefore, so-called multidisciplinary design of Mars airplane will be required in the future. However, it is not effective and not realistic to find the best design of Mars airplane considering various performances by changing many design parameters many times through trial and error based on only human knowledge, intuition and experience. The use of a design optimization approach, as described in Sec. 1.3, is one of good solutions to realize more global and effective search for better design of Mars airplane and

establish a new design concept for Mars airplanes.

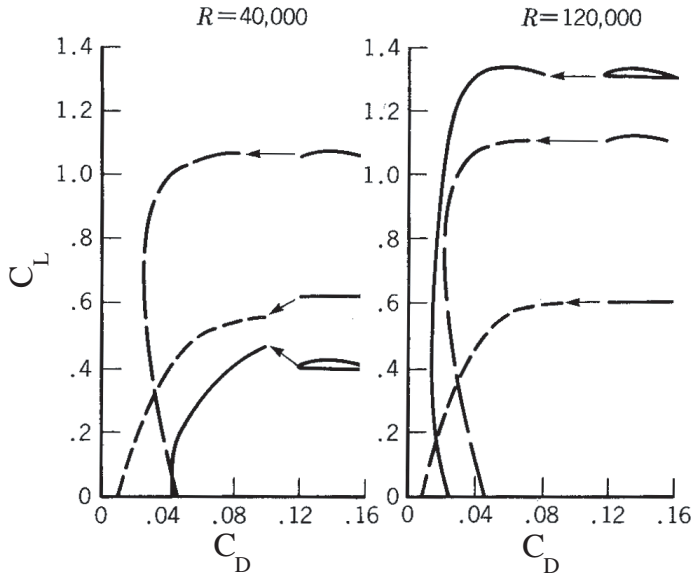


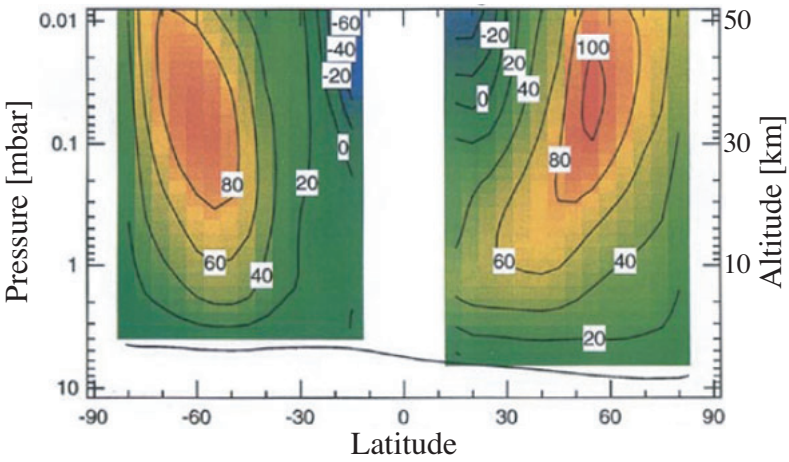
Figure 1.7 Comparison of polar curves of traditional airfoil, flat plate and curved plate between two cases of low Reynolds number conditions.[21]

Another reason why it is undesirable to design the Mars airplane just like the Earth ones is that the Mars airplane mission is very risky because this mission has no precedent and only one available chance to try flying over the Mars, and the Mars airplane must be controlled without a pilot. Therefore, it is desirable to expect and avoid every critical situation possible to occur during the Mars airplane mission in the initial design and development phases for success in completing this mission safely.

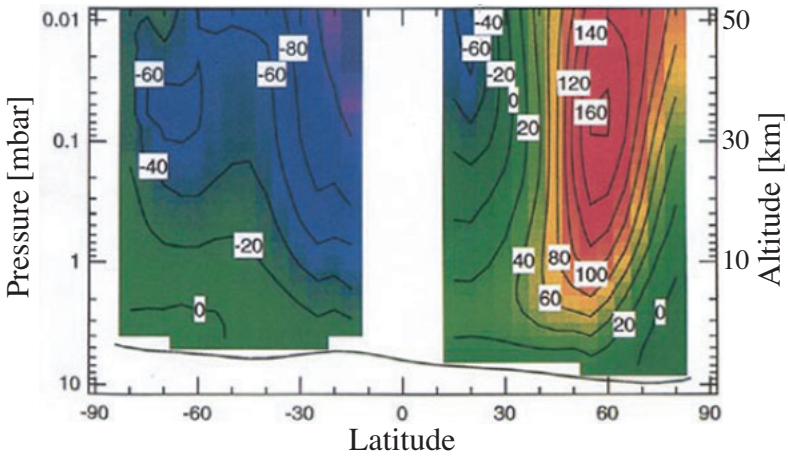
One of the most critical situations which must be avoided is the drastic deterioration in performance due to large wind variations on the Mars. It is well known that there exist large dairy and seasonal climate changes on the Mars because the Mars does not have the sea absorbing solar radiation heat, the Mars has large orbital eccentricity, and the Martian surface is undulating hardly with many mountains and craters, differently from the Earth. Figure 1.8 shows the mean meridional cross sections of east-westward gradient wind speeds, so-called the westerly speeds, in the northern hemisphere fall equinox and winter solstice on the Mars. This figure shows that the maximum westerly speed is more than 100 m/s at the altitude of about 30 km of high latitude regions, and also there exist relatively high wind speeds (about 20 m/s) near the Martian surface (at the altitude of several kilometers) where it is assumed for the Mars airplane to fly for exploring the structure of remnant magnetic

fields and lower atmosphere. Actually, it will be required for the Mars airplane to fly in both east-westward and north-southward directions for various Mars exploratory missions under the situation where the westerlies blow hard. In addition, comparing Figs. 1.8(a) and 1.8(b), it is clear that the westerly speeds change drastically in all seasons. Actually, it is not guaranteed that the Mars airplane can reach the Mars on schedule by a launcher in all missions, and some Mars airplanes may have to fly in unexpected season. Furthermore, it is known that the westerlies can produce not only the change of east-westward wind speeds but also that of vertical wind speeds caused by so-called the forced planetary wave, as shown in Fig. 1.9, when these blow over an hardly undulating Martian surface with mountains and craters. Some researchers in planetary meteorology say that the vertical wind speeds produced by the forced planetary wave may extend to about 20 m/s maximum on the Mars. Therefore, it is required to consider the effect of various possible wind variations which lead to various variations of flight conditions, such as Mach number, the angle of attack and the angle of side slip, for more realistic and reliable design of Mars airplane.

Figure 1.10 illustrates an example of the effect on aerodynamic performance by the variations of flight conditions. The solution A has the largest lift to drag ratio at design point. However, the lift to drag ratio of solution A decreases drastically with the increment in flight Mach number, and its decrement extends to the infeasible range. That is, it can be said that the solution A has the best but not robust aerodynamic performance against wind variations. On the other hand, the solutions B and C have slightly worse at the design point but much more robust aerodynamic performance against the wind variations than the solution A. The previous studies on NASA's AME[11] and ARES,[12–14] aimed at only improving the performance at design point (cruising flight conditions) and did not consider the effect on the aerodynamic performance by possible wind variations sufficiently. These facts indicate that NASA's AME and ARES may not have good performance at off-design points, like the solution A in Fig. 1.10, and may not be able to fly over the Mars well enough to complete these exploratory missions safely when wind variations are large on the Mars and the flight conditions of Mars airplane get off the assumed design point. As described above, it is desirable to use a design optimization approach for more global and effective search of the best design of Mars airplane. In addition, it is desirable to focus on not only the performance at design point but also the robustness of performance against wind variations for more realistic and reliable design of Mars



(a) Northern hemisphere fall equinox.



(b) Northern hemisphere winter solstice.

Figure 1.8 Mean meridional cross sections of east-westward gradient wind speeds on the Mars (positive wind speeds represent eastward).[22]

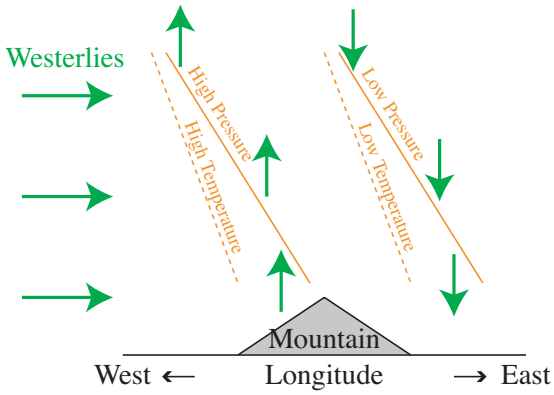


Figure 1.9 Forced planetary wave.[23]

airplane, like the solutions B and C in Fig. 1.10, too. One of solutions to realize such design is the use of a robust design optimization approach, as described in Sec. 1.4.

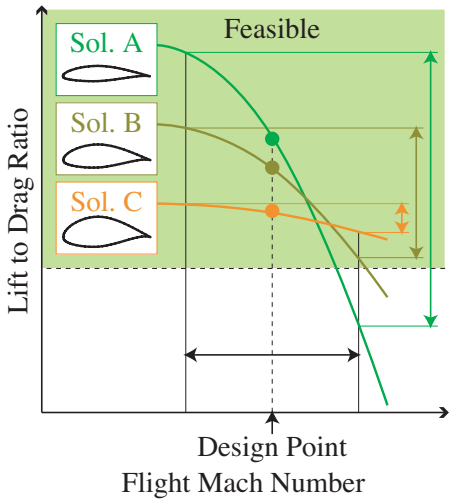


Figure 1.10 Example of the effect on aerodynamic performance by the variations of flight conditions

1.3 Past Studies of Design Optimization Approaches

A design optimization is a computer-aided design tool to search the design with the best performance globally and automatically. Up to the present, various design optimization approaches have been developed by many researchers, and applied to various engineering design problems about structure, dynamics, aerodynamics, etc. by many designers. As for aerodynamic design optimizations, it was the most conventional way to utilize theoretical formulas or experimental data for the evaluation

of aerodynamic performance carried out many times in an optimization process. Recently, the use of computational fluid dynamics (CFD) simulation[24,25] becomes popular for the evaluation of aerodynamic performance. Essentially, CFD simulations, especially viscous Navier-Stokes simulations, need much larger computation time and resources compared to the simulations of other engineering fields. However, now the aerodynamic design optimization using the CFD simulations has already reached at a practicable level according to recent fast improvement in computer power.

One of famous examples of aerodynamic design optimizations is the study conducted by Reuther *et al.*.[26,27] In their study, applications of an adjoint method[28] to the aerodynamic designs of transonic business jet configuration and supersonic transport configuration using an inviscid Euler flow solver were reported. The adjoint method is a kind of gradient-based methods (GM)[29,30] which are the most conventional design optimization approaches, and based on control theory. In this method, the gradient information at a single design point with respect to an arbitrary number of design variables: N can be obtained with the equivalent two flow calculations (one flow or state solution and one adjoint or costate solution), instead of the $N + 1$ flow calculations required in conventional GMs. Therefore, the use of adjoint method can reduce computation time drastically compared to the other kinds of GMs. Consequently, in their aerodynamic design optimization using the adjoint method where the total drag at cruising Mach number of 2.2 with lift coefficient of 0.105 must be minimized, the total pressure drag could be reduced by 8 % successfully. However, GMs such as the adjoint method can not guarantee that an obtained design is global optimal. Therefore it is required to carry out the optimizations using GMs many times with different initial designs to obtain a global optimal design.

Instead of the adjoint method, a genetic algorithm (GA)[31] has been paid attention to in the field of aerodynamic design optimization. A GA is a design optimization approach which imitates biological evolution process, and has superior capability of searching a global optimal design independently of initial designs compared to GMs. Essentially, however, GAs have poor convergency to a global optimal design. Oyama[32] engaged in the development of new efficient GA, called a real-coded adaptive range genetic algorithm (ARGA), and its application to the aerodynamic designs of airfoil configuration using viscous Navier-Stokes flow solver. In his study, it was shown that the real-coded ARGA can realize faster convergence to a global optimal design than conventional GAs in a test function problem, and transonic and supersonic aerodynamic design optimization problems of airfoil configuration. As for

the transonic aerodynamic design optimization using the ARGA where lift to drag ratio must be maximized and the constraint on bending moment must be satisfied at cruising Mach number of 0.8 and the angle of attack of 0 deg, the optimized design had better lift to drag ratio of 18.91, and also could satisfy the constraint on bending moment.

Usually, however, there exist not only a single objective, as described above, but also multiple objectives in real-world design problems. In general multi-objective design optimization problems, an improvement of one objective requires a compromise of the other objective, and such trade-off relation can be represented by the curve which consists of multiple Pareto-optimal solutions. GMs can not treat multi-objective optimization problems effectively. On the other hand, GAs can be applied to the multi-objective optimization problems easily. A multi-objective genetic algorithm (MOGA)[33] was developed for efficient multi-objective design optimizations and acquisition of realistic design information about trade-off relation among competing objectives. One of good examples of aerodynamic design optimizations using a MOGA is conducted by Sasaki.[34] In his study, a real-coded adaptive range multi-objective genetic algorithm (ARMOGA) was developed and validated in test function problems, and it was applied to aerodynamic designs of low pressure compression system, supersonic transport wing-fuselage configuration, and supersonic transport canard-wing-fuselage configuration using viscous Navier-Stokes flow solver. In the aerodynamic design optimization of canard-wing-fuselage configurations of supersonic transport, drag coefficient had to be minimized and equivalent area distribution had to matched Darsen's theoretical distribution which can achieve low sonic boom at cruising Mach number of 1.6 with fixed lift coefficient of 0.125. Consequently, this study could reveal trade-off relation between minimizations of drag coefficient and sonic boom, and then obtain design information about configuration for reduction in sonic boom and that for reduction in drag coefficient successfully.

1.4 Past Studies of Robust Design Optimization Approaches

Instead of a traditional design optimization considering only optimality as described in Sec. 1.3, a new design optimization called a robust design optimization has been proposed in recent years. In real-world engineering designs, inevitably there exist errors and uncertainties in design process, manufacturing process, and operating

condition. Therefore, a design optimized by a traditional design optimization approach may not be able to achieve its expected performance due to such errors and uncertainties, like the solution A in Fig. 1.10. A robust design optimization considers both optimality and robustness, and can find the design with moderately good performance at design point and also robust performance against the errors and uncertainties, like the solutions B and C in Fig. 1.10.

Optimality and robustness are usually competing in real-world design problems. Therefore, there exist multiple compromised solutions between the optimality and the robustness, as shown in Fig. 1.11, in robust design optimization problems. Objectives of robust design optimizations are to find these compromised solutions to reveal trade-off information, and to give chance to the upper-level decision maker to select one solution from these compromised solutions with other considerations.

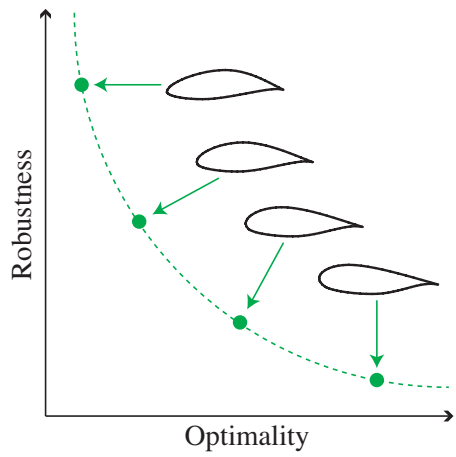


Figure 1.11 Example of trade-off relation between optimality and robustness.

Up to the present, some robust optimization approaches have been proposed and investigated to obtain robust optimal solutions. Youn and Choi[35] basically categorized robust design optimization approaches (called reliability-based design optimization (RBDO) methods in their article) into three; the approximate moment approach (AMA), the reliability index approach (RIA), and the performance measure approach (PMA). They recommended PMA among these approaches in terms of accuracy and stability. However, actually PMA is still not popular in real-world engineering design field because it is not easy for users to understand the essence of the formulation of PMA.

On the other hand, the design for six sigma (DFSS),[36] which belongs to an

AMA category, is one of popular robust design optimization approaches in real-world engineering design field. The formulation of DFSS is very simpler than that of other approaches, and can be understood easily and intuitively by users. In fact, DFSS has been successfully applied to various robust optimization problems.[37,38] However, DFSS has some drawbacks. First, DFSS finds only one robust optimal solution by one case. Second, DFSS does not guarantee that the obtained robust optimal solution satisfies the user-specified sigma level which is one of the very popular and useful measures of robustness to the designers in reliability engineering field. Therefore, users must repeat many cases by using DFSS with different input parameter set until a satisfactory robust optimal solutions, and eventually many robust optimal solutions are obtained enough to reveal trade-off information between optimality and robustness, as shown in Fig. 1.11. Deb and Gupta[39] also proposed other robust design optimization approach. In their study, a traditional design optimization problem was converted to a robust design optimization problem where the mean value of objective function must be minimized and also the perturbation of the value of objective function must be smaller than user-specified limit against the variation of design variables. This approach was almost similar to DFSS because of the need of specification of the advance constraint limit, therefore it can be said that this approach is also not effective and not useful to obtain trade-off information between optimality and robustness.

Rai[40] reported a new idea of robust design optimization approach. He treated a robust design optimization problem of turbine rotor airfoil shape as a multi-objective design optimization problem with two separate measures of optimality and robustness; to maximize the wedge angle at the trailing edge, and to minimize the variance in pressure against small airfoil shape change. As a result, many robust optimal solutions enough to reveal trade-off relation between these measures had been obtained. In his study, however, he discussed only the extreme solutions and detailed discussion about the obtained trade-off relation, which is very useful design information from the designer's viewpoint, was not conducted sufficiently. In his approach, in addition, only the variation of dispersive performance against the variation of design variables was used as the measure of robustness. However, there exist other kinds of measures of robustness or reliability which are very important and often used in real-world designs, *e.g.*, the safety probability where the dispersive performance is included in the acceptable range. In his study, further discussion about the measures of robustness except for the variation of dispersive performance was not conducted

sufficiently.

As the examples of robust aerodynamic design optimizations against the variation of configuration, flight condition, etc., some studies have been reported by some researchers.[40–44] In any of those studies, however, each proposed approach was applied to only idealized design problems, and investigated in terms of not a real-world design tool but a numerical algorithm. In addition, the detailed discussions about efficiency, usefulness and versatility of the proposed approaches were not conducted sufficiently. Unfortunately, most of the proposed robust design optimization approaches are not applicable easily and flexibly to various real-world engineering designs. Anyway, undoubtedly it can be said that a robust design optimization is now in a developing stage and has not reached a practicable level for the designers in various real-world engineering fields yet.

1.5 Research Objectives

Based on the backgrounds and past studies on Mars exploratory airplane, design optimization and robust design optimization, as described above, now I formulate my research objectives of this dissertation as follows:

- **To develop a new efficient and useful robust design optimization approach.**

  Firstly, I develop a new robust optimization approach "design for multi-objective six sigma (DFMOSS)" to solve the drawbacks of a traditional robust optimization approach. DFMOSS does not require to specify input parameters carefully in advance, and can obtain trade-off information between optimality and robustness efficiently. Then, the new robust optimization approach is applied to some robust design optimization problems to investigate the efficiency and the usefulness of the proposed approach.

- **To realize aerodynamic design optimizations of Mars exploratory airplane wing considering the effects of wind variations.**

  Secondly, I perform robust aerodynamic design optimizations of Mars exploratory airplane wing considering the effects of wind variations by using the DFMOSS. I start from robust design problems of airfoil configuration where the aerodynamic performance against the variations of flight Mach number or angle of attack must be optimized, and then extend to more realistic robust

design problem of airfoil and wing planform configurations where the aerodynamic performance against the variations of flight Mach number, angle of attack, and angle of side slip must be optimized.

- **To obtain realistic design information about the trade-off relation between the optimality and the robustness of aerodynamic performance of Mars exploratory airplane.**

  Finally, I try to obtain detailed design information about the trade-off relation between the optimality and the robustness of aerodynamic performance of Mars exploratory airplane from the obtained robust optimal solutions, and reveal the physical principles under this trade-off relation. Final goal of the present study is to provide one of useful guides to more reliable design of Mars exploratory airplanes with fixed wing and more reliable Mars exploratory missions using the airplanes.

1.6 Outline of Dissertation

The outline of this dissertation is as follows. In Chap. 2, the numerical methodologies for the present robust aerodynamic design optimization are described. Firstly, design optimization approaches are described in Sec. 2.2. Traditional design optimization approaches considering only optimality such as a gradient-based method (GM), a genetic algorithm (GA) and a multi-objective genetic algorithm (MOGA) are described in Subsec. 2.2.1. Then, robust design optimization approaches considering not only optimality but also robustness are described in Subsec. 2.2.2. In the present study, a new robust design optimization approach "design for multi-objective six sigma (DFMOSS)" is developed to solve the drawbacks of a traditional robust design optimization approach "design for six sigma (DFSS)". Secondly, computational fluid dynamics (CFD) simulation approach used for the evaluation of aerodynamic performance in the design optimization process is described in Sec. 2.3.

In Chap. 3, the newly developed robust optimization approach DFMOSS is applied to four simple examples of robust design optimization problems. In each example, the numerical results obtained by using the DFMOSS are compared to those obtained by using the DFSS to investigate the efficiency and the usefulness of the DFMOSS.

In Chap. 4, robust aerodynamic design optimizations of airfoil configuration for Mars exploratory airplane considering the effects of wind variations are performed

by using the DFMOSS and the CFD simulation approach. In the present study, three robust aerodynamic design optimizations of airfoil configuration are carried out. In each case, the numerical results obtained by the robust optimization using DFMOSS are compared to those obtained by a traditional one-point optimization or those obtained by the robust optimization using the DFSS to investigate efficiency and usefulness of the DFMOSS, and the differences in optimized designs. Then, the obtained trade-off relations between the optimality and the robustness of aerodynamic performance of airfoil configuration for Mars exploratory airplane are discussed.

In Chap. 5, the robust aerodynamic design optimizations of airfoil and wing planform configurations for Mars exploratory airplane are performed to obtain more realistic and more useful trade-off information between the optimality and the robustness of aerodynamic performance from the designer's viewpoint. The numerical results obtained by the robust optimization using DFMOSS are compared to those obtained by a traditional one-point optimization in order to investigate efficiency and usefulness of the DFMOSS and the differences in optimized designs. Then, the obtained trade-off relations between the optimality and the robustness of aerodynamic performance of airfoil and wing planform configurations of Mars exploratory airplane are discussed. Furthermore, one of useful guides to more reliable design of Mars exploratory airplanes with fixed wing and more reliable Mars exploratory missions using the airplanes is provided.

And finally in Chap. 6, the numerical results of this dissertation are summarized with conclusions.

Chapter 2

Numerical Methodology for Robust Aerodynamic Design Optimization

2.1 Introduction

In this chapter, numerical methodology for the robust aerodynamic design optimization of Mars exploratory airplane wing is described. The present methodology consists of mainly two components; optimizer and evaluator. In the optimizer, firstly, a design problem where the values of the values of certain performance parameters must be improved is defined, and initial values of input design parameters of each design candidate are generated. For each design candidate, corresponding output performance parameters are evaluated by using the evaluator. These input design parameters of each design candidate are modified so that the values of these performance parameters will be improved more by using a certain design optimization approach. Iterating these processes, finally, the design candidate which has the best performance parameters is obtained. Traditional design optimization approaches are describes in Subsec. 2.2.1, and robust design optimization approaches are described in Subsec. 2.2.2. In the evaluator, input design parameters are received from the optimizer, and corresponding output performance parameters are evaluated and sent back to the optimizer. In the present aerodynamic design optimization, computational fluid dynamics (CFD) simulations[24,25] are used for the evaluation of aerodynamic performance of Mars exploratory airplane. The present CFD simulation approach is described in Sec. 2.3.

2.2 Design Optimization Approaches

2.2.1 Traditional Design Optimization Approaches

2.2.1.1 General Theory of Traditional Design Optimization

Here, for example, consider the design problem where we aim at finding wing configuration parameters (thickness, camber, aspect ratio, sweepback angle, etc.) which has minimum drag in a certain flight condition. Here in this problem, consider in terms of structural strength that the bending moment of wing must not exceed an acceptable upper limit to avoid destruction, too. Firstly, note that the drag changes depending on the wing configuration parameters, i.e., the drag can be represented as the function of wing configuration parameters. Therefore, the drag can be minimized by searching the direction to reduce the drag and changing the wing configuration parameters in this direction. Secondly, note that the bending moment can be also represented as the function of wing configuration parameters. Therefore, the wing configuration parameters must be changed within the range that the bending moment does not exceed an acceptable upper limit. In the design optimization field, the wing configuration parameters of this problem which must be changed is called the design variables, the drag which must be minimized is called the objective function, and the bending moment which must not exceed the upper limit is called the constraint. That is, it can be said that the design optimization problem aim at finding the values of design variables which have minimum value of objective function defined as the function of design variables, and also satisfies the constraint defined as the function of design variables.

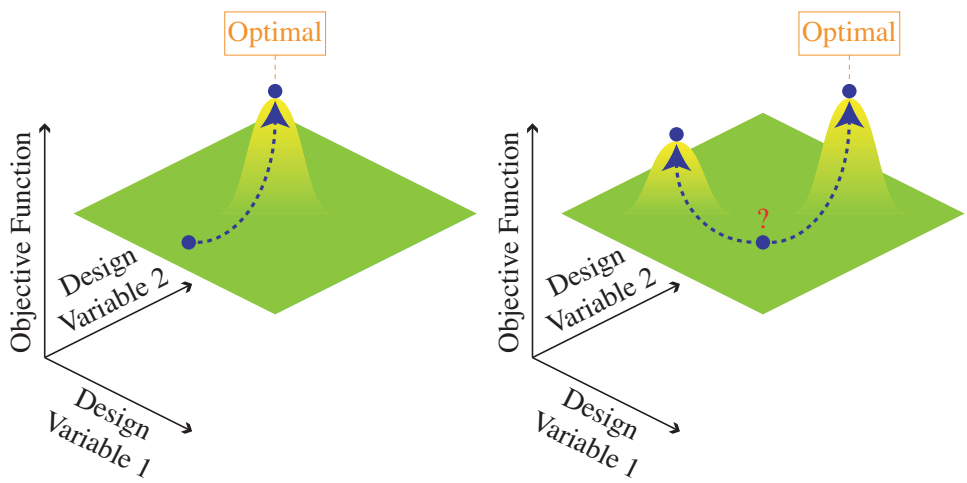
General formulation of traditional optimization problem is written as follows:

$$\begin{aligned} \text{Minimize: } & \boldsymbol{f}(\boldsymbol{x}) \\ \text{Subject to: } & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0} \end{aligned} \tag{2.1}$$

This formulation represents that we aim at finding a solution $\boldsymbol{x}$ which minimizes $\boldsymbol{f}(\boldsymbol{x})$ and also satisfies $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$. $\boldsymbol{x}$ is the vector of n design variables: $\boldsymbol{x} = [x_1, x_2, \cdots, x_n]^T$. $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$, defined as the functions of $\boldsymbol{x}$, are the vector of M objective functions: $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x})]^T$, and that of L inequality constraint functions: $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), \cdots, g_L(\boldsymbol{x})]^T$, respectively.

The optimization problems can be classified in terms of the number of peaks of objective function, as shown in Fig. 2.1. In the case of the single-peak problem where the value of objective function must be maximized, as shown in Fig. 2.1(a),

an optimal solution necessarily can be found by simply searching the direction to improve the value of objective function and changing the design variables in this direction, independently of initial value of solution. In the case of the multi-peak problem, which is more realistic case, as shown in Fig. 2.1(b), on the other hand, it may be possible to mistake right direction which should be searched and consequently find not a right solution which is global optimal but a wrong solution which is local optimal, depending on the initial value of solution. Therefore, it is most important to avoid local optimal solutions and find a global optimal solution for accurate design optimizations.



(a) Single-peak objective function. (b) Multi-peak objective function.

Figure 2.1 Classification of optimization problems in terms of the number of peaks of objective function.

In addition, the optimization problems can be classified in terms of the number of objective functions, as shown in Fig. 2.2, too. In the case of single-objective optimization problem where drag must be minimized, as shown in Fig. 2.2(a), it is clear that there exists only one optimal solution B. Next, consider the case of multi-objective optimization problem, which is more realistic case, where not only drag must be minimized but also lift must be maximized as shown in Fig. 2.2(b). Comparing four solutions A, B, C and D, the solution A has smallest drag but smallest lift. The solution B has larger lift but larger drag than the solution A, and the solution C has even larger lift but even larger drag than the solution B. Therefore, these three solutions A, B and C can be considered as good solutions

because these solutions are better in terms of either lift or drag and not worse in terms of both lift and drag than one another, i.e., we can not determine which is the best in terms of both lift and drag among the solutions A, B and C. The solution D, on the other hand, has not only larger drag but also smaller lift than the solution B. Therefore, the solution D can be considered as a bad solution because this solution is worse in terms of both lift and drag than the solution B. In the optimization field, the solutions A, B and C in Fig. 2.2(b) are said to be not dominated by any other solution, and such non-dominated solutions are called Pareto-optimal solutions. The terms "dominance" and "Pareto-optimality" can be defined mathematically for multi-objective optimization problems as follows:[33]

Definition 1: Dominance In the multi-objective optimization problem where multiple objective functions $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x})]^T$ must be minimized, a solution $\boldsymbol{x}_i$ is said to dominate the other solution $\boldsymbol{x}_j$ if all the following inequalities are satisfied:

$$\begin{aligned} f_k(\boldsymbol{x}_i) &\leq f_k(\boldsymbol{x}_j), \quad {}^{\forall}k = 1, 2, \cdots, M \\ f_k(\boldsymbol{x}_i) &< f_k(\boldsymbol{x}_j), \quad {}^{\exists}k = 1, 2, \cdots, M \end{aligned} \tag{2.2}$$

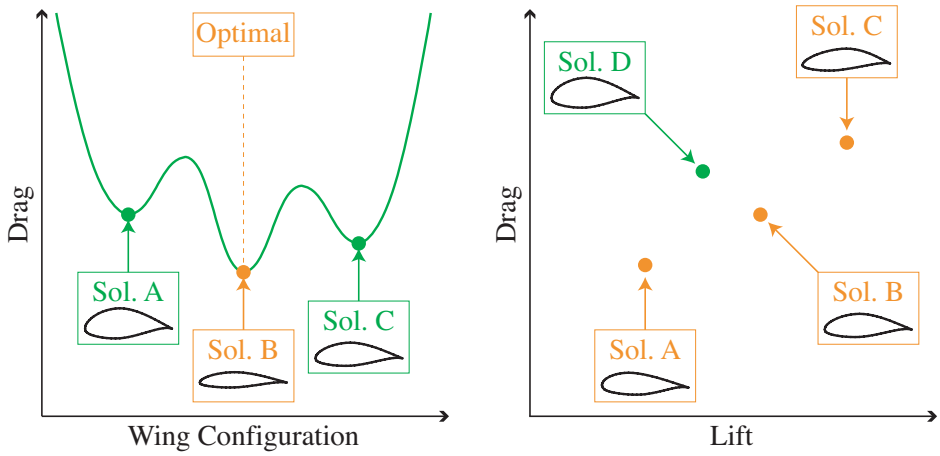
Definition 2: Pareto-Optimality A solution $\boldsymbol{x}_i \in \Omega$ is said to be Pareto-optimal if there is no other solution $\boldsymbol{x}_j \in \Omega$ which dominates $\boldsymbol{x}_i$.

In general multi-objective optimization problems, an improvement of one objective function requires a compromise of the other objective function, and such trade-off relation can be represented by the curve which consists of multiple Pareto-optimal solutions, as shown in Fig. 2.3. Therefore, it is most important to compare multiple solutions based on the Pareto-optimality concept, find multiple various Pareto-optimal solutions, and reveal the trade-off information among multiple objective functions for realistic multi-objective design optimizations.

There exist mainly two kinds of numerical approaches to solve optimization problems; one is a deterministic or mathematical approach such as the gradient-based method described in Subsubsec. 2.2.1.2, and the other is a heuristic or statistical approach such as the genetic algorithm described in Subsubsec. 2.2.1.3.

2.2.1.2 Gradient-Based Method

Among the optimization approaches, the gradient-based method (GM)[29,30] has been most widely used. GM is based on a simple concept so-called hill-climbing strategy



(a) Single-objective. (b) Multi-objective.

Figure 2.2 Classification of optimization problems in terms of the number of objective functions.

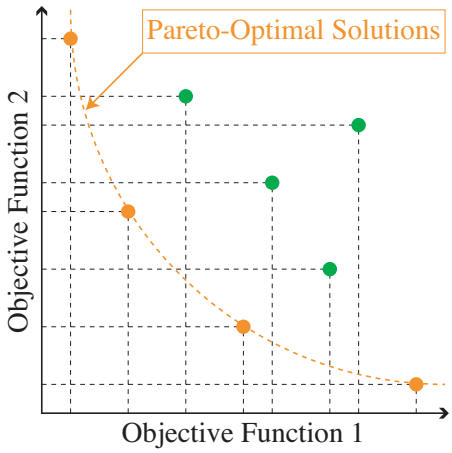


Figure 2.3 Pareto-optimal solutions.

that we can reach the top of hill from a certain point by climbing the hill in the steepest direction at this point. Generally in GM, an optimal solution $\boldsymbol{x}$ is consequently obtained by iterating a following equation:

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^{k} + \alpha \boldsymbol{d} \tag{2.3}$$

where $\bullet^k$ represents the iteration number. $\boldsymbol{d}$ is a vector search direction and α is the distance that we want to move in the direction $\boldsymbol{d}$. The flowchart of basic GM is shown in Fig. 2.4. The optimization process using GM consists of mainly three components; picking of search direction $\boldsymbol{d}$, decision of distance α, and convergence check. In the case of constrained optimization problems, $\boldsymbol{d}$ and α must be set so that $\boldsymbol{x}$ can be included in the feasible region at each iteration, such as an modified method of feasible direction (MMFD),[29,30] a sequential linear programming (SLP)[29,30] a sequential quadratic programming (SQP),[29,30] etc..

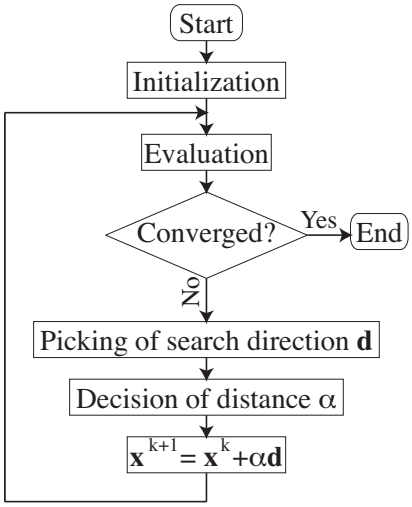


Figure 2.4 Flowchart of gradient-based method.

A global optimum solution can be obtained by GM if objective and constraint functions are differentiable and convex, i.e., have only one smooth peak. In practical, however, it is very difficult to prove differentiability and convexness. The final solution obtained by GM depends on the initial values of solution, and GM only can guarantee that the obtained solution is local optimal in the neighborhood of the initial solution. Therefore, the optimizations using GM must be carried out many times with different initial solutions to obtain a global optimal solution. In addition, it is required to calculate gradients of objective and constraint functions with respect to design variables in GM. This means that the total number of evaluations depends

on the number of design variables seriously, and GM needs even more computation time as the number of design variables becomes larger. In this sense, GM is not practical in real-world design problems which have multi-peak objective and constraint functions and many design variables.

2.2.1.3 Genetic Algorihm

Evolution is a phenomenon of adapting to a certain environment and passing genes to next generations. Darwin, who did not know about the underlying genetics, identified three basic principles driving natural evolution; reproduction, natural selection and diversity of individuals, maintained by variations from one to the next generation.[45] The genetic algorithms (GA) was originally proposed by Holland to understand adaptation phenomena in both natural and artificial systems by imitating biological evolution process.[46] Then, GA was developed by Goldberg as a numerical approach to solve optimization problems.[31]

The flowchart of basic GA is shown in Fig. 2.5. GA simulates the mechanism of natural evolution where a biological population which consists of multiple individuals $1, 2, \cdots, N$ evolves over generations to adapt to an certain environment by genetic operators such as selection and reproduction, and consequently can bear the best individual adapting to the environment, i.e., produce an optimal solution in a design optimization problem. Fitness, individual and genes in the evolutionary theory correspond to the objective function, solution (design candidate) and design variables in design optimization problems, respectively.

In recent years, GA has been applied to various design optimization problems due to some advantages which GA has. First, GA can be applied to any design optimization problem independently of differentiability and convexness of objective and constraint functions. Second, GA can solve the optimization problems without any special knowledge of the problems if only the objective and constraint functions are given mathematically. These advantages are due to the characteristic that the optimizations using GA have only to evaluate the values of objective and constraint functions, while those using GM need to calculate the gradients of these functions, too. Third, GA can avoid local optimal solutions and find a global optimal solution independently of initial values of solution because GA deals with multiple solutions simultaneously during the optimization process.

Details of GA are described as follows.

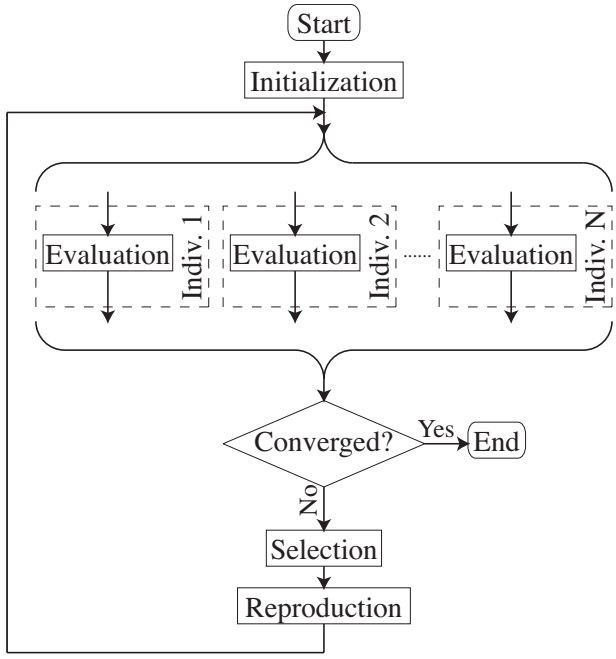


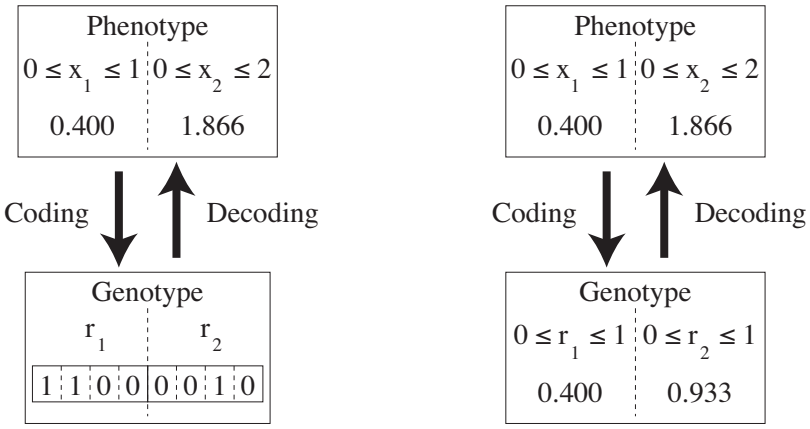
Figure 2.5 Flowchart of genetic algorithm.

Coding and Decoding GA requires both phenotype and genotype of design variables. The phenotype represents actual design variables, and the genotype represents the design variables transformed like genes. The transformation from phenotype into genotype is called coding and the inverse transformation is called decoding. Traditionally, GA represent the genotype by bit string (binary GA), as shown in Fig. 2.6(a). In the binary GA, however, there exist some problems; a discrete genotype can not correspond to any continuous phenotype, simulation accuracy is reduced due to rounding error produced at coding and decoding, and it becomes difficult to control the scale of variation of design variables by genetic operators, etc.. Therefore, it is desirable to represent the genotype by a real number (real-coded GA), as shown in Fig. 2.6(b), for more accurate and effective genetic operators in the physical sense. In the real-coded GA, the i-th design variable (phenotype) x_i is coded into the genotype r_i, which is normalized in $[0, 1]$:

$$r_i = \frac{x_i - x_{imin}}{x_{imax} - x_{imin}} \tag{2.4}$$

where x_{imax} and x_{imin} are the upper and lower limits of i-th design variable, respectively.

Initialization An initial population which consists of multiple individuals is generated. This is often accomplished by sampling values of design variables randomly



(a) Binary coding. (b) Real coding.

Figure 2.6 Coding and decoding methods.

within the design space.

Evaluation For each individual $i = 1, 2, \cdots, N$ in the population, objective and constraint function values $f(\boldsymbol{x}_i)$ and $\boldsymbol{g}(\boldsymbol{x}_i)$ are evaluated. Then, the fitness value F_i, which is the measure of superiority and inferiority in the population, is evaluated from $f(\boldsymbol{x}_i)$ and $\boldsymbol{g}(\boldsymbol{x}_i)$ for all individuals. Larger F_i indicates a better individual than the other ones. In the single-objective non-constrained optimizations where a objective function $f(\boldsymbol{x})$ must be minimized, F_i is simply given by $f(\boldsymbol{x}_i)$, *e.g.*, as

$$F_i = -f(\boldsymbol{x}_i) \tag{2.5}$$

How to evaluate the fitness value in the case of multi-objective or constrained optimizations is described in Subsubsec. 2.2.1.4.

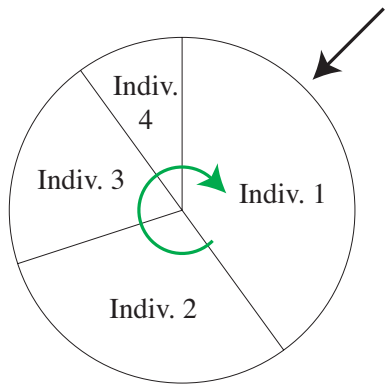
Selection GA selects better individuals as parents to generate new individuals for the next generation. Therefore, the type of selection method has a large influence on search performance of GA. Usually, the probability Prob_i where each individual $i = 1, 2, \cdots, N$ is selected is assigned on the bias of its fitness value F_i as

$$\mathrm{Prob}_i = \frac{F_i}{\sum_{j=1}^{N} F_j} \tag{2.6}$$

Then, the parents are selected according to Prob_i by either roulette-wheel selection (RWS)[31] as shown in Fig. 2.7(a), or stochastic universal sampling (SUS)[47] as shown in Fig. 2.7(b). The RWS simply selects N individuals by creating N random

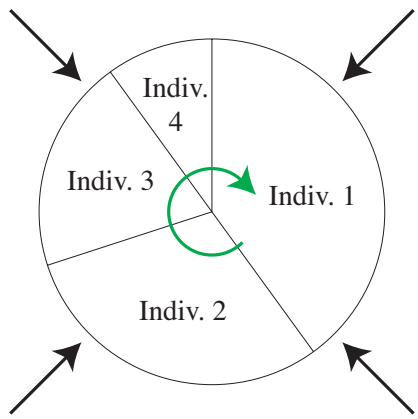
numbers individually, therefore the same individual may be selected many times if the individual has larger fitness value. The SUS, on the other hand, requires only one random number for whole selection process. A set of N equi-spaced numbers is created so that many different individuals will be selected to maintain diversity.

Individual	Fitness value	Selection probability [%]
1	4	40
2	3	30
3	2	20
4	1	10



(a) Roulette-wheel selection.

Individual	Fitness value	Selection probability [%]
1	4	100
2	3	100
3	2	80
4	1	40



(b) Stochastic universal sampling.

Figure 2.7 Selection methods.

Reproduction Based on the selected parents, new individuals are generated for the next generation. The reproduction operator consists of two operators; crossover and mutation.

Crossover Crossover is a operator which exchanges part of genotype of the selected parents and produces new individuals with the intent of improving the fitness value of the next generation. The type of crossover method remarkably affects the

search performance of GA.

For binary GA, crossover exchanges part of bit string of the selected parents with more than one cross sites. Figure 2.8 shows the example of two-point crossover. In the two-point crossover, two different cross sites are selected at random, and the bit string is divided into three substrings, and finally new bit strings of two children are generated by exchanging the middle substring between two parents.

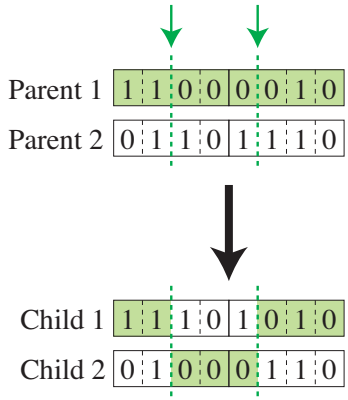


Figure 2.8 Two-point crossover.

For real-coded GA, many crossover methods are proposed. A well-known is the blended crossover (BLX-α) method.[48] The BLX-α generates the genotypes of i-th design variable of two children: C_{1i} and C_{2i}, from those of two parents: P_{1i} and P_{2i}, as

$$\begin{aligned} C_{1i} &= \gamma P_{1i} + (\gamma - 1) P_{2i} \\ C_{2i} &= (\gamma - 1) P_{1i} + \gamma P_{2i} \end{aligned} \tag{2.7}$$

where

$$\gamma = U(1 + 2\alpha) - \alpha \tag{2.8}$$

U is a uniform random number in $[0, 1]$. The children generated by the BLX-α exist in the rectangular region whose diagonal is the axis between the parents with Euclidean length of d and may be extended equally on both sides determined by a user-specified parameter α, as shown in Fig. 2.9.

Mutation Mutation maintains diversity and expands the search space by changing part of genotype of the generated child. If the mutation rate is high, a GA search is close to a random search and results in slow convergence. Therefore, an adequate value is required for the mutation rate.

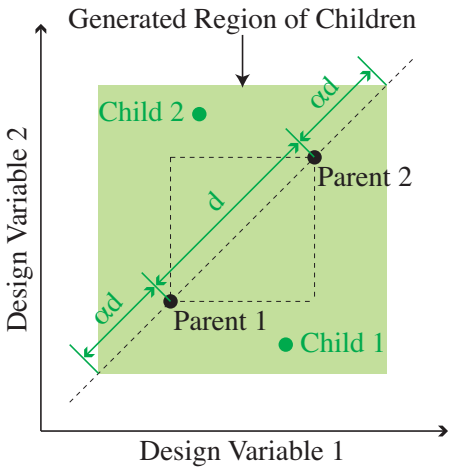


Figure 2.9 Blended crossover.

For binary GA, mutation is performed to reverse bit numbers. Figure. 2.10 shows the bit-wise mutation which reverses each bit number with a certain probability.

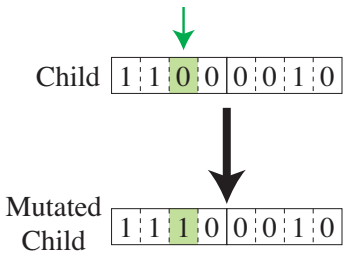


Figure 2.10 Bit-wise mutation.

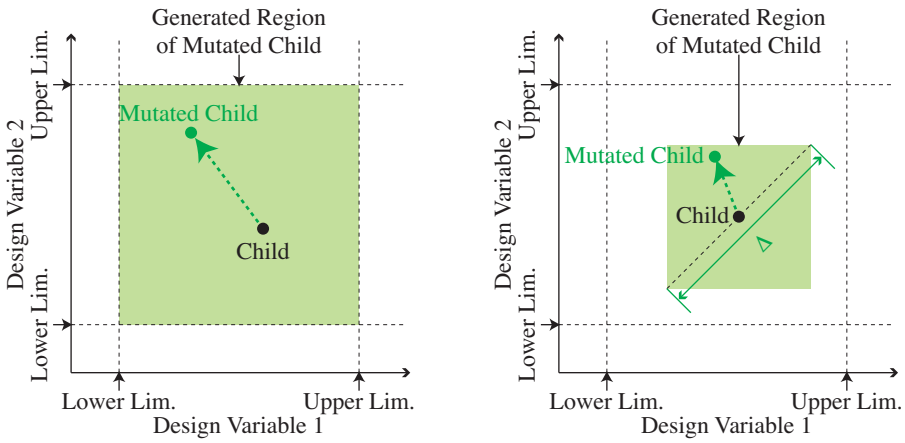
For real-coded GA, mutation is realized by adding perturbations to the genotypes. The simplest is the random mutation[49] which reproduces a new genotype $C_i{}'$ in the whole design space, as shown in Fig. 2.11(a):

$$C_i{}' = U\left(r_{imax} - r_{imin}\right) \tag{2.9}$$

This operator is independent of the original child and is equivalent to a random initialization. Instead of generating $C_i{}'$ in the whole design space, $C_i{}'$ can be also generated in the vicinity of the original child C_i with a uniform probability distribution,[33] as shown in Fig. 2.11(b):

$$C_i{}' = C_i + \left(U - \frac{1}{2}\right)\Delta_i \tag{2.10}$$

where Δ_i is a user-specified maximum perturbation allowed in the genotype of the i-th design variable.



(a) Random mutation. (b) Uniform mutation.

Figure 2.11 Mutation methods.

Alternation of Generations From both the parents and children of the present generation, the individuals surviving in the next generation are selected. In the present study, the best-N selection[50,51] is used to model the alternation of generations. The best-N selection constructs the population of the next generation which consist of N better individuals selected from total $2N$ individuals which consist of N parents and N children of the present generation. This selection brings selection pressure according to their fitness values. In general, GA can not guarantee a monotonic improvement in fitness value without some elitist strategies.[52] The use of best-N selection ensures a monotonic improvement in fitness value because the best individual in the all past generations can necessarily survive in the next generation.

2.2.1.4 Multi-Objective Genetic Algorithm

In the case of multi-objective deign optimizations, it is required to compare multiple solutions based on Pareto-optimality concept, as described in Subsubsec. 2.2.1.1. When using GM to solve multi-objective optimization problems, the original multi-objective problem (Eq. 2.1) must be transformed into the single-objective optimization problem, because GM can treat only single-objective optimization problems, as follows:

$$\begin{aligned} \text{Minimize: } & f'(\boldsymbol{x}) = \sum_{i=1}^{M} w_i f_i(\boldsymbol{x}) \\ \text{Subject to: } & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0} \end{aligned} \tag{2.11}$$

where w_i is a user-specified weighting factor of the i-th objective function $f_i(\boldsymbol{x})$ and represents the measure of evaluation balance of each $f_i(\boldsymbol{x})$. However, such weighted summation method has some difficulties. First, it is required to specify appropriate values of weighting factors w_i in advance. Obtained solution depends on the values of w_i, but the trade-off information among multiple objective functions is unknown. Therefore, users can not help specifying the values of w_i without any trade-off information. Second, only one Pareto-optimal solution is obtained by one case because this method is based on the formulation of single-objective optimization problem. Therefore, users must carry out many cases with different values of w_i or initial solution to obtain the trade-off information among multiple objective functions. Third, this method can not guarantee uniformity of obtained Pareto-optimal solutions. Fourth, this method can not search whole region of Pareto-optimal front in the case of non-convex Pareto-optimal front.

On the other hand, GA can be extended easily to an effective optimization approach to solve multi-objective optimization problems. As described in Subsubsec. 2.2.1.1, it is required to compare multiple solutions based on Pareto-optimality concept in multi-objective optimization problems. GA deals with multiple solutions simultaneously and evaluates only the values of objective and constraint functions of each individual. Therefore, GA can evolve multiple individuals adapting to the environment with multiple objective functions without any transformation into a single objective function as seen in GM, i.e., obtain trade-off information among the objective functions efficiently by only one case if the individuals are compared based on the Pareto-optimality concept in terms of the objective functions.

According to this idea, the multi-objective genetic algorithm (MOGA) was developed by Deb.[33] The difference between the single-objective GA (SOGA) and MOGA exists in how to evaluate the fitness value. Details of the evaluation of fitness value in MOGA are described as follows.

Ranking Now consider the case of non-constrained optimizations. In SOGA, the fitness value of each individual is simply given by the objective function value of its individual independently of the other individuals, *e.g.*, as Eq. 2.5. In MOGA, on the other hand, the fitness value of each individual must be determined based on Pareto-optimality concept, i.e., from the distribution of all treated individuals in the objective function space, as shown in Fig. 2.3.

Fonseca and Fleming developed the Pareto-ranking method[53] to assign the fitness

value to each individual based on Pareto-optimality concept. Firstly, this method assigns a rank R_i which is a measure of Pareto-optimality to each individual as

$$R_i = 1 + N_{di} \tag{2.12}$$

where N_{di} represents the number of individuals which dominate the i-th individual. Figure 2.12 shows the example of Pareto-ranking method. N_{di} is zero and R_i is one for Pareto-optimal solutions, while other dominated solutions have larger R_i than one corresponding to larger N_{di} than zero. Therefore, Pareto-optimal solutions can be obtained by assigning larger fitness value to the individual which has smaller rank. In addition, all individuals which have the same rank are assigned the same fitness value, i.e., have the same probability to survive in the next generation. Therefore, multiple various Pareto-optimal solutions can be obtained effectively by one case.

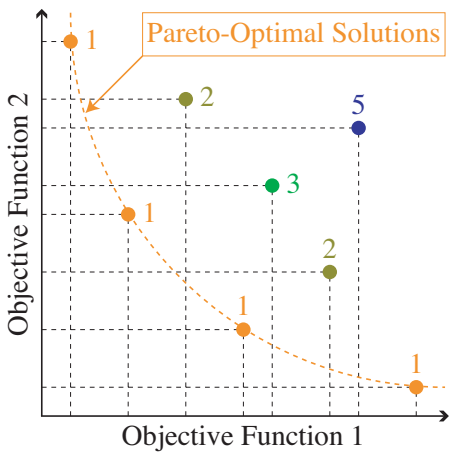


Figure 2.12 Parero-ranking method.

Fitness Assignment Next, the fitness value F_i are assigned to each individual from the assigned rank R_i. The simplest way is, *e.g.*, to use the reciprocal of R_i as

$$F_i = \frac{1}{R_i} \tag{2.13}$$

The use of this method, however, leads to the undesirable situation that only the individuals of $R_i = 1$ are selected as parents. In the present study, F_i is assigned by using Michaleswicz's nonlinear function[49] to maintain diversity of selected parents as

$$F_i = c\,(1 - c)^{R_i - 1} \tag{2.14}$$

where c is a user-defined parameter which corresponds to the probability where the individuals of $R_i = 1$ are selected, and this value is set to be 0.1 in the present study.

Niching In the multi-objective optimization problems, furthermore, it is also desirable to obtain Pareto-optimal solutions distributed uniformly and widely in the whole design space for easy understanding and acquisition of rich information about trade-off relation among multiple objective functions. Therefore, so-called niching procedure is required to maintain uniformity of obtained solutions. In the present study, the fitness sharing proposed by Fonseca and Fleming[33,53] is used for niching. In this method, the assigned fitness value of the i-th individual: F_i is modified to the shared fitness value $F_i{}'$ as

$$F_i{}' = \frac{F_i}{nc_i} \tag{2.15}$$

where nc_i is called niche count of the i-th individual, which is summation of sharing function $Sh(d_{ij})$ among all N individuals in the present generation as

$$\begin{aligned} nc_i &= \sum_{j=1}^{N} Sh(d_{ij}) \\ Sh(d_{ij}) &= \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma_{share}}\right)^{\alpha} & \text{if } d_{ij} \leq \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{2.16}$$

d_{ij} is the normalized Euclidean distance in the objective function space between the i-th and j-th individuals as

$$d_{ij} = \sqrt{\sum_{k=1}^{M} \frac{f_{ki} - f_{kj}}{f_{kmax} - f_{kmin}}} \tag{2.17}$$

where M is the number of objective functions, and f_{kmax} and f_{kmin} are the maximum and minimum values of the k-th objective function among all individuals in the present generation, respectively. σ_{share} is called the normalized niche size, and estimated by the following equation:

$$(1 + \sigma_{share})^M - 1 = N{\sigma_{share}}^M \tag{2.18}$$

In the present study, σ_{share} is calculated by solving this equation with Newton-Raphson method. And, α is the sharing parameter and set to be 1 in the present study. Eq. 2.16 indicates that $Sh(d_{ij})$, and then nc_i increase to reduce $F_i{}'$ if d_{ij} is smaller than σ_{share}, i.e., the fitness value of the individual close to the other individuals within the distance of σ_{share} in the objective function space is reduced and this individual becomes hard to be selected as parents. Therefore, the use of the fitness sharing can scatter the locally concentrated individuals and maintain uniformity of obtained solutions in the whole design space.

Constraint Handling In many real-world design optimization problems, usually there exist some constraints. The selection of constraint handling technique is a very important issue in GA because it is not easy for GA to solve constrained optimization problems compared to GM.

For single-objective optimizations using GA, traditional constraint handling approach is the penalty function method.[33] In the single-objective constrained optimization problem where a objective function $f(\boldsymbol{x})$ must be minimized and constraints $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), \cdots, g_L(\boldsymbol{x})]^T \leq \boldsymbol{0}$ must be satisfied, this method defines the fitness value of the i-th individual: F_i as weighted summation of objective function value and the amount of constraint violation as

$$F_i = -f(\boldsymbol{x}_i) - \sum_{j=1}^{L} \alpha_j \max\left[g_j(\boldsymbol{x}_i), 0\right] \tag{2.19}$$

where L is the number of constraints and α_j is the positive penalty function coefficient of the j-th constraint. This method, however, requires a careful specification of α_j to obtain a feasible optimal solution. For example, if α_j is too small, the obtained optimal solution may not satisfy the constrains. On the other hand, if α_j is too large, the obtained optimal solution may not have satisfactory objective function value. Not only the balance between objective and constraint functions, but also that among multiple constraints must be tuned carefully so that the obtained optimal solution can satisfy all constraints. Moreover, the penalty function method does not intend to deal with multi-objective optimization problems, i.e., this method does not consider how to combine multiple constraints with multiple objective functions.

Deb proposed an attracting approach for constraint-handling in multi-objective optimizations using MOGA based on the Pareto-optimality concept.[54] This method assigns rank to each individual by using the following definition:

Definition 3: Constrain-Domination by Deb A solution $\boldsymbol{x}_i$ is said to constrain-dominate the other solution $\boldsymbol{x}_j$ if any of the following conditions is true:

1. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both feasible and $\boldsymbol{x}_i$ dominates $\boldsymbol{x}_j$.
2. $\boldsymbol{x}_i$ is feasible and $\boldsymbol{x}_j$ is not.
3. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both infeasible, but $\boldsymbol{x}_i$ has a smaller constraint violation than $\boldsymbol{x}_j$.

This method does not require specification of penalty function coefficient as long as the number of constraint is one. In this sense, this method is very useful for the

design optimization using GA. However, this method still requires careful tuning of the weighting coefficients of the constraints when multiple constraints exist.

Another interesting approach is proposed by Coello Coello.[55] Essence of this method is usage of the following definition of constrain-domination:

Definition 4: Constrain-Domination by Coello Coello A solution $\boldsymbol{x}_i$ is said to constrain-dominate the other solution $\boldsymbol{x}_j$ if any of the following conditions is true:

1. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both feasible and $\boldsymbol{x}_i$ dominates $\boldsymbol{x}_j$.
2. $\boldsymbol{x}_i$ is feasible and $\boldsymbol{x}_j$ is not.
3. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both infeasbile and $\boldsymbol{x}_i$ violates less number of constraints than $\boldsymbol{x}_j$.
4. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both infeasible and $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ violate the same number of constraints, but $\boldsymbol{x}_i$ has a total amount of constraint function violation smaller than the constraint violation of $\boldsymbol{x}_j$ where the total amount of constraint violation for the i-th individual: G_i is given by

$$G_i = \sum_{k=1}^{L} \max\left[g_k(\boldsymbol{x}_i), 0\right] \tag{2.20}$$

Advantage of this method is that it does not use any coefficient to be tuned even if multiple constraints exist. However, this method may not be very efficient when the magnitudes of violation value of constraints $g_k(\boldsymbol{x})$ are significantly different because the total amount of constraint violation of each individual is simple summation of violation values of $g_k(\boldsymbol{x})$ without weighting coefficients.

In the present study, the Pareto-optimality-based constraint handling (PBCH) technique developed by Oyama *et al.*[56,57] is adopted. This method assigns rank to each individual by using the following definition:

Definition 5: Constrain-Domination by Oyama *et al.* A solution $\boldsymbol{x}_i$ is said to constrain-dominate the other solution $\boldsymbol{x}_j$ if any of the following conditions is true:

1. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both feasible and $\boldsymbol{x}_i$ dominates $\boldsymbol{x}_j$ in objective function space.
2. $\boldsymbol{x}_i$ is feasible and $\boldsymbol{x}_j$ is not.

3. $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are both infeasible, but $\boldsymbol{x}_i$ dominates $\boldsymbol{x}_j$ in constraint space, where dominance in objective function space is defined as Definition 1 while dominance in constraint space is defined as Definition 6.

Definition 6: Dominance in Constraint Space A solution $\boldsymbol{x}_i$ is said to dominate the other solution $\boldsymbol{x}_j$ in constraint space if all the following inequalities are satisfied.

$$\begin{aligned} g_k{}'(\boldsymbol{x}_i) \leq g_k{}'(\boldsymbol{x}_j), \quad {}^{\forall}k = 1, 2, \cdots, L \\ g_k{}'(\boldsymbol{x}_i) < g_k{}'(\boldsymbol{x}_j), \quad {}^{\exists}k = 1, 2, \cdots, L \end{aligned} \tag{2.21}$$

where

$$g_k{}'(\boldsymbol{x}_i) = \max\left[g_k(\boldsymbol{x}_i), 0\right] \tag{2.22}$$

This method simply introduces the idea of Pareto-optimality concept in the objective function space to the constraint function space. This idea can be used for most GAs. For example, any ranking procedure can be used for ranking among feasible individuals as well as infeasible ones. Use of stochastic ranking[58] may further improve efficiency and robustness. In addition, robustness is further improved by applying any niching mechanism in the constraint space as well as the objective function space for severely constrained design optimization problems. This method has a number of advantages: First, application to multi-objective multi-constraint optimization problems is straightforward. Second, it is efficient and robust even when the magnitude of violation of each constraint is very different because total amount of constraint violation is not used. Third, it does not require any coefficient to be specified and tuned. Fourth, number of objective functions is not increased because non-dominance ranking is applied to feasible individuals and infeasible individuals separately. And finally, implementation is easy.

2.2.2 Robust Design Optimization Approaches

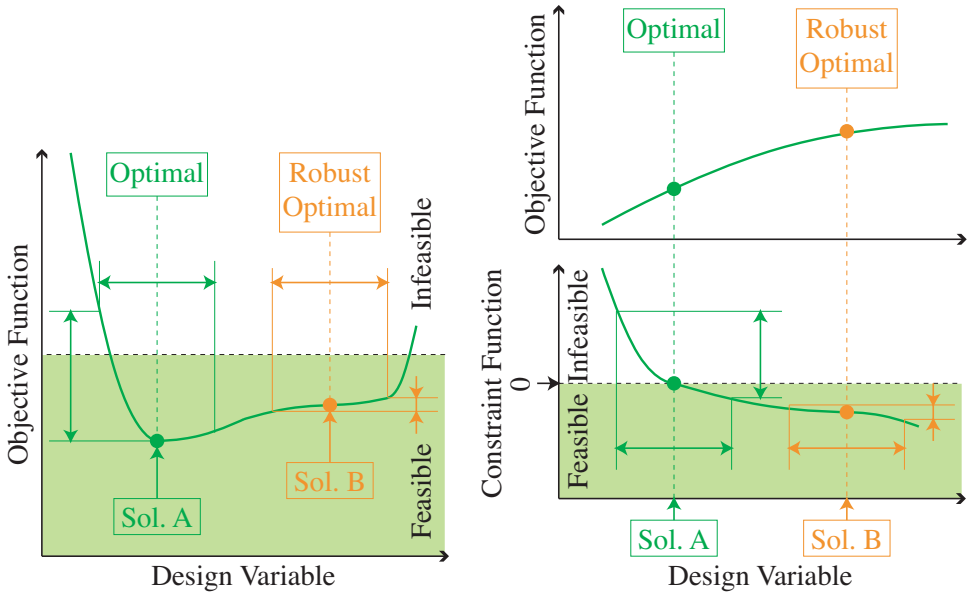
2.2.2.1 General Theory of Robust Design Optimization

In real-world engineering designs, inevitably there exist errors and uncertainties in design process, manufacturing process, and operating condition. Therefore, actually a design optimized by a traditional design optimization approach considering only optimality, as described in Subsec. 2.2.1, may not be able to achieve its expected performance due to such errors and uncertainties. Thus, the idea of robust optimization considering both optimality and robustness has been paid attention to for real-world design problems in recent years.

Here, robustness is defined as stability of system against uncertainties, *e.g.*, the dispersion of performance parameters (values of objective or constraint functions) due to the dispersion of design parameters (values of design variables) caused by design errors or uncertainties. Figure 2.13 illustrates comparisons between a traditional optimization and a robust optimization. First, focus on the case when considering the robustness in objective function value, as shown in Fig. 2.13(a). In the single-objective optimization problem where the value of objective function must be minimized, the solution A obtained by a traditional optimization is the best in terms of the optimality because this solution has the smallest value of objective function. However, the solution A disperses widely in terms of the objective function against the dispersion of design variable, and this dispersion may extend to the infeasible range. On the other hand, the solution B obtained by a robust optimization is moderately good in terms of the optimality and also good in terms of the robustness because the dispersion of objective function against that of design variable is narrow at this solution.

Next, focus on the case when considering the robustness in constraint function violation, as shown in Fig. 2.13(b). In the single-objective optimization problem where the value of objective function must be minimized and also the value of constraint function must be smaller than zero, the solution A obtained by a traditional optimization is the best in terms of optimality because this solution has the smallest value of objective function within the feasible range where the value of constraint function is smaller than zero. Here, note that usually the value of constraint function of a traditional optimal solution becomes zero, called active, as shown in Fig. 2.13(b) in general constrained optimization problems. Therefore, the solution A disperses easily to the infeasible range in terms of the constraint function against the disper-

sion of design variable. On the other hand, the solution B obtained by a robust optimization is moderately good in terms of the optimality and also good in terms of the robustness because the dispersion of constraint function against that of design variable is included in the feasible range at this solution.



(a) Considering robustness in objective function value.

(b) Considering robustness in constraint function violation.

Figure 2.13 Comparisons between traditional optimization and robust optimization.

Optimality and robustness are usually competing in real-world design problems, like competing multiple objective functions in a multi-objective optimization problem. Therefore, there exist multiple compromised solutions between the optimality and the robustness. Objectives of robust design optimizations are to find these compromised solutions to reveal the trade-off information, and to give chance to the upper-level decision maker to select one solution from these compromised solutions with other considerations.

2.2.2.2 Design for Six Sigma

Design for six sigma (DFSS)[36] is one of conventional robust design optimization approaches. Here, the term "sigma" refers to standard deviation σ, which is a measure of dispersion. A DFSS is based on "six sigma concept", which is one of the management reform technique aiming at the establishment of business process with very

small dispersion such that the range of $\pm 6\sigma$ around the mean value μ of performance parameter is included in the acceptable range. The level of dispersion can be defined as "sigma level n", as shown in Fig. 2.14. Larger sigma level indicates smaller dispersion, i.e., more robust characteristic. Table 2.1 shows the safety probability, where the dispersive performance is included in the acceptable range, at each sigma level n in the case where dispersive performance follows the normal probability density distribution. Larger sigma level indicates larger safety probability, i.e., it results in normal operation more often. Here, note that "six sigma concept" is not limited to the normal probability density distribution case, and also can be applied to any case where dispersive performance follows other kind of probability density distribution.

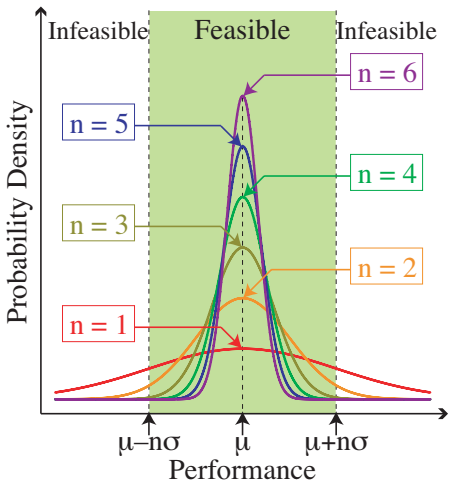


Figure 2.14 Characteristics of sigma level n.

Table 2.1 Safety probability at each sigma level for normal distribution.

Sigma Level n	Safety Probability [%]
1	68.25
2	95.46
3	99.73
4	99.9937
5	99.999943
6	99.9999998

Now, for example, consider the traditional single-objective non-constrained optimization problem where the value of objective function $f(\boldsymbol{x})$ of design variables $\boldsymbol{x}$ must be minimized. In DFSS, when considering the robustness in objective function

value, this traditional optimization problem is rewritten to the problem where the weighted summation of the mean value μ_f and the variance ${\sigma_f}^2$ of dispersive objective function $f(\boldsymbol{x})$ against the dispersion of design variables $\boldsymbol{x}$ must be minimized as follow:

$$\text{Minimize: } w_\mu \mu_f + w_\sigma {\sigma_f}^2 \tag{2.23}$$

where w_μ and w_σ are user-specified weighting factors of μ_f and ${\sigma_f}^2$, respectively. In DFSS, in addition, following inequality constraints must be satisfied so that the sigma level robustness quality which users expect, as shown in Fig. 2.14 and Table 2.1, will be realized.

$$\begin{aligned} \text{Subject to: } \mu_f - n\sigma_f &\geq \text{LSL} \\ \mu_f + n\sigma_f &\leq \text{USL} \end{aligned} \tag{2.24}$$

where n represents user-specified sigma level, and LSL and USL are user-specified lower and upper specification limits, respectively.

The flowchart of robust optimization using DFSS is shown in Fig. 2.15. First, parameters such as weighting factors w_μ and w_σ, sigma level n, and LSL and USL are specified by users, and then it proceeds to the optimization block. In this block, the mean value μ_f and standard deviation σ_f of objective function $f(\boldsymbol{x})$ at sample points around design variables $\boldsymbol{x}$ are estimated, and $w_\mu \mu_f + w_\sigma {\sigma_f}^2$ is treated as one objective function. Then, $\mu_f - n\sigma_f - \text{LSL}(\geq 0)$ and $\mu_f + n\sigma_f - \text{USL}(\leq 0)$ are evaluated as two constraint functions. $\boldsymbol{x}$ in next step is reproduced based on the evaluated objective and constraint functions, and this optimization process is iterated until $\boldsymbol{x}$ is converged. This single-objective optimization can be carried out by using any single-objective optimization approaches.

However, DFSS as shown in Fig. 2.15 has some drawbacks as follows.

- **It is necessary to set weighting factors w_μ and w_σ carefully in advance.** There exists arbitrariness in specification of the weighting factors in Eq. 2.23. Users need to specify value of these parameters according to balance between the optimality and the robustness which they expect. However, it is difficult for users to specify value of weighting factors appropriately in advance because the trade-off information between the optimality and the robustness is unknown. Eventually, users can not help carrying out many cases with different weighting factor values until satisfactory solution is obtained.

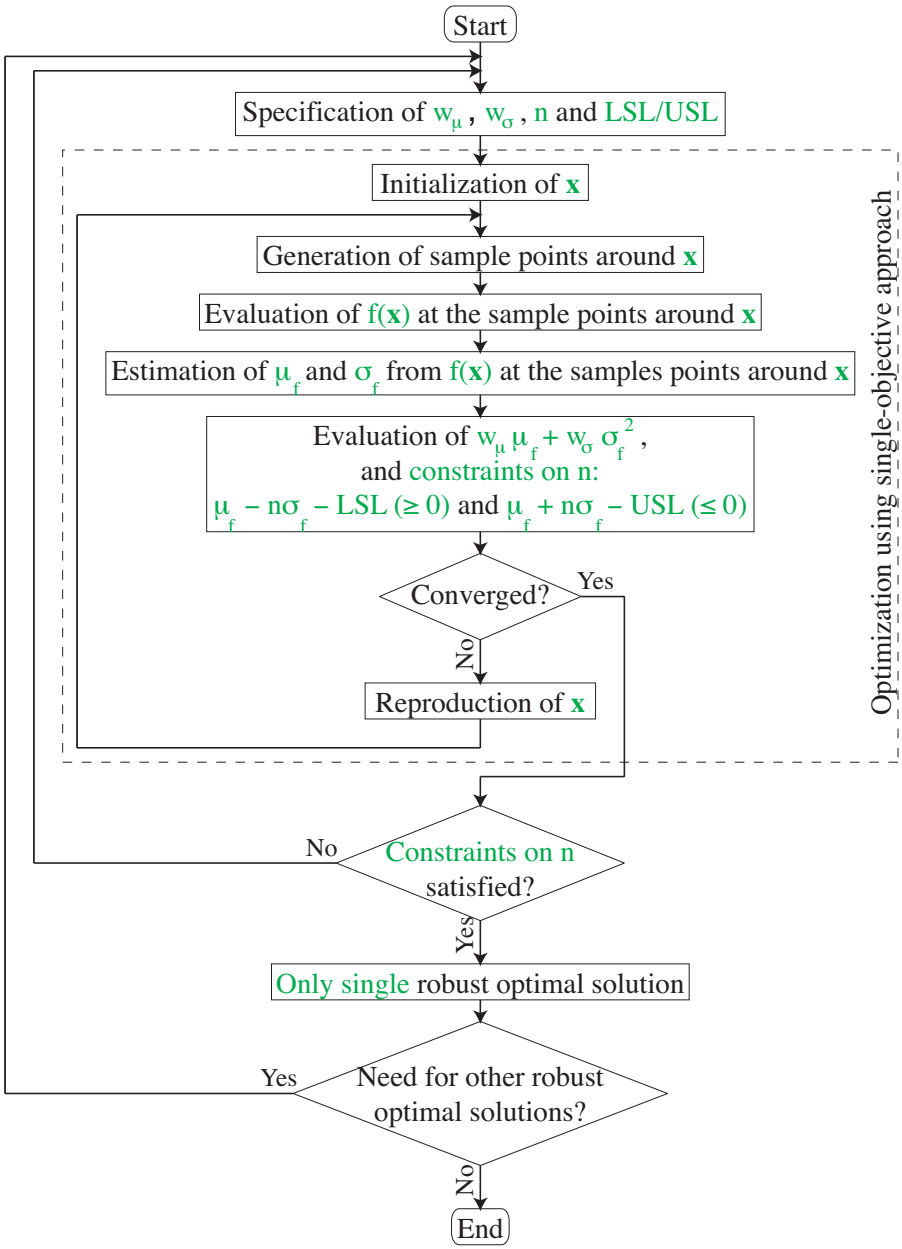


Figure 2.15 Flowchart of robust optimization using design for six sigma.

- **It is necessary to set appropriate sigma level n in advance.** Essentially, the sigma level satisfying Eq. 2.24 is known only after an optimization. However, users must specify the sigma level blindly without any information. If users specify the sigma level too large, a robust optimal solution may not be obtained by the optimization because the constraints on sigma level become too severe. On the other hand, if users set the sigma level too small, the robust optimal solution obtained by the optimization may have lack of reliability. Eventually, users need to carry out robust optimization repeatedly until a feasible robust optimal solution is found.

- **Only one robust optimal solution is obtained by one case.** Because DFSS deals with the single-objective optimization problem considering the weighted summation of the mean value and the variance of objective function ($w_\mu \mu_f + w_\sigma {\sigma_f}^2$) as a new objective function, only one robust optimal solution is obtained by one case. Therefore, users must carry out many cases with different weighting factor values or the sigma level to obtain multiple robust optimal solutions. Moreover, many cases can not necessarily derive trade-off relation between the optimality and the robustness.

2.2.2.3 Design for Multi-Objective Six Sigma

Drawbacks of the DFSS are mainly come from the fact that the DFSS deals with the single-objective optimization problem as shown in Eq. 2.23. Therefore, a new robust optimization approach named as "design for multi-objective six sigma (DFMOSS)" is developed in the present study. The idea of DFMOSS is to incorporate multi-objective genetic algorithm (MOGA)[33] into DFSS to solve the drawbacks of the DFSS. In DFMOSS, the mean value μ_f and the standard deviation σ_f of dispersive objective function $f(\boldsymbol{x})$ against the dispersion of design variables $\boldsymbol{x}$ are treated as new multiple objective functions and minimized separately as follows:

$$\begin{aligned} \text{Minimize: } & \mu_f \\ & \sigma_f \end{aligned} \tag{2.25}$$

Because the formulation of DFMOSS does not include the weighting factors w_μ and w_σ which are seen in the formulation of DFSS (Eq. 2.23), DFMOSS does not have difficulty in the advance specification of weighting factors. In addition, solving this multi-objective optimization problem by using MOGA results in many robust optimal solutions by only one case. Furthermore, DFMOSS does not consider the

constraints on sigma level n which are seen in the formulation of DFSS (Eq. 2.24) during optimization process. Thus, DFMOSS also does not have difficulty in the advance specification of sigma level n. The sigma level n satisfying Eq. 2.24 can be evaluated from the robust optimal solutions obtained by the optimization as post-processing, as described later.

Figure 2.16 shows the flowchart of robust optimization using DFMOSS. There is no need to specify parameters such as weighting factors w_μ and w_σ, sigma level n, and lower and upper specification limits LSL and USL before optimization block, which is seen in the DFSS (Fig. 2.15). In the optimization block using MOGA, multiple solutions $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N$ are dealt with simultaneously. For each individual $i = 1, 2, \cdots, N$, the mean value μ_{f_i} and the standard deviation σ_{f_i} of objective function $f(\boldsymbol{x})$ are evaluated as two separate objective functions from $f(\boldsymbol{x})$ at the sampled points around $\boldsymbol{x}_i$. Better solutions are selected based on the Pareto-optimality concept between μ_{f_i} and σ_{f_i} for $i = 1, 2, \cdots, N$. Solutions $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N$ in the next step are reproduced by crossover and mutation from the selected solutions. This optimization process is iterated until the trade-off relation between μ_f and σ_f is converged, and multiple robust optimal solutions are obtained.

In DFMOSS, the sigma level n satisfying Eq. 2.24 is evaluated from the obtained robust optimal solutions after the optimization block, as shown in Fig. 2.16. Figure 2.17 illustrates detail of post-evaluation of sigma level n. Now, it is assumed that four robust optimal solutions A, B, C and D are obtained by a robust optimization using DFMOSS. Painted area indicates the area satisfying the constraints on 6σ robustness quality. The solution C is included in this painted area, i.e., this solution has more than 6σ robustness quality. If the sigma level n becomes smaller, *e.g.* $n = 3$, the painted area becomes larger (the gradient of broken lines by which the painted area is bound becomes steeper), as illustrated in Fig. 2.17, and the solution B is included in the painted area, i.e., this solution has more than 3σ robust ness quality. Thus, the satisfied sigma level n of each solution can be evaluated easily and flexibly by using DFMOSS.

Here, it should be noted that above approach to evaluate the sigma level as post-processing can be very useful especially for preliminary conceptual designs where there exists no precedent and it is difficult to specify appropriate sigma level before the optimizations. Therefore, the robust design optimizations using the DFMOSS seem to be suitable for effective acquisition of information about robustness quality in the robust aerodynamic design of Mars exploratory airplane wing considered in

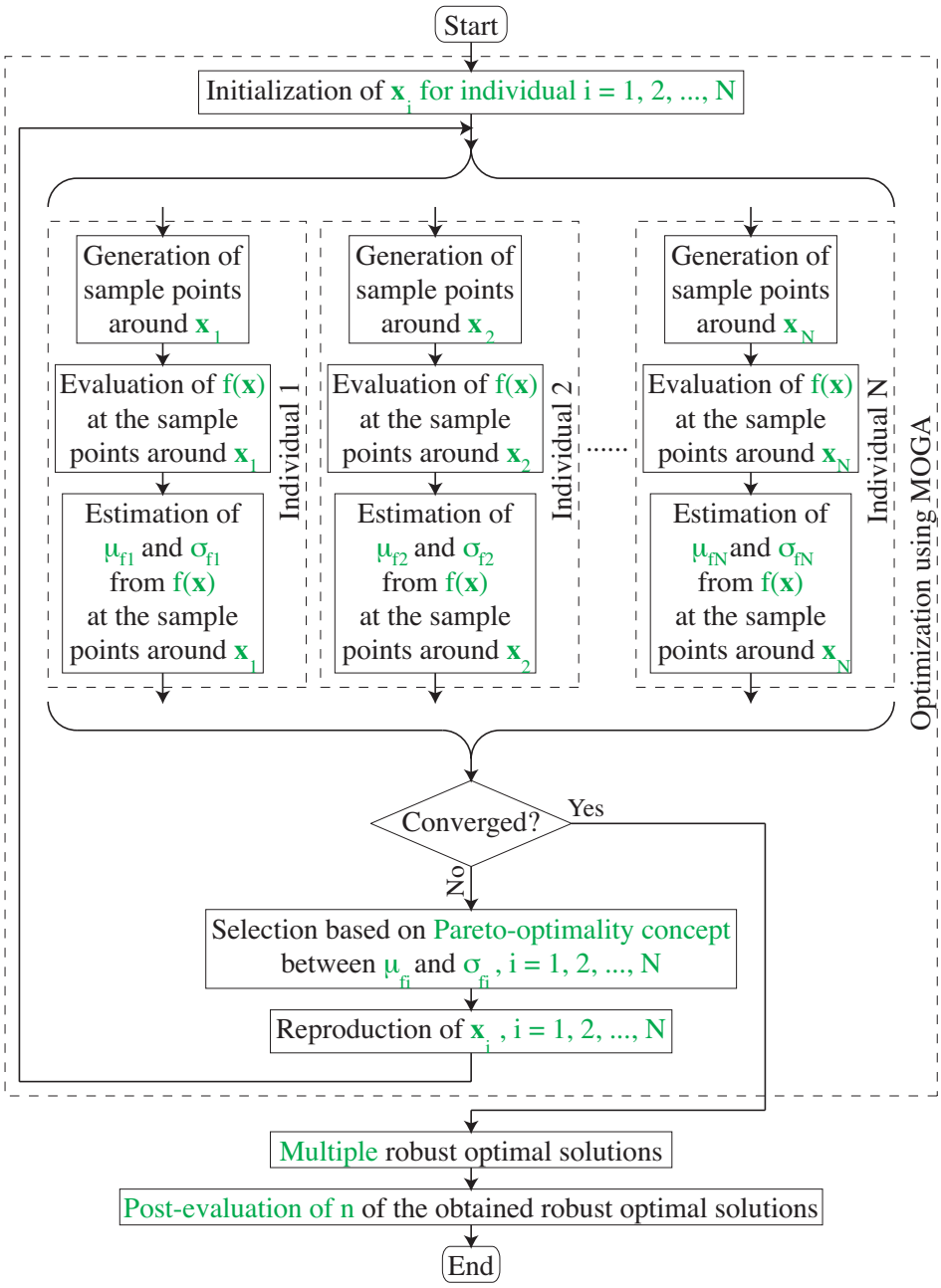


Figure 2.16 Flowchart of robust optimization using design for multi-objective six sigma.

the present study.

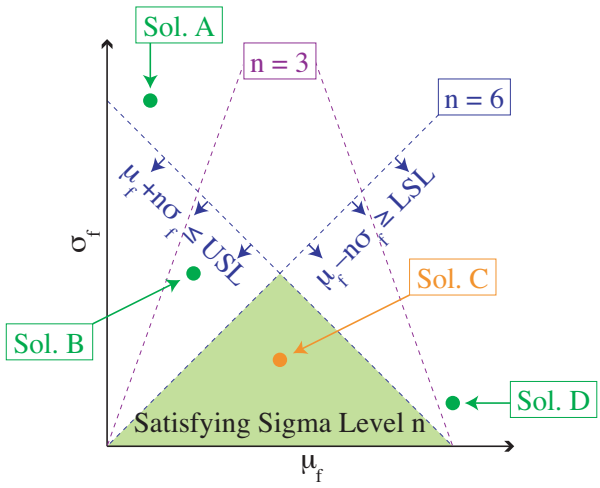


Figure 2.17 Post-evaluation of sigma level n in design for multi-objective six sigma.

The above descriptions of DFSS and DFMOSS are based on the case when considering the robustness in objective function value $f(\boldsymbol{x})$. Here, note that almost the same descriptions can be applied to the case when considering the robustness in constraint function violation, too. Now, consider the traditional single-objective constrained optimization problem where the value of objective function $f(\boldsymbol{x})$ of design variables $\boldsymbol{x}$ must be minimized and also the values of constraint functions $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), \cdots, g_L(\boldsymbol{x})]^T$ must be smaller than zero. In DFSS, when considering the robustness in constraint functions violation, this traditional optimization problem is rewritten as follows:

$$\begin{aligned} &\text{Minimize: } \mu_f \\ &\text{Subject to: } \begin{bmatrix} \mu_{g_1} + n_1\sigma_{g_1} \\ \mu_{g_2} + n_2\sigma_{g_2} \\ \vdots \\ \mu_{g_L} + n_L\sigma_{g_L} \end{bmatrix} \leq \mathbf{0} \\ &\qquad \boldsymbol{g}(\boldsymbol{\mu_x}) \leq \mathbf{0} \end{aligned} \tag{2.26}$$

where $\boldsymbol{\mu_x}$ is the mean value of $\boldsymbol{x}$, and μ_{g_i}, σ_{g_i} and n_i are the mean value, the standard deviation, and user-specified sigma levels of $g_i(\boldsymbol{x})$, respectively. This formulation does not include the weighting factors which are seen in the case when considering the robustness in objective function value (Eq. 2.23). However, this still has the drawbacks that there exists difficulty in the advance specification of sigma levels $n_1, n_2, \cdots, n_L$ without any information, and only one robust optimal solution is

obtained by one case.

In DFMOSS, on the other hand, it is proposed to use following multi-objective formulation for the robust optimization considering the robustness in constraint functions violation.

$$\begin{aligned} &\text{Minimize: } \mu_f \\ &\text{Maximize: } \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_L \end{bmatrix} = - \begin{bmatrix} \mu_{g_1}/\sigma_{g_1} \\ \mu_{g_2}/\sigma_{g_2} \\ \vdots \\ \mu_{g_L}/\sigma_{g_L} \end{bmatrix} \\ &\text{Subject to: } \boldsymbol{g}(\boldsymbol{\mu_x}) \leq \mathbf{0} \end{aligned} \tag{2.27}$$

Similar to the case when considering the robustness in objective function value (Eq. 2.25), solving this multi-objective optimization problem by using MOGA results in many robust optimal solutions by only one case. In addition, this formulation does not have difficulty in the advance specification of sigma levels $n_1, n_2, \cdots, n_L$ without any information. The sigma levels $n_1, n_2, \cdots, n_L$ are evaluated from the mean values $\mu_{g_1}, \mu_{g_2}, \cdots, \mu_{g_L}$ and the standard deviations $\sigma_{g_1}, \sigma_{g_2}, \cdots, \sigma_{g_L}$ of constraint functions $\boldsymbol{g}(\boldsymbol{x})$ during an optimization process.

Here, it should be noted that the DFMOSS can be used for not only the robust design optimizations where optimized design variables such as configuration parameters disperse, but also those where non-optimized environmental variables such as operating condition parameters disperse. It indicates that the DFMOSS has capability of treating various real-world design problems including errors and uncertainties in design process, manufacturing process, and operating conditions.

2.2.2.4 Estimation of Statistical Values

In the robust optimizations using DFSS and DFMOSS, it is required to estimate statistical values such as the mean value (μ_f or $\mu_{g_1}, \mu_{g_2}, \cdots .\mu_{g_L}$) and the standard deviation (σ_f or $\sigma_{g_1}, \sigma_{g_2}, \cdots .\sigma_{g_L}$) of dispersive objective or constraint functions against the dispersion of design variables. In general, these statistical value can be estimated by generating sample points around the focused solution $\boldsymbol{x}$, and evaluating the value of objective function $f(\boldsymbol{x})$ or constraint functions $\boldsymbol{g}(\boldsymbol{x})$ at each sample point. There exist mainly two kinds of approaches to estimate the statistical values; one is a sensitivity-based variability estimation approach, and the other is Monte Carlo simulation.

Sensitivity-Based Variability Estimation A sensitivity-based variability estimation approach uses gradients for objective and constraint functions with respect to the dispersive design variables. For example, based on Taylor's series expansions, an objective function $f(\boldsymbol{x})$ can be rewritten as

$$f(\boldsymbol{x}+\Delta\boldsymbol{x}) = f(\boldsymbol{x}) + \left(\frac{\partial f}{\partial \boldsymbol{x}}\right)^T \Delta\boldsymbol{x} + \frac{1}{2}\Delta\boldsymbol{x}^T \frac{\partial^2 f}{\partial \boldsymbol{x}^2}\Delta\boldsymbol{x} + O(\Delta\boldsymbol{x}^3) \tag{2.28}$$

Then, the mean value μ_f and the standard deviation σ_x of objective function $f(\boldsymbol{x})$ can be derived from this equation. For example, neglecting higher-order terms than the first one, μ_f and σ_f are given as

$$\begin{aligned} \mu_f &= f(\boldsymbol{\mu_x}) \\ \sigma_f &= \sqrt{\sum_{i=1}^{n}\left(\frac{\partial f}{\partial x_i}\right)^2 {\sigma_{x_i}}^2} \end{aligned} \tag{2.29}$$

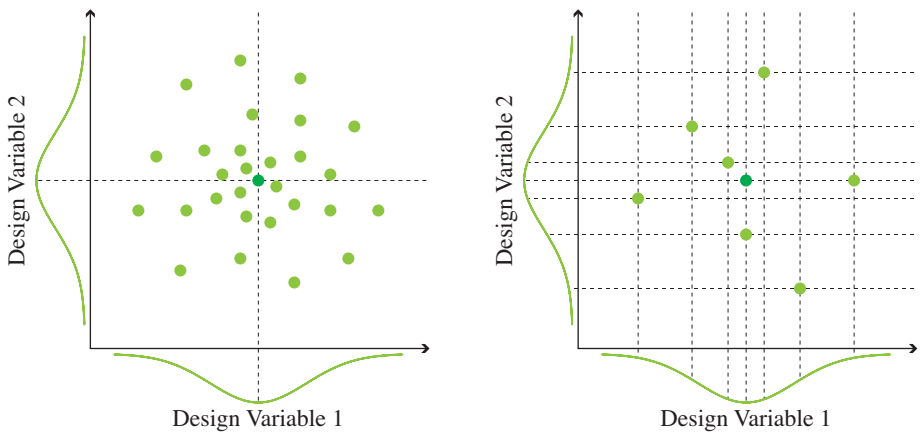
And, neglecting higher-order terms than the second one, μ_f and σ_f are given as

$$\begin{aligned} \mu_f &= f(\boldsymbol{\mu_x}) + \frac{1}{2}\sum_{i=1}^{n}\frac{\partial^2 f}{\partial {x_i}^2}\sigma_{x_i} \\ \sigma_f &= \sqrt{\sum_{i=1}^{n}\left(\frac{\partial f}{\partial x_i}\right)^2 {\sigma_{x_i}}^2 + \frac{1}{2}\sum_{i,j=1}^{n}\left(\frac{\partial^2 f}{\partial x_i \partial x_j}\right)^2 {\sigma_{x_i}}^2 {\sigma_{x_j}}^2} \end{aligned} \tag{2.30}$$

where n is the number of dispersive design variables (note that it is not the total number of all design variables). $\boldsymbol{\mu_x}$ is user-specified mean values of dispersive design variables $\boldsymbol{x}$, and σ_{x_i} is user-specified standard deviation of the i-th dispersive design variable. For small n, Taylor's expansion approach is more efficient than Monte Carlo simulation because the sample points required to estimate μ_f and σ_f is small. The first-order expansion approach requires only $n+1$ sample points The second-order approach requires $(n+1)(n+2)/2$ sample points, but this expense is reduced to $2n+1$ if cross terms in Eq. 2.30 are neglected. However, this approach becomes less efficient with increasing n. In the present study, the second-order Taylor's expansion neglecting the cross terms is used for relatively accurate estimation and reduction in computation time.

Monte Carlo Simulation Monte Carlo simulation (MCS) is implemented by randomly generating the sample points following user-specified probability density distribution with user-specified mean value and standard deviation of each dispersive design variable. Simple random sampling (SRS) is the basic and most commonly

used MCS technique. In MCS with SRS technique, as shown in Fig. 2.18(a), a defined number of evaluations of objective function $f(\boldsymbol{x})$ are performed by sampling design variables around the focused solution $\boldsymbol{x}$. The number of sample points required for accurate estimation of mean value μ_f and standard deviation σ_f of $f(\boldsymbol{x})$ using SRS technique is usually more than desirable and often more than practical (often $1,000 \sim 10,000$ or more).



(a) Simple random sampling. (b) Descriptive sampling.

Figure 2.18 Monte Carlo simulation techniques.

As other sampling technique, descriptive sampling (DS)[59] has been developed to reduce the sample size without sacrificing the quality of the statistical description of the behavior of system. In MCS with DS technique, as shown in Fig. 2.18(b), the space defined by each dispersive design variable is divided into subsets of equal probability, and the evaluation of $f(\boldsymbol{x})$ is performed with each subset of one dispersive design variable combined with only one subset of each dispersive design variable. The i-th value of divided subset xd_i are generated by

$$xd_i = \Phi^{-1}\left(\frac{i-0.5}{nd}\right) \tag{2.31}$$

where nd is the number of divided subsets per a dispersive design variable. $\Phi(x), x \in (-\infty, \infty)$ is the cumulative distribution function of specified probability density $\phi(x)$, i.e., $\Phi(x) = \int_{-\infty}^{x} \phi(u)\mathrm{d}u$. When the inverse function $\Phi^{-1}(U), U \in [0,1]$ is not available, numerical or functional approximations must be required. For example, Ramberg and Schmeiser[60] proposed the approximate inverse function for the lambda

distribution, as

$$\Phi^{-1}(U) = \lambda_1 + \frac{U^{\lambda_3} - (1-U)^{\lambda_3}}{\lambda_2} \tag{2.32}$$

where, for normal distribution with the mean value of μ and standard deviation of σ,

$$\lambda_1 = \mu, \quad \lambda_2 = \frac{0.1975}{\sigma}, \quad \lambda_3 = 0.135 \tag{2.33}$$

This sampling technique is similar to Latin hypercube sampling technique.[61] As shown in Fig. 2.18(b), each row and column in the discretized two design variable space is sampled only once at random. The number of sample points required for accurate estimation of mean value μ_f and standard deviation σ_f of $f(\boldsymbol{x})$ using DS technique is smaller (often $100 \sim 1,000$ or more) than that using SRS technique. Even so, the computation cost of MCS for accurate estimation is generally much higher than that of Taylor's expansion approach. However, MCS is consequently the most accurate.

2.3 CFD Simulation Approach

2.3.1 Governing Equations

2.3.1.1 Navier-Stokes Equations in Cartesian Coordinate System

The governing equations of fluid dynamics are the Navier-Stokes equations. The conservation form of the compressible Navier-Stokes equations in the three-dimensional Cartesian coordinate system is written as follow:

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \frac{\partial \boldsymbol{E}}{\partial x} + \frac{\partial \boldsymbol{F}}{\partial y} + \frac{\partial \boldsymbol{G}}{\partial z} = \frac{\partial \boldsymbol{E_v}}{\partial x} + \frac{\partial \boldsymbol{F_v}}{\partial y} + \frac{\partial \boldsymbol{G_v}}{\partial z} \tag{2.34}$$

The independent variables are time t and the Cartesian coordinates x, y and z. $\boldsymbol{Q}$ represents the conservative variable vector, $\boldsymbol{E}$, $\boldsymbol{F}$ and $\boldsymbol{G}$ represent the inviscid flux vectors in x, y and z directions, respectively, and $\boldsymbol{E_v}$, $\boldsymbol{F_v}$ and $\boldsymbol{G_v}$ represent the viscous flux vectors in x, y and z directions, respectively, given by

$$\begin{aligned}
&\boldsymbol{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{bmatrix}, \quad \boldsymbol{E} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u w \\ (e+p)\,u \end{bmatrix}, \quad \boldsymbol{F} = \begin{bmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho v w \\ (e+p)\,v \end{bmatrix}, \quad \boldsymbol{G} = \begin{bmatrix} \rho w \\ \rho w u \\ \rho w v \\ \rho w^2 + p \\ (e+p)\,w \end{bmatrix}, \\
&\boldsymbol{E_v} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ \beta_x \end{bmatrix}, \quad \boldsymbol{F_v} = \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ \beta_y \end{bmatrix}, \quad \boldsymbol{G_v} = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ \beta_z \end{bmatrix} \\
&\beta_x = \tau_{xx} u + \tau_{xy} v + \tau_{xz} w - q_x \\
&\beta_y = \tau_{yx} u + \tau_{yy} v + \tau_{yz} w - q_y \\
&\beta_z = \tau_{zx} u + \tau_{zy} v + \tau_{zz} w - q_z
\end{aligned} \tag{2.35}$$

The first row of Eq. 2.34 represents the conservation of mass, the second, third and fourth ones represent that of momentums in x, y and z directions, respectively, and the fifth one represents that of energy. u, v and w is the velocity components in x, y and z directions, respectively, while ρ is the density and e is the total energy per unit volume. The static pressure p is related to ρ, u, v, w and e by the equation of state for an ideal gas:

$$\begin{aligned}
p &= \rho R T \\
&= (\gamma - 1)\left[e - \frac{1}{2}\left(u^2 + v^2 + w^2\right)\right]
\end{aligned} \tag{2.36}$$

where R, T, γ represent the gas constant, static temperature and ratio of specific heats, respectively. For air at standard conditions, $R = 287$ [m^2/(s^{2}K)] and $\gamma = 1.4$.

τ_{ij} represents the viscous stress tensor as shown in Fig. 2.19. We assume that τ_{ij} in Eq. 2.35 is a linear function of the rate of strain tensor. This assumption is closely satisfied in most cases of interest with possible deviation only in the flow through very strong shock wave.[62] τ_{ij} for Newtonian fluid becomes

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \delta_{ij} \lambda \frac{\partial u_k}{\partial x_k} \tag{2.37}$$

where μ is the dynamic viscosity coefficient and λ is the second viscosity coefficient. And, τ_{ii} becomes

$$\tau_{ii} = (2\mu + 3\lambda) \frac{\partial u_i}{\partial x_i} \tag{2.38}$$

The coefficient $2\mu + 3\lambda$ is called bulk viscosity coefficient, and assumed to be zero in the Stokes's hypothesis: $(\tau_{xx} + \tau_{yy} + \tau_{zz})/3 = 0$. Thus, λ can be written as

$$\lambda = -\frac{2}{3}\mu \tag{2.39}$$

and, τ_{ij} becomes

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij} \frac{\partial u_k}{\partial x_k} \right) \tag{2.40}$$

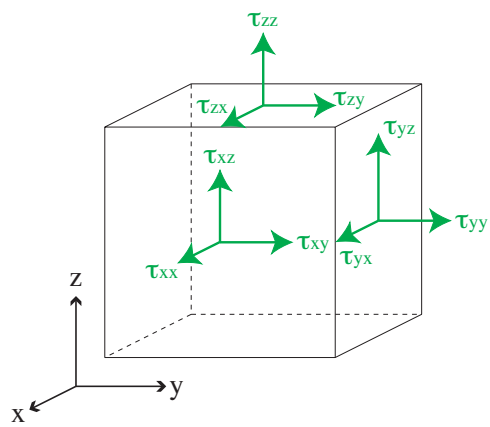


Figure 2.19 Viscous stress tensor.

Similar to the viscous stress tensor, we assume a linear relation between the heat flux q_j in Eq. 2.35 and the gradient of the temperature. q_j is obtained by the Fourier's law:

$$\begin{aligned} q_j &= -\kappa \frac{\partial T}{\partial x_j} \\ &= -\frac{\mu}{(\gamma - 1)\, Pr} \frac{\partial}{\partial x_j} \left(a^2 \right) \end{aligned} \tag{2.41}$$

κ is the heat conductivity coefficient. Pr is the Prandtl number defined as

$$Pr = \frac{\mu c_p}{\kappa} \tag{2.42}$$

where c_p represent the specific heat at constant pressure: $c_p = \gamma R/(\gamma - 1)$. For air at standard conditions, $Pr = 0.72$. a is the speed of sound:

$$\begin{aligned} a &= \sqrt{\gamma R T} \\ &= \sqrt{\frac{\gamma p}{\rho}} \end{aligned} \tag{2.43}$$

In general, μ and κ are related to T, *e.g.*, by the Sutherland's law for viscosity and heat conductivity. In the present study, however, μ and κ are assumed to be constant to freestream values because we deal with only subsonic flow with small change of T.

2.3.1.2 Non-Dimensionalization of Navier-Stokes Equations

The Navier-Stokes equations are often put into non-dimensional form because the characteristic parameter such as Mach number, Reynolds number and Prandtl number can be varied independently. In addition, by non-dimensionalizing the equations, the flow variables are normalized so that their values fall between certain prescribed limits such as 0 and 1. In the present study, flow variables are non-dimensionalized by introducing their values at freestream conditions as follows:

$$\begin{aligned} &t^* = \frac{t}{L_{ref}/a_\infty}, \quad {x_i}^* = \frac{x_i}{L_{ref}}, \quad \rho^* = \frac{\rho}{\rho_\infty}, \quad {u_i}^* = \frac{u}{a_\infty}, \\ &p^* = \frac{p}{\rho_\infty {a_\infty}^2} = \frac{p}{\gamma p_\infty}, \quad e^* = \frac{e}{\rho_\infty {a_\infty}^2} \\ &\mu^* = \frac{\mu}{\mu_\infty}, \quad {\tau_{ij}}^* = \frac{\tau_{ij}}{\mu_\infty a_\infty / L_{ref}}, \quad {q_j}^* = \frac{q_j}{\mu_\infty {a_\infty}^2 / L_{ref}} \end{aligned} \tag{2.44}$$

where $\bullet^*$ denotes non-dimensional flow variables, $\bullet_\infty$ denotes their values at freestream conditions, and L_{ref} is the reference length. By substituting these non-dimensional variables into Eq. 2.34, the non-dimensional conservation form of the compressible Navier-Stokes equations in the Cartesian coordinate system are obtained:

$$\frac{\partial \boldsymbol{Q}^*}{\partial t^*} + \frac{\partial \boldsymbol{E}^*}{\partial x^*} + \frac{\partial \boldsymbol{F}^*}{\partial y^*} + \frac{\partial \boldsymbol{G}^*}{\partial z^*} = \frac{1}{Re}\left(\frac{\partial {\boldsymbol{E_v}}^*}{\partial x^*} + \frac{\partial {\boldsymbol{F_v}}^*}{\partial y^*} + \frac{\partial {\boldsymbol{G_v}}^*}{\partial z^*}\right) \tag{2.45}$$

where

$$\boldsymbol{Q^*} = \begin{bmatrix} \rho^* \\ \rho^* u^* \\ \rho^* v^* \\ \rho^* w^* \\ e^* \end{bmatrix}, \quad \boldsymbol{E^*} = \begin{bmatrix} \rho^* u^* \\ \rho^* u^{*2} + p^* \\ \rho^* u^* v^* \\ \rho^* u^* w^* \\ (e^* + p^*)\, u^* \end{bmatrix}, \quad \boldsymbol{F^*} = \begin{bmatrix} \rho^* v^* \\ \rho^* v^* u^* \\ \rho^* v^{*2} + p^* \\ \rho^* v^* w^* \\ (e^* + p^*)\, v^* \end{bmatrix},$$

$$\boldsymbol{G^*} = \begin{bmatrix} \rho^* w^* \\ \rho^* w^* u^* \\ \rho^* w^* v^* \\ \rho^* w^{*2} + p^* \\ (e^* + p^*)\, w^* \end{bmatrix}, \quad \boldsymbol{E_v}^* = \begin{bmatrix} 0 \\ {\tau_{xx}}^* \\ {\tau_{xy}}^* \\ {\tau_{xz}}^* \\ {\beta_x}^* \end{bmatrix}, \quad \boldsymbol{F_v}^* = \begin{bmatrix} 0 \\ {\tau_{yx}}^* \\ {\tau_{yy}}^* \\ {\tau_{yz}}^* \\ {\beta_y}^* \end{bmatrix}, \quad \boldsymbol{G_v}^* = \begin{bmatrix} 0 \\ {\tau_{zx}}^* \\ {\tau_{zy}}^* \\ {\tau_{zz}}^* \\ {\beta_z}^* \end{bmatrix} \tag{2.46}$$

$${\beta_x}^* = {\tau_{xx}}^* u^* + {\tau_{xy}}^* v^* + {\tau_{xz}}^* w^* - {q_x}^*$$

$${\beta_y}^* = {\tau_{yx}}^* u^* + {\tau_{yy}}^* v^* + {\tau_{yz}}^* w^* - {q_y}^*$$

$${\beta_z}^* = {\tau_{zx}}^* u^* + {\tau_{zy}}^* v^* + {\tau_{zz}}^* w^* - {q_z}^*$$

and

$$p^* = (\gamma - 1)\left[e^* - \frac{1}{2}\rho^*\left({u^*}^2 + {v^*}^2 + {w^*}^2\right)\right] \tag{2.47}$$

Re in Eq. 2.45 represents the Reynolds number based on L_{ref}:

$$\begin{aligned} Re &= \frac{\rho_\infty a_\infty L_{ref}}{\mu_\infty} \\ &= \frac{1}{M_\infty}\frac{\rho_\infty u_\infty L_{ref}}{\mu_\infty} \end{aligned} \tag{2.48}$$

where M is the Mach number: $M = u/a$. Here, note that Eq. 2.34 and Eq. 2.45 are identical except for $1/Re$ in the right hand side. For convenience, hereafter, $\bullet^*$ is dropped from the non-dimensional variables.

2.3.1.3 General Transformation of Navier-Stokes Equations

The governing equations of fluid dynamics are valid for any coordinate system. For practical applications to the flow in complicated and moving geometries, it is more convenient to transform the Cartesian coordinate system (physical domain) to any general curvilinear and moving coordinate system (computational domain) so that the moving boundary surfaces in the physical domain can be easily mapped onto planes in the computational domain, as shown in Fig. 2.20. The Navier-Stokes equations can be written in terms of general coordinate system by using the general

transformation[63,64] as follows:

$$\begin{aligned}\xi &= \xi\left(x, y, z, t\right)\\ \eta &= \eta\left(x, y, z, t\right)\\ \zeta &= \zeta\left(x, y, z, t\right)\\ \tau &= t\end{aligned} \tag{2.49}$$

where ξ, η and ζ are the general coordinates, and τ is time. The differential form of Cartesian coordinate systems (x, y, z, t) is

$$\begin{bmatrix}\mathrm{d}x\\ \mathrm{d}y\\ \mathrm{d}z\\ \mathrm{d}t\end{bmatrix} = \begin{bmatrix}x_\xi & x_\eta & x_\zeta & x_\tau\\ y_\xi & y_\eta & y_\zeta & y_\tau\\ z_\xi & z_\eta & z_\zeta & z_\tau\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\mathrm{d}\xi\\ \mathrm{d}\eta\\ \mathrm{d}\zeta\\ \mathrm{d}\tau\end{bmatrix} \tag{2.50}$$

where $t_\xi = t_\eta = t_\zeta = 0$ and $t_\tau = 1$. Similarly, the differential form of general coordinate system (ξ, η, ζ, τ) is

$$\begin{bmatrix}\mathrm{d}\xi\\ \mathrm{d}\eta\\ \mathrm{d}\zeta\\ \mathrm{d}\tau\end{bmatrix} = \begin{bmatrix}\xi_x & \xi_y & \xi_z & \xi_t\\ \eta_x & \eta_y & \eta_z & \eta_t\\ \zeta_x & \zeta_y & \zeta_z & \zeta_t\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\mathrm{d}x\\ \mathrm{d}y\\ \mathrm{d}z\\ \mathrm{d}t\end{bmatrix} \tag{2.51}$$

where $\tau_x = \tau_y = \tau_z = 0$ and $\tau_t = 1$.

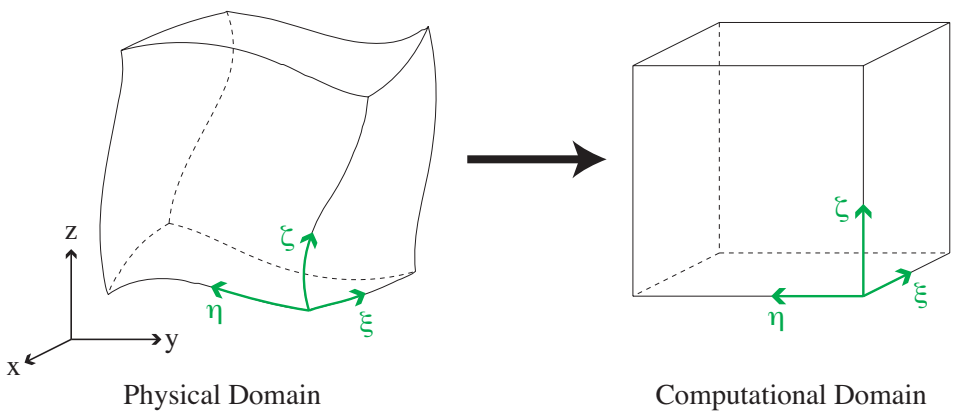


Figure 2.20 General transformation.

Therefore, we have the relations between (x, y, z, t) and (ξ, η, ζ, τ) from Eqs. 2.50 and 2.51, and the metrics ξ_x, ξ_y, etc., are formed from the derivatives x_ξ, x_η, etc.,

as follows:

$$\begin{bmatrix} \xi_x & \xi_y & \xi_z & \xi_t \\ \eta_x & \eta_y & \eta_z & \eta_t \\ \zeta_x & \zeta_y & \zeta_z & \zeta_t \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta & x_\zeta & x_\tau \\ y_\xi & y_\eta & y_\zeta & y_\tau \\ z_\xi & z_\eta & z_\zeta & y_\tau \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1}$$

$$= J \begin{bmatrix} y_\eta z_\zeta - y_\zeta z_\eta & z_\eta x_\zeta - z_\zeta x_\eta & x_\eta y_\zeta - x_\zeta y_\eta & -x_\tau (y_\eta z_\zeta - y_\zeta z_\eta) - y_\tau (z_\eta x_\zeta - z_\zeta x_\eta) - z_\tau (x_\eta y_\zeta - x_\zeta y_\eta) \\ y_\zeta z_\xi - y_\xi z_\zeta & z_\zeta x_\xi - z_\xi x_\zeta & x_\zeta y_\xi - x_\xi y_\zeta & -x_\tau (y_\zeta z_\xi - y_\xi z_\zeta) - y_\tau (z_\zeta x_\xi - z_\xi x_\zeta) - z_\tau (x_\zeta y_\xi - x_\xi y_\zeta) \\ y_\xi z_\eta - y_\eta z_\xi & z_\xi x_\eta - z_\eta x_\xi & x_\xi y_\eta - x_\eta y_\xi & -x_\tau (y_\xi z_\eta - y_\eta z_\xi) - y_\tau (z_\xi x_\eta - z_\eta x_\xi) - z_\tau (x_\xi y_\eta - x_\eta y_\xi) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.52}$$

Here, note that the vectors $\boldsymbol{S_\xi}$, $\boldsymbol{S_\eta}$ and $\boldsymbol{S_\zeta}$ shown as

$$\boldsymbol{S_\xi} = \frac{1}{J} \begin{bmatrix} \xi_x \\ \xi_y \\ \xi_z \end{bmatrix}, \quad \boldsymbol{S_\eta} = \frac{1}{J} \begin{bmatrix} \eta_x \\ \eta_y \\ \eta_z \end{bmatrix}, \quad \boldsymbol{S_\zeta} = \frac{1}{J} \begin{bmatrix} \zeta_x \\ \zeta_y \\ \zeta_z \end{bmatrix} \tag{2.53}$$

correspond to the area vectors of the cell interface normal to ξ, η and ζ directions, respectively, and ξ_t/J, η_t/J and ζ_t/J correspond to the volume swept by the moving cell interface normal to ξ, η and ζ directions, respectively. J is the transformation Jacobian from (x, y, z, t) to (ξ, η, ζ, τ):

$$\begin{aligned} J &= \frac{\partial (\xi, \eta, \zeta)}{\partial (x, y, z)} = \left[\frac{\partial (x, y, z)}{\partial (\xi, \eta, \zeta)} \right]^{-1} = \det \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix}^{-1} \\ &= \frac{1}{x_\xi y_\eta z_\zeta + x_\eta y_\zeta z_\xi + x_\zeta y_\xi z_\eta - x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\zeta - x_\zeta y_\eta z_\xi} \end{aligned} \tag{2.54}$$

and corresponds to the ratio of cell volume in the computational domain to that in the physical domain.

In addition, using the chain rule of partial differentiation, Eq. 2.49 becomes

$$\begin{bmatrix} \partial/\partial x \\ \partial/\partial y \\ \partial/\partial z \\ \partial/\partial t \end{bmatrix} = \begin{bmatrix} \xi_x & \eta_x & \zeta_x & 0 \\ \xi_y & \eta_y & \zeta_y & 0 \\ \xi_z & \eta_z & \zeta_z & 0 \\ \xi_t & \eta_t & \zeta_t & 1 \end{bmatrix} \begin{bmatrix} \partial/\partial \xi \\ \partial/\partial \eta \\ \partial/\partial \zeta \\ \partial/\partial \tau \end{bmatrix} \tag{2.55}$$

By substituting these partial derivatives into Eq. 2.45, the non-dimensional conservation form of the compressible Navier-Stokes equations in the general coordinate system are obtained:

$$\frac{\partial \hat{\boldsymbol{Q}}}{\partial \tau} + \frac{\partial \hat{\boldsymbol{E}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta} = \frac{1}{Re} \left(\frac{\partial \hat{\boldsymbol{E}}_{\boldsymbol{v}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}_{\boldsymbol{v}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}_{\boldsymbol{v}}}{\partial \zeta} \right) \tag{2.56}$$

where

$$\hat{\boldsymbol{Q}} = \frac{1}{J}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ e\end{bmatrix}, \quad \hat{\boldsymbol{E}} = \frac{1}{J}\begin{bmatrix}\rho U\\ \rho u U + \xi_x p\\ \rho v U + \xi_y p\\ \rho w U + \xi_z p\\ (e+p)\,U - \xi_t p\end{bmatrix}, \quad \hat{\boldsymbol{F}} = \frac{1}{J}\begin{bmatrix}\rho V\\ \rho u V + \eta_x p\\ \rho v V + \eta_y p\\ \rho w V + \eta_z p\\ (e+p)\,V - \eta_t p\end{bmatrix},$$

$$\hat{\boldsymbol{G}} = \frac{1}{J}\begin{bmatrix}\rho W\\ \rho u W + \zeta_x p\\ \rho v W + \zeta_y p\\ \rho w W + \zeta_z p\\ (e+p)\,W - \zeta_t p\end{bmatrix}, \quad \hat{\boldsymbol{E}}_{\boldsymbol{v}} = \frac{1}{J}\begin{bmatrix}0\\ \xi_x\tau_{xx} + \xi_y\tau_{yx} + \xi_z\tau_{zx}\\ \xi_x\tau_{xy} + \xi_y\tau_{yy} + \xi_z\tau_{zy}\\ \xi_x\tau_{xz} + \xi_y\tau_{yz} + \xi_z\tau_{zz}\\ \xi_x\beta_x + \xi_y\beta_y + \xi_z\beta_z\end{bmatrix}, \tag{2.57}$$

$$\hat{\boldsymbol{F}}_{\boldsymbol{v}} = \frac{1}{J}\begin{bmatrix}0\\ \eta_x\tau_{xx} + \eta_y\tau_{yx} + \eta_z\tau_{zx}\\ \eta_x\tau_{xy} + \eta_y\tau_{yy} + \eta_z\tau_{zy}\\ \eta_x\tau_{xz} + \eta_y\tau_{yz} + \eta_z\tau_{zz}\\ \eta_x\beta_x + \eta_y\beta_y + \eta_z\beta_z\end{bmatrix}, \quad \hat{\boldsymbol{G}}_{\boldsymbol{v}} = \frac{1}{J}\begin{bmatrix}0\\ \zeta_x\tau_{xx} + \zeta_y\tau_{yx} + \zeta_z\tau_{zx}\\ \zeta_x\tau_{xy} + \zeta_y\tau_{yy} + \zeta_z\tau_{zy}\\ \zeta_x\tau_{xz} + \zeta_y\tau_{yz} + \zeta_z\tau_{zz}\\ \zeta_x\beta_x + \zeta_y\beta_y + \zeta_z\beta_z\end{bmatrix}$$

U, V and W are the so-called contravariant velocity components in ξ, η and ζ directions, respectively:

$$\begin{aligned} U &= \xi_t + \xi_x u + \xi_y v + \xi_z w \\ V &= \eta_t + \eta_x u + \eta_y v + \eta_z w \\ W &= \zeta_t + \zeta_x u + \zeta_y v + \zeta_z w \end{aligned} \tag{2.58}$$

The metrics and Jacobian (Eqs. 2.52 and 2.54) evaluated in the finite volume fashion satisfy the geometric conservation law (GCL) in the discretized form of the Navier-Stokes equations.[24,65]

2.3.1.4 Thin-Layer Approximation

In high Reynolds number flow, one usually has only enough grid points to resolve viscous terms in a thin layer near body surfaces. Typically, grid lines are clustered near a body surface and resolution along the body surface is similar to what is needed in inviscid flow. Even though the full Navier-Stokes equations may be programmed, viscous derivatives along the body surface are not resolved in general unless the grid spacing along the body surface sufficiently small, in many cases of $O(1/\sqrt{Re})$ based on the effective viscous coefficient. Consequently, a thin-layer approximation approximation[64] is used: all viscous derivatives in ξ and η directions (along the body surface) are neglected, while terms in ζ direction are retained and the body surface is mapped onto $\zeta = \text{const}$ plane. Thus, Eq. 2.56 simplifies to the thin-layer

Navier-Stokes equations:

$$\frac{\partial \hat{\boldsymbol{Q}}}{\partial \tau} + \frac{\partial \hat{\boldsymbol{E}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta} = \frac{1}{Re}\frac{\partial \hat{\boldsymbol{G}}_{\boldsymbol{v}}}{\partial \zeta} \tag{2.59}$$

where

$$\hat{\boldsymbol{G}}_{\boldsymbol{v}} = \frac{1}{J}\begin{bmatrix} 0 \\ \mu m_1 \frac{\partial u}{\partial \zeta} + \frac{1}{3}\mu m_2 \zeta_x \\ \mu m_1 \frac{\partial v}{\partial \zeta} + \frac{1}{3}\mu m_2 \zeta_y \\ \mu m_1 \frac{\partial w}{\partial \zeta} + \frac{1}{3}\mu m_2 \zeta_z \\ \mu m_1 m_3 + \frac{1}{3}\mu m_2 \left(\zeta_x u + \zeta_y v + \zeta_z w\right) \end{bmatrix} \tag{2.60}$$

with

$$\begin{aligned} m_1 &= {\zeta_x}^2 + {\zeta_y}^2 + {\zeta_z}^2 \\ m_2 &= \zeta_x \frac{\partial u}{\partial \zeta} + \zeta_y \frac{\partial v}{\partial \zeta} + \zeta_z \frac{\partial w}{\partial \zeta} \\ m_3 &= \frac{1}{2}\frac{\partial}{\partial \zeta}\left(u^2 + v^2 + w^2\right) + \frac{1}{(\gamma - 1)\, Pr}\frac{\partial}{\partial \zeta}\left(a^2\right) \end{aligned} \tag{2.61}$$

In the thin-layer approximation, a boundary-layer-like coordinate is adopted and the viscous terms which are dropped in the boundary-layer theory are eliminated. It should be stressed that this approximation is valid only for high Reynolds number flows and very large turbulent viscosity coefficient could conceivably invalidate this approximation. In the present study, the compressible thin-layer Navier-Stokes equations are used as the governing equations of fluid dynamics.

2.3.2 Discretization of Governing Equations

A temporal and spatial discretization of governing equations is required to numerically model these equations whose form is partial differential. In Eq. 2.59, firstly, temporal difference of the conservative variable vector $\hat{\boldsymbol{Q}}$ is evaluated by the trapezoidal differencing using λ, as follow:

$$\hat{\boldsymbol{Q}}^{n+1} - \hat{\boldsymbol{Q}}^{n} = \lambda \boldsymbol{R}(\hat{\boldsymbol{Q}}^{n+1}) + (1 - \lambda)\, \boldsymbol{R}(\hat{\boldsymbol{Q}}^{n}) \tag{2.62}$$

where

$$\boldsymbol{R}(\hat{\boldsymbol{Q}}) = -\Delta t \left(\frac{\partial \hat{\boldsymbol{E}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta} - \frac{1}{Re}\frac{\partial \hat{\boldsymbol{G}}_{\boldsymbol{v}}}{\partial \zeta} \right) \tag{2.63}$$

with $\Delta\tau = \Delta t$ for simplicity. The temporal accuracy of this scheme varies with λ; $\lambda = 0$ yields the first-order Euler explicit scheme, $\lambda = 1/2$ yields the second-order Crank-Nicolson scheme, and $\lambda = 1$ yields the first-order Euler implicit scheme. Next,

spatial differences of the flux vectors $\hat{\boldsymbol{E}}$, $\hat{\boldsymbol{F}}$, $\hat{\boldsymbol{G}}$ and $\hat{\boldsymbol{G}}_{\boldsymbol{v}}$ in Eq. 2.63 are evaluated by the central differencing, as follows:

$$\begin{aligned} \left(\frac{\partial \hat{\boldsymbol{E}}}{\partial \xi}\right)_j &= \tilde{\boldsymbol{E}}_{j+1/2} - \tilde{\boldsymbol{E}}_{j-1/2}, & \left(\frac{\partial \hat{\boldsymbol{F}}}{\partial \eta}\right)_k &= \tilde{\boldsymbol{F}}_{k+1/2} - \tilde{\boldsymbol{F}}_{k-1/2}, \\ \left(\frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta}\right)_l &= \tilde{\boldsymbol{G}}_{l+1/2} - \tilde{\boldsymbol{G}}_{l-1/2}, & \left(\frac{\partial \hat{\boldsymbol{G}}_{\boldsymbol{v}}}{\partial \zeta}\right)_l &= \tilde{\boldsymbol{G}}_{\boldsymbol{v}l+1/2} - \tilde{\boldsymbol{G}}_{\boldsymbol{v}l-1/2} \end{aligned} \tag{2.64}$$

where $\Delta\xi = \Delta\eta = \Delta\zeta = 1$ for simplicity. $\tilde{\bullet}$ denotes the numerical fluxes defined at the cell interface. The spatial accuracy of this scheme depends on the evaluation of the numerical fluxes.

2.3.3 Evaluation of Numerical Fluxes

2.3.3.1 Inviscid Terms

Flux difference splitting (FDS) by Roe[66] and flux vector splitting (FVS) by van Leer[67] are widely used for the evaluation of inviscid terms of numerical fluxes $\tilde{\boldsymbol{E}}$, $\tilde{\boldsymbol{F}}$ and $\tilde{\boldsymbol{G}}$ in Eq. 2.64, in the simulations of compressible flows. The FDS scheme gives good resolutions to both shock waves and boundary layers because the numerical dissipation which is produced by the upwinding becomes small in boundary layers.[68] This scheme is suitable for the simulation of viscous flows, but less robust and suffers from numerical instability at the strong compression and expansion. Other way of introducing the upwinding is the FVS. This scheme is more robust against the strong compression and expansion and needs less computation time than the FDS. This scheme, however, is not suitable for the simulation of viscous flows because it produces excessive numerical dissipation in boundary layers. There is another effort to develop less dissipative upwind schemes in which the excessive dissipation of van Leer's FVS scheme is reduced by introducing the flavor of Roe's FDS into the FVS scheme. Liou and Steffen proposed a scheme named advection upstream splitting method (AUSM).[69] The AUSM scheme not only produces small numerical dissipation in boundary layers but also is robust enough to calculate the strong compression and expansion. The AUSM, however, bears a slight numerical overshoot at the shock wave. Inspired by the AUSM scheme, therefore, many schemes have been proposed due to overcome the slight disadvantage of AUSM scheme. One of them is the simple high-resolution upwind scheme (SHUS) by Shima and Jounouchi.[70] In the present study, the inviscid terms of numerical fluxes are evaluated by the SHUS scheme which is a family of AUSM type schemes with sufficient grid resolution to avoid numerical

instability at the strong compression and expansion while maintaining the resolution of the shock wave and the boundary layer.

The AUSM scheme splits the inviscid term of numerical flux, *e.g.* $\tilde{\boldsymbol{E}}$, into two parts; one from the contribution of the convective term $\tilde{\boldsymbol{E}}_{\boldsymbol{c}}$, and the other from the pressure term $\tilde{\boldsymbol{P}}$ as follow:

$$\tilde{\boldsymbol{E}} = \tilde{\boldsymbol{E}}_{\boldsymbol{c}} + \tilde{\boldsymbol{P}} \tag{2.65}$$

where

$$\begin{aligned} \boldsymbol{E}_{\boldsymbol{c}} &= \left|\boldsymbol{S}_{\boldsymbol{\xi}}\right| \rho V_n \boldsymbol{\Phi} \\ \boldsymbol{P} &= \left|\boldsymbol{S}_{\boldsymbol{\xi}}\right| p \boldsymbol{N} \end{aligned} \tag{2.66}$$

with

$$\begin{aligned} V_n &= n_x u + n_y v + n_z w = \frac{U}{\sqrt{{\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2}} \\ \boldsymbol{\Phi} &= \begin{bmatrix} 1 \\ u \\ v \\ w \\ H \end{bmatrix}, \quad \boldsymbol{N} = \begin{bmatrix} 0 \\ n_x \\ n_y \\ n_z \\ 0 \end{bmatrix}, \quad \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \frac{1}{\sqrt{{\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2}} \begin{bmatrix} \xi_x \\ \xi_y \\ \xi_z \end{bmatrix} \end{aligned} \tag{2.67}$$

Although only numerical flux in ξ direction is presented in Eq. 2.65, similar procedure can be applied to the numerical fluxes in the other directions η and ζ. $\boldsymbol{E}_{\boldsymbol{c}}$ indicates the convection of passive scalar quantities in $\boldsymbol{\Phi}$ transported by the mass flux ρV_n, therefore $\tilde{\boldsymbol{E}}_{\boldsymbol{c}}$ is evaluated by the upwind differencing:

$$\begin{aligned} \tilde{\boldsymbol{E}}_{\boldsymbol{c}} &= \left|\boldsymbol{S}_{\boldsymbol{\xi}}\right| \left[\frac{\tilde{V}_n + \left|\tilde{V}_n\right|}{2} (\rho \boldsymbol{\Phi})^L + \frac{\tilde{V}_n - \left|\tilde{V}_n\right|}{2} (\rho \boldsymbol{\Phi})^R \right] \\ &= \left|\boldsymbol{S}_{\boldsymbol{\xi}}\right| \left[\frac{\tilde{M}_n + \left|\tilde{M}_n\right|}{2} (\rho a \boldsymbol{\Phi})^L + \frac{\tilde{M}_n - \left|\tilde{M}_n\right|}{2} (\rho a \boldsymbol{\Phi})^R \right] \end{aligned} \tag{2.68}$$

where $M_n = V_n / a$. $\bullet^L$ and $\bullet^R$ denote the physical states at the left and right side of the cell interface, respectively. According to van Leer's FVS, $\tilde{M}_n$ is defined as follow:

$$\begin{aligned} \tilde{M}_n &= {M_n}_+^L + {M_n}_-^R \\ M_{n\pm} &= \begin{cases} \pm \left(M_n \pm 1\right)^2 / 4 & \text{if } \left|M_n\right| \leq 1 \\ \left(M_n \pm \left|M_n\right|\right) / 2 & \text{otherwise} \end{cases} \end{aligned} \tag{2.69}$$

Similarly, $\tilde{\boldsymbol{P}}$ is defined as follow:

$$\tilde{\boldsymbol{P}} = \left|\boldsymbol{S}_{\boldsymbol{\xi}}\right| \tilde{p} \boldsymbol{N} \tag{2.70}$$

where

$$\tilde{p} = \beta_+^L p^L + \beta_-^R p^R$$
$$\beta_\pm = \begin{cases} (M_n \pm 1)^2 (2 \mp M_n)/4 & \text{if } |M_n| \leq 1 \\ (M_n \pm |M_n|)/(2|M_n|) & \text{otherwise} \end{cases} \tag{2.71}$$

The SHUS scheme evaluates the convection term $\tilde{\boldsymbol{E}}_{\boldsymbol{c}}$ as follow:

$$\tilde{\boldsymbol{E}}_{\boldsymbol{c}} = |\boldsymbol{S}_{\boldsymbol{\xi}}| \left(\frac{\tilde{m} + |\tilde{m}|}{2} \boldsymbol{\Phi}^L + \frac{\tilde{m} - |\tilde{m}|}{2} \boldsymbol{\Phi}^R \right) \tag{2.72}$$

where m is the mass flux: $m = \rho V_n$, and $\tilde{m}$ is evaluated at the cell interface. The difference between the other AUSM type schemes is the expression of $\tilde{m}$. $\tilde{m}$ of the SHUS is given by Roe's FDS scheme using the finite difference of primitive variables as follow:

$$\begin{aligned} \tilde{m} = & \frac{1}{2} \Bigg[(\rho V_n)^L + (\rho V_n)^R - |\bar{V}_n| \Delta\rho - \frac{|\bar{M}+1| - |\bar{M}-1|}{2} \bar{\rho} \Delta V_n \\ & - \frac{|\bar{M}+1| + |\bar{M}-1| - 2|\bar{M}|}{2} \frac{\Delta p}{\bar{a}} \Bigg] \end{aligned} \tag{2.73}$$

where $\Delta\bullet = \bullet^R - \bullet^L$ and

$$\begin{aligned} & \bar{\rho} = \frac{\rho^L + \rho^R}{2}, \quad \bar{p} = \frac{p^L + p^R}{2}, \quad \bar{V}_n = \frac{{V_n}^L + {V_n}^R}{2}, \\ & \bar{a} = \sqrt{\frac{\gamma \bar{p}}{\bar{\rho}}}, \quad \bar{M} = \frac{\bar{V}_n}{\bar{a}} \end{aligned} \tag{2.74}$$

$\bar{\rho}$, $\bar{p}$ and $\bar{V}_n$ are arithmetic average of normal density, pressure and velocity, respectively. In the SHUS scheme, the pressure term $\tilde{\boldsymbol{P}}$ is evaluated by Eqs. 2.70 and 2.71.

The SHUS scheme (Eq.2.72) evaluates the convective term $\tilde{\boldsymbol{E}}_{\boldsymbol{c}}$ by averaging the mass flux m, while other AUSM type schemes (Eq.2.68) evaluate $\tilde{\boldsymbol{E}}_{\boldsymbol{c}}$ by averaging the Mach number M_n or velocity V_n. Therefore, the difference between the SHUS and other AUSM type schemes is how to handle the density ρ. The difference in the behavior between SHUS scheme and other AUSM type schemes can be clearly understood considering the response of the mass flux $\tilde{m}$ (Eq. 2.73) to the pressure difference Δp. $\tilde{m}$ of other AUSM type schemes is zero if the convective velocity $\tilde{V}_n$ is zero. In the SHUS scheme, on the other hand, the term proportional to Δp exists in $\tilde{m}$. Therefore, the SHUS scheme can avoid a numerical overshoot at shock wave which occurs on other AUSM type schemes.

In addition, the physical properties at the both sides of the cell interface are evaluated by the monotone upstream-centered schemes for conservation law (MUSCL)

interpolation[71,72] based on the primitive variables with van Albada differentiable limiter[67] to achieve a high-order spatial accuracy, as follow:

$$\begin{aligned} \boldsymbol{q}^L_{j+1/2} &= \boldsymbol{q}_j + \frac{s}{4}\left[(1-\kappa s)\,\boldsymbol{\Delta}_- + (1+\kappa s)\,\boldsymbol{\Delta}_+\right]_j \\ \boldsymbol{q}^R_{j+1/2} &= \boldsymbol{q}_{j+1} - \frac{s}{4}\left[(1-\kappa s)\,\boldsymbol{\Delta}_+ + (1+\kappa s)\,\boldsymbol{\Delta}_-\right]_{j+1} \end{aligned} \tag{2.75}$$

where $\boldsymbol{q}$ represents the primitive variable vector $\boldsymbol{q} = (\rho, u, v, w, p)^T$, and

$$\begin{aligned} s &= \frac{2\Delta_+\Delta_- + \epsilon}{{\Delta_+}^2 + {\Delta_-}^2 + \epsilon} \\ \boldsymbol{\Delta}_{+j} &= \boldsymbol{q}_{j+1} - \boldsymbol{q}_j \\ \boldsymbol{\Delta}_{-j} &= \boldsymbol{q}_j - \boldsymbol{q}_{j-1} \end{aligned} \tag{2.76}$$

The spatial accuracy of the scheme varies with the value of κ. $\kappa = 1$ yields the second-order central difference scheme, $\kappa = -1$ yields the second-order fully-upwind scheme, and $\kappa = 1/3$ yields the third-order upwind-biased scheme on an evenly spaced grid. In the present study, κ is set to be 1/3. ϵ is a very small number (10^{-6}) to prevent division by zero in regions of null gradients, such as the uniform region where $\Delta_+ = \Delta_- = 0$.

2.3.3.2 Viscous Terms

In the present study, spatial difference of the viscous flux vector $\boldsymbol{G_v}$ in Eq. 2.63 is evaluated by the second-order central differencing. That is, the inviscid term of numerical flux $\tilde{\boldsymbol{G}}_{\boldsymbol{v}}$ is given by simply averaging and differencing the physical properties at both grid points next to the cell interface.

2.3.4 Time Integration

There are mainly two choices of the time integration scheme; one is the explicit time integration method, and the other is the implicit time integration method. Even the simplest Euler explicit scheme has at least the first-order temporal accuracy, and it is easy to extend temporal accuracy of the explicit scheme using Runge-Kutta method. However, the time step size Δt for the explicit scheme is strongly restricted by the local Courant-Friedrichs-Lewy (CFL) number which indicates stability of the time integration scheme, and the restriction becomes very strict when the grid points are clustered near the wall to resolve the boundary layer. Consequently, this restriction derives the long total simulation time until convergence. Therefore, from the viewpoint of aerodynamic optimization which needs many evaluations of aerodynamic

performance, the use of implicit scheme is desirable to overcome the restriction of Δt and shorten the simulation time required to evaluate the aerodynamic performance.

2.3.4.1 LU-ADI Factorization Algorithm

The lower-upper alternate directional implicit (LU-ADI) factorization algorithm[73] is one of implicit time integration methods. The flowfields considered in the present study are essentially steady, so the inner-iteration method such as Newton-Raphson iteration method is not used.

When λ in Eq. 2.62 is set to be 1, the equations can be rewritten as

$$\hat{\boldsymbol{Q}}^{n+1} - \hat{\boldsymbol{Q}}^{n} = -\Delta t \left(\frac{\partial \hat{\boldsymbol{E}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta} \right)^{n+1} \tag{2.77}$$

where viscous term $Re^{-1}\partial \hat{\boldsymbol{G}}_{\boldsymbol{v}}/\partial \zeta$ is neglected as treating explicitly, as described in Subsubsec. 2.3.4.3. Then, locally linearize the inviscid fluxes $\hat{\boldsymbol{E}}$, $\hat{\boldsymbol{F}}$ and $\hat{\boldsymbol{G}}$,[74] such as

$$\hat{\boldsymbol{E}}^{n+1} = \hat{\boldsymbol{E}}^{n} + \left(\frac{\partial \hat{\boldsymbol{E}}}{\partial \hat{\boldsymbol{Q}}} \right)^{n} \left(\hat{\boldsymbol{Q}}^{n+1} - \hat{\boldsymbol{Q}}^{n} \right) + O(\Delta t^2) \tag{2.78}$$

and Eq 2.77 can be written as follow:

$$\begin{aligned} \left[\boldsymbol{I} + \Delta t \left(\frac{\partial \hat{\boldsymbol{A}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{B}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{C}}}{\partial \zeta} \right) \right]^{n} \Delta \hat{\boldsymbol{Q}}^{n} &= -\Delta t \left(\frac{\partial \hat{\boldsymbol{E}}}{\partial \xi} + \frac{\partial \hat{\boldsymbol{F}}}{\partial \eta} + \frac{\partial \hat{\boldsymbol{G}}}{\partial \zeta} \right)^{n} \\ &= \boldsymbol{R}^{n} \end{aligned} \tag{2.79}$$

where $\Delta \hat{\boldsymbol{Q}}^{n} = \hat{\boldsymbol{Q}}^{n+1} - \hat{\boldsymbol{Q}}^{n}$ and $\boldsymbol{R}^{n}$ is defined by Eq. 2.63. $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$ and $\hat{\boldsymbol{C}}$ are the Jacobian matrices of the inviscid fluxes:

$$\hat{\boldsymbol{A}} = \frac{\partial \hat{\boldsymbol{E}}}{\partial \hat{\boldsymbol{Q}}}, \quad \hat{\boldsymbol{B}} = \frac{\partial \hat{\boldsymbol{F}}}{\partial \hat{\boldsymbol{Q}}}, \quad \hat{\boldsymbol{C}} = \frac{\partial \hat{\boldsymbol{G}}}{\partial \hat{\boldsymbol{Q}}} \tag{2.80}$$

and given by

$$\begin{aligned} &\hat{\boldsymbol{A}} \text{ or } \hat{\boldsymbol{B}} \text{ or } \hat{\boldsymbol{C}} \\ &= \left[\begin{array}{ccccc} \kappa_t & \kappa_x & \kappa_y & \kappa_z & 0 \\ \kappa_x \phi^2 - u\theta & \kappa_t + \theta - (\gamma - 2)\,\kappa_x u & \kappa_y u - (\gamma - 1)\,\kappa_x v & \kappa_z u - (\gamma - 1)\,\kappa_x w & (\gamma - 1)\,\kappa_x \\ \kappa_y \phi^2 - v\theta & \kappa_x v - (\gamma - 1)\,\kappa_y u & \kappa_t + \theta - (\gamma - 2)\,\kappa_y v & \kappa_z v - (\gamma - 1)\,\kappa_y w & (\gamma - 1)\,\kappa_y \\ \kappa_z \phi^2 - w\theta & \kappa_x w - (\gamma - 1)\,\kappa_z u & \kappa_y w - (\gamma - 1)\,\kappa_z v & \kappa_t + \theta - (\gamma - 2)\,\kappa_z w & (\gamma - 1)\,\kappa_z \\ -\theta \left(\frac{\gamma e}{\rho} - 2\phi^2 \right) & \kappa_x \left(\frac{\gamma e}{\rho} - \phi^2 \right) - (\gamma - 1)\,\theta u & \kappa_y \left(\frac{\gamma e}{\rho} - \phi^2 \right) - (\gamma - 1)\,\theta v & \kappa_z \left(\frac{\gamma e}{\rho} - \phi^2 \right) - (\gamma - 1)\,\theta w & \kappa_t + \gamma\theta \end{array} \right] \end{aligned} \tag{2.81}$$

where

$$\begin{aligned}\theta &= \kappa_x u + \kappa_y v + \kappa_z w \\ \phi^2 &= \frac{1}{2}(\gamma - 1)\left(u^2 + v^2 + w^2\right)\end{aligned} \tag{2.82}$$

with $\kappa = \xi$, η or ζ for $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$ or $\hat{\boldsymbol{C}}$, respectively. The implicit operator (ADI operator) inside [•] of the left hand side of Eq. 2.79 is sparse but block non-band matrix, and it is a tough work to inverse it. The implicit operator of Eq. 2.79 can be factored by the approximate factorization[74] as

$$\left(\boldsymbol{I} + \Delta t \frac{\partial \hat{\boldsymbol{A}}}{\partial \xi}\right)^n \left(\boldsymbol{I} + \Delta t \frac{\partial \hat{\boldsymbol{B}}}{\partial \eta}\right)^n \left(\boldsymbol{I} + \Delta t \frac{\partial \hat{\boldsymbol{C}}}{\partial \zeta}\right)^n \Delta \hat{\boldsymbol{Q}}^n = \boldsymbol{R}^n \tag{2.83}$$

Now, each implicit operator reduces to block tridiagonal matrix when the spatial differences are evaluated by the second-order central differencing. Thus, the equations can be written as

$$\begin{aligned}&\left(\boldsymbol{I} + h\delta_\xi \hat{\boldsymbol{A}} - \epsilon_i \boldsymbol{J}^{-1} \nabla_\xi \Delta_\xi \boldsymbol{J}\right)^n \left(\boldsymbol{I} + h\delta_\eta \hat{\boldsymbol{B}} - \epsilon_i \boldsymbol{J}^{-1} \nabla_\eta \Delta_\eta \boldsymbol{J}\right)^n \\ &\times \left(\boldsymbol{I} + h\delta_\zeta \hat{\boldsymbol{C}} - \epsilon_i \boldsymbol{J}^{-1} \nabla_\zeta \Delta_\zeta \boldsymbol{J}\right)^n \Delta \hat{\boldsymbol{Q}}^n \\ =&\boldsymbol{R}^n\end{aligned} \tag{2.84}$$

where $h = \Delta t/\Delta\xi = \Delta t/\Delta\eta = \Delta t/\Delta\zeta = \Delta t$. $\delta_\kappa \bullet$ denotes the second-order central difference operator, and $\Delta_\kappa \bullet$ and $\nabla_\kappa \bullet$ denote the first-order forward and backward difference operators, respectively. The term including ϵ_i inserted into each implicit operator in the left hand side of Eq. 2.84 is the implicit numerical dissipation term to maintain the numerical stability of the simulation. Next, each implicit operator can be written in the diagonal form,[75] *e.g.* for ξ direction:

$$\begin{aligned}\boldsymbol{I} + h\delta_\xi \hat{\boldsymbol{A}} - \epsilon_i \boldsymbol{J}^{-1} \nabla_\xi \Delta_\xi \boldsymbol{J} &= \boldsymbol{I} + h\delta_\xi \left(\boldsymbol{T_\xi} \boldsymbol{\Lambda_\xi} \boldsymbol{T_\xi}^{-1}\right) - \epsilon_i \boldsymbol{J}^{-1} \nabla_\xi \Delta_\xi \boldsymbol{J} \\ &\doteq \boldsymbol{T_\xi} \left(\boldsymbol{I} + h\delta_\xi \boldsymbol{\Lambda_\xi} - \epsilon_i \boldsymbol{J}^{-1} \nabla_\xi \Delta_\xi \boldsymbol{J}\right) \boldsymbol{T_\xi}^{-1}\end{aligned} \tag{2.85}$$

where $\hat{\boldsymbol{A}} = \boldsymbol{T_\xi} \boldsymbol{\Lambda_\xi} \boldsymbol{T_\xi}^{-1}$. $\boldsymbol{\Lambda_\xi}$ is the diagonal matrix of eigenvalues of $\hat{\boldsymbol{A}}$, and $\boldsymbol{T_\xi}$ is the matrix of right eigenvectors of $\hat{\boldsymbol{A}}$. Similar diagonalization can be applied to the other implicit operators: $\hat{\boldsymbol{B}} = \boldsymbol{T_\eta} \boldsymbol{\Lambda_\eta} \boldsymbol{T_\eta}^{-1}$ and $\hat{\boldsymbol{C}} = \boldsymbol{T_\zeta} \boldsymbol{\Lambda_\zeta} \boldsymbol{T_\zeta}^{-1}$. $\boldsymbol{\Lambda_\xi}$, $\boldsymbol{\Lambda_\eta}$ and $\boldsymbol{\Lambda_\zeta}$ are given by

$$\begin{aligned}\boldsymbol{\Lambda_\xi} &= \mathrm{diag}\left[U, U, U, U + a\sqrt{{\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2}, U - a\sqrt{{\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2}\right] \\ \boldsymbol{\Lambda_\eta} &= \mathrm{diag}\left[V, V, V, V + a\sqrt{{\eta_x}^2 + {\eta_y}^2 + {\eta_z}^2}, V - a\sqrt{{\eta_x}^2 + {\eta_y}^2 + {\eta_z}^2}\right] \\ \boldsymbol{\Lambda_\zeta} &= \mathrm{diag}\left[W, W, W, W + a\sqrt{{\zeta_x}^2 + {\zeta_y}^2 + {\zeta_z}^2}, W - a\sqrt{{\zeta_x}^2 + {\zeta_y}^2 + {\zeta_z}^2}\right]\end{aligned} \tag{2.86}$$

$\boldsymbol{T_\kappa}$ and $\boldsymbol{T_\kappa}^{-1}$ are given by

$$
\begin{aligned}
&\boldsymbol{T_\kappa} \\
&= \begin{bmatrix}
\tilde{\kappa}_x & \tilde{\kappa}_y & \tilde{\kappa}_z & \alpha & \alpha \\
\tilde{\kappa}_x u & \tilde{\kappa}_y u - \rho\tilde{\kappa}_z & \tilde{\kappa}_z u + \rho\tilde{\kappa}_y & \alpha\left(u + \tilde{\kappa}_x a\right) & \alpha\left(u - \tilde{\kappa}_x a\right) \\
\tilde{\kappa}_x v + \rho\tilde{\kappa}_z & \tilde{\kappa}_y v & \tilde{\kappa}_z v - \rho\tilde{\kappa}_x & \alpha\left(v + \tilde{\kappa}_y a\right) & \alpha\left(v - \tilde{\kappa}_y a\right) \\
\tilde{\kappa}_x w - \rho\tilde{\kappa}_y & \tilde{\kappa}_y w + \rho\tilde{\kappa}_x & \tilde{\kappa}_z w & \alpha\left(w + \tilde{\kappa}_z a\right) & \alpha\left(w - \tilde{\kappa}_z a\right) \\
\left[\frac{\tilde{\kappa}_x\phi^2}{\gamma-1} + \rho\left(\tilde{\kappa}_z v - \tilde{\kappa}_y w\right)\right] & \left[\frac{\tilde{\kappa}_y\phi^2}{\gamma-1} + \rho\left(\tilde{\kappa}_x w - \tilde{\kappa}_z u\right)\right] & \left[\frac{\tilde{\kappa}_z\phi^2}{\gamma-1} + \rho\left(\tilde{\kappa}_y u - \tilde{\kappa}_x v\right)\right] & \alpha\left(\frac{\phi^2+a^2}{\gamma-1} + \tilde{\theta}a\right) & \alpha\left(\frac{\phi^2+a^2}{\gamma-1} - \tilde{\theta}a\right)
\end{bmatrix} \\
&\boldsymbol{T_\kappa}^{-1} \\
&= \begin{bmatrix}
\tilde{\kappa}_x\left(1 - \frac{\phi^2}{a^2}\right) - \frac{\tilde{\kappa}_z v - \tilde{\kappa}_y w}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_x u}{a^2} & \frac{(\gamma-1)\tilde{\kappa}_x v}{a^2} + \frac{\tilde{\kappa}_z}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_x w}{a^2} - \frac{\tilde{\kappa}_y}{\rho} & -\frac{(\gamma-1)\tilde{\kappa}_x}{a^2} \\
\tilde{\kappa}_y\left(1 - \frac{\phi^2}{a^2}\right) - \frac{\tilde{\kappa}_x w - \tilde{\kappa}_z u}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_y u}{a^2} - \frac{\tilde{\kappa}_z}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_y v}{a^2} & \frac{(\gamma-1)\tilde{\kappa}_y w}{a^2} + \frac{\tilde{\kappa}_x}{\rho} & -\frac{(\gamma-1)\tilde{\kappa}_y}{a^2} \\
\tilde{\kappa}_z\left(1 - \frac{\phi^2}{a^2}\right) - \frac{\tilde{\kappa}_y u - \tilde{\kappa}_x v}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_z u}{a^2} + \frac{\tilde{\kappa}_y}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_z v}{a^2} - \frac{\tilde{\kappa}_x}{\rho} & \frac{(\gamma-1)\tilde{\kappa}_z w}{a^2} & -\frac{(\gamma-1)\tilde{\kappa}_z}{a^2} \\
\beta\left(\phi^2 - \tilde{\theta}a\right) & -\beta\left[(\gamma-1)u - \tilde{\kappa}_x a\right] & -\beta\left[(\gamma-1)v - \tilde{\kappa}_y a\right] & -\beta\left[(\gamma-1)w - \tilde{\kappa}_z a\right] & (\gamma-1)\beta \\
\beta\left(\phi^2 + \tilde{\theta}a\right) & -\beta\left[(\gamma-1)u + \tilde{\kappa}_x a\right] & -\beta\left[(\gamma-1)v + \tilde{\kappa}_y a\right] & -\beta\left[(\gamma-1)w + \tilde{\kappa}_z a\right] & (\gamma-1)\beta
\end{bmatrix}
\end{aligned}
\tag{2.87}
$$

where

$$
\begin{aligned}
&\alpha = \frac{\rho}{\sqrt{2}a}, \quad \beta = \frac{1}{\sqrt{2}\rho a} \\
&\tilde{\kappa}_x = \frac{\kappa_x}{\sqrt{{\kappa_x}^2 + {\kappa_y}^2 + {\kappa_z}^2}}, \quad \tilde{\kappa}_y = \frac{\kappa_y}{\sqrt{{\kappa_x}^2 + {\kappa_y}^2 + {\kappa_z}^2}}, \quad \tilde{\kappa}_z = \frac{\kappa_z}{\sqrt{{\kappa_x}^2 + {\kappa_y}^2 + {\kappa_z}^2}} \\
&\tilde{\theta} = \frac{\theta}{\sqrt{{\kappa_x}^2 + {\kappa_y}^2 + {\kappa_z}^2}}
\end{aligned}
\tag{2.88}
$$

and $\boldsymbol{T_\kappa}^{-1}\boldsymbol{T_\lambda}$ is given by

$$
\boldsymbol{T_\kappa}^{-1}\boldsymbol{T_\lambda} = \begin{bmatrix}
m_1 & m_2 & m_3 & -\mu m_4 & \mu m_4 \\
-m_2 & m_1 & m_4 & \mu m_3 & -\mu m_3 \\
-m_3 & -m_4 & m_1 & -\mu m_2 & \mu m_2 \\
\mu m_4 & -\mu m_3 & \mu m_2 & \mu^2\left(1 + m_1\right) & \mu^2\left(1 - m_1\right) \\
-\mu m_4 & \mu m_3 & -\mu m_2 & \mu^2\left(1 - m_1\right) & \mu^2\left(1 + m_1\right)
\end{bmatrix}
\tag{2.89}
$$

where

$$\begin{aligned}
\mu &= \frac{1}{\sqrt{2}} \\
m_1 &= \tilde{\kappa}_x\tilde{\lambda}_x + \tilde{\kappa}_y\tilde{\lambda}_y + \tilde{\kappa}_z\tilde{\lambda}_z \\
m_2 &= \tilde{\kappa}_x\tilde{\lambda}_y - \tilde{\kappa}_y\tilde{\lambda}_x, \quad m_3 = \tilde{\kappa}_x\tilde{\lambda}_z - \tilde{\kappa}_z\tilde{\lambda}_x, \quad m_4 = \tilde{\kappa}_y\tilde{\lambda}_z - \tilde{\kappa}_z\tilde{\lambda}_y
\end{aligned} \tag{2.90}$$

Then, $\delta_\xi \boldsymbol{\Lambda}_\xi$ in Eq. 2.85 can be decomposed to the forward and the backward differencings by introducing the idea of the FVS scheme:[67]

$$\boldsymbol{I} + h\delta_\xi\hat{\boldsymbol{A}} - \epsilon_i\boldsymbol{J}^{-1}\nabla_\xi\Delta_\xi\boldsymbol{J} = \boldsymbol{T_\xi}\left(\boldsymbol{I} + \nabla_\xi\boldsymbol{\Lambda_\xi}^+ + \Delta_\xi\boldsymbol{\Lambda_\xi}^-\right)\boldsymbol{T_\xi}^{-1} \tag{2.91}$$

where $\boldsymbol{\Lambda_\xi}^+$ and $\boldsymbol{\Lambda_\xi}^-$ are the diagonal matrices with positive and negative components of the eigenvalues of $\boldsymbol{A}$, respectively:

$$\boldsymbol{\Lambda_\xi}^\pm = \frac{h}{2}\left(\boldsymbol{\Lambda_\xi} \pm |\boldsymbol{\Lambda_\xi}|\right) \pm \bar{J}^{-1}\epsilon_i\boldsymbol{J} \tag{2.92}$$

$\bar{J}$ is taken to be the Jacobian at the central point corresponding to Eq. 2.85. The implicit operator for ξ direction can be rewritten as

$$\boldsymbol{I} + h\delta_\xi\hat{\boldsymbol{A}} - \epsilon_i\boldsymbol{J}^{-1}\nabla_\xi\Delta_\xi\boldsymbol{J} = \boldsymbol{T_\xi}\left(\boldsymbol{L_\xi} + \boldsymbol{M_\xi} + \boldsymbol{N_\xi}\right)\boldsymbol{T_\xi}^{-1} \tag{2.93}$$

where $\boldsymbol{L_\xi}$, $\boldsymbol{M_\xi}$ and $\boldsymbol{N_\xi}$ are the left-lower triangular, the diagonal and the right-upper triangular matrices, respectively, as

$$\begin{aligned}
\boldsymbol{L_\xi}_j &= -\boldsymbol{\Lambda_\xi}^+{}_{j-1} \\
\boldsymbol{M_\xi}_j &= \boldsymbol{I} + \boldsymbol{\Lambda_\xi}^+{}_j - \boldsymbol{\Lambda_\xi}^-{}_j \\
\boldsymbol{N_\xi}_j &= \boldsymbol{\Lambda_\xi}^-{}_{j+1}
\end{aligned} \tag{2.94}$$

Furthermore, $\boldsymbol{L_\xi} + \boldsymbol{M_\xi} + \boldsymbol{N_\xi}$ in the implicit operator of Eq. 2.93 can be factored by the diagonally dominant factorization:[76]

$$\boldsymbol{L_\xi} + \boldsymbol{M_\xi} + \boldsymbol{N_\xi} = \left(\boldsymbol{L_\xi} + \boldsymbol{M_\xi}\right)\boldsymbol{M_\xi}^{-1}\left(\boldsymbol{M_\xi} + \boldsymbol{N_\xi}\right) + O(h^2) \tag{2.95}$$

since $\boldsymbol{M_\xi} = O(1)$ and $\boldsymbol{L_\xi}, \boldsymbol{N_\xi} = O(h)$. Thus, the LU factorization for an ADI operator can be obtained as

$$\boldsymbol{I} + h\delta_\xi\hat{\boldsymbol{A}} - \epsilon_i\boldsymbol{J}^{-1}\nabla_\xi\Delta_\xi\boldsymbol{J} = \boldsymbol{T_\xi}\left(\boldsymbol{L_\xi} + \boldsymbol{M_\xi}\right)\boldsymbol{M_\xi}^{-1}\left(\boldsymbol{M_\xi} + \boldsymbol{N_\xi}\right)\boldsymbol{T_\xi}^{-1} \tag{2.96}$$

Here, note that the block tridiagonal matrix $\boldsymbol{I} + h\delta_\xi\hat{\boldsymbol{A}} - \epsilon_i\boldsymbol{J}^{-1}\nabla_\xi\Delta_\xi\boldsymbol{J}$ is decomposed to the block diagonal matrices $\boldsymbol{T_\xi}$ and $\boldsymbol{T_\xi}^{-1}$, scalar diagonal matrix $\boldsymbol{M_\xi}^{-1}$, and lower and upper scalar bidiagonal matrices $\boldsymbol{L_\xi} + \boldsymbol{M_\xi}$ and $\boldsymbol{M_\xi} + \boldsymbol{N_\xi}$. The inversion of the

implicit operator of Eq. 2.96 does not require any actual matrix inversion, but only requires one forward sweep for the inversion of $\boldsymbol{L_\xi}+\boldsymbol{M_\xi}$, one backward sweep for that of $\boldsymbol{M_\xi}^{-1}\left(\boldsymbol{M_\xi}+\boldsymbol{N_\xi}\right)$, and two matrix multiplications of $\boldsymbol{T_\xi}$ and $\boldsymbol{T_\xi}^{-1}$. Therefore, the LU-ADI scheme is very efficient.

Finally, the LU-ADI scheme can be described by using the similar decomposition for the implicit operators of the other directions η and ζ in Eq. 2.84, as follow:

$$\begin{aligned}&\boldsymbol{T_\xi}\left(\boldsymbol{L_\xi}+\boldsymbol{M_\xi}\right)\boldsymbol{M_\xi}^{-1}\left(\boldsymbol{M_\xi}+\boldsymbol{N_\xi}\right)\left(\boldsymbol{T_\xi}^{-1}\boldsymbol{T_\eta}\right)\left(\boldsymbol{L_\eta}+\boldsymbol{M_\eta}\right)\boldsymbol{M_\eta}^{-1}\left(\boldsymbol{M_\eta}+\boldsymbol{N_\eta}\right)\\&\times\left(\boldsymbol{T_\eta}^{-1}\boldsymbol{T_\zeta}\right)\left(\boldsymbol{L_\zeta}+\boldsymbol{M_\zeta}\right)\boldsymbol{M_\zeta}^{-1}\left(\boldsymbol{M_\zeta}+\boldsymbol{N_\zeta}\right)\boldsymbol{T_\zeta}^{-1}\Delta\hat{\boldsymbol{Q}}^n\\=&\boldsymbol{R}^n\end{aligned}\tag{2.97}$$

The LU-ADI scheme introduces several approximations to inverse the implicit operator efficiently, and its temporal accuracy is less than the first-order. However, this temporal accuracy is sufficient to resolve the almost steady flowfields considered in the present study.

2.3.4.2 ADI-SGS Factorization Algorithm

The ADI-SGS factorization algorithm is the time integration method based on the combination of the ideas of lower-upper alternate directional implicit (LU-ADI) scheme and lower-upper symmetric Gauss Seidel (LU-SGS) scheme.[77] The ADI-SGS scheme has both efficient characteristic involved by the LU-ADI scheme and stable characteristic involved by the LU-SGS scheme.

In the ADI-SGS scheme, we introduce the idea of the FVS scheme[67] into the implicit operators in Eq. 2.83 as

$$\begin{aligned}&\left[\boldsymbol{I}+\Delta t\left(\frac{\partial}{\partial\xi}\hat{\boldsymbol{A}}^+ + \frac{\partial}{\partial\xi}\hat{\boldsymbol{A}}^-\right)\right]^n\left[\boldsymbol{I}+\Delta t\left(\frac{\partial}{\partial\eta}\hat{\boldsymbol{B}}^+ + \frac{\partial}{\partial\eta}\hat{\boldsymbol{B}}^-\right)\right]^n\\&\times\left[\boldsymbol{I}+\Delta t\left(\frac{\partial}{\partial\zeta}\hat{\boldsymbol{C}}^+ + \frac{\partial}{\partial\zeta}\hat{\boldsymbol{C}}^-\right)\right]^n\Delta\hat{\boldsymbol{Q}}^n\\=&\boldsymbol{R}^n\end{aligned}\tag{2.98}$$

where $\hat{\boldsymbol{A}}^+$, $\hat{\boldsymbol{B}}^+$ and $\hat{\boldsymbol{C}}^+$ are the Jacobian matrices with positive components of the eigenvalues of $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$ and $\hat{\boldsymbol{C}}$, and $\hat{\boldsymbol{A}}^-$, $\hat{\boldsymbol{B}}^-$ and $\hat{\boldsymbol{C}}^-$ are those with negative components of these eigenvalues, respectively. Here, Eq. 2.98 can be rewritten by introducing the first-order upwind differencing as

$$\begin{aligned}&\left[\boldsymbol{I}+h\left(\nabla_\xi\hat{\boldsymbol{A}}^+ + \Delta_\xi\hat{\boldsymbol{A}}^-\right)\right]^n\left[\boldsymbol{I}+h\left(\nabla_\eta\hat{\boldsymbol{B}}^+ + \Delta_\eta\hat{\boldsymbol{B}}^-\right)\right]^n\\&\times\left[\boldsymbol{I}+h\left(\nabla_\zeta\hat{\boldsymbol{C}}^+ + \Delta_\zeta\hat{\boldsymbol{C}}^-\right)\right]^n\Delta\hat{\boldsymbol{Q}}^n\\=&\boldsymbol{R}^n\end{aligned}\tag{2.99}$$

where $h = \Delta t/\Delta\xi = \Delta t/\Delta\eta = \Delta t/\Delta\zeta = \Delta t$. $\Delta_\kappa\bullet$ and $\nabla_\kappa\bullet$ denote the first-order forward and backward difference operators, respectively. Next, each implicit operator in the left hand side of Eq. 2.99 can be factored by the lower-diagonal-upper (LDU) factorization, *e.g.* for ξ direction:

$$\begin{aligned} \boldsymbol{I} + h\left(\nabla_\xi \hat{\boldsymbol{A}}^+ + \Delta_\xi \hat{\boldsymbol{A}}^-\right) &= \boldsymbol{L_\xi} + \boldsymbol{D_\xi} + \boldsymbol{U_\xi} \\ &\doteq \left(\boldsymbol{L_\xi} + \boldsymbol{D_\xi}\right) \boldsymbol{D_\xi}^{-1} \left(\boldsymbol{D_\xi} + \boldsymbol{U_\xi}\right) \end{aligned} \tag{2.100}$$

where $\boldsymbol{L_\xi}$, $\boldsymbol{D_\xi}$ and $\boldsymbol{U_\xi}$ are the left-lower triangular, the diagonal and the right-upper triangular matrices, respectively, as

$$\begin{aligned} \boldsymbol{L_\xi} &= -h\hat{\boldsymbol{A}}^+{}_{j-1} \\ \boldsymbol{D_\xi} &= \boldsymbol{I} + h\left(\hat{\boldsymbol{A}}^+{}_j - \hat{\boldsymbol{A}}^-{}_j\right) \\ \boldsymbol{U_\xi} &= h\hat{\boldsymbol{A}}^-{}_{j+1} \end{aligned} \tag{2.101}$$

Though there exist several ways to evaluate $\hat{\boldsymbol{A}}^\pm$, following definition is much easier and cost effective:

$$\hat{\boldsymbol{A}}^\pm = \frac{\hat{\boldsymbol{A}} \pm \sigma_\xi \boldsymbol{I}}{2} \tag{2.102}$$

where σ_ξ is the maximum eigenvalue, so-called spectral radius, of $\hat{\boldsymbol{A}}$ as

$$\sigma_\xi = |U| + a\sqrt{{\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2} \tag{2.103}$$

In Eq. 2.100, note that the block tridiagonal matrix $\boldsymbol{I} + h\left(\nabla_\xi \hat{\boldsymbol{A}}^+ + \Delta_\xi \hat{\boldsymbol{A}}^-\right)$ is decomposed to the scalar diagonal matrix $\boldsymbol{D_\xi}^{-1}$, and lower and upper scalar bidiagonal matrices $\boldsymbol{L_\xi} + \boldsymbol{D_\xi}$ and $\boldsymbol{D_\xi} + \boldsymbol{U_\xi}$. The inversion of the implicit operator of Eq. 2.100 requires only one forward sweep for the inversion of $\boldsymbol{L_\xi} + \boldsymbol{D_\xi}$ and one backward sweep for that of $\boldsymbol{D_\xi}^{-1}\left(\boldsymbol{D_\xi} + \boldsymbol{U_\xi}\right)$.

Finally, the ADI-SGS scheme can be described by using the similar decomposition for the other implicit operators of the other directions η and ζ in Eq. 2.99, as follow:

$$\begin{aligned} &\left(\boldsymbol{L_\xi} + \boldsymbol{D_\xi}\right) \boldsymbol{D_\xi}^{-1} \left(\boldsymbol{D_\xi} + \boldsymbol{U_\xi}\right) \left(\boldsymbol{L_\eta} + \boldsymbol{D_\eta}\right) \boldsymbol{D_\eta}^{-1} \left(\boldsymbol{D_\eta} + \boldsymbol{U_\eta}\right) \\ &\times \left(\boldsymbol{L_\zeta} + \boldsymbol{D_\zeta}\right) \boldsymbol{D_\zeta}^{-1} \left(\boldsymbol{D_\zeta} + \boldsymbol{U_\zeta}\right) \Delta\hat{\boldsymbol{Q}}^n \\ =&\boldsymbol{R}^n \end{aligned} \tag{2.104}$$

The ADI-SGS scheme eliminates the need for the inversions of block diagonal matrices, as seen in the LU-ADI scheme, without using a diagonalization procedure, and can achieve further reduction of calculation processes compared to the LU-ADI scheme. In addition, the ADI-SGS scheme is more diagonally dominant, i.e., more stable than the LU-ADI scheme due to the approximation of the split Jacobian matrices with those spectral radii.

2.3.4.3 Treatment of Viscous Term

According to the conservation in the discretized form of the governing equations, viscous term $Re^{-1}\partial\hat{\boldsymbol{G}}_{\boldsymbol{v}}/\partial\zeta$ should be introduced to the right hand side of Eq. 2.77. The viscous term can be treated by defining and decomposing the Jacobian matrix of $\hat{\boldsymbol{G}}_{\boldsymbol{v}}$, and adding this matrix to the implicit operator as well as $\hat{\boldsymbol{E}}$, $\hat{\boldsymbol{F}}$ and $\hat{\boldsymbol{G}}$.[63,64] However, accurate formulation of the implicit operator is not required because various approximations are introduced to the implicit operator. For simplicity and efficiency, in the present study, the split Jacobian matrices $\hat{\boldsymbol{C}}^{\pm}$ are modified to $\hat{\boldsymbol{C}}_{\boldsymbol{\nu}}{}^{\pm}$ as

$$\begin{aligned}\hat{\boldsymbol{C}}_{\boldsymbol{\nu}}{}^{\pm} &= \hat{\boldsymbol{C}}^{\pm} \pm \nu \boldsymbol{I} \\ &= \boldsymbol{T}_{\boldsymbol{\zeta}} \left(\boldsymbol{\Lambda}_{\boldsymbol{\zeta}}{}^{\pm} \pm \nu \boldsymbol{I}\right) \boldsymbol{T}_{\boldsymbol{\zeta}}{}^{-1}\end{aligned} \tag{2.105}$$

where ν is the approximate operator of the viscous term introduced by Obayashi *et al.*,[78] which is similar to the MacCormack's implicit scheme,[79] as

$$\nu = \frac{2\mu\left(\zeta_x{}^2 + \zeta_y{}^2 + \zeta_z{}^2\right)}{Re\rho\Delta\zeta} \tag{2.106}$$

2.3.4.4 Space Variable Time Stepping

As described above, the restriction on available time step size Δt becomes very severe near the wall where the grid points are clustered to resolve the boundary layer. If Δt is set to be constant at all grid points, called physical time stepping technique, the governing equations must be solved with very small Δt at all grid points, and it results in very long total simulation time. Therefore, it is more effective to solve the governing equations by setting an appropriate available Δt at each grid point individually. The space variable time stepping technique changes the time step size at each grid point: $\Delta t_{j,k,l}$ according to the cell volume, as

$$\Delta t_{j,k,l} = \frac{\Delta t_{ref}}{1 + \sqrt{J_{j,k,l}}} \tag{2.107}$$

where Δt_{ref} is the reference time step size which is given as an input parameter. This technique reduces small $\Delta t_{j,k,l}$ at the grid point with small cell volume (large $J_{j,k,l}$) while large $\Delta t_{j,k,l}$ is maintained at the grid point with large cell volume (small $J_{j,k,l}$). Consequently, this technique can reduce the total simulation time until the convergence to steady flowfield.

2.3.4.5 Multigrid Method

Multigrid method has derived from the observation that many numerical methods for solving discretized partial differential equations are adept at smoothing high frequency error components. The early stage of convergence history shows a region of rapid error reduction by removing high frequency components. Then, however, a long region of slow error reduction corresponding to the slow expulsion of the low frequency modes follows. The multigrid method dissipates the lower frequency errors sufficiently using a sequence of grid levels. Grid level 1 is the finest grid from which coarser grid s can be formed successively by deleting every other grid line in all coordinate directions. On coarser grids, the fine-grid low frequency error components are resolved as higher frequencies due to the increased grid spacing, and thus they are damped effectively. When this correction is interpolated back to the fine grid, high frequency errors can be generated but they are damped quickly by the fine-grid smoothing iteration. An Example of sequence of grids is shown in Fig. 2.21

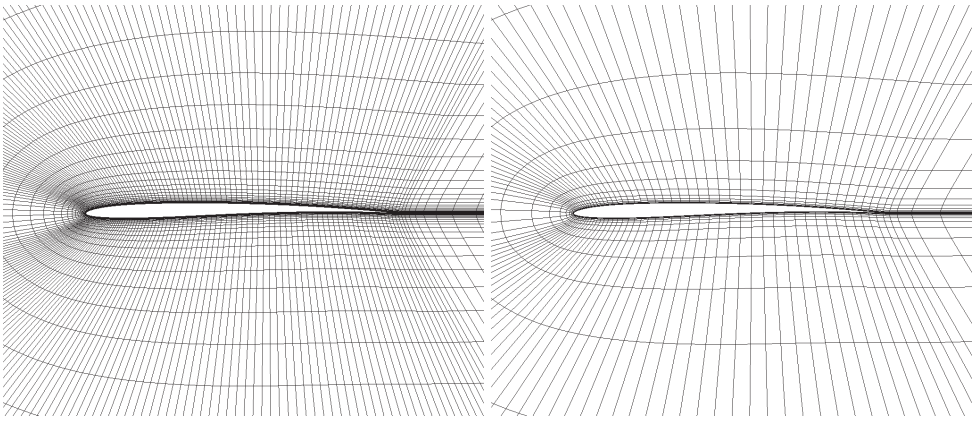
The multigrid method used in the present study is the full approximation scheme (FAS).[80], and the cycling strategy used in the present study is a V-cycle with three grid levels as shown in Fig. 2.22. At the each grid level, two time steps proceed.

The multigrid method can be illustrated by a grid sequence between two grid levels. The Euler or Navier-Stokes equations are solved approximately on a fine grid level h by discretizing the computational domain into cells yielding a system of algebraic equations as

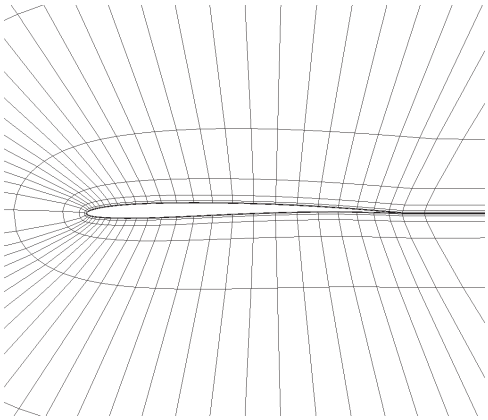
$$\boldsymbol{M}_h \Delta \hat{\boldsymbol{Q}}_h = \boldsymbol{R}_h(\hat{\boldsymbol{Q}}_h) \tag{2.108}$$

where $\boldsymbol{M}$ is the explicit or implicit relaxation operator matrix of the considered scheme, and $\boldsymbol{R}(\hat{\boldsymbol{Q}})$ is the explicit steady state (residual) vector corresponding the current conservative variable (dependent variable) vector $\hat{\boldsymbol{Q}}$. $\bullet_h$ indicates the matrices and vectors on grid level h. After calculating Eq. 2.108, the residuals and dependent variables on the first coarse grid level $h+1$ are calculated by restricting the corresponding fine-grid values. A volume weighted restriction operator $\boldsymbol{I}_h^{h+1}$ transfers values on the fine grid level h to the coarse grid level $h+1$ as

$$\begin{aligned} \boldsymbol{I}_h^{h+1}\hat{\boldsymbol{Q}}_h &= \frac{1}{J_{h+1}} \frac{\sum_{\in h} \hat{\boldsymbol{Q}}}{\sum_{\in h} 1/J} \\ \boldsymbol{I}_h^{h+1}\boldsymbol{R}_h &= \frac{1}{J_{h+1}} \frac{\sum_{\in h} \boldsymbol{R}}{\sum_{\in h} 1/J} \end{aligned} \tag{2.109}$$



(a) Level 1 (249 × 49). (b) Level 2 (125 × 25).



(c) Level 3 (63 × 13).

Figure 2.21 Example of grid sequence of multigrid method.

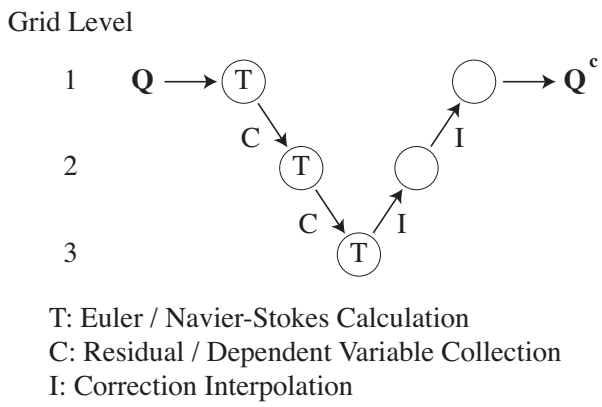


Figure 2.22 Flowchart of multigrid method with V-cycle strategy.

where the summations in Eq. 2.109 are taken over the entire fine grid cells which make up the coarse grid cell and J is the Jacobian of the fine-grid cell. This restriction operator is conservative and thus the surface integral of the fluxes crossing the cell interfaces on the coarse grid are the same as the corresponding integral on the fine grid. The relative truncation error vector $\boldsymbol{\tau}$ on the coarse grid is calculated from

$$\boldsymbol{\tau}_{h+1} = \boldsymbol{R}_{h+1}(\boldsymbol{I}_h^{h+1}\hat{\boldsymbol{Q}}_h) - \boldsymbol{I}_h^{h+1}\boldsymbol{R}_h \tag{2.110}$$

On the coarse grid, a few iterations of the approximate factorization scheme can be conducted by adding the relative errors as

$$\boldsymbol{M}_{h+1}\Delta\hat{\boldsymbol{Q}}_{h+1} = \boldsymbol{R}_{h+1}(\hat{\boldsymbol{Q}}_{h+1}) - \boldsymbol{\tau}_{h+1} \tag{2.111}$$

Because $\boldsymbol{\tau}_{h+1}$ has been previously calculated, the residuals $\boldsymbol{R}$ on the coarse grid is calculated easily by simply calculating $\boldsymbol{R}_{h+1}(\hat{\boldsymbol{Q}}_{h+1})$ from the most current values on the coarse grid and subtracting $\boldsymbol{\tau}_{h+1}$, as

$$\boldsymbol{R}_{h+1} = \boldsymbol{R}_{h+1}(\hat{\boldsymbol{Q}}_{h+1}) - \boldsymbol{\tau}_{h+1} \tag{2.112}$$

The correction $\boldsymbol{V}$ on the coarse grid is calculated as

$$\boldsymbol{V}_{h+1} = \hat{\boldsymbol{Q}}_{h+1} - \boldsymbol{I}_h^{h+1}\hat{\boldsymbol{Q}}_h \tag{2.113}$$

This correction is transferred to the fine grid by using trilinear interpolation operator $\boldsymbol{I}_{h+1}^h$ as

$$\hat{\boldsymbol{Q}}^c{}_h = \hat{\boldsymbol{Q}}_h + \boldsymbol{I}_{h+1}^h\boldsymbol{V}_{h+1} \tag{2.114}$$

The fine-grid solution is fully updated by one more smoothing iteration using Eq. 2.108.

Compared to the conventional time-marching scheme, the V-cycle strategy requires additional computations on coarser grids. However, on the first coarse grid level 2, the computational efforts are reduced to 1/8 of those on the finest grid level 1 in three-dimensional cases because the number of nodes in reduced to 1/2 in each coordinate direction. Thus, the computational cost on coarser grids is extremely small. Similarly, it becomes 1/64 on the grid level 3. Thus, the total computational cost of the present V-cycle strategy becomes $2 + 2/8 + 2/64 \doteq 2.3$ while six time steps are computed in one cycle.

Another merit of multigrid method is that the time step size limited by the CFD number can be doubled as the grid spacing is doubled from the previous grid. Then, the multigrid cycle proceeds the total time step size of $2\times\Delta t+2\times(2\Delta t)+2\times(4\Delta t) = 14\Delta t$ compared to $6\Delta t$ for the singlegrid method, where Δt is the time step size on the finest grid.

2.3.5 Turbulence Modeling

The flowfields where the Reynolds number is very high is considered to be mainly turbulent. The minimum length scale of turbulence is much larger than the length scale of molecular motion except for extreme cases such as the rarefied gas, therefore the turbulent flow can be treated as continuum motion and essentially simulated by using the unsteady Navier-Stokes equations. In practical case where the Reynolds number is high, however, the minimum scale of turbulence, called Kolmogoroff scale, is extremely small. Therefore, even if we use up-to-date computers, it is unpractical to resolve the minimum scale of turbulence directly because vast simulation time and vast memory size are required. And, statistical data processing of vast simulation results is also required to obtain useful information from the turbulent flowfields. As for the engineering statistical processing for turbulent flow, it is easier and more usual to time-average the governing equations in advance before the simulations to eliminate the irregular fluctuations associated with the turbulence because most engineers do not treat turbulence phenomena themselves directly in general. In the present study, therefore, the time-averaged governing equations are used for the simulations of turbulent flow.

2.3.5.1 Reynolds Averaging

In the turbulent flow, any physical variable $f(t)$ is decomposed into a time-averaged value $\overline{f}$ and a fluctuation $f'(t)$ around $\overline{f}$ as

$$f(t) \quad = \overline{f} + f'(t) \tag{2.115}$$

where $\overline{\bullet}$ denotes the time-averaged value called Reynolds average and defined, *e.g.* for f, as

$$\overline{f} = \frac{1}{\Delta t}\int_{t}^{t+\Delta t} f(t)\mathrm{d}t \tag{2.116}$$

Here, Δt is required to be large compared to the period of the random fluctuations associated with the turbulence, but small with respect to the time constant for any slow variations in the flowfield associated with ordinary unsteady flows. And, note that $\overline{f'} = 0$ by definition.

2.3.5.2 Favre Averaging

The Reynolds-averaging is reasonable for the incompressible Navier-Stokes equations. For the compressible Navier-Stokes equations, on the other hand, the Reynolds

averaging produces the correlation terms such as $\overline{\rho' u_j'}$ which can not be treated easily in the simulations, and the system of the governing equations is not closed. Therefore, mass-weighted time-averaging called Favre Averaging is convenient to eliminate the troublesome correlation terms. Similar to the Reynolds averaging, $f(t)$ is decomposed into a mass-weighted time-averaged value $\widetilde{f}$ and a fluctuation $f''(t)$ around $\widetilde{f}$ as

$$f(t) = \widetilde{f} + f''(t) \tag{2.117}$$

where $\widetilde{\bullet}$ denotes the Favre average and defined, *e.g.* for f, as

$$\widetilde{f} = \frac{\overline{\rho f}}{\overline{\rho}} \tag{2.118}$$

Here, note that $\overline{\rho f''} = 0$ by definition. Important differences between the Reynolds and the Favre averagings are; $\overline{f'} = 0$ and $\overline{\rho f'} \neq 0$ for the Reynolds averaging, and $\overline{f''} \neq 0$ and $\overline{\rho f''} = 0$ for the Favre averaging.

Applying the Favre averaging to the compressible Navier-Stokes equations, several correlation terms are produced. Among these terms, so-called Reynolds stress term $-\overline{\rho u_i'' u_j''}$ is modeled by introducing the turbulent viscosity coefficient μ_t, based on the Boussinesq approximation, as

$$-\overline{\rho u_i'' u_j''} = \mu_t \left(\frac{\partial \widetilde{u_i}}{\partial x_j} + \frac{\partial \widetilde{u_j}}{\partial x_i} \right) - \frac{2}{3} \delta_{ij} \frac{\partial \widetilde{u_k}}{\partial x_k} \tag{2.119}$$

and turbulent heat flux term $\overline{\rho u_j'' H''}$ is modeled by introducing the turbulent Prandtl number Pr_t as

$$\overline{\rho u_j'' H''} = -\frac{\mu_t}{(\gamma - 1)\, Pr_t} \frac{\partial}{\partial x_j} \left(\widetilde{a}^2 \right) \tag{2.120}$$

The other correlation terms are considered to be negligible in the present study.

Consequently, the Favre averaged compressible Navier-Stokes equations can be solved by simply replacing the dynamic viscosity coefficient μ and the Prandtl number Pr in Eq. 2.59 with

$$\begin{aligned} \mu &= \mu_l + \mu_t \\ \frac{\mu}{Pr} &= \frac{\mu_l}{Pr_l} + \frac{\mu_t}{Pr_t} \end{aligned} \tag{2.121}$$

μ_l and μ_t are the laminar and turbulent viscosity coefficients, respectively. In this study, μ_l is assumed to be constant to freestream value because we deal with only subsonic flow with small change of temperature, and μ_t is evaluated by the Baldwin-Lomax algebraic turbulence model. Pr_l and Pr_t are the laminar and turbulent Prandtl number and set to be 0.72 and 0.9, respectively.

2.3.5.3 Baldwin-Lomax Algebraic Turbulence Model

The Baldwin-Lomax algebraic turbulence model[81] is developed for the two-dimensional attached and steady boundary layer, and evaluates the turbulent viscosity coefficient μ_t from the velocity profile along the line normal to the wall. This model treats the wall boundary layer consisting of inner and outer regions where μ_t is defined as

$$\mu_t = \begin{cases} \mu_{t_{inner}} & \text{if } y \leq y_{crossover} \\ \mu_{t_{outer}} & \text{otherwise} \end{cases} \tag{2.122}$$

where y is the normal distance from the wall, and $y_{crossover}$ is defined as the smallest value of y at which $\mu_{t_{inner}} = \mu_{t_{outer}}$.

The Prandtl-Van Driest formulation is used for $\mu_{t_{inner}}$:

$$\mu_{t_{inner}} = \rho l^2 |\omega| \tag{2.123}$$

where

$$l = ky\left[1 - \exp\left(-y^+/A^+\right)\right] \tag{2.124}$$

$|\omega|$ is the magnitude of the vorticity:

$$|\omega| = \sqrt{\left(\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial z} - \frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial x} - \frac{\partial u}{\partial z}\right)^2} \tag{2.125}$$

and

$$y^+ = \frac{\sqrt{\rho_w \tau_w}}{\mu_w} y \tag{2.126}$$

$\bullet_w$ denotes the properties evaluated at the wall, and τ_w is the skin friction at the wall.

$\mu_{t_{outer}}$ is given by the Clauser formulation:

$$\mu_{t_{outer}} = K C_{cp} \rho F_{wake} F_{kleb}(y) \tag{2.127}$$

where K is the Clauser constant, C_{cp} is an additional constant, and

$$F_{wake} = \min\left(y_{max} F_{max}, C_{wk} y_{max} {u_{dif}}^2 / F_{max}\right) \tag{2.128}$$

The quantities y_{max} and F_{max} are determined from the function:

$$F(y) = y|\omega|\left[1 - \exp\left(-y^+/A^+\right)\right] \tag{2.129}$$

In wakes, the exponential term of this equation is set equal to zero. F_{max} is the maximum value of $F(y)$ that occurs in a profile, and y_{max} is the value of y at which it occurs. $F_{kleb}(y)$ is the Klebanoff intermittency factor given by

$$F_{kleb}(y) = \left[1 + 5.5\left(\frac{C_{kleb}y}{y_{max}}\right)^6\right]^{-1} \tag{2.130}$$

u_{dif} is the difference between maximum and minimum total velocity in the profile:

$$u_{dif} = \sqrt{u^2 + v^2 + w^2}_{max} - \sqrt{u^2 + v^2 + w^2}_{min} \tag{2.131}$$

For the flowfiled with an attached boundary layer, $|\omega|$ outside the boundary layer is small, and $F(y)$ takes its maximum F_{max} at the outer edge of the boundary layer, i.e., y_{max} represents the boundary layer thickness. In the Baldwin-Lomax algebraic turbulence model, therefore, the distribution of vorticity is used to determine length scale so that the necessity for finding the outer edge of the boundary layer is removed compared with the algebraic turbulence model by Cebeci.[82]

The constant appearing in the foregoing relations are determined by requiring agreement with the Cebeci formulation for constant pressure boundary layers at transonic speeds:

$$\begin{aligned} A^+ = 26, \quad C_{cp} = 1.6, \quad C_{kleb} = 0.3, \\ C_{wk} = 0.25, \quad k = 0.4, \quad K = 0.0168 \end{aligned} \tag{2.132}$$

2.3.6 Boundary Conditions

2.3.6.1 Solid Wall Boundary

Solid wall boundary on a body surface is treated as non-slip wall. When $l = 1$ plane is a body surface, the velocity components on $l = 1$ are given as

$$u_{l=1} = x_{\tau l=1}, \quad v_{l=1} = y_{\tau l=1}, \quad w_{l=1} = z_{\tau l=1} \tag{2.133}$$

where x_τ, y_τ and z_τ represent the velocities of the grid points on the body surface, and these values become zero for a stationary body surface, as considered in the present study. The density is extrapolated from the adjacent node:

$$\rho_{l=1} = \rho_{l=2} \tag{2.134}$$

Static pressure is defined from the equilibrium condition of the momentum normal to the body surface (normal momentum equation). In this case, ζ direction is normal

to the body surface, therefore the normal momentum equation is expressed in the generalized coordinate system as follows:

$$A\frac{\partial p}{\partial \xi} + B\frac{\partial p}{\partial \eta} + \frac{\partial p}{\partial \zeta} = C \tag{2.135}$$

where

$$\begin{aligned}
A =& \frac{1}{D}\left(\xi_x\zeta_x + \xi_y\zeta_y + \xi_z\zeta_z\right) \\
B =& \frac{1}{D}\left(\eta_x\zeta_x + \eta_y\zeta_y + \eta_z\zeta_z\right) \\
C =& \frac{1}{D}\left[\rho\left(\frac{\partial \zeta_t}{\partial \tau} + u\frac{\partial \zeta_x}{\partial \tau} + v\frac{\partial \zeta_y}{\partial \tau} + w\frac{\partial \zeta_z}{\partial \tau}\right)\right. \\
& \left. -\rho U\left(\zeta_x\frac{\partial u}{\partial \xi} + \zeta_y\frac{\partial v}{\partial \xi} + \zeta_z\frac{\partial w}{\partial \xi}\right) - \rho V\left(\zeta_x\frac{\partial u}{\partial \eta} + \zeta_y\frac{\partial v}{\partial \eta} + \zeta_z\frac{\partial w}{\partial \eta}\right)\right] \\
D =& {\zeta_x}^2 + {\zeta_y}^2 + {\zeta_z}^2
\end{aligned} \tag{2.136}$$

The first term of C in the right hand side of Eq. 2.135 is the term due to accelerated motion and deformation of grid, and becomes zero for a stationary grid as considered in the present study. Unknown variables in Eq. 2.135 are only differential terms of pressure, and pressure distribution over the flowfield ($l \geq 2$) is known. Applying the second-order one-sided differencing to $\partial p/\partial \zeta$ as

$$\left(\frac{\partial p}{\partial \zeta}\right)_{l=1} = \frac{-3p_{l=1} + 4p_{l=2} - p_{l=3}}{2\Delta\zeta} \tag{2.137}$$

then, Eq. 2.135 becomes as

$$-\hat{A}\frac{\partial p}{\partial \xi} - \hat{B}\frac{\partial p}{\partial \eta} + p_{l=1} = \hat{C} \tag{2.138}$$

where

$$\begin{aligned}
\hat{A} &= \frac{2}{3}\Delta\zeta A \\
\hat{B} &= \frac{2}{3}\Delta\zeta B \\
\hat{C} &= -\frac{1}{3}\left(2\Delta\zeta C - 4p_{l=2} + p_{l=3}\right)
\end{aligned} \tag{2.139}$$

The following approximate factorization is applied to the left hand side of Eq. 2.138, written in Matrix form, as,

$$\left(\boldsymbol{I} - \hat{A}\delta_\xi\right)\left(\boldsymbol{I} - \hat{B}\delta_\eta\right)\boldsymbol{p}_{l=1} = \hat{\boldsymbol{C}} + \hat{A}\hat{B}\delta_\xi\delta_\eta\boldsymbol{p}_{l=1} \tag{2.140}$$

where $\delta_\kappa\bullet$ denotes the second-order central difference operator. The scalar tridiagonal equation (Eq. 2.138) can be solved through a sequence of operations, and then the pressures on the body surface: $\boldsymbol{p}_{l=1}$ are given.

2.3.6.2 External Boundary

When $l = l_{max}$ plane is an external boundary, the conditions on $l = l_{max}$ are decided according to freestream Mach number M_∞ and the contravariant velocity in ζ direction: W, as follows:

Subsonic Case ($M_\infty < 1$)

Outflow Condition ($W_{l=lmax-1} \geq 0$) Pressure is fixed to freestream value p_∞, and the other physical properties are extrapolated from the adjacent node as

$$\boldsymbol{Q}_{l=lmax} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \frac{p_\infty}{\gamma-1} + \frac{1}{2}\rho\left(u^2+v^2+w^2\right) \end{bmatrix}_{l=lmax-1} \tag{2.141}$$

Inflow Condition ($W_{l=lmax-1} < 0$) All physical properties are fixed to freestream values as

$$\boldsymbol{Q}_{l=lmax} = \boldsymbol{Q}_\infty \tag{2.142}$$

Supersonic Case ($M_\infty \geq 1$)

Outflow Condition ($W_{l=lmax-1} \geq 0$) All physical properties are extrapolated from the adjacent node as

$$\boldsymbol{Q}_{l=lmax} = \boldsymbol{Q}_{l=lmax-1} \tag{2.143}$$

Inflow Condition ($W_{l=lmax-1} < 0$) All physical properties are fixed to freestream values as

$$\boldsymbol{Q}_{l=lmax} = \boldsymbol{Q}_\infty \tag{2.144}$$

2.3.6.3 Symmetric Boundary

In the case without side slip flow, the flowfield around only a half of body is simulated for the reduction in computation time. When $k = 2$ plane is a symmetry plane,

physical properties on $l = 1$ are given by

$$\boldsymbol{Q}_{k=1} = \begin{bmatrix} \rho \\ \rho u \\ -\rho v \\ \rho w \\ e \end{bmatrix}_{k=3} \tag{2.145}$$

2.3.6.4 Wake Singularity

Two dimensional C type grid and three dimensional C-H type grid around an airfoil used in the present study have singular points, where multiple points overlap each other, in rear wake and side wake regions. When $(j, k, 1)$ and $(jmax - j + 1, k, 1)$ points overlap each other, physical properties on these points are averaged from those at the points adjacent to each overlapped point.

$$\begin{aligned} \boldsymbol{Q}_{j,k,1} &= \frac{1}{2}\left(\boldsymbol{Q}_{j,k,2} + \boldsymbol{Q}_{jmax-j+1,k,2}\right) \\ \boldsymbol{Q}_{jmax-j+1,k,1} &= \frac{1}{2}\left(\boldsymbol{Q}_{j,k,2} + \boldsymbol{Q}_{jmax-j+1,k,2}\right) \end{aligned} \tag{2.146}$$

Chapter 3

Validation of Design for Multi-Objective Six Sigma Approach

3.1 Introduction

In this chapter, a newly developed robust design optimization approach "design for multi-objective six sigma (DFMOSS)" is applied to simple robust optimization problems. The numerical results obtained by using the DFMOSS are compared to those obtained by using a traditional robust design optimization approach "design for six sigma (DFSS)" to investigate efficiency and usefulness of the DFMOSS. The investigations of DFMOSS in the optimization problems considering the robustness in objective function value against the variations of optimized design variables are described in Sec. 3.2, and those in the optimization problems considering the robustness in constraint function violation against the variations of optimized design variables are described in Sec. 3.3.

3.2 Case 1: Robustness in Objective Function Value

3.2.1 Case 1.1: Test Function Optimization Problem

3.2.1.1 Problem Definition

Consider the test function optimization problem where the value of objective function $f(x)$ of one design variable x must be minimized:

$$\text{Minimize: } f(x) = -\exp\left(-\frac{|x|}{5}\right)\cos\left(\frac{2\pi x}{|x|^{0.1}}\right) \tag{3.1}$$

where the range of design variable is $-0.5 \leq x \leq 5$. Now, convert this problem to the robust optimization problem where both the mean value μ_f and standard deviation

σ_f of objective function $f(x)$ must be minimized:

$$\begin{aligned} \text{Minimize: } & \mu_f \\ & \sigma_f \end{aligned} \tag{3.2}$$

Figure 3.1 shows the test function distribution ($f(x)$ against x). In a traditional optimization problem (Eq. 3.1), it is clear that an optimal solution is $x = 0$. In the robust optimization problem (Eq. 3.2), on the other hand, $x = 0$ is the best in terms of the optimality, but not good in terms of the robustness because the objective function $f(x)$ disperse widely against the dispersion of design variable x around $x = 0$. The optimality becomes worse but the robustness becomes better in order of hollows $x = 0$, 1, 2.16, 3.38 and 4.66 because the hollow becomes shallower and gentler. Therefore, all these five hollows can be the robust optimal solutions in this robust optimization problem.

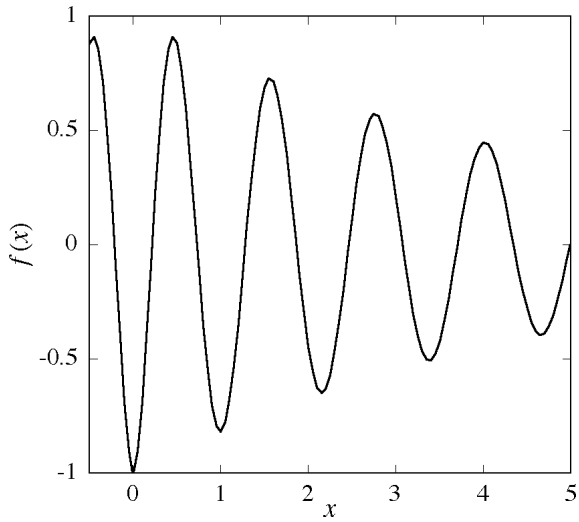


Figure 3.1 Test function distribution of Case 1.1.

3.2.1.2 Optimization Methods and Conditions

This robust optimization problem is solved by using the multi-objective genetic algorithm (MOGA)[33] coupled with the newly developed robust optimization approach DFMOSS, and the single-objective genetic algorithm (SOGA)[31] coupled with the traditional robust optimization approach DFSS,[36] respectively. In both robust optimizations, fitness values are evaluated by using the Pareto-ranking method,[53] the fitness sharing,[33,53] and Michaleswicz's nonlinear function.[49] Parents are selected by the roulette-wheel selection (RWS),[31] children are reproduced by the blended

crossover (BLX-0.5)[48] and random mutation,[49] and the alternation of generations is performed by the Best-N selection.[50,51] In the robust optimization using DFSS, the constraints on sigma level are treated by using the Pareto-optimality-based constraint-handling (PBCH) technique.[56,57] The statistical values of objective function (μ_f and σ_f) are evaluated by the Monte Carlo simulation (MCS) with descriptive sampling (DS).[59]

Table 3.1 shows the present optimization conditions. The mean value of MCS random design variable is set as the design variable of each solution. The standard deviation of MCS random design variable, on the other hand, is fixed at 0.1. In the robust optimizations using DFSS, the sigma level n is set as 3σ in advance, and total seven cases with different combination of weighting factors w_μ and w_σ are carried out. In the robust optimization using DFMOSS, on the other hand, it is not necessary to set the sigma level in advance, and only one case is carried out without the advance setting of weighting factors.

Table 3.1 Optimization conditions of Case 1.1.

for SOGA/MOGA	Population size	32
	Number of generations	100
	Mutation rate [%]	20
for MCS random design variable	Sample size	1000
	Probability density distribution	normal distribution
	Mean value	value of each solution
	Standard deviation	0.1
for DFSS	USL	-0.2
	LSL	N/A
	Sigma level n	3σ
	$w_\mu : w_\sigma$	1 : 1000, 1 : 100, 1 : 10, 1 : 1, 10 : 1, 100 : 1, 1000 : 1

3.2.1.3 Numerical Results

Figure 3.2 shows the robust optimal solution distributions (σ_f against μ_f) and Table 3.2 shows the numerical data of the robust optimal solutions of the present case obtained by using the DFSS and the DFMOSS, respectively. In the robust optimization using DFSS, only two robust optimal solutions with more than 3σ robustness quality ($x = 1$ and 3.39) are obtained though total seven-time optimizations with different combination of weighting factors are carried out. Here note that essentially

there exist total three robust optimal solutions with more than 3σ robustness quality ($x = 1$, 2.16 and 3.38). These results indicate that the DFSS does not have enough capability of finding all robust optimal solutions. In this study, two robust optimal solutions can be obtained because the advance setting of sigma level as 3σ is appropriate by chance. However, it is not always guaranteed for the DFSS to obtain the robust optimal solutions according to the advance setting of sigma level.

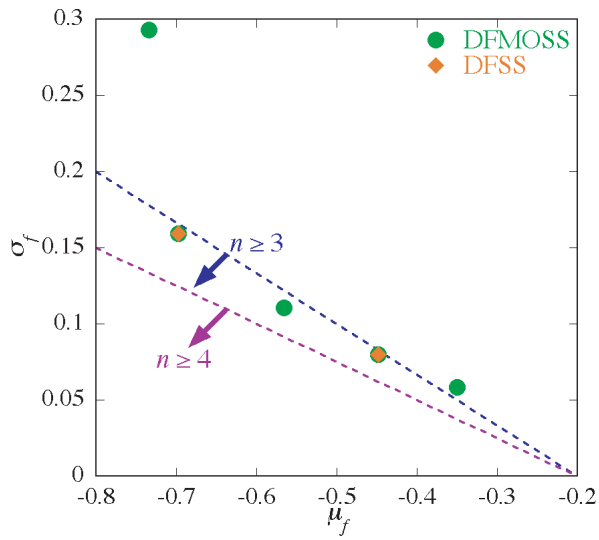


Figure 3.2 Robust optimal solution distributions of Case 1.1.

Table 3.2 Numerical data of robust optimal solutions of Case 1.1.

	$w_\mu : w_\sigma$	x	μ_f	σ_f
DFSS	1 : 1000	3.3864	−0.44811	0.079737
	1 : 100	3.3863	−0.44812	0.079737
	1 : 10	0.99987	−0.69769	0.1592
	1 : 1	0.99808	−0.69776	0.15927
	10 : 1	0.99724	−0.69777	0.15934
	100 : 1	0.99713	−0.69777	0.15935
	1000 : 1	0.99712	−0.69777	0.15935
DFMOSS		0.001737	−0.73406	0.29317
		1.0023	−0.69747	0.15923
		2.159	−0.56617	0.11038
		3.3835	−0.44815	0.079798
		4.6596	−0.3499	0.058399

In the robust optimization using DFMOSS, on the other hand, all five robust optimal solutions ($x = 0$, 1, 2.16, 3.38 and 4.66) can be obtained successfully and effectively in only one calculation, and it is understood easily that the maximum

sigma level of the obtained robust optimal solutions is more than 3σ by the post-evaluation.

3.2.2 Case 1.2: Welded Beam Design Problem

3.2.2.1 Problem Definition

Next, the DFMOSS is applied to the welded beam design problem.[83,84] The present welded beam structure is shown in Fig. 3.3. The welded beam consists of a beam and a weld required to secure the beam to the member. The objective of the design is to find a feasible set of dimensions h, l, t and b (denoted by $\boldsymbol{x} = [x_1, x_2, x_3, x_4]^T$) to carry a certain load (P) and still have a minimum total fabricating cost.

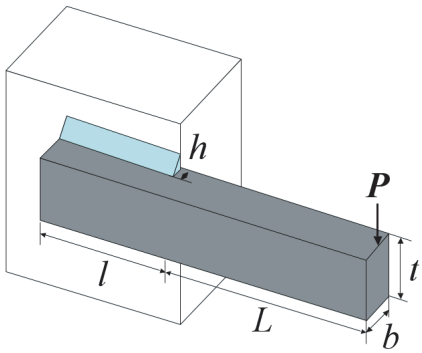


Figure 3.3 Welded beam structure.

The objective function $f(\boldsymbol{x})$ is the total fabricating cost which mainly comprises of the set-up cost, welding labor cost and material cost:

$$\text{Minimize: } f(\boldsymbol{x}) = (1 + c_1)\, {x_1}^2 x_2 + c_2 x_3 x_4 \,(L + x_2) \tag{3.3}$$

where c_1 and c_2 are the cost of unit volume of weld material and bar stock, respectively. The associated functional constraints $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), g_3(\boldsymbol{x}), g_4(\boldsymbol{x}), g_5(\boldsymbol{x}), g_6(\boldsymbol{x})]^T$ are:

$$\begin{aligned}
\text{Subject to: } & g_1(\boldsymbol{x}) = \tau(\boldsymbol{x}) - \tau_{max} \leq 0 \\
& g_2(\boldsymbol{x}) = \sigma(\boldsymbol{x}) - \sigma_{max} \leq 0 \\
& g_3(\boldsymbol{x}) = x_1 - x_4 \leq 0 \\
& g_4(\boldsymbol{x}) = c_1 {x_1}^2 + c_2 x_3 x_4 \,(L + x_2) - 5 \leq 0 \\
& g_5(\boldsymbol{x}) = \delta(\boldsymbol{x}) - \delta_{max} \leq 0 \\
& g_6(\boldsymbol{x}) = P - P_c(\boldsymbol{x}) \leq 0
\end{aligned} \tag{3.4}$$

where

$$\begin{aligned}
\tau(\boldsymbol{x}) &= \sqrt{\tau'^2 + 2\tau'\tau''\frac{x_2}{2R} + \tau''^2}, \quad \tau' = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{MR}{J}, \\
M &= P\left(L + \frac{x_2}{2}\right), \quad R = \sqrt{\frac{{x_2}^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}, \\
J &= 2\left\{\sqrt{2}x_1x_2\left[\frac{{x_2}^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}, \\
\sigma(\boldsymbol{x}) &= \frac{6PL}{x_4{x_3}^2}, \quad \delta(\boldsymbol{x}) = \frac{4PL^3}{E{x_3}^3x_4}, \quad P_c(\boldsymbol{x}) = \frac{4.013E\sqrt{{x_3}^2{x_4}^6/36}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)
\end{aligned} \tag{3.5}$$

and

$$\begin{aligned}
c_1 &= 0.10471, \quad c_2 = 0.04811, \\
P &= 6 \times 10^3[\text{lb}], \quad L = 14[\text{in}], \quad E = 3 \times 10^7[\text{psi}], \quad G = 1.2 \times 10^7[\text{psi}], \\
\delta_{max} &= 0.25[\text{in}], \quad \tau_{max} = 1.36 \times 10^4[\text{psi}], \quad \sigma_{max} = 3 \times 10^4[\text{psi}]
\end{aligned} \tag{3.6}$$

where $\tau(\boldsymbol{x})$, $\sigma(\boldsymbol{x})$, $\delta(\boldsymbol{x})$ and $P_c(\boldsymbol{x})$ are weld shear stress, bar bending stress, bar end deflection and bar buckling load, respectively. The ranges of design variables are $0.125 \leq x_1 \leq 5$[in], $0.1 \leq x_2 \leq 10$[in], $0.1 \leq x_3 \leq 10$[in] and $0.1 \leq x_4 \leq 5$[in]. Now, convert this problem to the robust optimization problem that both the mean value μ_f and the standard deviation σ_f of total fabricating cost $f(\boldsymbol{x})$ must be minimized, and also six constraint functions $\boldsymbol{g}(\boldsymbol{x})$ must be satisfied at the mean values $\boldsymbol{\mu_x} = [\mu_{x_1}, \mu_{x_2}, \mu_{x_3}, \mu_{x_4}]^T$ of design variables $\boldsymbol{x}$:

$$\begin{aligned}
\text{Minimize: } & \mu_f \\
& \sigma_f \\
\text{Subject to: } & \boldsymbol{g}(\boldsymbol{\mu_x}) \leq \boldsymbol{0}
\end{aligned} \tag{3.7}$$

3.2.2.2 Optimization Methods and Conditions

The optimization methods of the present case are the same as those used in Case 1.1 described in Subsec. 3.2.1. Table 3.3 shows the present optimization conditions. In the robust optimization using DFSS, the sigma level n is set as 6σ in advance, and total seven cases with different combinations of weighting factors w_μ and w_σ are carried out. In the robust optimization using DFMOSS, on the other hand, it is not necessary to set the sigma level in advance, and only one case is carried out without the advance setting of weighting factors.

Table 3.3 Optimization conditions of Case 1.2.

for SOGA/MOGA	Population size	50
	Number of generations	1000
	Mutation rate [%]	20
for MCS random design variables	Sample size	1000
	Probability density distribution	normal distribution
	Mean value	value of each solution
	Standard deviation	0.01
for DFSS	USL	3
	LSL	N/A
	Sigma level n	6σ
	$w_\mu : w_\sigma$	1 : 1000, 1 : 100, 1 : 10, 1 : 1, 10 : 1, 100 : 1, 1000 : 1

3.2.2.3 Numerical Results

Figure 3.4 shows the robust optimal solution distributions (standard deviation σ_f against mean value μ_f of total fabricating cost $f(\boldsymbol{x})$) of the present case obtained by using the DFSS and the DFMOSS. In the robust optimization using DFSS, robust optimal solutions with more than 6σ robustness quality are obtained. However, these solutions distribute very locally though total seven-time optimizations with different combination of weighting factors are carried out. These results indicate that the DFSS does not have enough capability of finding robust optimal solutions globally. In this study, the robust optimal solutions can be obtained because the advance setting of sigma level as 6σ is appropriate by chance. However, it should be noted that it is not always guaranteed for the DFSS to obtain the robust optimal solutions according to the advance setting of sigma level.

In the robust optimization using DFMOSS, on the other hand, many robust optimal solutions can be obtained effectively in only one optimization, and these solutions distribute globally and uniformly. Also it is understood easily that the maximum sigma level of the obtained robust optimal solutions is more than 6σ and total twenty-one robust optimal solutions have more than 6σ robustness quality by the post-evaluation.

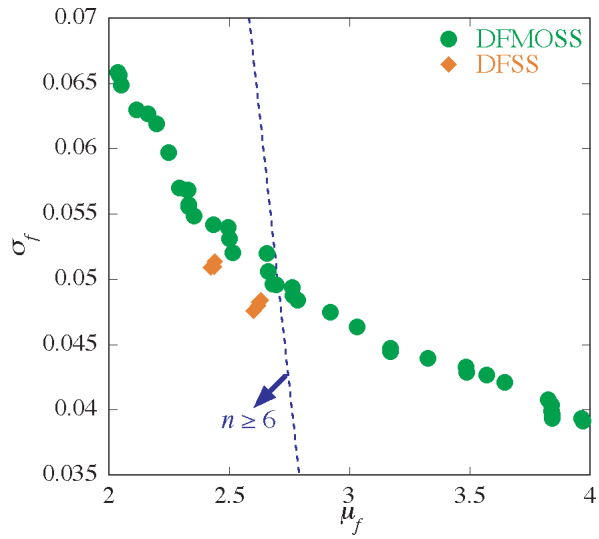


Figure 3.4 Robust optimal solution distributions of Case 1.2.

3.3 Case 2: Robustness in Constraint Function Violation

3.3.1 Case 2.1: Test Function Optimization Problem

3.3.1.1 Problem Definition

Consider the test function problem where the value of objective function $f(\boldsymbol{x})$ of two design variables $\boldsymbol{x} = [x_1, x_2]^T$:

$$\text{Minimize: } f(\boldsymbol{x}) = \sqrt{{x_1}^2 + {x_2}^2} \tag{3.8}$$

and also the constraint function $g(\boldsymbol{x})$ must be satisfied:

$$\text{Subject to: } g(\boldsymbol{x}) = -x_1 - x_2 + \sqrt{2} \leq 0 \tag{3.9}$$

where the ranges of design variables are $0 \leq x_1 \leq 2$ and $0 \leq x_2 \leq 2$. Now, convert this problem to the robust optimization problem where the mean value μ_f of objective function $f(\boldsymbol{x})$ must be minimized, the sigma level n of constraint function $g(x)$ must be maximized, and also the constraint function $g(\boldsymbol{x})$ must be satisfied at the mean value $\boldsymbol{\mu_x} = [\mu_{x_1}, \mu_{x_2}]^T$ of design variable $\boldsymbol{x}$:

$$\begin{aligned} \text{Minimize: } & \mu_f \\ \text{Maximize: } & n = -\mu_g/\sigma_g \\ \text{Subject to: } & g(\boldsymbol{\mu_x}) \leq 0 \end{aligned} \tag{3.10}$$

where μ_g and σ_g are the mean value and the standard deviation of constraint function $g(\boldsymbol{x})$.

Figure 3.1 shows the contour plots of objective function $f(\boldsymbol{x})$ and constraint function $g(\boldsymbol{x})$ on x_1–x_2 plane. In a traditional optimization problem (Eqs. 3.8 and 3.9), it is clear that an optimal solution is $\boldsymbol{x} = \left[1/\sqrt{2}, 1/\sqrt{2}\right]^T$ and also the constraint function $g(\boldsymbol{x})$ is active ($g(\boldsymbol{x}) = 0$) at the optimal solution. In the robust optimization problem (Eq. 3.10), on the other hand, $\boldsymbol{x} = \left[1/\sqrt{2}, 1/\sqrt{2}\right]^T$ is the best in terms of the optimality, but not good in terms of the robustness because this solution disperses easily outside the feasible region in terms of the constraint function $g(\boldsymbol{x})$ against the dispersion of design variables $\boldsymbol{x}$ around $\boldsymbol{x} = \left[1/\sqrt{2}, 1/\sqrt{2}\right]^T$. Therefore, the optimality becomes worse but the robustness becomes better as the solution shifts away from the constraint function limit along the direction where the value of constraint function $g(\boldsymbol{x})$ increases in this robust optimization problem.

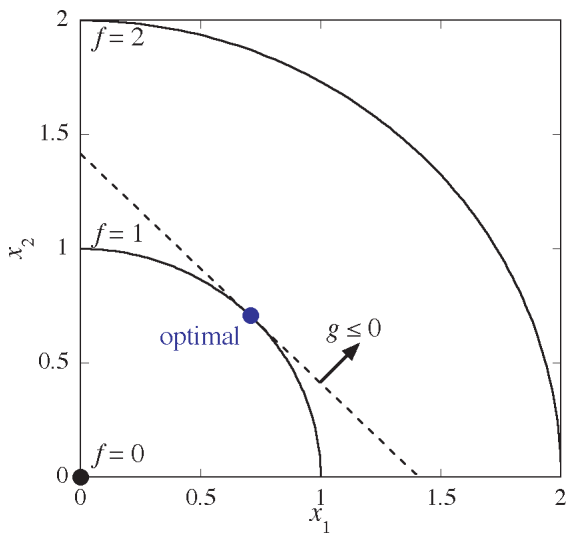


Figure 3.5 Test function contour plots of Case 2.1.

3.3.1.2 Optimization Methods and Conditions

The optimization methods of the present case are the same as those used in Case 1.1 described in Subsec. 3.2.1. Table 3.4 shows the present optimization conditions. In the robust optimization using DFSS, total seven cases with different sigma levels n are carried out. In the robust optimization using DFMOSS, on the other hand, only one case is carried out without the advance setting of sigma level.

3.3.1.3 Numerical Results

Figures 3.6 and 3.7 show the robust optimal solution distributions (n against μ_f) and corresponding design variable distributions (x_2 against x_1), respectively, of the

Table 3.4 Optimization conditions of Case 2.1.

for SOGA/MOGA	Population size	32
	Number of generations	100
	Mutation rate [%]	20
for MCS random design variables	Sample size	1000
	Probability density distribution	normal distribution
	Mean value	value of each solution
	Standard deviation	0.3
for DFSS	Sigma level n	1σ, 2σ, 3σ, 4σ, 5σ, 6σ, 7σ

present case obtained by using the DFSS and the DFMOSS. In the robust optimization using DFSS, six robust optimal solutions with 1σ, 2σ, 3σ, 4σ, 5σ and 6σ robustness quality are obtained, and the obtained solutions shift away from the constraint function limit with the increment of sigma level. However, a robust optimal solution can not be found in the case when the sigma level is specified as 7σ. This result indicates that the DFSS has difficulty in the advance specification of sigma level. In addition, the robust optimal solutions obtained by the DFSS distribute sparsely because the specified sigma levels are discrete integers. This result indicates that the robust optimization using DFSS is not effective to obtain robust optimal solutions distributed uniformly in the whole design space.

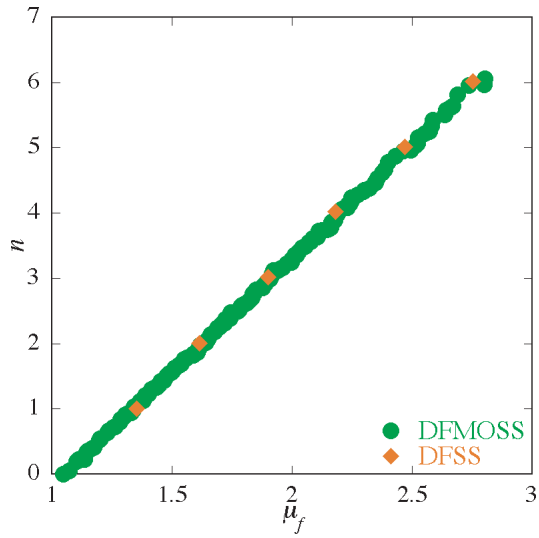


Figure 3.6 Robust optimal solution distributions of Case 2.1.

In the robust optimization using DFMOSS, on the other hand, many robust optimal solutions can be obtained effectively by only one case without the advance

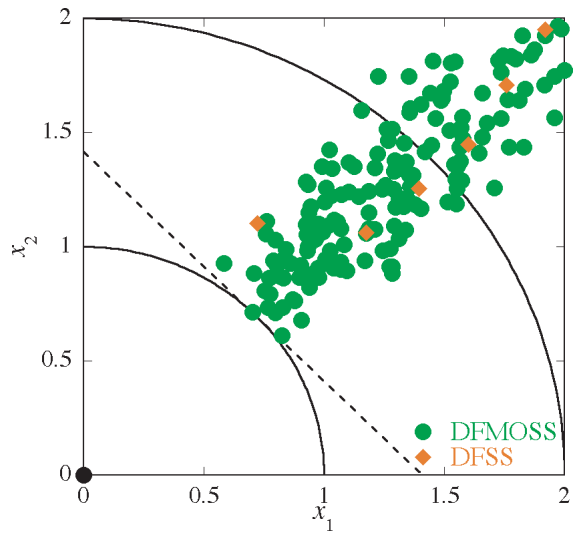


Figure 3.7 Design variable distributions of robust optimal solutions of Case 2.1.

specification of sigma level, and these solutions distribute uniformly in the whole design space. Also it is understood easily that this robust optimization problem has continuous trade-off relation between the mean value of objective function and the sigma level of constraint function from 0σ to 6σ, from the obtained robust optimal solution distribution.

3.3.2 Case 2.2: Welded Beam Design Problem

3.3.2.1 Problem Definition

Again, the DFMOSS is applied to the welded beam design problem,[83,84] as described in Subsec. 3.2.2. In this case, convert this problem (Eqs. 3.3 and 3.4) to the robust optimization problem where the mean value μ_f of total fabricating cost $f(\boldsymbol{x})$ must be minimized, the sigma levels $\boldsymbol{n} = [n_1, n_2, n_3, n_4, n_5, n_6]^T$ of six constraint functions $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), g_3(\boldsymbol{x}), g_4(\boldsymbol{x}), g_5(\boldsymbol{x}), g_6(\boldsymbol{x})]^T$, respectively, must be maximized, and also the constraints $\boldsymbol{g}(\boldsymbol{x})$ must be satisfied at the mean

value $\boldsymbol{\mu_x} = [\mu_{x1}, \mu_{x2}, \mu_{x3}, \mu_{x4}]^T$ of design variables $\boldsymbol{x}$:

$$\begin{aligned} \text{Minimize: } & \mu_f \\ \text{Maximize: } & n_1 = -\mu_{g_1}/\sigma_{g_1} \\ & n_2 = -\mu_{g_2}/\sigma_{g_2} \\ & n_3 = -\mu_{g_3}/\sigma_{g_3} \\ & n_4 = -\mu_{g_4}/\sigma_{g_4} \\ & n_5 = -\mu_{g_5}/\sigma_{g_5} \\ & n_6 = -\mu_{g_6}/\sigma_{g_6} \\ \text{Subject to: } & \boldsymbol{g}(\boldsymbol{\mu_x}) \leq \mathbf{0} \end{aligned} \tag{3.11}$$

where $\boldsymbol{\mu_g} = [\mu_{g_1}, \mu_{g_2}, \mu_{g_3}, \mu_{g_4}, \mu_{g_5}, \mu_{g_6}]^T$ and $\boldsymbol{\sigma_g} = [\sigma_{g_1}, \sigma_{g_2}, \sigma_{g_3}, \sigma_{g_4}, \sigma_{g_5}, \sigma_{g_6}]^T$ are the mean values and the standard deviations of constraint functions $\boldsymbol{g}(\boldsymbol{x})$, respectively.

3.3.2.2 Optimization Methods and Conditions

The optimization methods of the present case are the same as those used in Case‘1.1 described in Subsec. 3.2.1. Table 3.4 shows the present optimization conditions. In the robust optimization using DFSS, total three cases with different sigma levels n are carried out. In the robust optimization using DFMOSS, on the other hand, only one case is carried out without the advance setting of sigma level.

Table 3.5 Optimization conditions of Case 2.2.

for SOGA/MOGA	Population size	100
	Number of generations	200
	Mutation rate [%]	10
for MCS random design variables	Sample size	1000
	Probability density distribution	normal distribution
	Mean value	value of each solution
	Standard deviation	0.1
for DFSS	Sigma level n	1σ, 2σ, 3σ

3.3.2.3 Numerical Results

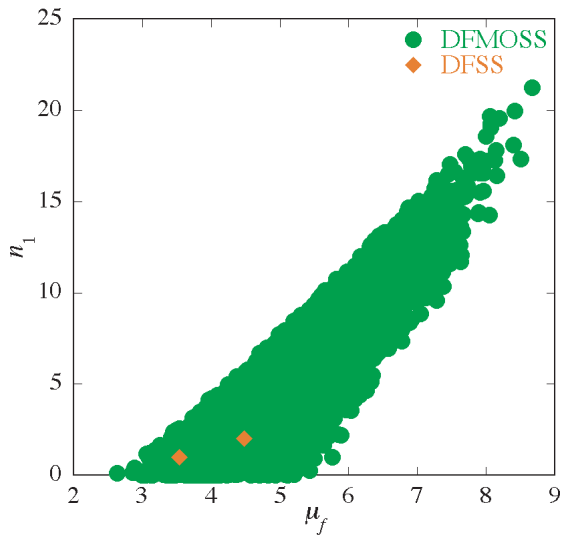
Figure 3.8(a) shows the distributions of all robust optimal solutions (sigma level n_1 of the first constraint function $g_1(\boldsymbol{x})$ against mean value μ_f of total fabricating cost $f(\boldsymbol{x})$) of the present case obtained by using the DFSS and the DFMOSS, and Figs. 3.8(b) and 3.8(c) show similar distributions of robust optimal solutions whose

all sigma levels $\boldsymbol{n}$ are larger than 1 and 2, respectively. In the robust optimization using DFSS, two robust optimal solutions with 1σ, 2σ robustness quality are obtained. However, a robust optimal solution can not be found in the case when all sigma levels are specified as 3σ. This result indicates that the DFSS has difficulty in the advance specification of sigma level. In addition, the robust optimal solutions obtained by the DFSS distribute sparsely because the specified sigma levels are discrete integers. This result indicates that the robust optimization using DFSS is not effective to obtain robust optimal solutions distributed uniformly in the whole design space.

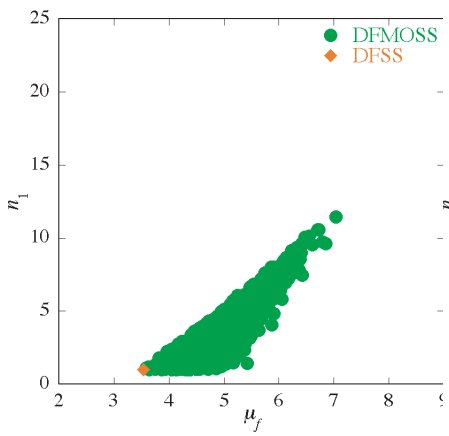
In the robust optimization using DFMOSS, on the other hand, many robust optimal solutions can be obtained effectively by only one case without the advance specification of sigma levels, and these solutions distribute uniformly. In addition, it is understood easily that this robust optimization problem has continuous trade-off relation between the mean value of total fabricating cost and the sigma level of the first constraint function. Furthermore, though only the trade-off relations between μ_f and n_1 are shown here, it is also possible to reveal the other trade-off relations between all combinations of the sigma levels of all constraint functions by further investigation.

3.4 Summary

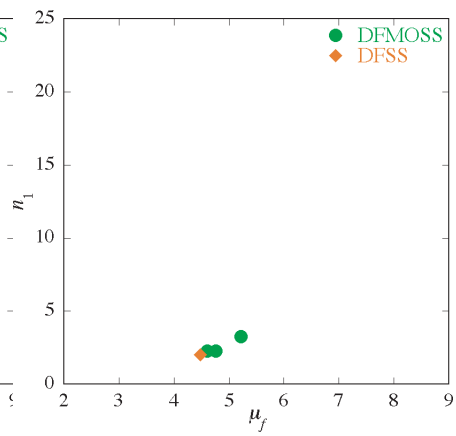
In this chapter, a newly developed robust optimization approach DFMOSS was applied to simple robust optimization problems to investigate the efficiency and usefulness of the DFMOSS. These studies showed that the DFMOSS has some advantages compared to the DFSS. Firstly, the DFMOSS does not have difficulty in the advance specification of input parameters such as weighting factors and sigma level. Secondly, the DFMOSS can obtain multiple robust optimal solutions effectively by only one case and reveal trade-off information between optimality and robustness surely, compared to the DFSS. Thirdly, the satisfied sigma level of each obtained robust optimal solution can be evaluated easily and flexibly as post-processing in the robust optimizations using DFMOSS. These results indicated clearly that the DFMOSS had more effective and more useful characteristics than the DFSS.



(a) All solutions.



(b) Solutions with all sigma levels larger than 1σ.



(c) Solutions with all sigma levels larger than 2σ.

Figure 3.8 Robust optimal solution distributions of Case 2.2.

Chapter 4

Robust Aerodynamic Design of Airfoil Configuration for Mars Exploratory Airplane

4.1 Introduction

In this chapter, robust aerodynamic design optimizations of airfoil configuration for Mars exploratory airplane considering the effects of wind variations are performed by using the DFMOSS and the CFD simulation. Differently from the robust optimization problems against the variations of optimized design variables as described in Chap. 3, the present robust aerodynamic design optimization problems correspond to those against the variations of non-optimized environmental variables such as wind speeds and directions. In the present study, three cases of the robust aerodynamic design optimizations of airfoil configuration are carried out; the robust design optimization of lift to drag ratio against the variation of flight Mach number is described in Sec. 4.2, that of pitching moment coefficient against the variation of flight Mach number is described in Sec. 4.3, and that of lift to drag ratio against the variation of angle of attack is described in Sec. 4.4. In each case, the numerical results obtained by the robust optimization using DFMOSS are compared to those obtained by a traditional one-point optimization or those obtained by the robust optimization using DFSS in order to compare efficiency and usefulness of the DFMOSS and the differences in optimized designs. Then, the obtained trade-off relations between the optimality and the robustness of aerodynamic performance of airfoil configuration for Mars exploratory airplane are discussed.

4.2 Case 1: Robust Design Optimization of Lift to Drag Ratio against the Variation of Flight Mach Number

4.2.1 Design Problem Definition

Firstly, consider the traditional aerodynamic design optimization problem where airfoil configuration must be optimized so that lift to drag ratio L/D will be maximized at design point:

$$\text{Maximize: } L/D \tag{4.1}$$

In the present study, cruising conditions of NASA's AME,[11] as shown in Table 4.1, is used as the design point. Now, convert this problem to the robust aerodynamic design optimization problem where the mean value of dispersive L/D must be maximized and also the standard deviation of dispersive L/D must be minimized when the flight Mach number M_∞ disperses around design point:

$$\begin{aligned} &\text{Maximize: mean value of } L/D \\ &\text{Minimize: standard deviation of } L/D \end{aligned} \tag{4.2}$$

This robust aerodynamic optimization problem aims at finding the airfoil configuration not only with maximum L/D but also with robust L/D characteristic that the Mars exploratory airplane can avoid failure in flying over an expected range when it flies in east-westward direction without side and vertical winds under the situation where westerlies as shown in Fig. 1.8 blow stably. Table 4.2 shows the assumed statistical characteristics of dispersive flight Mach number M_∞ for NASA's AME. The value of 0.1 as the standard deviation of M_∞ corresponds to the variation of wind speeds in the moving direction of about 22 m/s on the Mars, and it is nearly equal to the dairy and seasonal variation of westerly speeds near the Martian surface (at the altitude of several kilometers) where it is assumed for the Mars airplane to fly for exploring the structure of remnant magnetic fields and lower atmosphere, as shown in Fig. 1.8.

Table 4.1 Two-dimensional cruising conditions of NASA's AME.[11]

Reynolds number Re (based on root chord length c)	1.0×10^5
Flight Mach number M_∞	0.4735
Angle of attack α [deg]	2.0

Table 4.2 Statistical characteristics of dispersive flight Mach number M_∞ for NASA's AME.

Mean value	0.4735
Standard deviation	0.1

Selection of how to define airfoil configuration is an important step for the design optimization of airfoil configuraion.[85] In the present study, airfoil configuration is defined by the B-spline curves. The definition based on the B-spline curves has some advantages; second-order derivative is continuous, various airfoil configurations can be expressed, and definition of initial design space is intuitive. Figure 4.1 shows the present definition of airfoil configuration based on the B-spline curves where x and y are the coordinates in the chordwise and the vertical directions, respectively. Six control points of the B-spline curves are given as well as three fixed point corresponding to the leading and trailing edges. The B-spline curves are generated by approximating x and y coordinates of these given points where the generated B-spline curves are scaled up or down in x direction so that the minimum x coordinate on this curve corresponding to the leading edge becomes zero. Note that x and y coordinates of the leading edge obtained by the present definition of airfoil configuration are not necessary $[0, 0]$ exactly. Table 4.3 shows the design variables and those ranges of the present aerodynamic design optimization problem. The design variables are x and y coordinates of six control points, and the total number of design variables is twelve. Such specification of range of design variables may leads to the intersection of upper and lower airfoil surfaces. Therefore, a geometric constraint is imposed in the present aerodynamic design optimization so that the upper and lower surfaces of airfoil do not intersect each other. The structural constraint on airfoil thickness is not considered because we want to discuss an aerodynamic effect purely in the present study.

4.2.2 Numerical Methods and Conditions

4.2.2.1 Optimization

In the present case, four aerodynamic design optimizations are performed and the obtained numerical results are compared:

- **One-point optimization using a single-objective genetic algorithm (SOGA)[31]**

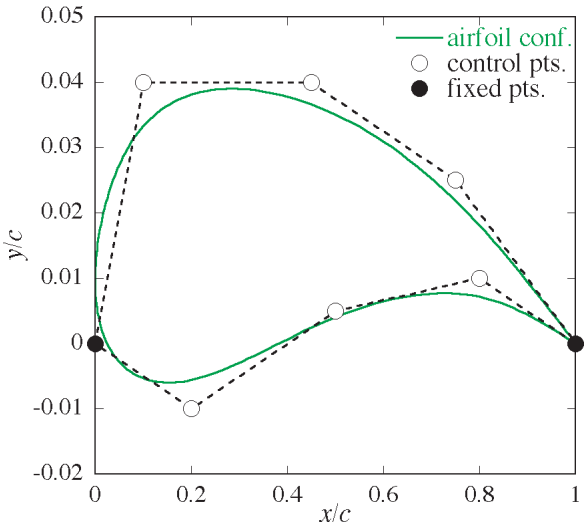


Figure 4.1 Definition of airfoil configuration based on the B-spline curves of Case 1.

Table 4.3 Design variables and corresponding ranges of Case 1.

Design variable		lower limit	upper limit
Coordinate x/c	B-spline control point 1	0.66667	0.99
	B-spline control point 2	0.33333	0.66667
	B-spline control point 3	0.01	0.33333
	B-spline control point 4	0.01	0.33333
	B-spline control point 5	0.33333	0.66667
	B-spline control point 6	0.66667	0.99
Coordinate y/c	B-spline control point 1	-0.15	0.15
	B-spline control point 2	-0.15	0.15
	B-spline control point 3	-0.15	0.15
	B-spline control point 4	-0.15	0.15
	B-spline control point 5	-0.15	0.15
	B-spline control point 6	-0.15	0.15

When $M_\infty = 0.4735$ and $\alpha = 2.0$ [deg],

$$\text{Maximize: } L/D \tag{4.3}$$

- **Two-point optimization using a multi-objective genetic algorithm (MOGA)[33]**

  When $M_\infty = 0.3735$ and $\alpha = 2.0$ [deg],

$$\text{Maximize: } \quad L/D \tag{4.4}$$

  and also when $M_\infty = 0.5735$ and $\alpha = 2.0$ [deg],

$$\text{Maximize: } \quad L/D \tag{4.5}$$

- **Robust optimization using the SOGA coupled with the DFSS[36]**

- **Robust optimization using the MOGA coupled with the DFMOSS**

  When M_∞ disperses around 0.4735 with its standard deviation of 0.1,

$$\begin{aligned} &\text{Maximize: mean value of } L/D \\ &\text{Minimize: standard deviation of } L/D \end{aligned} \tag{4.6}$$

In all optimizations, fitness value are evaluated by using the Pareto-ranking method,[53] the fitness sharing,[33,53] and Michaleswicz's nonlinear function.[49] Parents are selected by the stochastic universal sampling (SUS),[47] children are reproduced by the blended crossover (BLX-0.5)[48] and uniform mutation[33] with 10 % of the range of each design variable as the maximum perturbation, and the alternation of generations is performed by the Best-N selection.[50,51] The constraints are treated by using the Pareto-optimality-based constraint-handling (PBCH) technique.[56,57] In the robust aerodynamic design optimizations using DFSS or DFMOSS, the statistical values of aerodynamic performance parameters are evaluated by the second-order Taylor's series expansion approach, i.e., three CFD simulations at $M_\infty = 0.3735$, 0.4735 and 0.5735 are required for one evaluation of statistical values.

Table 4.4 shows the present optimization conditions. In the robust optimizations using DFSS, the lower specification limit LSL and the sigma level n of L/D is set as 42 and 3σ in advance, respectively, and total three cases with different combination of weighting factors w_μ and w_σ are carried out. In the robust optimization using DFMOSS, on the other hand, it is not necessary to set the sigma level in advance, and only one case is carried out without the advance setting of weighting factors.

Table 4.4 Optimization conditions of Case 1.

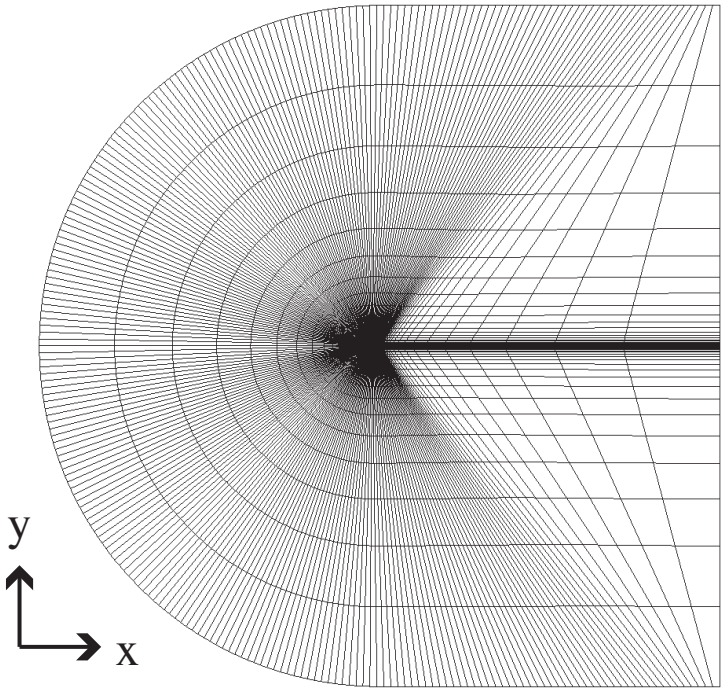
for SOGA/MOGA	Population size	64
	Number of generations	100
	Mutation rate [%]	20
for DFSS	USL	N/A
	LSL	42
	Sigma level n	3σ
	$w_\mu : w_\sigma$	1 : 10, 1 : 1, 10 : 1

4.2.2.2 CFD Simulation

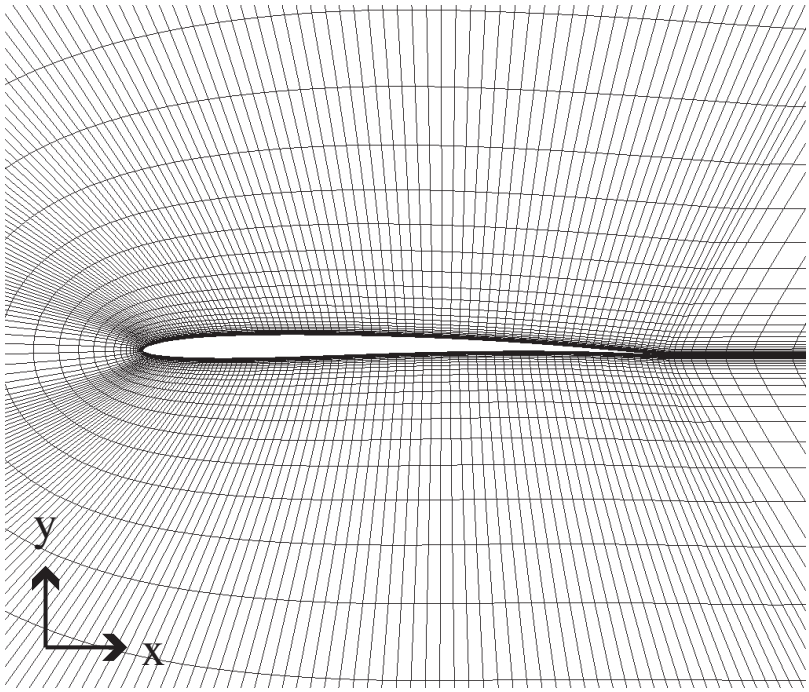
The governing equations for the CFD simulation are the two-dimensional Favre-averaged compressible thin-layer Navier-Stokes equations. The LU-ADI factorization algorithm[73] is used for the time integration with the physical time step size of 0.01 (maximum local CFL number of about 650) and the number of iteration steps of 40,000. Aerodynamic performance parameters such as lift, drag and moment are evaluated by averaging these values for the last 10,000 steps. The inviscid terms of numerical fluxes are evaluated by the SHUS scheme.[70] In the inviscid terms, high-order accuracy is obtained by the third-order upwind-biased MUSCL interpolation[71,72] based on the primitive variables with van Albada differentiable limiter.[67] The viscous terms are evaluated by the second-order central differencing, and the turbulent viscosity is modeled by the Baldwin-Lomax algebraic turbulence model.[81]

Figure 4.2 shows the grid distributions. In the present study, C type grid is generated algebraically around the airfoil, and grid points are clustered near the leading edge, the trailing edge and an airfoil surface. The number of grid points is 251 in the direction around the airfoil (211 points over the airfoil surface), 51 in the direction normal to the airfoil surface, and the total number of grid points is 12,801. The minimum grid spacing near the airfoil surface is about 1.0×10^{-5} based on the root chord length c.

It should be noted that a lot of grid points are required if we attempt to simulate laminar flows accurately without the turbulence model. Unrealistic vorticities may occur when laminar flows are simulated with not so many grid points used in the present study. Therefore, in the present study, the flowfields are assumed to be fully turbulent by using the Baldwin-Lomax algebraic turbulence model for the reduction in computation time.



(a) Overall view.



(b) Close-up view.

Figure 4.2 Grid distributions.

The computation time required for one evaluation of aerodynamic performance of airfoil using the CFD simulation is about five minutes with one processor of NEC SX-6 computing system owned by the Institute of Space and Astronautical Science (ISAS) of JAXA. In the present study, the optimizer distributes the multiple evaluators corresponding to the multiple individuals of GA into 32 processors of this computing system in parallel. Therefore, the total computation time required for one case of the present robust aerodynamic design optimization using DFMOSS can be reduced to about 56 hours.

4.2.3 Numerical Results

Figure 4.3 compares the robust optimal solution distributions (standard deviation against mean value of L/D) obtained by using the DFSS and the DFMOSS with the optimal solution distributions obtained by the one-point and the two-point optimizations neglecting the robustness of L/D. Compared to the robust optimal solutions, the one-point optimal solution has larger standard deviation of L/D. It indicates that the one-point optimization can not find a solution with robust characteristic of L/D. The two-point optimal solutions have smaller standard deviations of L/D than the one-point optimal solution, but still has less robust characteristics than the robust optimal solutions. In addition, the two-point optimal solutions do not distribute globally and can not reveal trade-off relation between optimality (mean value of L/D) and robustness (standard deviation of L/D) sufficiently. The robust optimizations using the DFSS can find three robust optimal solutions with smaller standard deviation of L/D than those of one-point and two-point optimal solutions and more than 3σ robustness quality. However, these solutions distribute narrowly though three cases are carried out with different combination of weighting factors. This indicates that the DFSS has lack in capability of revealing global trade-off relation between optimality and robustness. On the other hand, the robust optimization using the DFMOSS can successfully and effectively find total eighteen robust optimal solutions by only one optimization run, and these solutions distribute globally and uniformly. Also it is understood easily that the maximum sigma level of L/D of the obtained robust optimal solutions is more than 6σ by the post-evaluation when the lower specification limit of L/D is set as 42. It indicates that the DFMOSS can give rich trade-off information between optimality and robustness which various designers can utilize flexibly. Though it appeared that the DFSS find better robust optimal solutions than the DFMOSS, note that, in the robust design opti-

mizations, it is important not to find the robust optimal solutions themselves, but to reveal trade-off information between the optimality and the robustness of performance and give chance to the upper-level decision maker to select one solution from the obtained multiple robust optimal solutions with other considerations from the designer's viewpoint. Also it should be noted that the robust optimization using the DFSS requires more cases with different combinations of weighting factors when the number of objective functions increases, and eventually leads to unpractical vast computation time in real-world design problems with multiple objective functions.

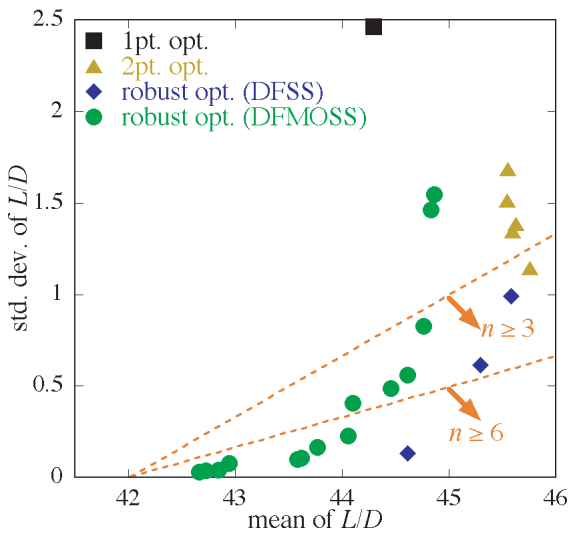


Figure 4.3 Robust optimal solution distributions of Case 1.

Hereafter, three robust optimal solutions with 1σ, 3σ and 6σ robustness qualities obtained by using the DFMOSS, as shown in Fig. 4.4, are compared with the one-point optimal solution. Figure 4.5 shows the histories of L/D against M_∞ at $\alpha = 2.0$ [deg]. The one-point optimal solution has the largest L/D at the design point $M_\infty = 0.4735$. However, L/D of this solution decreases drastically with the increment in M_∞, and it falls below its lower specification limit 42 at high M_∞. On the other hand, three robust optimal solutions have slightly smaller L/D at the design point but more stable characteristics of L/D against the increment in M_∞ than the one-point optimal solution, and keep large L/D until high M_∞. In addition, comparing the three robust optimal solutions, the solution with larger sigma level of L/D has more stable characteristic of L/D against the increment in M_∞. These results indicate that the robust aerodynamic design optimization using the DFMOSS can actually find the multiple designs with various robust characteristics of L/D against the variation of M_∞ by only one optimization run.

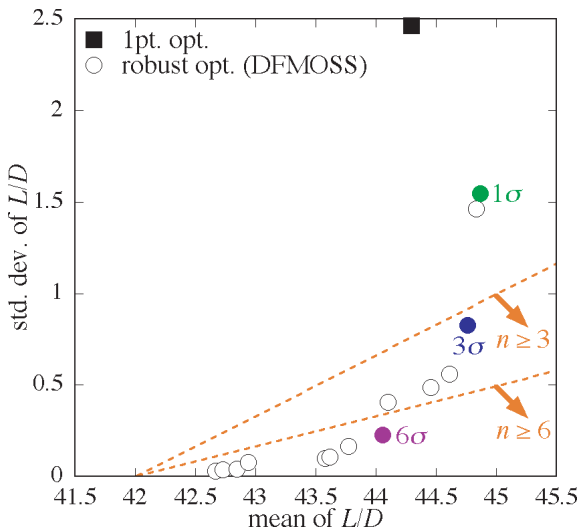


Figure 4.4 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 1.

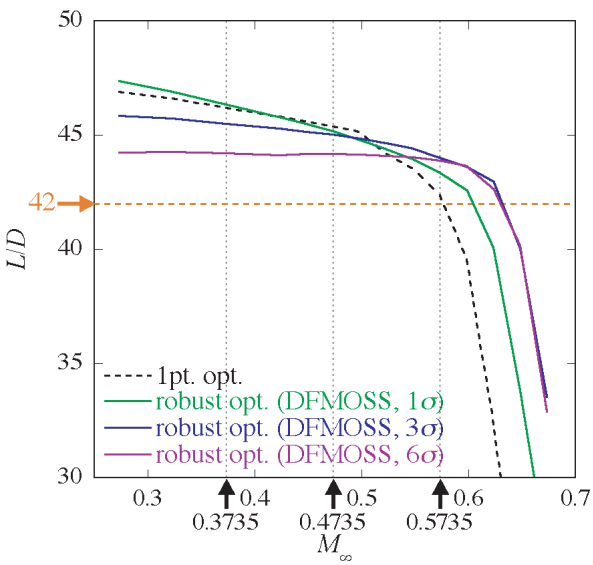


Figure 4.5 Histories of L/D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1.

Figure 4.6 compares the airfoil configurations. In the present case without the structural constraint on airfoil thickness, the airfoils of all these solutions have almost zero thickness in order to produce large difference of pressure between the upper and lower surfaces which leads to large lift. The airfoil of one-point optimal solution has the largest maximum camber to generate strong expansion region over the upper surface. The airfoils of three robust optimal solutions, on the other hand, have smaller maximum camber than that of one-point optimal solution. In addition, comparing the three robust optimal solutions, the maximum camber becomes smaller as the sigma level of L/D becomes larger.

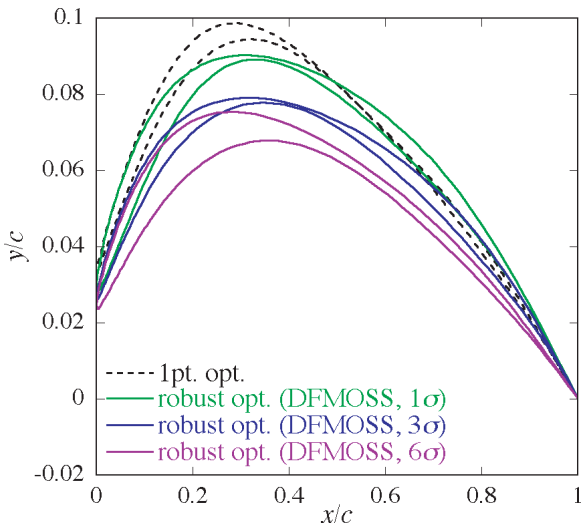


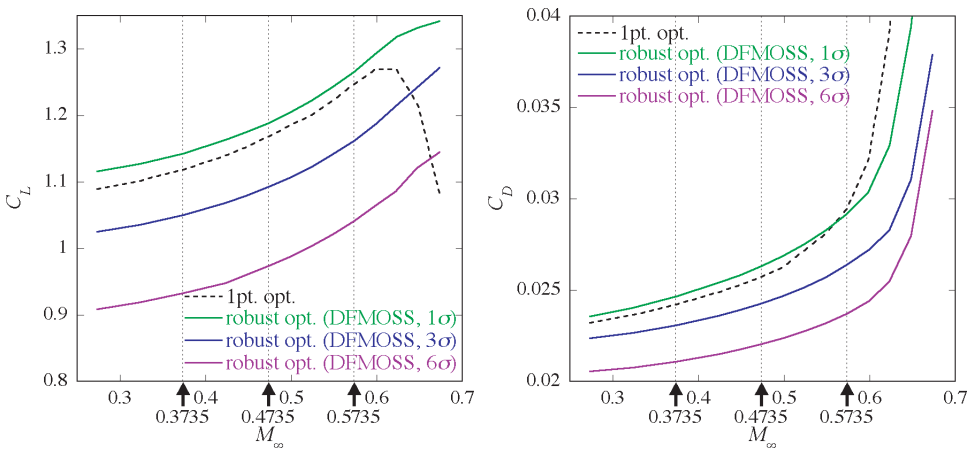
Figure 4.6 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1.

Figure 4.7 compares the histories of lift and drag coefficients C_L and C_D against M_∞ at $\alpha = 2.0$ [deg], and Table 4.5 shows the ratio of aerodynamic performance parameters L/D, C_L and C_D at $M_\infty = 0.5735$ and $\alpha = 2.0$ [deg] to those at $M_\infty = 0.3735$ and $\alpha = 2.0$ [deg]. Here, C_L and C_D are defined as

$$\begin{aligned} C_L &= \frac{L}{\frac{1}{2}\rho_\infty {u_\infty}^2 S_{ref}} \\ C_D &= \frac{D}{\frac{1}{2}\rho_\infty {u_\infty}^2 S_{ref}} \end{aligned} \tag{4.7}$$

where L and D are lift and drag, and ρ_∞ and u_∞ are the density and the velocity at freestream condition, respectively. S_{ref} is the reference area and the wing area is used as S_{ref} in the present study. Qualitative tendencies of C_L–M_∞ and C_D–M_∞ histories appear almost the same among these solutions, except for quantitative differences

of C_L and C_D at the same M_∞ among these solutions and the stall characteristic at high M_∞ of one-point optimal solution. However, Table 4.5 shows that the increment in C_D against the increment in M_∞ becomes smaller as the solution has more robust characteristic of L/D against the variation of M_∞, while the increment in C_L against the increment in M_∞ is almost the same independently of the robustness of L/D. These results indicate that the increment in C_D has more dominant effect on the decrement in L/D against the increment in M_∞, i.e., the robustness of L/D, than that of C_L.



(a) C_L. (b) C_D.

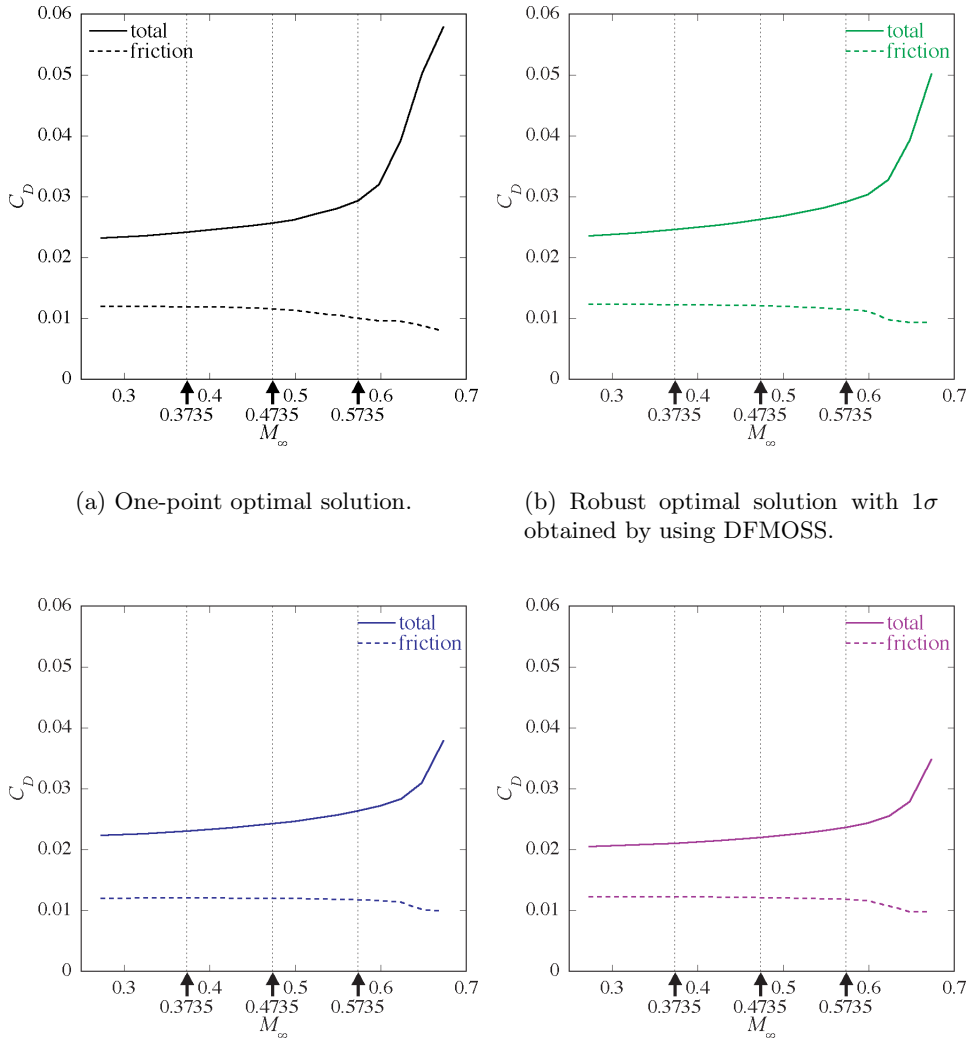
Figure 4.7 Histories of C_L and C_D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 1.

Table 4.5 Ratio of aerodynamic performance parameters at $M_\infty = 0.5735$ and $\alpha = 2.0$ [deg] to those at $M_\infty = 0.3735$ and $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 1 [%].

		L/D	C_L	C_D
One-point optimization		−8.2967	+11.514	+21.607
Robust optimization using DFMOSS	1σ	−6.4243	+10.781	+18.385
	3σ	−3.2611	+10.687	+14.416
	6σ	−0.55509	+9.7848	+10.393

Figure 4.8 shows the histories of the breakdown of C_D (total and friction drags) against M_∞ at $\alpha = 2.0$ [deg]. These solutions have the same histories of friction

drag. On the other hand, the solutions have different histories of pressure drag (total drag − friction drag), and the pressure drag becomes smaller over the whole range of M_∞ as the solution has more robust characteristic of L/D against the variation of M_∞. This result indicates that the pressure drag has more dominant effect on the robustness of L/D against the variation of M_∞ than the friction drag.



(a) One-point optimal solution.

(b) Robust optimal solution with 1σ obtained by using DFMOSS.

(c) Robust optimal solution with 3σ obtained by using DFMOSS.

(d) Robust optimal solution with 6σ obtained by using DFMOSS.

Figure 4.8 Histories of breakdown of C_D against M_∞ at $\alpha = 2.0$ [deg] of Case 1.

Figures 4.9 and 4.10 show the local mach number contour plots around the airfoil of the one-point optimal solution and the robust optimal solution with 6σ robustness quality obtained by using the DFMOSS, respectively. In the one-point optimal solu-

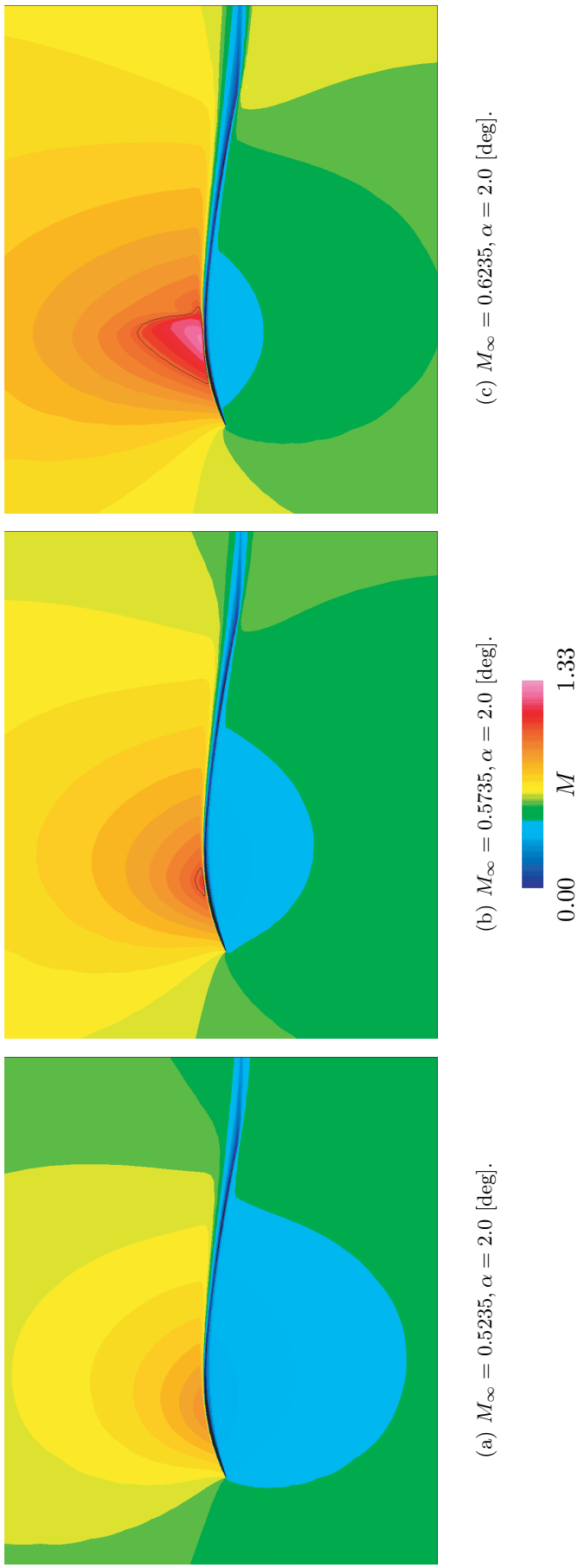
tion, supersonic expansion region is generated over the upper surface and the shock wave occurs behind this region at M_∞ of higher than 0.5735. In the robust optimal solution with 6σ robustness quality, on the other hand, supersonic expansion region is generated at M_∞ of higher than 0.6235 due to smaller maximum camber, and the shock wave generated behind this region is weaker than that of one-point optimal solution at the same M_∞. The difference of the strength of generated supersonic expansion region and shock wave is because the airfoil of robust optimal solution has smaller maximum camber than that of one-point optimal solution, as shown in Fig. 4.6.

Figure 4.11 shows the chordwise pressure coefficient C_p distributions over the airfoil surface at various freestream Mach numbers $M_\infty = 0.4735$, 0.5235, 0.5735 and 0.6235, and $\alpha = 2.0$ [deg]. Here, C_p is defined as

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty {u_\infty}^2} \tag{4.8}$$

where p is the local pressure, and p_∞ is the pressure at freestream condition. In all solutions, the expansion region over the upper surface of airfoil becomes larger, the negative pressure peak becomes higher, and the pressure recovery behind this peak becomes steeper, i.e., the shock wave behind the expansion region becomes stronger with the increment in M_∞. Among these solutions, the negative pressure peak magnitude of one-point optimal solution changes the most drastically. Comparing the three robust optimal solutions, the change of negative pressure peak becomes smaller as the sigma level of L/D becomes larger. This result indicates that the airfoil with more robust characteristic of L/D has weaker growth of the generated shock wave against the increment in M_∞.

It is well known that the wave drag is produced when the shock wave occurs, and it becomes larger as the shock wave becomes stronger. It indicates that the solution with more robust characteristic of L/D has smaller increment in wave drag against the increment in M_∞. Therefore, the whole numerical results of the present case show that an airfoil configuration with smaller maximum camber can improve the robustness of L/D against the variation of flight Mach number M_∞ because such airfoil can suppress the growth of shock wave, i.e., realize smaller increment in pressure drag (wave drag) against the increment in flight Mach number M_∞, and it results in smaller decrement in L/D against the increment in M_∞.

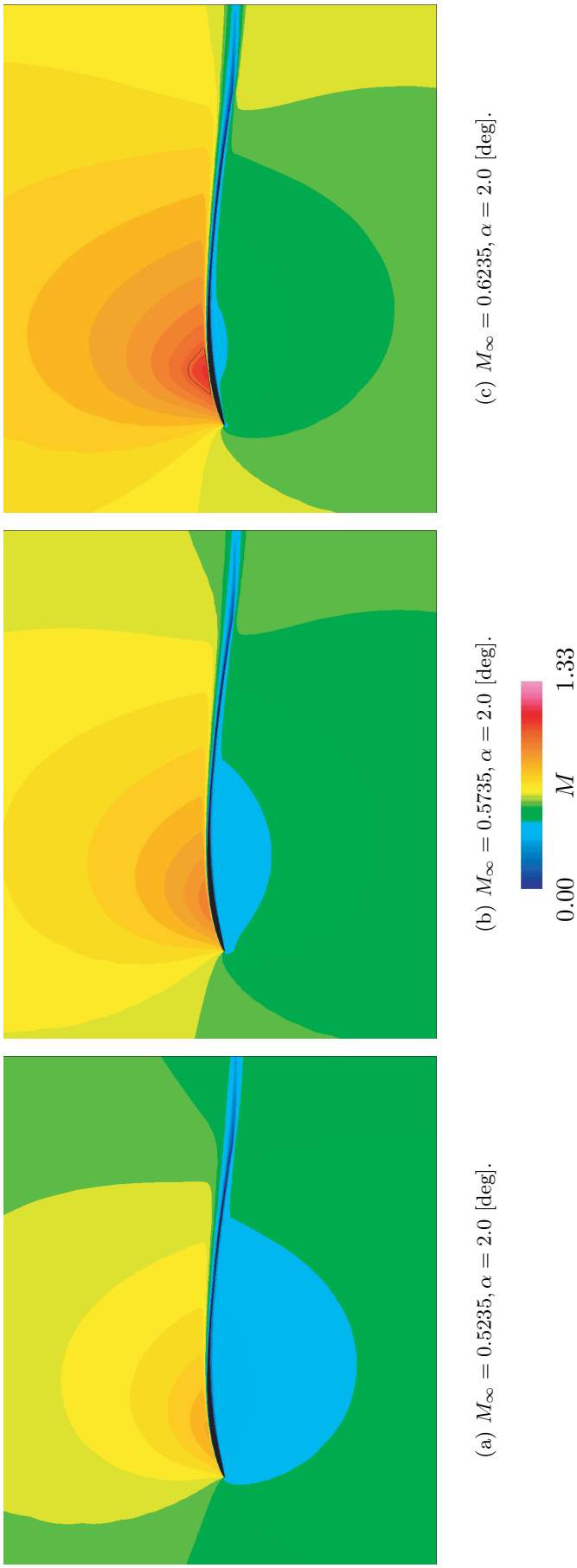


(a) $M_\infty = 0.5235, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.5735, \alpha = 2.0$ [deg].

(c) $M_\infty = 0.6235, \alpha = 2.0$ [deg].

Figure 4.9 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 1 (black solid lines are sonic lines).

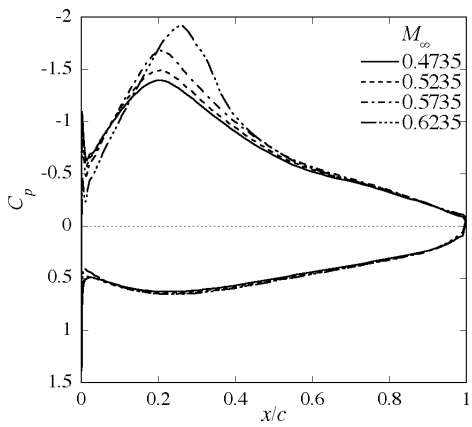


(a) $M_\infty = 0.5235, \alpha = 2.0$ [deg].

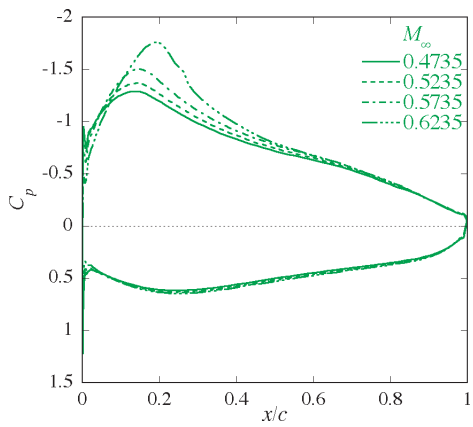
(b) $M_\infty = 0.5735, \alpha = 2.0$ [deg].

(c) $M_\infty = 0.6235, \alpha = 2.0$ [deg].

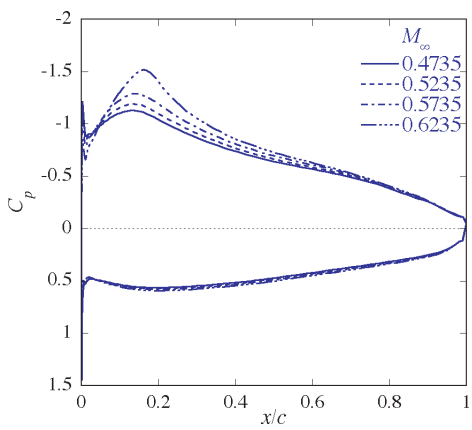
Figure 4.10 Local Mach number contour plots around the airfoil of robust optimal solution with 6σ robustness quality obtained by using DFMOSS of Case 1 (black solid lines are sonic lines).



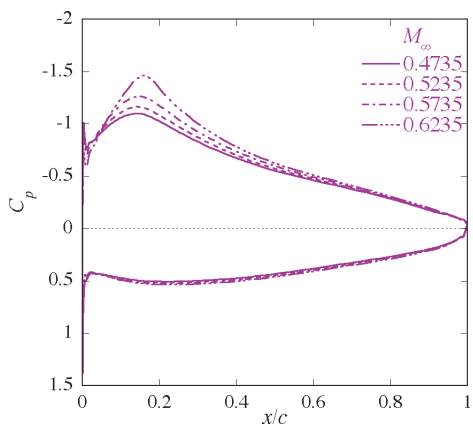
(a) One-point optimal solution.



(b) Robust optimal solution with 1σ obtained by using DFMOSS.



(c) Robust optimal solution with 3σ obtained by using DFMOSS.



(d) Robust optimal solution with 6σ obtained by using DFMOSS.

Figure 4.11 Chordwise C_p distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 1.

4.3 Case 2: Robust Design Optimization of Pitching Moment Coefficient against the Variation of Flight Mach Number

4.3.1 Design Problem Definition

In Case 1, the robust aerodynamic design optimization of airfoil configuration considering the optimality and the robustness of dispersive lift to drag ratio L/D against the dispersion of flight Mach number M_∞ was performed. However, aerodynamic stability is also an important issue from the designer's viewpoint. It is well known that the airfoil with too large lift often involves too large pitching-down moment, and it may leads to the situation where the pitching motion of main wing can not be controlled by a horizontal tail wing. In the present case, therefore, the robust aerodynamic design optimization aiming at the improvements in the optimality and the robustness of both L/D and aerodynamic stability against the dispersion of M_∞ is performed.

Consider the traditional aerodynamic design optimization problem where airfoil configuration must be optimized so that lift to drag ratio L/D will be maximized and also the absolute value of pitching moment coefficient C_{M_p} will be lower than the upper limit at design point:

$$\begin{aligned} &\text{Maximize: } L/D \\ &\text{Subject to: } \left|C_{M_p}\right| \leq 0.13 \end{aligned} \tag{4.9}$$

Here, C_{M_p} is defined as

$$C_{M_p} = \frac{M_p}{\frac{1}{2}\rho_\infty {u_\infty}^2 S_{ref} c_{ref}} \tag{4.10}$$

where M_p is pitching moment around the center of gravity of airplane. In the present study, the center of gravity is assumed to be at 25% chord position ($[x/c, y/c]_{CG} = [0.25, 0]$). S_{ref} and c_{ref} are the reference area and length, and the wing area and the root chord length are used as S_{ref} and c_{ref}, respectively. Similar to Case 1, cruising conditions of NASA's AME,[11] as shown in Table 4.1, is used as the design point in the present case, too. The upper limit of C_{M_p} as 0.13 in Eq. 4.9 indicates the maximum value of C_{M_p} which the horizontal tail wing of NASA's AME assumes to be able to generate. Now, convert this problem to the robust aerodynamic design optimization problem where the mean value of dispersive L/D must be maximized and also the sigma level of the dispersive constraint function on pitching stability

$\left|C_{M_p}\right| \leq 0.13$ must be maximized when the flight Mach number M_∞ disperses around design point:

$$\begin{aligned} \text{Maximize: } & \text{mean value of } L/D \\ & \text{sigma level of } \left|C_{M_p}\right| \leq 0.13 \end{aligned} \tag{4.11}$$

and naturally, the constraint on pitching stability $\left|C_{M_p}\right| \leq 0.13$ must be satisfied at the design point:

$$\text{Subject to: } \left|C_{M_p}\right| \leq 0.13 \tag{4.12}$$

This robust aerodynamic optimization problem aims at finding the airfoil configuration not only with maximum L/D but also with robust C_{M_p} characteristic that the Mars exploratory airplane can avoid failure in controlling its pitching motion by its horizontal tail wing when it flies in east-westward direction without side and vertical winds under the situation where westerlies as shown in Fig. 1.8 blow stably. The assumed statistical characteristics of dispersive flight Mach number M_∞ of the present case is the same as those of Case 1, as shown in Table 4.2.

Similar to Case 1, airfoil configuration is defined by the B-spline curves with six control points as well as three fixed points, as shown in Fig. 4.1, in the present case, too. The design variables and those ranges are the same as those of Case 1, as shown in Table 4.3. In addition, the geometric constraint so that the upper and lower surfaces of the airfoil do not intersect each other is imposed and the structural constraint on airfoil thickness is not considered in the present case.

4.3.2 Numerical Methods and Conditions

In the present case, three aerodynamic design optimizations are performed and the obtained numerical results are compared:

- **One-point optimization using a single-objective genetic algorithm (SOGA)[31]**

  When $M_\infty = 0.4735$ and $\alpha = 2.0$ [deg],

$$\begin{aligned} \text{Maximize: } & L/D \\ \text{Subject to: } & \left|C_{M_p}\right| \leq 0.13 \end{aligned} \tag{4.13}$$

- **Robust optimization using the SOGA coupled with the DFSS[36]**

- **Robust optimization using the MOGA coupled with the DFMOSS**

When M_∞ disperses around 0.4735 with its standard deviation of 0.1,

$$\begin{aligned} \text{Maximize: } & \text{mean value of } L/D \\ & \text{sigma level of } \left|C_{M_p}\right| \leq 0.13 \end{aligned} \tag{4.14}$$

and also when $M_\infty = 0.4735$, $\alpha = 2.0$ [deg],

$$\text{Subject to: } \left|C_{M_p}\right| \leq 0.13 \tag{4.15}$$

The optimization methods of the present case are the same as those used in Case 1. Table 4.6 shows the present optimization conditions. In the robust optimizations using DFSS, total three cases with different sigma level n of $\left|C_{M_p}\right| \leq 0.13$: 1σ, 3σ and 6σ are carried out. In the robust optimization using DFMOSS, on the other hand, only one case is carried out without the advance setting of sigma level. In addition, the methods for CFD simulation are also the same as those used in Case 1.

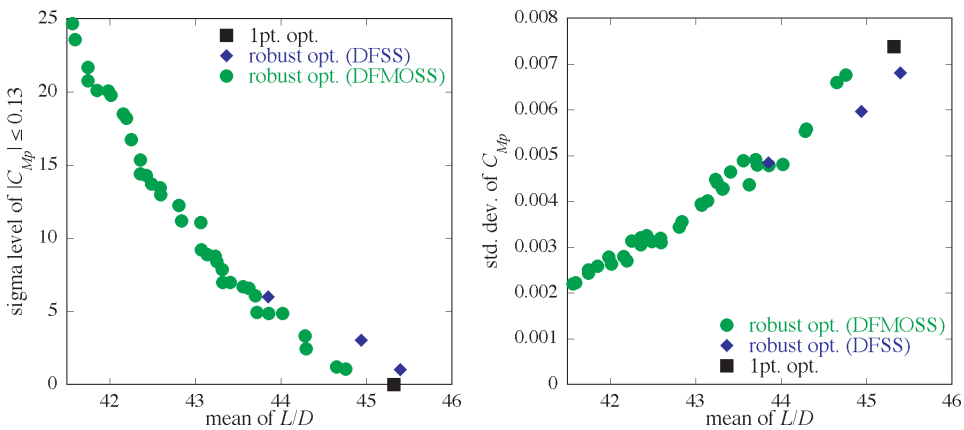
Table 4.6 Optimization conditions of Case 2.

for SOGA/MOGA	Population size	64
	Number of generations	100
	Mutation rate [%]	20
for DFSS	Sigma level n	1σ, 3σ, 6σ

4.3.3 Numerical Results

Figure 4.12 compares the robust optimal solution distributions (sigma level of $\left|C_{M_p}\right| \leq 0.13$ against mean value of L/D, and standard deviation of C_{M_p} against mean value of L/D) obtained by using the DFSS and the DFMOSS with the optimal solution distributions obtained by the one-point optimization neglecting the robustness of C_{M_p}. Compared to the robust optimal solutions, the one-point optimal solution has 0σ robustness quality and larger standard deviation of C_{M_p}. It indicates that the one-point optimization can not find a solution with robust characteristic of C_{M_p}. The robust optimizations using the DFSS can find the robust optimal solutions with smaller standard deviation of C_{M_p} than that of one-point optimal solution, and 1σ, 3σ and 6σ robustness qualities, respectively. However these solutions distribute narrowly and sparsely because the specified sigma levels are discrete integers. This

indicates that the DFSS has lack in capability of revealing the global and uniform trade-off relation between optimality (mean value of L/D) and robustness (standard deviation of C_{M_p} or sigma level of $\left|C_{M_p}\right| \leq 0.13$). On the other hand, the robust optimization using the DFMOSS can successfully and effectively find total forty robust optimal solutions by only one optimization run, and these solutions distribute globally and uniformly. It indicates that the DFMOSS can give rich trade-off information between optimality and robustness which various designers can utilize flexibly.



(a) Sigma level of $\left|C_{M_p}\right| \leq 0.13$ against mean value of L/D.

(b) Standard deviation of C_{M_p} against mean value of L/D.

Figure 4.12 Robust optimal solution distributions of Case 2.

Hereafter, three robust optimal solutions with 1σ, 4σ and 8σ robustness qualities obtained by using the DFMOSS, as shown in Fig. 4.13, are compared with the one-point optimal solution. Figures 4.14(a) and 4.14(b) show the histories of L/D and C_{M_p} against M_∞ at $\alpha = 2.0$ [deg]. The one-point optimal solution has the largest L/D over almost the whole range of M_∞, and its constraint on pitching stability becomes active, i.e., $C_{M_p} = -0.13$ at the design point $M_\infty = 0.4735$. C_{M_p} of this solution decreases with the increment in M_∞ and falls below its lower limit -0.13. Therefore, in this solution, the constraint on pitching stability is not satisfied at high M_∞. On the other hand, three robust optimal solutions have smaller L/D over almost the whole range of M_∞ but a little more stable characteristics of C_{M_p} against the increment in M_∞ than the one-point optimal solution, and also those histories of C_{M_p} against M_∞ shift up from its lower limit. Therefore, in these solutions, the constraint on pitching stability is satisfied even at high M_∞. In addition, comparing the three robust optimal solutions, the solution with larger sigma level of $\left|C_{M_p}\right| \leq$

0.13 is easier to satisfy the constraint on pitching stability even at high M_∞. These results indicate that the robust aerodynamic design optimization using the DFMOSS can actually find the multiple designs with various robust characteristics of C_{M_p} against the variation of M_∞ by only one optimization run.

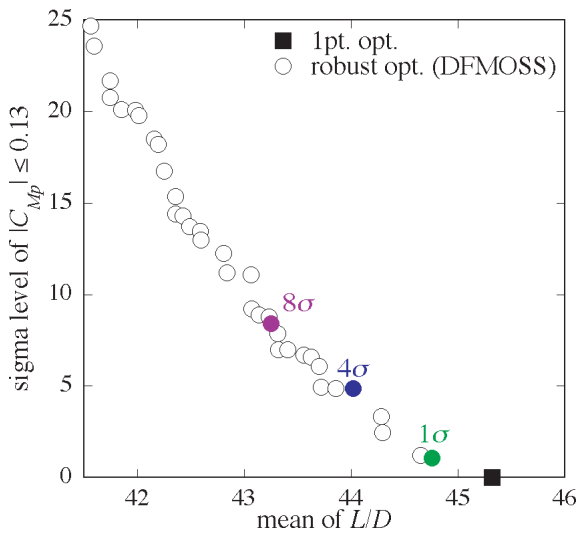
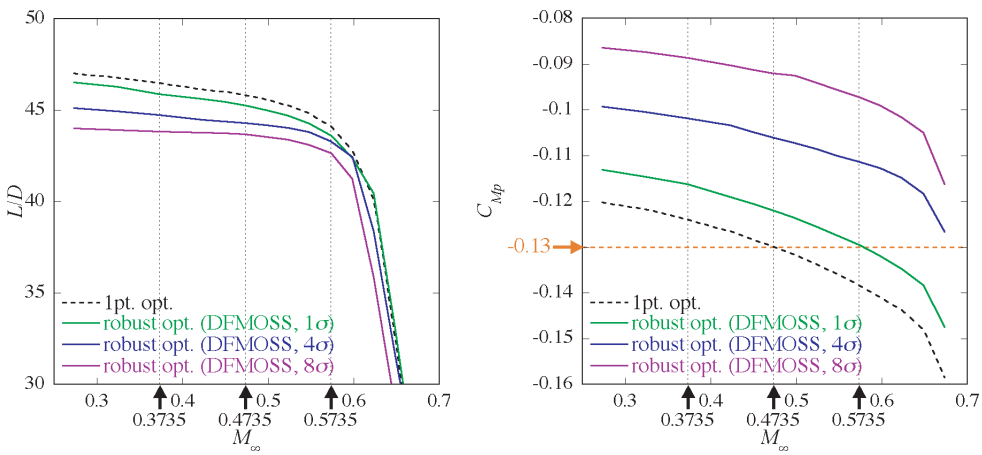


Figure 4.13 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 2.

Figure 4.15 compares the airfoil configurations. Similar to Case 1, the airfoils of all these solutions have small thickness and large maximum camber in order to produce large lift. However, in the present case, it is different from Case 1 that the airfoils with the robust optimal solutions are folded down at the front part (about 15 % chord position). In addition, comparing the three robust optimal solutions, the airfoil is folded down more greatly as the sigma level of $\left|C_{M_p}\right| \leq 0.13$ becomes larger.

Figure 4.16 compares the histories of lift and drag coefficients C_L and C_D against M_∞ at $\alpha = 2.0$ [deg]. Qualitative tendencies of C_L–M_∞ and C_D–M_∞ histories are the same among these solutions, except for quantitative differences of C_L and C_D at the same M_∞ among these solutions. Differently from Case 1, the increments in C_L and C_D have almost the same effects on decrement in L/D against the increment in M_∞. In addition, there exists no difference in the histories of the breakdown of C_D (pressure and friction drags) against M_∞ among these solutions.

Figures 4.17 and 4.18 show the local mach number contour plots around the airfoil of the one-point optimal solution and the robust optimal solution with 8σ robustness quality obtained by using the DFMOSS, respectively. Comparing the



(a) L/D. (b) C_{M_p}.

Figure 4.14 Histories of L/D and C_{M_p} against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2.

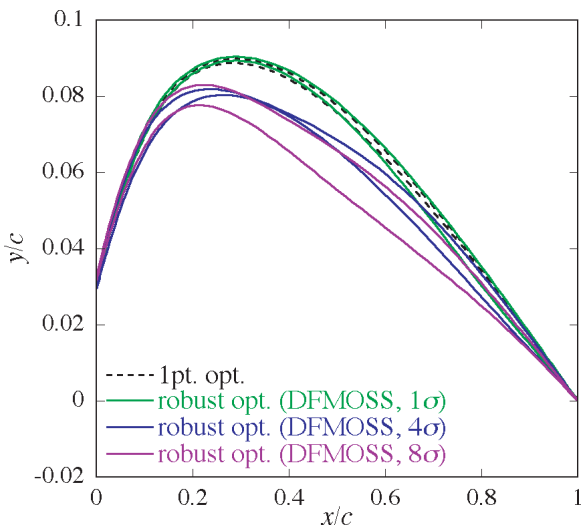
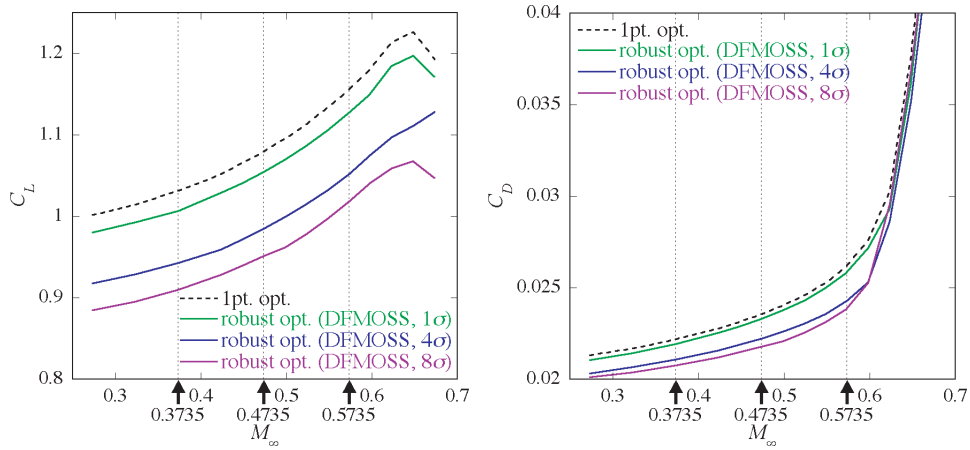


Figure 4.15 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2.

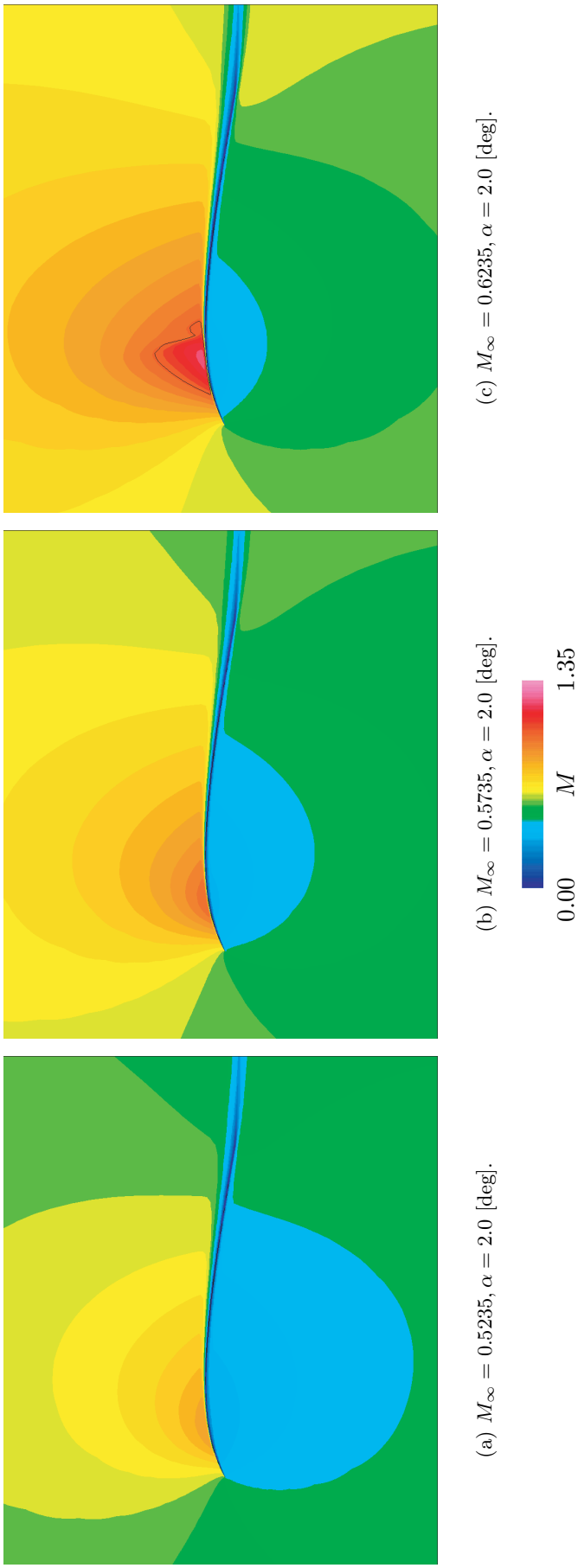


(a) C_L. (b) C_D.

Figure 4.16 Histories of C_L and C_D against M_∞ at $\alpha = 2.0$ [deg] of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 2.

contour plots of these solutions at $M_\infty = 0.5735$, small supersonic expansion region is generated over the upper surface of the airfoil of the robust optimal solution with 8σ robustness quality, while there exist no supersonic region around the airfoil of the one-point optimal solution. Comparing the contour plots of these solutions at $M_\infty = 0.6235$, on the other hand, supersonic expansion regions are generated over the upper surfaces of the airfoils of both solutions, and the generated supersonic expansion region of the one-point optimal solution is a little larger than that of the robust optimal solution with 8σ robustness quality. From these results, it is expected that the airfoil with more robust characteristic of C_{M_p} has smaller backward movement of generated shock wave against the increment in M_∞.

Figure 4.19 shows the chordwise pressure coefficient C_p distributions over the airfoil surface at various freestream Mach numbers $M_\infty = 0.4735$, 0.5235, 0.5735 and 0.6235, and $\alpha = 2.0$ [deg]. Similar to Case 1, in all solutions, the expansion region over the upper surface of airfoil becomes larger, the negative pressure peak becomes higher, and the pressure recovery behind this peak becomes steeper, i.e., the shock wave behind the expansion region becomes stronger with the increment in M_∞. However, it is different from Case 1 that the changes of the negative pressure peak magnitudes against the increment in M_∞ are almost the same among these solutions. On the other hand, the position of negative pressure peak of one-point optimal solution moves backward the most drastically. Comparing the three robust

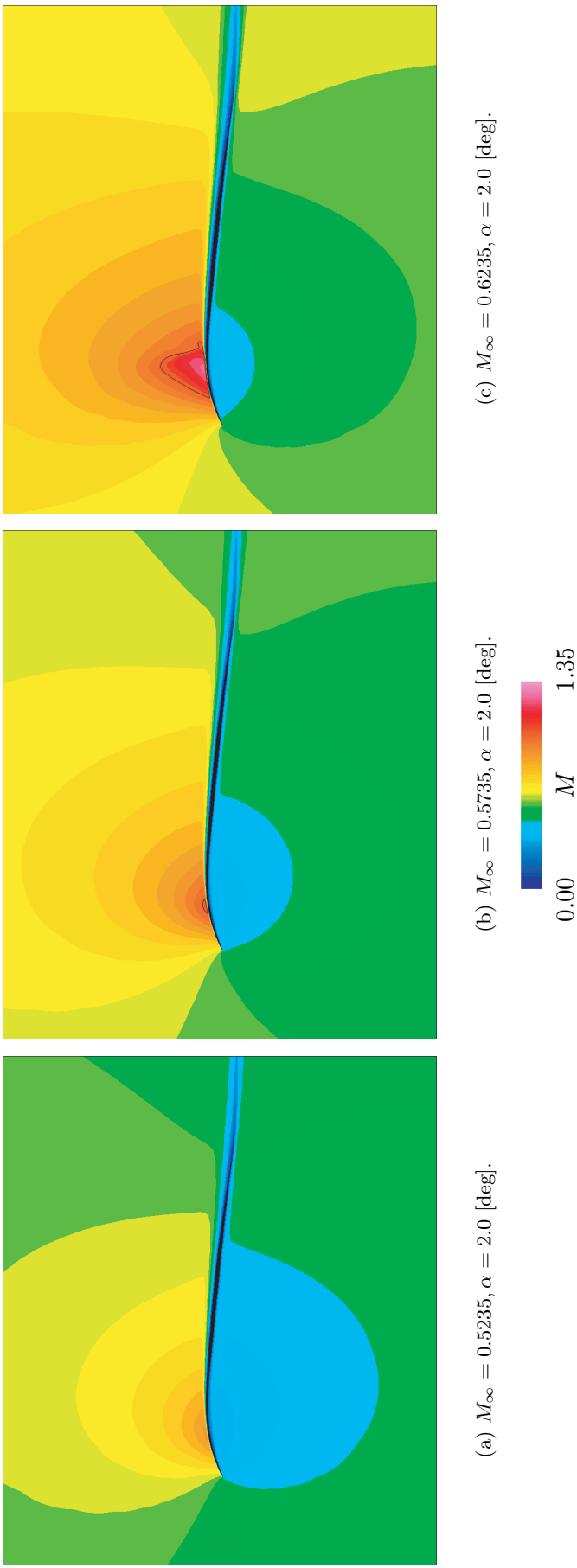


(a) $M_\infty = 0.5235, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.5735, \alpha = 2.0$ [deg].

(c) $M_\infty = 0.6235, \alpha = 2.0$ [deg].

Figure 4.17 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 2 (black solid lines are sonic lines).



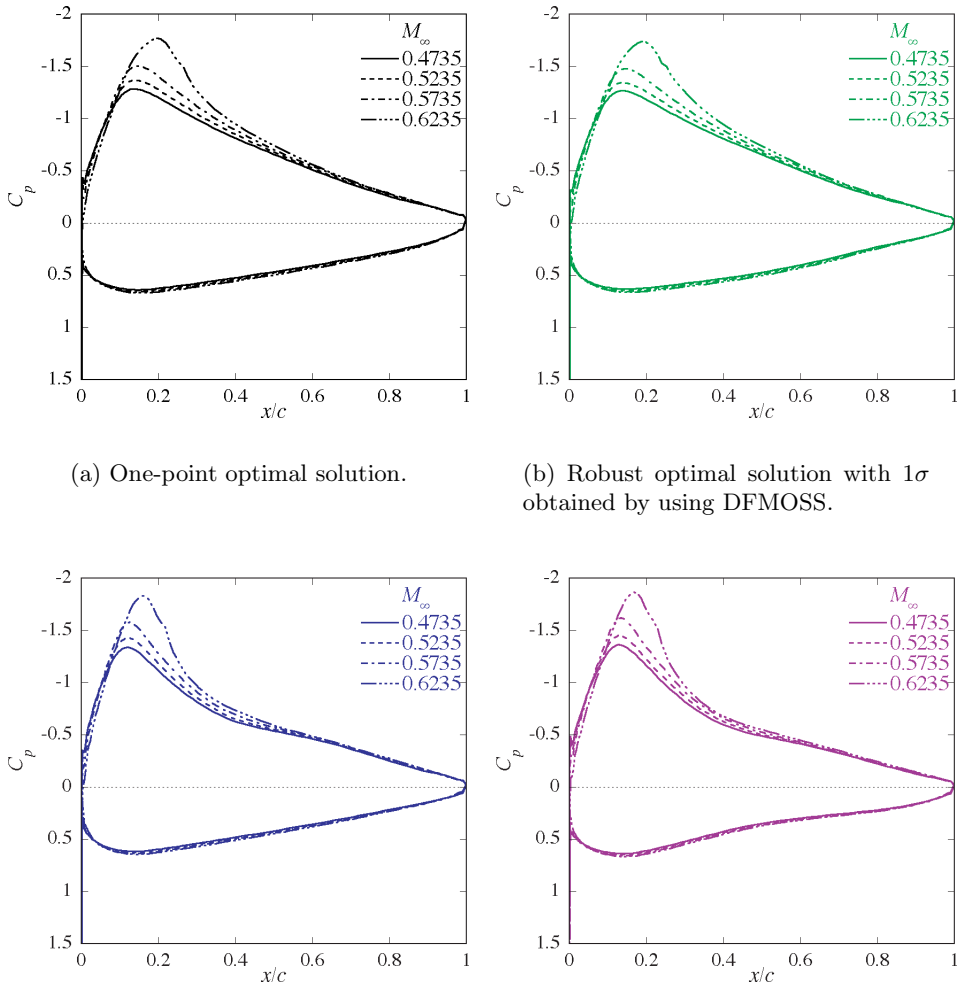
(a) $M_\infty = 0.5235, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.5735, \alpha = 2.0$ [deg].

(c) $M_\infty = 0.6235, \alpha = 2.0$ [deg].

Figure 4.18 Local Mach number contour plots around the airfoil of robust optimal solution with 8σ robustness quality obtained by using DFMOSS of Case 2 (black solid lines are sonic lines).

optimal solutions, the backward movement of negative pressure peak becomes smaller as the sigma level of $\left|C_{M_p}\right| \leq 0.13$ becomes larger. This is because the airfoil with more robust characteristic of C_{M_p} against the variation of M_∞ has larger curvature locally at the front part due to its folded-down configuration, as shown in Fig. 4.15, and it can suppress the backward movement of shock wave occurred over the upper surface more forcedly than that of one-optimal solution.



(a) One-point optimal solution.

(b) Robust optimal solution with 1σ obtained by using DFMOSS.

(c) Robust optimal solution with 4σ obtained by using DFMOSS.

(d) Robust optimal solution with 8σ obtained by using DFMOSS.

Figure 4.19 Chordwise C_p distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 2.

Figure 4.20 shows the chordwise pressure coefficient $C_p \times$ moment arm $(x - x_{CG})/c$ distributions over the airfoil surface at various freestream Mach numbers $M_\infty =$ 0.4735, 0.5235, 0.5735 and 0.6235, and $\alpha = 2.0$ [deg] of the three robust optimal

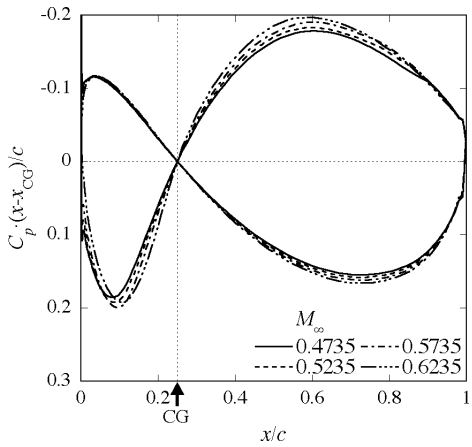
solutions obtained by using the DFMOSS and the one-point optimal solution. In these figures, the area between upper and lower lines represents the pitching moment coefficient of the airfoil, and the areas lower and greater than x_{CG} correspond to the pitching-up and pitching-down moments, respectively. Among these solutions, the one-point optimal solution has the greatest increment in the pitching-down moment produced in the rear part of airfoil with the increment in M_∞ due to large movement of shock wave occurred over the upper surface of the airfoil. On the other hand, the three robust optimal solutions have smaller changes in pitching-down moments against the increment in M_∞ than that of one-point optimal solution. Comparing the three robust optimal solutions, the solution with more robust C_{M_p} characteristic against the variation of M_∞ has smaller change in pitching-down moment.

The whole numerical results of the present case show that an airfoil configuration which is folded-down in the front part can improve the robustness of pitching moment coefficient C_{M_p} against the variation of flight Mach number M_∞. This is because such airfoil configuration involves large curvature locally at its folded-down position and its locally large curvature can suppress the backward movement of shock wave occurred over the upper surface of airfoil against the increment in flight Mach number M_∞. Eventually, it results in smaller change in pitching-down moment produced in the rear part of airfoil against the increment in M_∞.

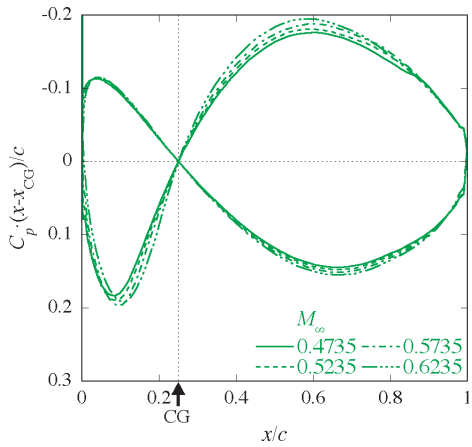
4.4 Case 3: Robust Design Optimization of Lift to Drag ratio against the Variation of Angle of Attack

4.4.1 Design Problem Definition

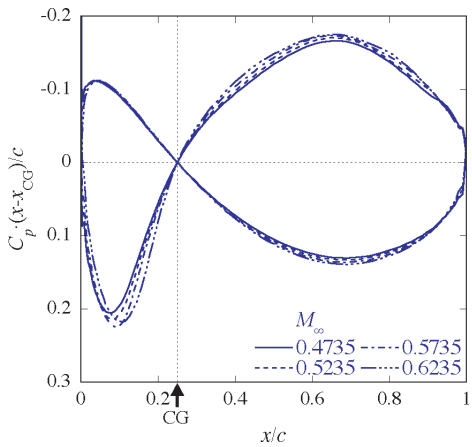
In Cases 1 and 2, the robust aerodynamic design optimizations of airfoil configuration considering the optimality and the robustness of the dispersive aerodynamic performance parameters such lift to drag ratio L/D and pitching stability against the dispersion of flight Mach number M_∞ were performed. However, there exists not only the flight Mach number M_∞ but also the angle of attack α as the dispersive flight condition parameters. It is known that the westerlies can produce not only the change of east-westward wind speeds but also that of vertical wind speeds caused by so-called the forced planetary wave, as shown in Fig. 1.9, when these blow over an hardly undulating Martian surface with mountains and craters. It is expected that the most robust design against the dispersion of M_∞ is different from the most robust one against the dispersion of α. Therefore, in the present case, the



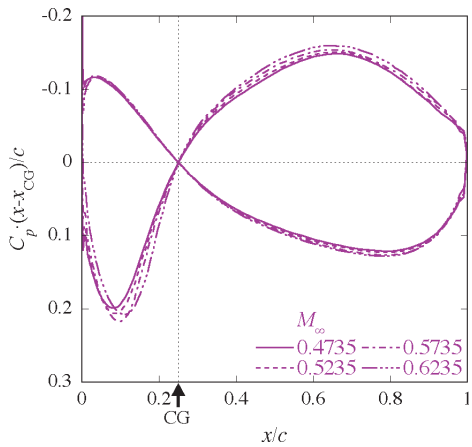
(a) One-point optimal solution.



(b) Robust optimal solution with 1σ obtained by using DFMOSS.



(c) Robust optimal solution with 4σ obtained by using DFMOSS.



(d) Robust optimal solution with 8σ obtained by using DFMOSS.

Figure 4.20 Chordwise $C_p \times$ moment arm distributions over the airfoil surface at various M_∞ and $\alpha = 2.0$ [deg] of Case 2.

robust aerodynamic design optimization considering the optimality and the robustness of dispersive aerodynamic performance parameters against the dispersion of α is performed.

Similar to Case 2, consider the traditional aerodynamic design optimization problem where airfoil configuration must be optimized so that lift to drag ratio L/D will be maximized and also the absolute value of pitching moment coefficient C_{M_p} will be lower than 0.13 at design point, as shown in Eq. 4.9. Similar to Cases 1 and 2, cruising conditions of NASA's AME,[11] as shown in Table 4.1, is used as the design point in the present case, too. Now, convert this problem to the robust aerodynamic design optimization problem where the mean value of dispersive L/D must be maximized and also the standard deviation of dispersive L/D must be minimized when the angle of attack α disperses around design point:

$$\begin{aligned} &\text{Maximize: mean value of } L/D \\ &\text{Minimize: standard deviation of } L/D \end{aligned} \tag{4.16}$$

and naturally, the constraint on pitching stability $\left|C_{M_p}\right| \leq 0.13$ must be satisfied at the design point:

$$\text{Subject to: } \left|C_{M_p}\right| \leq 0.13 \tag{4.17}$$

This robust aerodynamic optimization problem aims at finding the airfoil configuration not only with maximum L/D but also with robust L/D characteristic that the Mars exploratory airplane can avoid failure in flying over an expected range when it flies in east-westward direction without side winds under the situation where the vertical winds caused by the forced planetary wave as shown in Fig. 1.9 blow stably. Table 4.7 shows the assumed statistical characteristics of dispersive angle of attack α for NASA's AME. The value of 3.0 deg as the standard deviation of α corresponds to the variation of vertical wind speeds of about 5.5 m/s for NASA's AME with the flight Mach number of 0.4735. This value is smaller than its expected value (maximum 20 m/s), as described in Sec. 1.2. However, there exists no evident measurement data about the forced planetary wave on the Mars. Therefore, in the present study, the value 3.0 deg is adopted as the standard deviation of dispersive α to avoid overestimating the robustness of dispersive aerodynamic performance against the dispersion of α.

Similar to Cases 1 and 2, airfoil configuration is defined by the B-spline curves with six control points as well as three fixed points in the present case. However, it

Table 4.7 Statistical characteristics of dispersive angle of attack α for NASA's AME [deg].

Mean value	2.0
Standard deviation	3.0

is different from Cases 1 and 2 that the generated B-spline curves are rotated around the trailing edge so that x and y coordinates of leading edge become $[0, 0]$ exactly, as shown in Fig. 4.21. The design variables and those ranges are the same as those of Cases 1 and 2, as shown in Table 4.3. In addition, the geometric constraint so that the upper and lower surfaces of the airfoil do not intersect each other is imposed and the structural constraint on airfoil thickness is not considered in the present case.

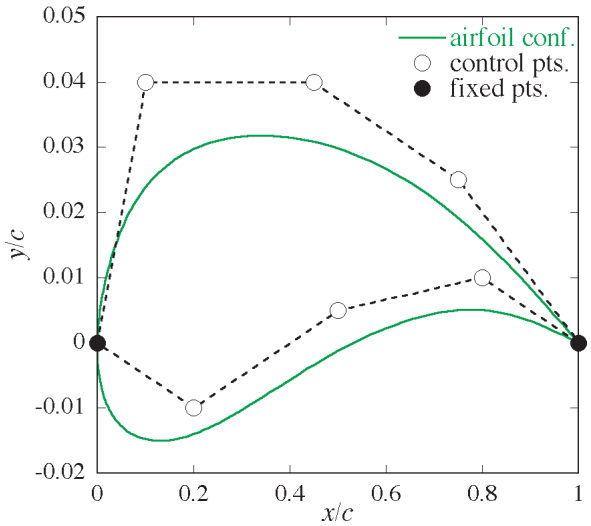


Figure 4.21 Definition of airfoil configuration based on the B-spline curves of Case 3.

4.4.2 Numerical Methods and Conditions

In the present case, two aerodynamic design optimizations are performed and the obtained numerical results are compared:

- **One-point optimization using a single-objective genetic algorithm (SOGA)[31]**

  When $M_\infty = 0.4735$ and $\alpha = 2.0$ [deg],

$$\begin{aligned} \text{Maximize: } & L/D \\ \text{Subject to: } & \left|C_{M_p}\right| \leq 0.13 \end{aligned} \tag{4.18}$$

- **Robust optimization using the MOGA coupled with the DFMOSS**

When α disperses around 2.0 deg with its standard deviation of 3.0 deg,

$$\begin{aligned} \text{Maximize: mean value of } L/D \\ \text{Minimize: standard deviation of } L/D \end{aligned} \tag{4.19}$$

and also when $M_\infty = 0.4735$ and $\alpha = 2.0$ [deg],

$$\text{Subject to: } \left|C_{M_p}\right| \leq 0.13 \tag{4.20}$$

The optimization methods of the present case are the same as those used in Cases 1 and 2. In the robust aerodynamic design optimization using DFMOSS, the statistical values of L/D are evaluated by the second-order Taylor's series expansion approach, i.e., three CFD simulations at $\alpha = -1.0$, 2.0 and 5.0 [deg] are required for one evaluation of statistical values. Table 4.8 shows the present optimization conditions. In the robust optimization using DFMOSS, it is not necessary to set the sigma level in advance, and only one case is carried out without the advance setting of weighting factors.

Table 4.8 Optimization conditions of Case 3.

for SOGA/MOGA	Population size	64
	Number of generations	100
	Mutation rate [%]	20

4.4.3 Numerical Results

Figure 4.22 compares the robust optimal solution distributions (standard deviation against mean value of L/D) obtained by using the DFMOSS with the optimal solution distributions obtained by the one-point optimization neglecting the robustness of L/D. Compared to the robust optimal solutions, the one-point optimal solution has larger standard deviation of L/D. It indicates that the one-point optimization can not find a solution with robust characteristic of L/D. On the other hand, the robust optimization using the DFMOSS can successfully and effectively find total forty-four robust optimal solutions by only one optimization run, and these solutions distribute globally and uniformly. Also it is understood easily that the maximum sigma level of L/D of the obtained robust optimal solutions is more than 1σ by the post-evaluation when the lower specification limit of L/D is set as 20. It indicates

that the DFMOSS can reveal rich trade-off information between optimality (mean value of L/D) and robustness (standard deviation of L/D) which various designers can utilize flexibly.

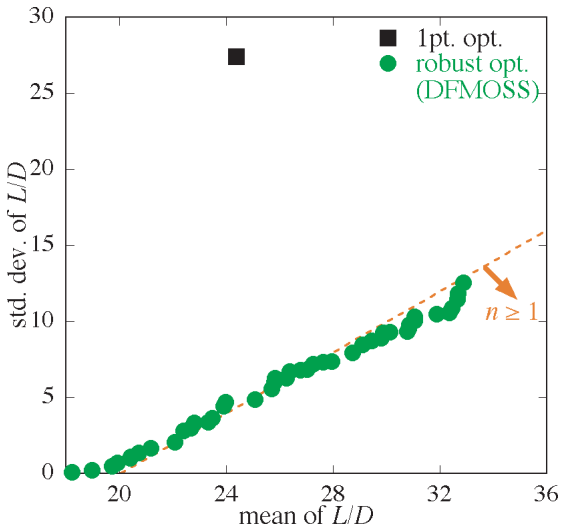


Figure 4.22 Robust optimal solution distributions of Case 3.

Hereafter, three robust optimal solutions with various robustness qualities (larger, medium and smaller standard deviations of dispersive L/D against the variation of α) obtained by using the DFMOSS, as shown in Fig. 4.23, are compared one another with the one-point optimal solution. Figure 4.24 shows the histories of L/D against α at $M_\infty = 0.4735$ of. The one-point optimal solution has the largest L/D at the design point $\alpha = 2.0$ [deg]. However, L/D of this solution decreases drastically with the increment or decrement in α from the design point, and it falls below its lower specification limit 20 at high or low α. On the other hand, three robust optimal solutions have smaller L/D at the design point but more stable characteristics of L/D against the variation of α than the one-point optimal solution, and keep large L/D over almost the whole range of α. In addition, comparing the three robust optimal solutions, the solution with smaller standard deviation of L/D has more stable characteristic of L/D against the variation of α. These results indicate that the robust aerodynamic design optimization using the DFMOSS can actually find the multiple designs with various robust characteristics of L/D against the variation of α by only one optimization run.

Figure 4.25 compares the airfoil configurations. Similar to Cases 1 and 2, the airfoil of one-point optimal solution has almost zero thickness and large maximum camber in order to produce large lift. The airfoils of three robust optimal solutions,

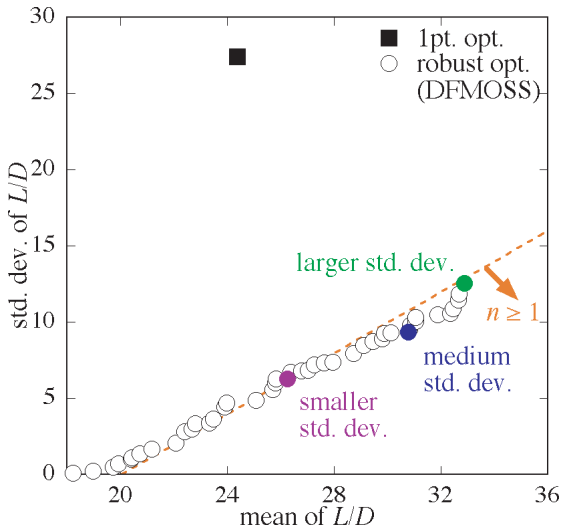


Figure 4.23 Compared three robust optimal solutions obtained by using the DFMOSS and one-point optimal solution of Case 3.

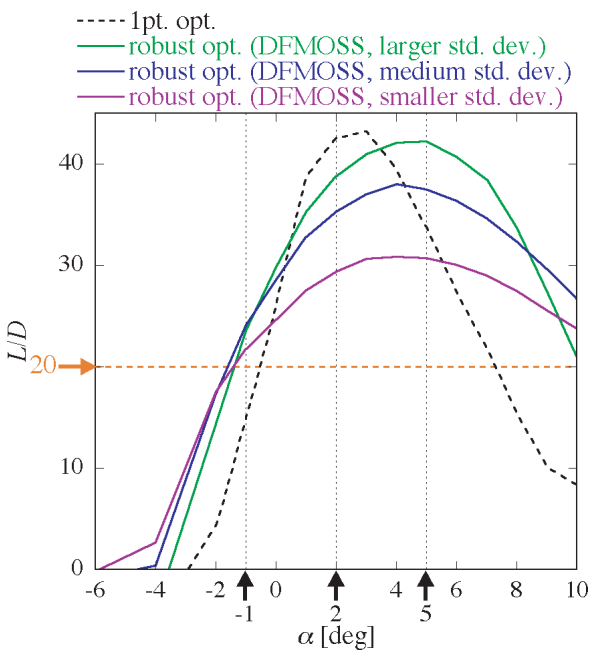


Figure 4.24 Histories of L/D against α at $M_\infty = 0.4735$ of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3.

on the other hand, have larger thickness and blunter leading edge than that of one-point optimal solution. In addition, comparing the three robust optimal solutions, the leading edge of airfoil becomes blunter as the standard deviation of L/D against the variation of α becomes smaller. Furthermore, only the robust optimal solution has blunt trailing edge. This is to reduce the pitching-down moment produced in the rear part of airfoil so that the constraint on pitching stability $\left|C_{M_p}\right| \leq 0.13$ will be satisfied at the design point.

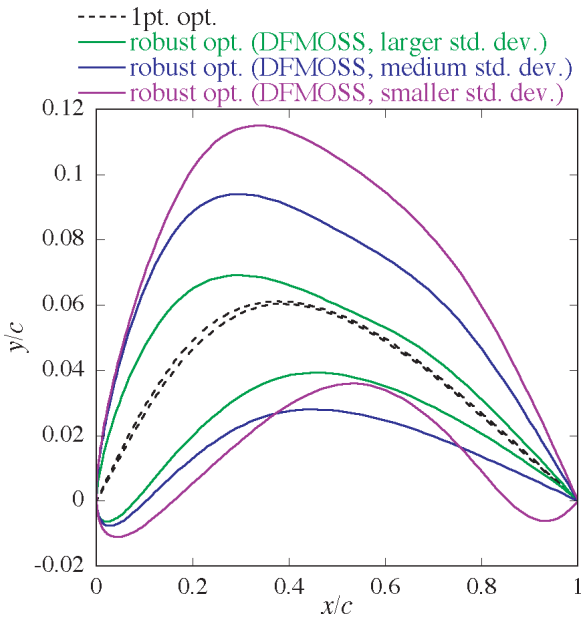
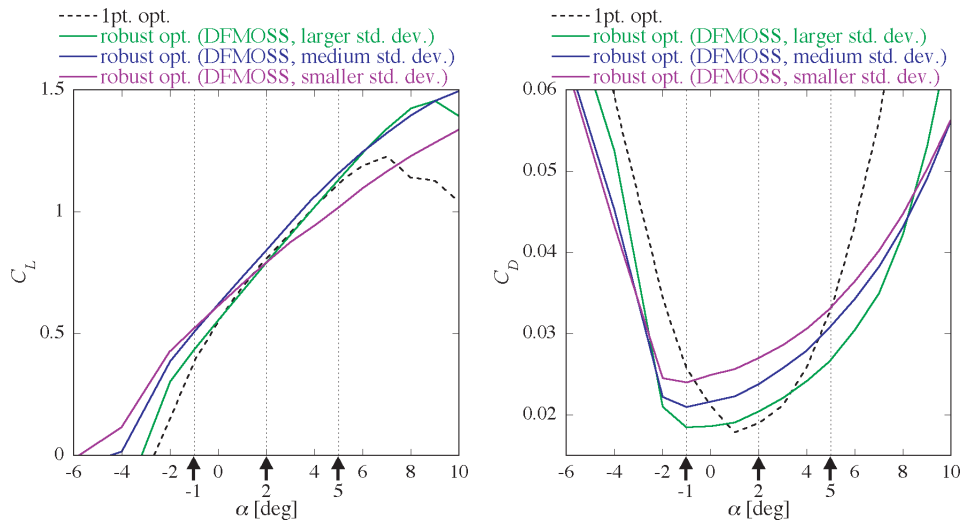


Figure 4.25 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3.

Figure 4.26 compares the histories of lift and drag coefficients C_L and C_D against α at $M_\infty = 0.4735$. Firstly, comparing C_L–α histories, these solutions have almost the same C_L at the design point $\alpha = 2.0$ [deg]. However, the airfoil of one-point optimal solution stalls at high α. In the three robust optimal solutions, on the other hand, C_L keeps increasing with the increment in α. Next, comparing C_D–α histories, the one-point optimal solution has the smallest C_D at the design point, but C_D increases drastically against the variation of α. The three robust optimal solutions, on the other hand, have larger C_D at the design point but more stable characteristic of C_D against the variation of α than the one-point optimal solution. In addition, comparing the three robust optimal solutions, C_D characteristic becomes more stable as the solution has more robust characteristic of L/D against the variation of α.

Figure 4.27 shows the histories of the breakdown of C_D (total and friction drags)

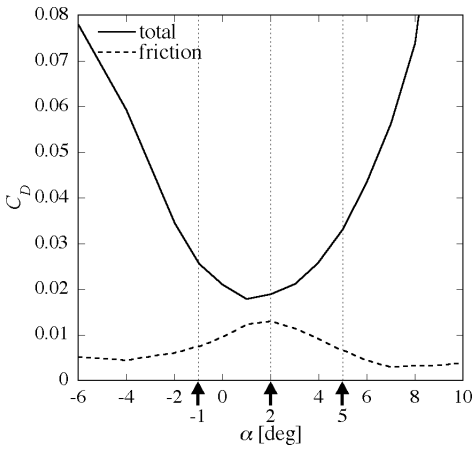


(a) C_L. (b) C_D.

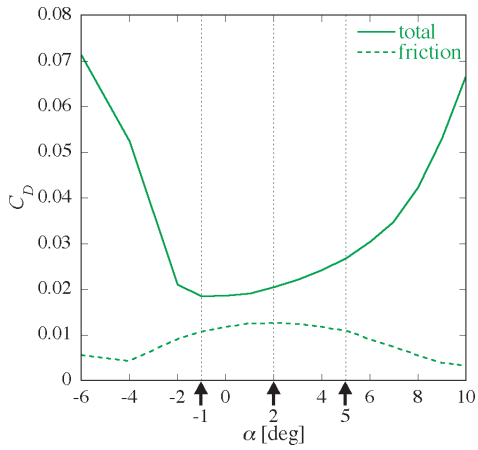
Figure 4.26 Histories of C_L and C_D against α at $M_\infty = 0.4735$ of three robust optimal solutions obtained by using DFMOSS and one-point optimal solution of Case 3.

against α at $M_\infty = 0.4735$. The friction drag becomes a little larger at high α as the solution has more robust characteristic of L/D against the variation of α. On the other hand, the pressure drag becomes considerably smaller as the solution has more robust characteristic of L/D against the variation of α. This result indicates that the pressure drag has more dominant effect on the robustness of L/D against the variation of α than the friction drag.

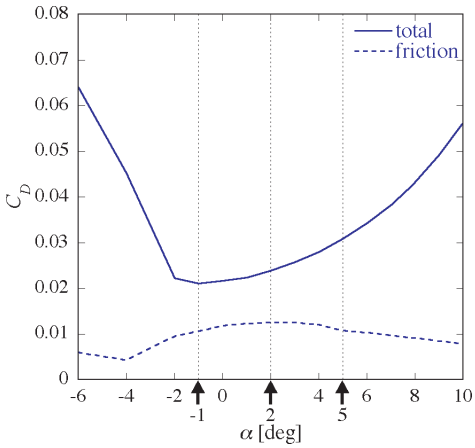
Figures 4.28 and 4.29 show the local mach number contour plots around the airfoil, and Figs 4.30 and 4.31 show the streamline patterns near the leading edge of the airfoil of the one-point optimal solution and the robust optimal solution with smaller standard deviation of L/D obtained by using the DFMOSS, respectively. In these solutions, there exists no supersonic region and no shock wave at any α, as seen in Cases 1 and 2. In the one-point optimal solution, the flow is fully attached at the design point $\alpha = 2.0$ [deg], but the flow begins to separate from the leading edge and reattached to the upper surface of airfoil at $\alpha = 5.0$ [deg]. This phenomenon produces the separation bubble near the leading edge, and the separation bubble becomes longer in the downstream direction as α becomes larger. In the robust optimal solution with smaller standard deviation of L/D, on the other hand, the flow near the leading edge is attached and the separation bubble is not generated



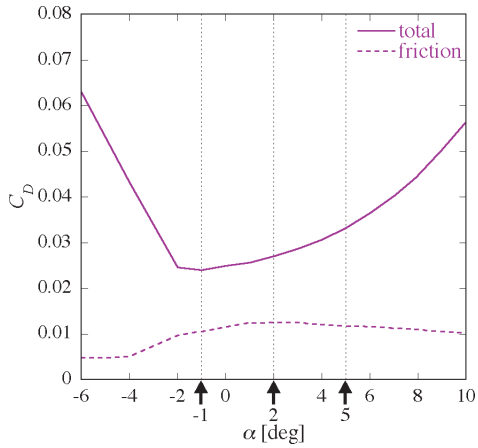
(a) One-point optimal solution.



(b) Robust optimal solution with larger standard deviation of L/D obtained by using DFMOSS.



(c) Robust optimal solution with medium standard deviation of L/D obtained by using DFMOSS.



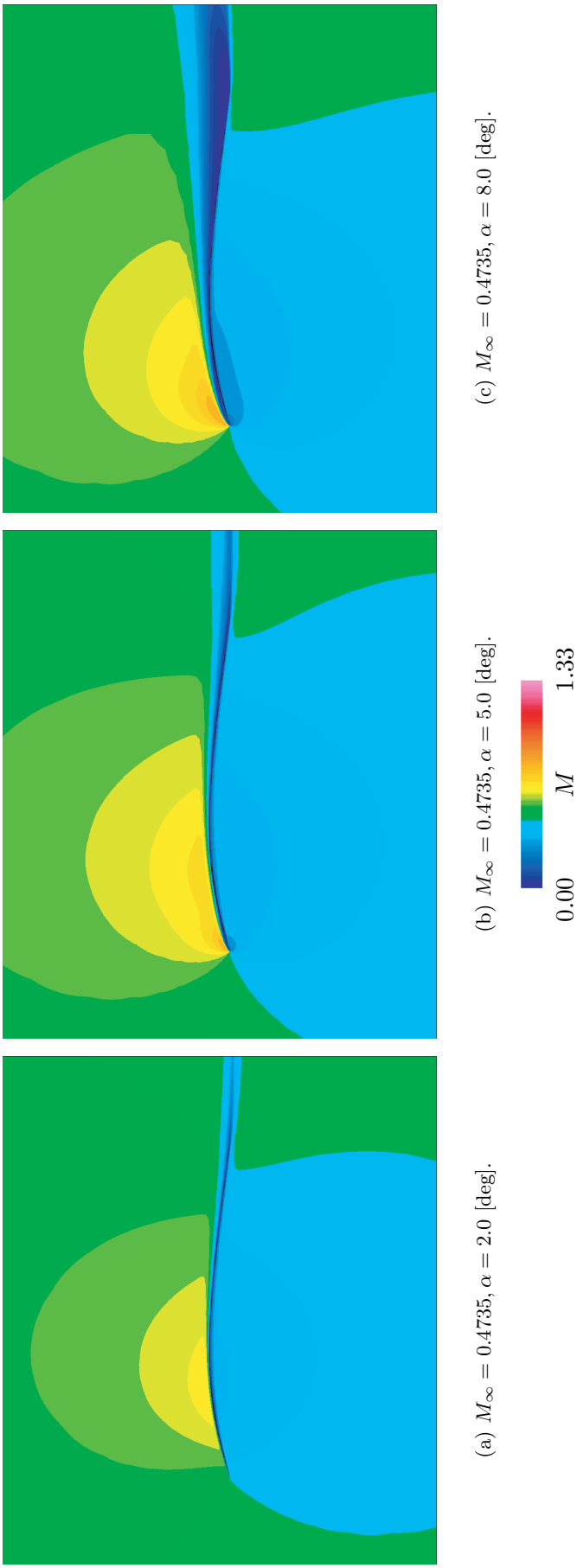
(d) Robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS.

Figure 4.27 Histories of breakdown of C_D against α at $M_\infty = 0.4735$ of Case 3.

at any α. This is because the airfoil of robust optimal solution has blunter leading edge than that of one-point optimal solution.

Figure 4.32 shows the chordwise pressure coefficient C_p distributions over the airfoil surface at various angle of attack $\alpha = 2.0$, 4.0, 6.0 and 8.0 [deg], and $M_\infty = 0.4735$. In all solutions, the expansion region over the upper surface of airfoil becomes larger, and the negative pressure peak becomes higher with the increment in α. In the one-point optimal solution, the negative pressure level changes drastically in the front part near the leading edge. In the three robust optimal solutions, on the other hand, the changes of negative pressure level in the front part near the leading edge are gentler than that of one-point optimal solution. Comparing the three robust optimal solutions, the change of negative pressure level in the front part near the leading edge becomes gentler as the standard deviation of L/D against the variation of α becomes smaller. Figure 4.33 compares the chordwise C_p distributions over the airfoil surface at $\alpha = 2.0$ and 6.0 [deg] between the robust optimal solution with smaller standard deviation of L/D and the one-point optimal solution. This figure clearly shows that the robust optimal solution with smaller standard deviation has gentler change of negative pressure level in the front part near the leading edge with the increment in α. This is because the airfoil with solution with more robust characteristic of L/D against the variation of α has gentler leading edge, as shown in Fig.4.25, and it can suppress the growth of separation bubble generated near the leading edge against the increment in α.

The whole numerical results of the present case show that an airfoil configuration with blunter leading edge can improve the robustness of lift to drag ratio L/D against the variation of angle of attack α. This is because such airfoil configuration can suppress the growth of the separation bubble generated near the leading edge against the increment in α. It leads to gentler change of negative pressure level in the front part near the leading edge against the increment in α. Higher negative pressure level on the upper surface of airfoil involves larger drag at high α. Therefore, gentler change of negative pressure level in the front part near the leading edge results in smaller increment in drag, and eventually smaller change of L/D against the variation of α.

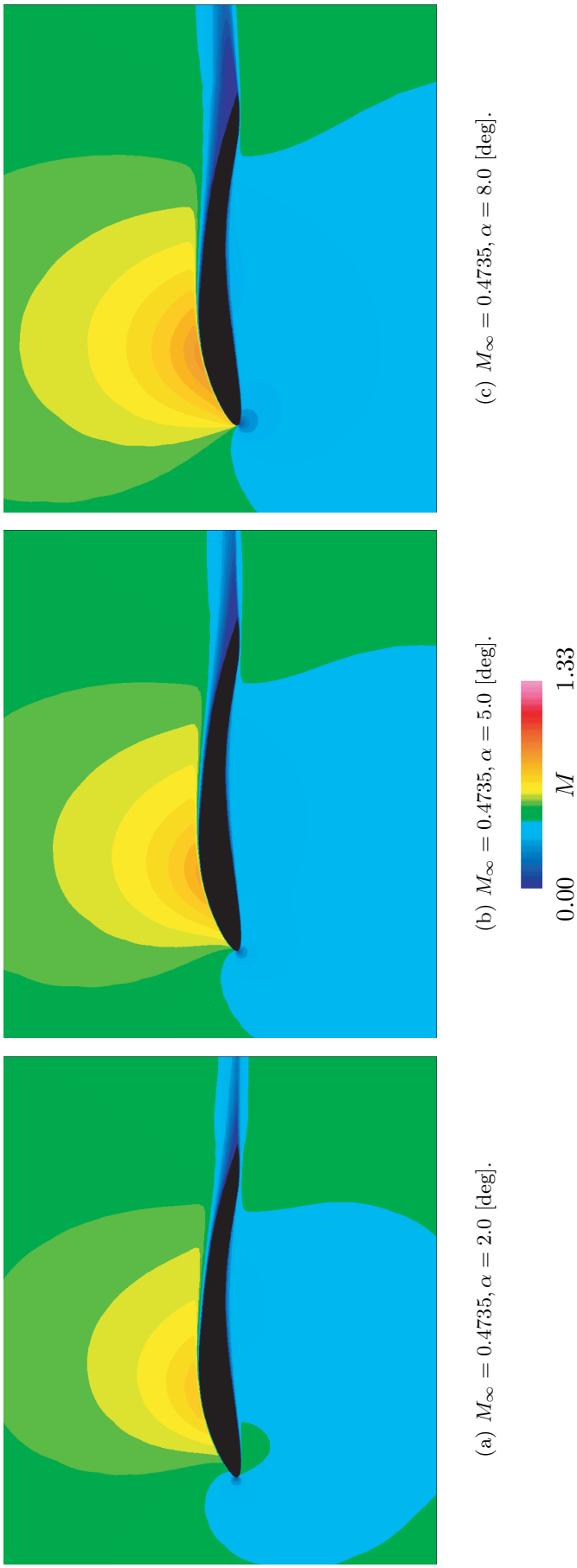


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg].

Figure 4.28 Local Mach number contour plots around the airfoil of one-point optimal solution of Case 3.

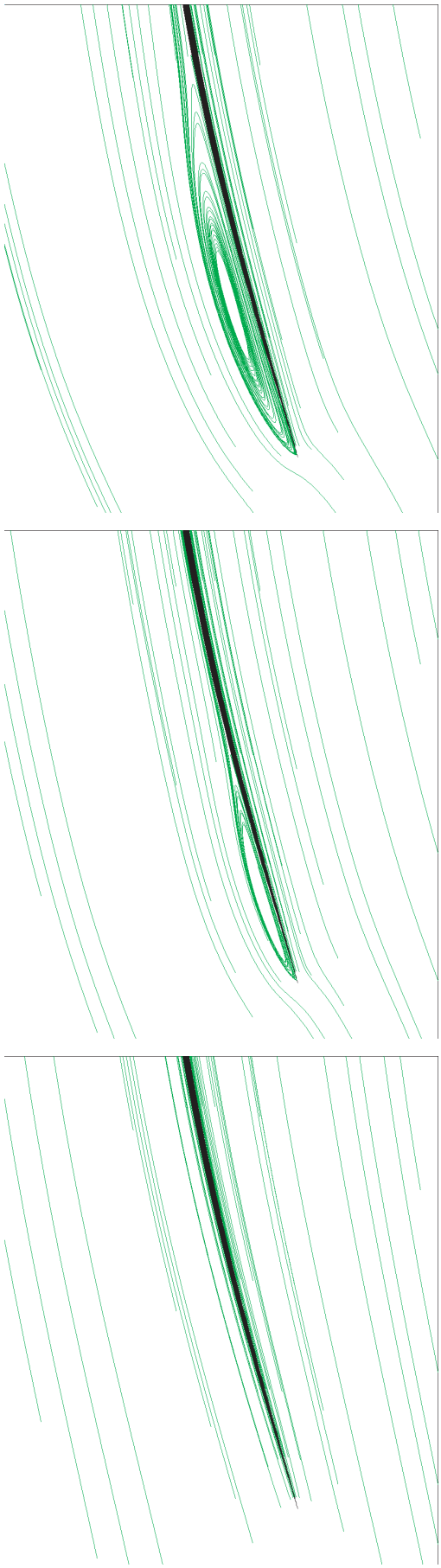


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg].

Figure 4.29 Local Mach number contour plots around the airfoil of robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3.

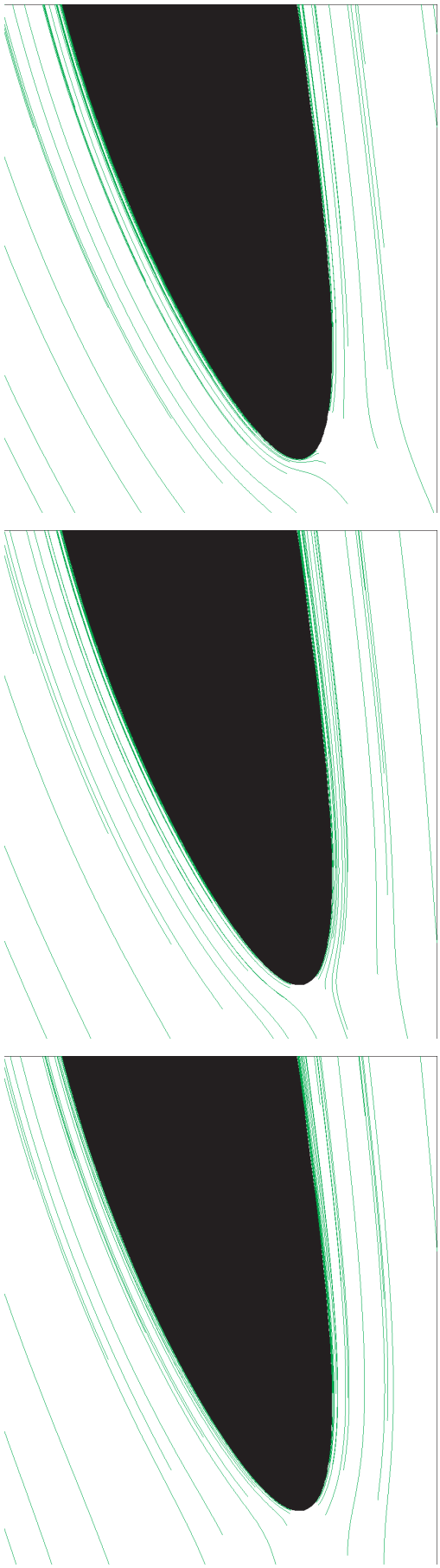


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg].

Figure 4.30 Streamline patterns near the leading edge of the airfoil of one-point optimal solution of Case 3.

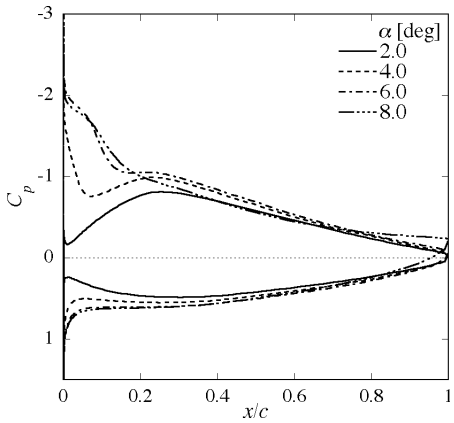


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg].

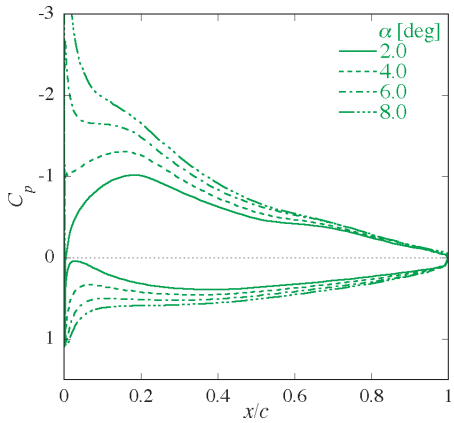
(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg].

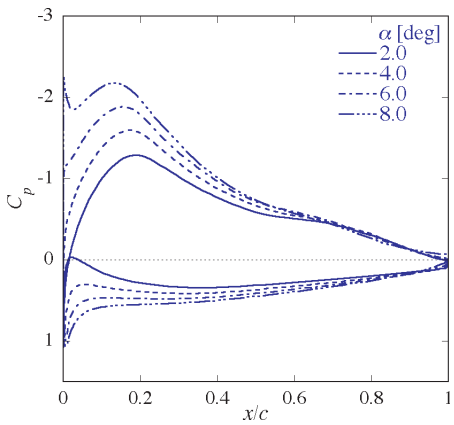
Figure 4.31 Streamline patterns near the leading edge of the airfoil of robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3.



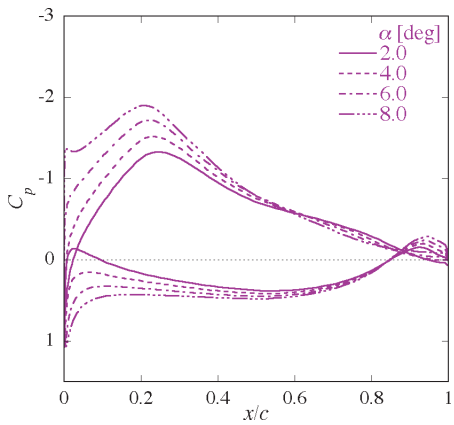
(a) One-point optimal solution.



(b) Robust optimal solution with larger standard deviation of L/D obtained by using DFMOSS.



(c) Robust optimal solution with medium standard deviation of L/D obtained by using DFMOSS.



(d) Robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS.

Figure 4.32 Chordwise C_p distributions over the airfoil surface at various α and $M_\infty = 0.4735$ of Case 3.

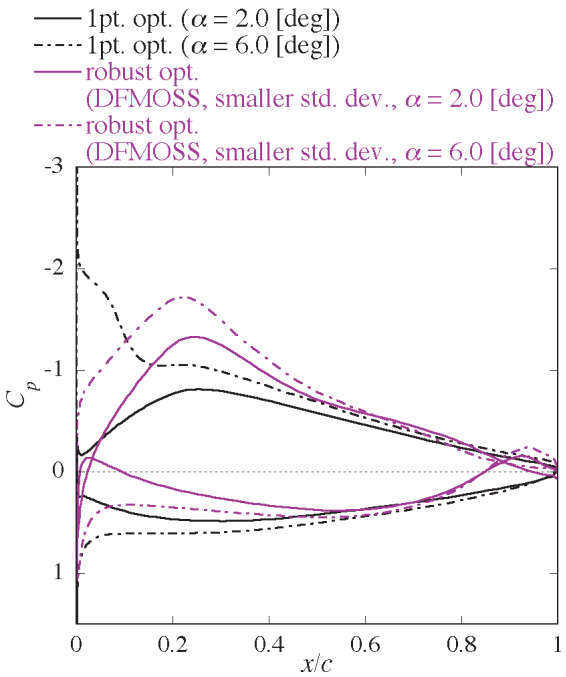


Figure 4.33 Comparison of chordwise C_p distributions over the airfoil surface at various α and $M_\infty = 0.4735$ between one-point optimal solution and robust optimal solution with smaller standard deviation of L/D obtained by using DFMOSS of Case 3.

4.5 Summary

In this chapter, three robust aerodynamic design optimizations of two-dimensional airfoil configuration for Mars exploratory airplane were carried out considering the effects of wind variations by using the DFMOSS and the CFD simulation. In all cases, the robust aerodynamic design optimization using the DFMOSS revealed trade-off relation between the optimality and the robustness of aerodynamic performance sufficiently by only one optimization run without advance specification of input parameters such as the weighting factors and the sigma level.

In the first case considering the robustness of lift to drag ratio against the variation of flight Mach number, it was shown that an airfoil configuration with smaller maximum camber can improve the robustness of lift to drag ratio against the variation of flight Mach number. This is because such airfoil can suppress the growth of the shock wave, i.e., realize smaller increment in pressure drag (wave drag) against increment in flight Mach number.

In the second case considering the robustness of pitching moment coefficient against the variation of flight Mach number, it was shown that an airfoil configura-

tion with larger curvature in the front part can improve the robustness of pitching moment coefficient against the variation of flight Mach number. This is because such airfoil can suppress the backward movement of shock wave occurred over the upper surface of airfoil against increment in flight Mach number, and eventually it results in smaller change in pitching-down moment produced in the rear part of airfoil against increment in flight Mach number.

Finally, in the third case considering the robustness of lift to drag ration against the variation of angle of attack, it was shown that an airfoil configuration with blunter leading edge can improve the robustness of lift to drag ratio against the variation of angle of attack. This is because such airfoil configuration can suppress the growth of the separation bubble generated near the leading edge against increment in angle of attack. It leads to gentler change of negative pressure level in the front part near the leading edge, i.e., smaller change of drag against increment in angle of attack, and eventually it results in smaller change of lift to drag ratio against the variation of angle of attack.

Chapter 5

Robust Aerodynamic Design of Airfoil and Wing Planform Configurations of Mars Exploratory Airplane

5.1 Introduction

In Chap. 4, the robust aerodynamic design optimizations of two-dimensional airfoil configuration of Mars exploratory airplane considering the effects of wind variations were performed by using the DFMOSS and the CFD simulation. In this chapter, robust aerodynamic design optimizations of airfoil and wing planform configurations for Mars exploratory airplane are performed to obtain more realistic and more useful trade-off information between the optimality and the robustness of aerodynamic performance. Differently from the two-dimensional case, not only the robustness of aerodynamic performance against the variations of flight Mach number and angle of attack but also that against the variation of angle of side slip is considered in the present case. In the present study, the numerical results obtained by the robust optimization using DFMOSS are compared to those obtained by the one-point optimization in order to investigate efficiency and usefulness of the DFMOSS and the differences in optimized designs. Then, the obtained trade-off relations between the optimality and the robustness of aerodynamic performance of airfoil and wing planform configurations for Mars exploratory airplane are discussed. Furthermore, one of useful guides to more reliable designs of Mars exploratory airplanes with fixed wing and more reliable Mars exploratory missions using the airplanes is provided.

5.2 Design Problem Definition

Consider the traditional aerodynamic design optimization problem where airfoil and wing planform configurations must be optimized so that lift L will be maximized, drag D will be minimized, and the absolute value of pitching moment around the center of gravity of airplane M_p will be minimized at design point:

$$\begin{aligned} \text{Maximize: } & L \\ \text{Minimize: } & D \\ & |M_p| \end{aligned} \tag{5.1}$$

In the present study, the center of gravity used for the definition of M_p is assumed to be at 25% root chord position ($[x/c, y/c, z/c]_{CG} = [0.25, 0, 0]$, c is the root chord length). Similar to the two-dimensional case as described in Chap. 4, cruising conditions of NASA'S AME,[11] as shown in Table 5.1, is used as the design point in the present case. In addition, it is required to convert non-dimensional aerodynamic coefficients obtained by the CFD simulations into the values with real units in the present case. These conversions are performed based on the reference values as shown in Table 5.2. These reference values are based on the real configuration data of NASA's AME[11] and Martian meteorological data.[19] Now, convert this problem to the robust aerodynamic design optimization problem where the mean value of dispersive L must be maximized, the standard deviation of dispersive L must be minimized, the mean value and the standard deviation of dispersive D must be minimized, and the absolute mean value and the standard deviation of dispersive M_p must be minimized when the flight Mach number M_∞, the angle of attack α and the angle of side slip β disperse around design point:

$$\begin{aligned} \text{Maximize: } & \text{mean value of } L \\ \text{Minimize: } & \text{standard deviation of } L \\ & \text{mean value of } D \\ & \text{standard deviation of } D \\ & |\text{mean value of } M_p| \\ & \text{standard deviation of } M_p \end{aligned} \tag{5.2}$$

This robust aerodynamic optimization problem aims at finding the wing configuration not only with maximum L, minimum D and M_p but also with robust L, D and M_p characteristics that the Mars exploratory airplane can avoid failure in supporting

its weight, flying over an expected range, and controlling its pitching motion by control devices such as a horizontal tail wing when it flies in various horizontal directions under the situation where the westerlies as shown in Fig. 1.8 and the vertical winds caused by the forced planetary wave as shown in Fig. 1.9 blow stably. Table 5.3 shows the assumed statistical characteristics of dispersive flight Mach number M_∞, angle of attack α and angle of side slip β for NASA's AME. As described in Chap. 4, the value of 0.1 as the standard deviation of M_∞ corresponds to the variation of wind speeds in the moving direction of about 22 m/s, and the value of 3.0 deg as the standard deviation of α corresponds to the variation of vertical wind speeds of about 5.5 m/s for NASA's AME with the flight Mach number of 0.4735. The value of 12.0 deg as the standard deviation of β corresponds to the variation of side wind speeds of about 22 m/s, and it is nearly equal to the dairy and seasonal variation of westerly speeds near the Martian surface (at the altitude of several kilometers) where it is assumed for the Mars airplane to fly for exploring the structure of remnant magnetic fields and lower atmosphere, as shown in Fig. 1.8.

Table 5.1 Three-dimensional cruising conditions of NASA's AME.[11]

Reynolds number Re (based on root chord length c)	1.0×10^5
Flight Mach number M_∞	0.4735
Angle of attack α [deg]	2.0
Angle of side slip β [deg]	0.0

Table 5.2 Reference values for conversions into the values with real units.[11,19]

Length	Root chord length c [m] (NASA's AME)	12.607
Density	Martian freestream density ρ_∞ [kg/m^3]	1.18×10^3
Velocity	Martian freestream speed of sound a_∞ [m/s]	220
Acceleration	Martian acceleration of gravity g [m/s^2]	3.78

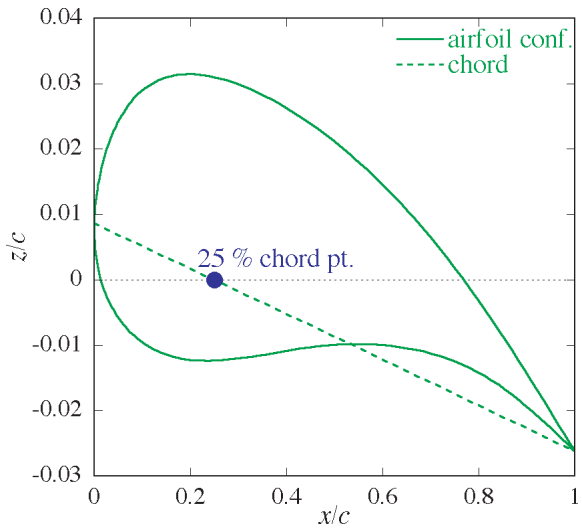
Figure 5.1 shows the present definition of airfoil and wing planform configurations. x, y and z are the coordinates in the chordwise, the spanwise and the vertical directions, respectively. In the present study, it is assumed that the half wing has three folds (named as 1st, 2nd and 3rd folds in order near the root), like the main wing of NASA's AME, to be stored into an aeroshell delivered by a launcher from the Earth to the Mars. Firstly, the airfoil configurations on x–z plane are defined at

Table 5.3 Statistical characteristics of three-dimensional flight conditions for NASA's AME.

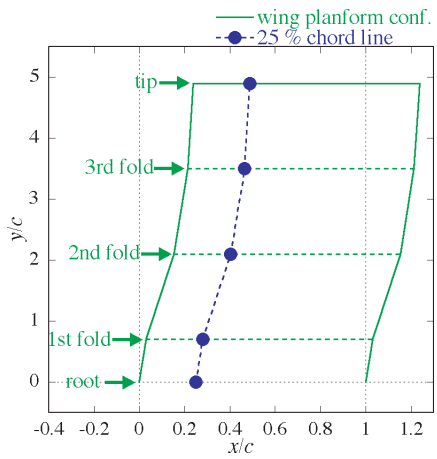
Flight Mach number M_∞	Mean value	0.4735
	Standard deviation	0.1
Angle of attack α [deg]	Mean value	2.0
	Standard deviation	3.0
Angle of side slip β [deg]	Mean value	0.0
	Standard deviation	12.0

five sections (root, 1st fold, 2nd fold, 3rd fold and tip), as shown in Fig. 5.1(a). For each section, the airfoil is generated by the B-spline curves with six control points as well as three fixed points, similar to the two-dimensional case as described in Chap. 4, and this configuration is twisted around the 25 % chord point by giving the twist angle. In the present case, it is assumed that the airfoil configurations are similar at all five sections except for the twist angles. Next, 25 % chord line of wing planform is defined as shown in Figs. 5.1(b) and 5.1(c). y coordinates of five sections are determined by giving the span lengths of four wing divisions (root $\sim$ 1st fold, 1st fold $\sim$ 2nd fold, 2nd fold $\sim$ 3rd fold, and 3rd fold $\sim$ tip). And, x and z coordinates of 25 % chord points at five sections are determined by giving the sweep and the dihedral angles of four wing divisions. Finally, wing planform is generated by linearly interpolating x, y and z coordinates of the generated airfoils at five sections along the generated 25 % chord line of wing planform. Table 5.4 shows the design variables and those ranges of the present aerodynamic design optimization problem. The total number of design variables is 29. In the present study, the taper ratio of each wing division is not considered as the design variable, i.e., the chord lengths are one at all five sections. This is because smaller taper ratio leads to smaller wing area at constant root chord length and constant wing span length and it involves smaller lift. In addition, smaller taper ratio leads to longer wing span length at constant root chord length and constant wing area, which is unfavorable in terms of wing storability into an aeroshell as described later.

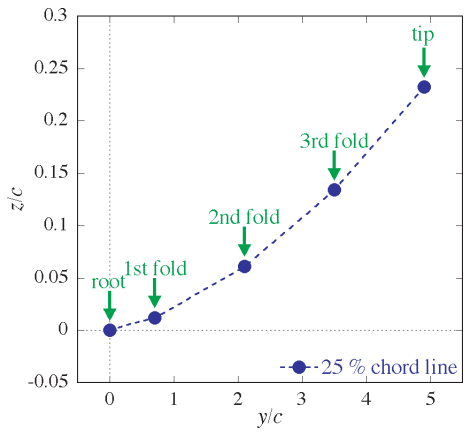
Similar to the two dimensional case as described in Chap. 4, a geometric constraint so that the upper and lower surfaces of the airfoil do not intersect each other is imposed. In addition, another geometric constraint so that the wing can be folded successfully without interference between the left and the right half wings, as shown in Fig. 5.2 is considered. That is, the spanwise position of one half folded wing must



(a) Airfoil configuration on x–z plane.



(b) Wing planform configuration and 25 % chord line on x–y plane.



(c) 25 % chord line on y–z plane.

Figure 5.1 Definition of airfoil and wing planform configurations.

Table 5.4 Design variables and corresponding ranges.

Design variable		lower limit	upper limit
for Airfoil configuration			
Coordinate x/c	B-spline control point 1	0.66667	0.99
	B-spline control point 2	0.33333	0.66667
	B-spline control point 3	0.01	0.33333
	B-spline control point 4	0.01	0.33333
	B-spline control point 5	0.33333	0.66667
	B-spline control point 6	0.66667	0.99
Coordinate y/c	B-spline control point 1	−0.15	0.15
	B-spline control point 2	−0.15	0.15
	B-spline control point 3	−0.15	0.15
	B-spline control point 4	−0.15	0.15
	B-spline control point 5	−0.15	0.15
	B-spline control point 6	−0.15	0.15
Twist angle [deg]	Section root	−10.0	10.0
	Section 1st fold	−10.0	10.0
	Section 2nd fold	−10.0	10.0
	Section 3rd fold	−10.0	10.0
	Section tip	−10.0	10.0
for 25 % chord line of wing planform configuration			
Span length (based on root chord length c)	Wing division root ∼ 1st fold	0.05	0.98362
	Wing division 1st fold ∼ 2nd fold	0.05	1.9672
	Wing division 2nd fold ∼ 3rd fold	0.05	1.9672
	Wing division 3rd fold ∼ tip	0.05	1.9672
Sweep angle [deg]	Wing division root ∼ 1st fold	−89.0	89.0
	Wing division 1st fold ∼ 2nd fold	−89.0	89.0
	Wing division 2nd fold ∼ 3rd fold	−89.0	89.0
	Wing division 3rd fold ∼ tip	−89.0	89.0
Dihedral angle [deg]	Wing division root ∼ 1st fold	−10.0	10.0
	Wing division 1st fold ∼ 2nd fold	−10.0	10.0
	Wing division 2nd fold ∼ 3rd fold	−10.0	10.0
	Wing division 3rd fold ∼ tip	−10.0	10.0

not exceed the spanwise position of 1st fold of the other half wing. Furthermore, a geometric constraint so that the folded wing can be stored into an aeroshell, as shown in Fig. 5.3, is also considered. Here, it is assumed that the internal diameter of aeroshell is 2.48 m, which is that of Viking-derivarive aeroshell.[13]

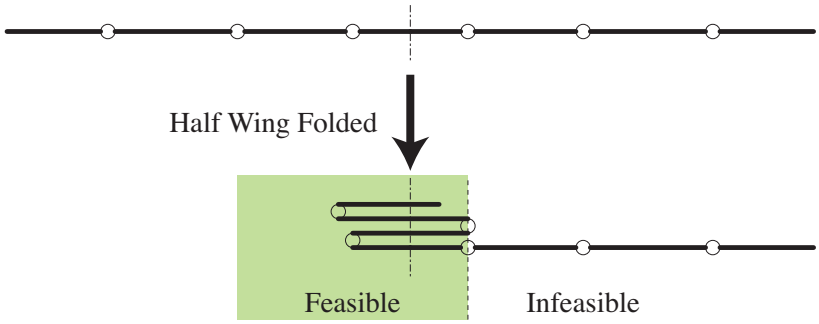


Figure 5.2 Geometric constraint on wing foldability.

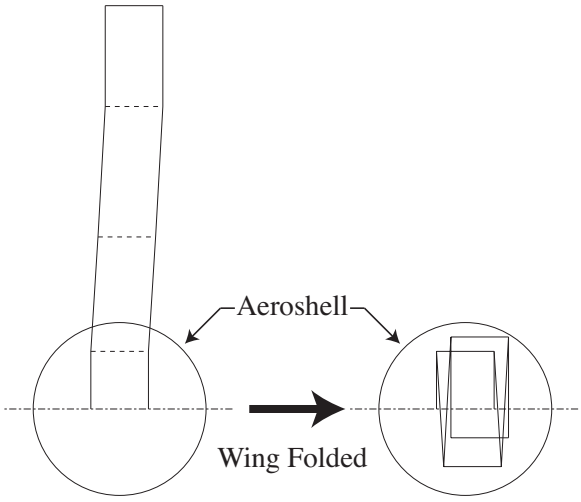


Figure 5.3 Geometric constraint on wing storability into aeroshell.

In the present aerodynamic design optimization problem, the following structural geometric constraint on required wing thickness to stand the bending moment due to aerodynamic force is considered. Here, it is assumed that the wing has the beam with thin hollow circular cross section, as shown in Fig. 5.4, along 25 % chord line. At an arbitrary spanwise position y, the local bending moment $M_b(y)$ can be calculated from the local aerodynamic force in z direction per unit span length $f_z(y)$ as

$$\frac{\mathrm{d}^2}{\mathrm{d}y^2} M_b(y) = -f_z(y) \tag{5.3}$$

This equation can be rewritten to an integral form as

$$M_b(y) = \int_y^{b/2} f_z(s)(s-y)\mathrm{d}s \tag{5.4}$$

where b is the wing span length. For simplicity, $f_z(y)$ distribution is replaced by spanwise concentrated loads. The maximum local bending stress $\sigma_{max}(y)$ is given as

$$\sigma_{max}(y) = \frac{M_b}{I}\frac{d}{2} \tag{5.5}$$

where d is the beam diameter. I is the geometrical moment of inertia, and I for the beam with thin hollow circular cross section is given as

$$I = \frac{\pi}{8}d^3 t \tag{5.6}$$

where t are the beam thickness. Then, the structural constraint so that σ_{max} must be less than its limit value σ_{lim} as

$$\sigma_{max}(y) \leq \sigma_{lim} \tag{5.7}$$

Using Eqs. 5.5 and 5.6, this constraint can be rewritten as

$$\begin{aligned} d &\geq d_{req} \\ d_{req} &= \sqrt{\frac{4M_b(y)}{\pi t \sigma_{lim}}} \end{aligned} \tag{5.8}$$

Consequently, the structural constraint so that airfoil thickness at 25 % chord position $t_{c/4}$ must be larger than the required beam diameter to stand the bending moment is obtained:

$$t_{c/4} \geq d_{req} \tag{5.9}$$

In the present study, it is assumed that the wing beam is made of aluminum alloy 2024-T3 ($\sigma_{lim} = 274.59$ [MPa]), and the beam thickness is 1.0 mm.

5.3 Numerical Methods and Conditions

5.3.1 Optimization

In the present case, two aerodynamic design optimizations are performed and the obtained numerical results are compared:

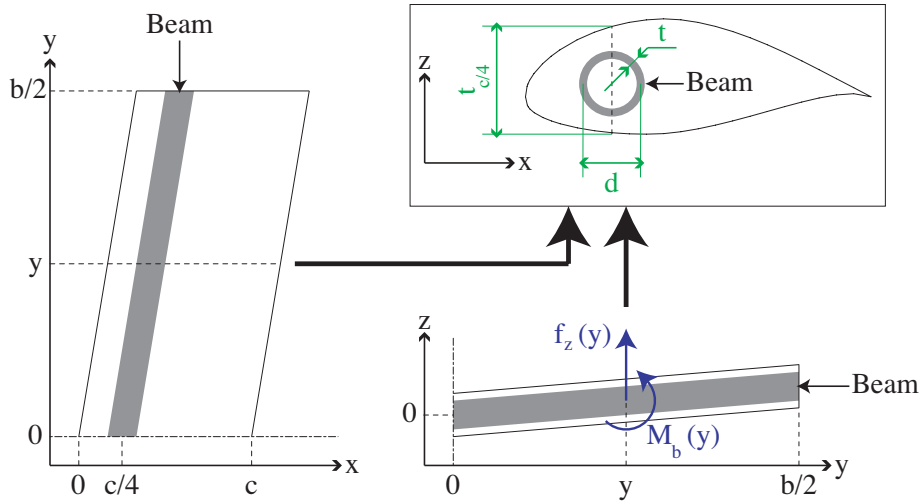


Figure 5.4 Wing beam structure.

- **One-point optimization using a multi-objective genetic algorithm (MOGA)[33]**

  When $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = 0.0$ [deg],

$$\begin{aligned} \text{Maximize: } & L \\ \text{Minimize: } & D \\ & |M_p| \end{aligned} \tag{5.10}$$

- **Robust optimization using the MOGA coupled with the DFMOSS**

  When M_∞ disperses around 0.4735 with its standard deviation of 0.1, α disperses around 2.0 deg with its standard deviation of 3.0 deg, and β disperses around 0.0 deg with its standard deviation of 12.0 deg,

$$\begin{aligned} \text{Maximize: } & \text{mean value of } L \\ \text{Minimize: } & \text{standard deviation of } L \\ & \text{mean value of } D \\ & \text{standard deviation of } D \\ & |\text{mean value of } M_p| \\ & \text{standard deviation of } M_p \end{aligned} \tag{5.11}$$

In all optimizations, fitness value are evaluated by using the Pareto-ranking method,[53] the fitness sharing,[33,53] and Michaleswicz's nonlinear function.[49] Parents are selected by the stochastic universal sampling (SUS),[47] children are reproduced by the blended

crossover (BLX-0.5)[48] and uniform mutation[33] with 10 % of the range of each design variable as the maximum perturbation, and the alternation of generations is performed by the Best-N selection.[50,51] The constraints are treated by using the Pareto-optimality-based constraint-handling (PBCH) technique.[56,57] In the robust aerodynamic design optimization using DFMOSS, the statistical values of aerodynamic performance parameters are evaluated by the second-order Taylor's series expansion approach, i.e., total six CFD simulations at

1. $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = 0.0$ [deg]
2. $M_\infty = 0.3735$, $\alpha = 2.0$ [deg] and $\beta = 0.0$ [deg]
3. $M_\infty = 0.5735$, $\alpha = 2.0$ [deg] and $\beta = 0.0$ [deg]
4. $M_\infty = 0.4735$, $\alpha = -1.0$ [deg] and $\beta = 0.0$ [deg]
5. $M_\infty = 0.4735$, $\alpha = 5.0$ [deg] and $\beta = 0.0$ [deg]
6. $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = 12.0$ [deg]

are required for one evaluation of statistical values. Here, note that the aerodynamic performance at $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = -12.0$ [deg] is the same as that at $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = 12.0$ [deg] in terms of L, D and M_p evaluated in the present robust optimization, therefore the CFD simulations at $M_\infty = 0.4735$, $\alpha = 2.0$ [deg] and $\beta = -12.0$ [deg] are omitted to reduce computation time.

Table 5.5 shows the present optimization conditions. In the robust optimization using DFMOSS, only one case is carried out without the advance setting of input parameters such as the weighting factors and sigma level. The robust optimization using DFMOSS starts from the initial population based on the optimal solutions obtained by the one-point optimization using MOGA to advance the convergence to the robust optimal solutions. This is why the number of evaluations required in robust optimization using DFMOSS is smaller than that required in the one-point optimization using MOGA.

5.3.2 CFD Simulation

The governing equations for the CFD simulation are the three-dimensional Favre-averaged compressible thin-layer Navier-Stokes equations. The ADI-SGS factorization algorithm is used for the time integration with the total number of iteration steps of 6,500. For the first 1,000 steps, the multigrid method[80] with V-cycle strategy

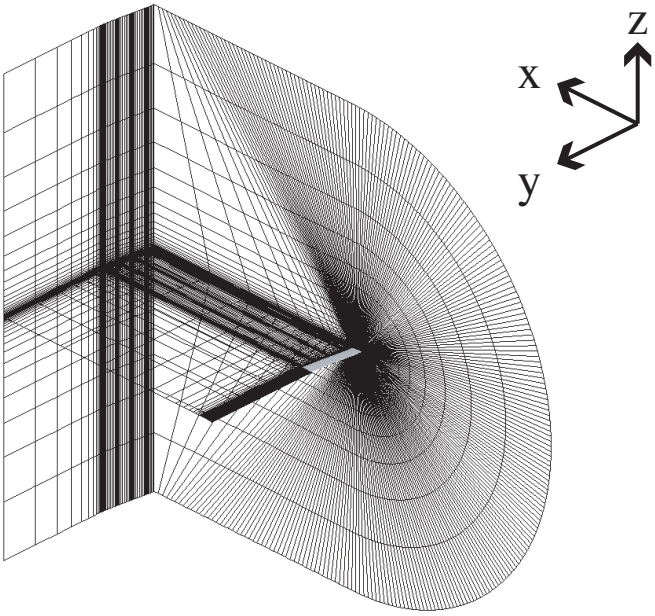
Table 5.5 Optimization conditions.

for One-point optimization using MOGA	Population size	32
	Number of generations	50
	Mutation rate [%]	20
for Robust optimization using DFMOSS	Population size	16
	Number of generations	30
	Mutation rate [%]	20

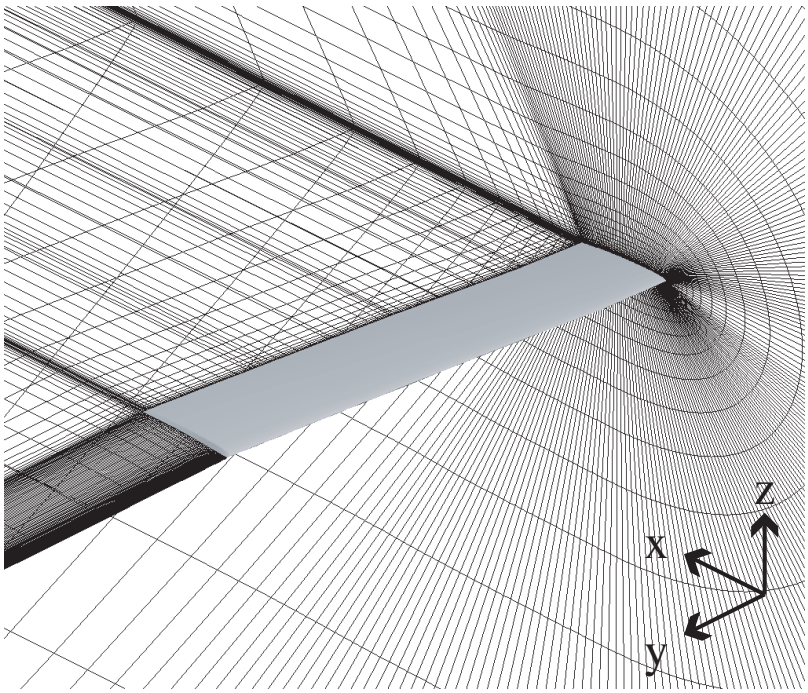
and three grid levels, and the space variable time stepping technique with reference time step size of 2.0 are used for fast convergence. And then, it switches to the usual singlegrid method with physical time step size of 0.01 (maximum local CFD number of about 350). Aerodynamic performance parameters such as lift, drag and moment are evaluated by averaging these values for the last 2,500 steps. The inviscid terms of numerical fluxes are evaluated by the SHUS scheme.[70] In the inviscid terms, high-order accuracy is obtained by the third-order upwind-biased MUSCL interpolation[71,72] based on the primitive variables with van Albada differentiable limiter.[67] The viscous terms are evaluated by the second-order central differencing, and the turbulent viscosity is modeled by the Baldwin-Lomax algebraic turbulence model.[81]

Figure 5.5 shows the grid distributions. In the present study, C–H type grid is generated algebraically around the wing, and grid points are clustered near the leading edge, the trailing edge, each fold, tip and a wing surface. In the case without side wind ($\beta = 0.0$ [deg]), the grid is generated and the flowfields are simulated around only the half wing, assuming that the flowfields are symmetric at the root. The number of grid points for the half wing case is 249 in the direction around the airfoil (209 points over the wing surface), 70 in the spanwise direction (54 points over the wing surface), 41 in the direction normal to the wing surface, and the total number of grid points is 714,630. In the case with side flow ($\beta \neq 0.0$ [deg]), on the other hand, the grid is generated and the flowfields are simulated around the full wing exactly. The number of grid points for the full wing case is 249 in the direction around the airfoil (209 points over the wing surface), 137 in the spanwise direction (105 points over the wing surface), 41 in the direction normal to the wing surface, and the total number of grid points is 1,398,633. The minimum grid spacing near the airfoil surface is about 1.0×10^{-4} based on the root chord length c.

The computation time required for one evaluation of aerodynamic performance



(a) Overall view.



(b) Close-up view.

Figure 5.5 Grid distributions.

of half wing without side wind using the CFD simulation is about one hour with one processor of NEC SX-6 computing system owned by the ISAS of JAXA. In the present study, the optimizer distributes the multiple evaluators corresponding to the multiple individuals of GA into 32 processors of this computing system in parallel. Therefore, the total computation time required for one case of the present robust aerodynamic design optimization using DFMOSS can be reduced to about 110 hours.

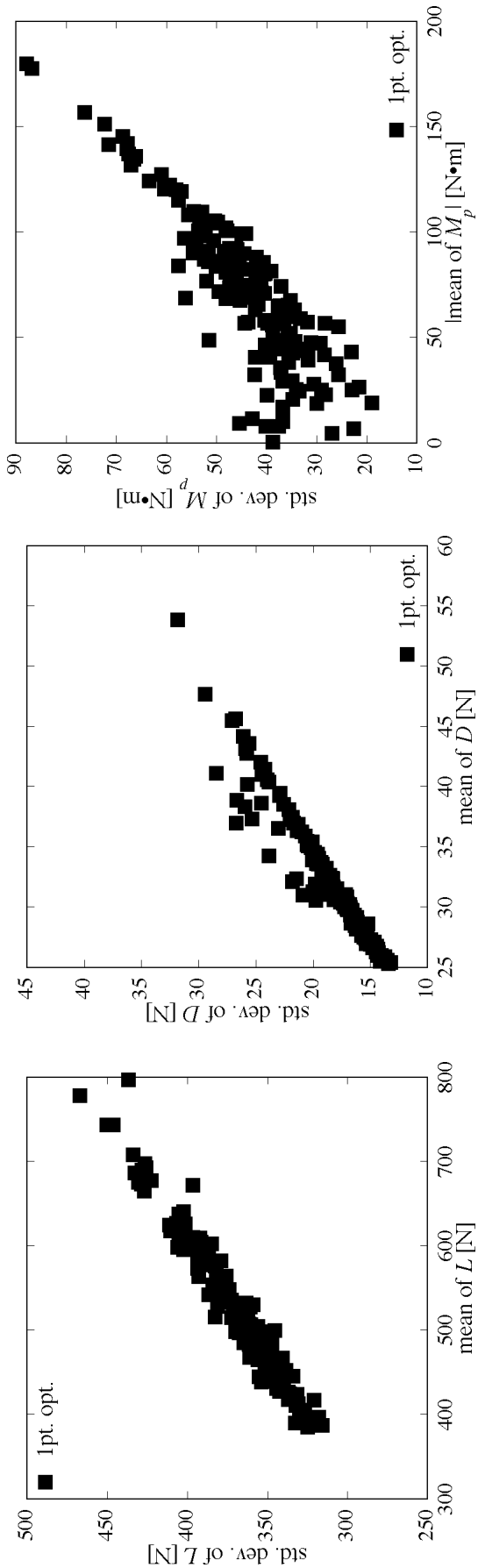
5.4 Numerical Results

Figures 5.6 and 5.7 compares the robust optimal solution distributions (projections onto standard deviation against mean value of L plane, standard deviation against mean value of D plane, and standard deviation against absolute mean value of M_p) obtained by using the DFMOSS with the optimal solution distributions obtained by the one-point optimization neglecting the robustness of L, D and M_p. The one-point optimization can only reveal the distinct relations that the standard deviations of L and D decrease with the decrements in the mean values of L and D. Such relations are worthless from the designer's viewpoint because it can be easily expected that the variations of L and D against the wind variations become larger as the magnitude of L and D becomes larger even if the optimizations are not performed. The robust optimization using the DFMOSS, on the other hand, can find not only such natural relation but also the solutions which have almost the same mean value of L and various standard deviations of L. Comparing the projections onto the standard deviation against the mean value of L plane between the one-point optimal solutions and the robust optimal solutions obtained by using the DFMOSS, there exist the solutions with smaller standard deviation of L than the one-point optimal solutions in the robust optimal solutions obtained by using the DFMOSS. The difference between the one-point optimization and the robust optimization using DFMOSS can be seen in the projections onto the standard deviation against the mean value of D plane. The robust optimization using the DFMOSS find the solutions with larger mean value and larger standard deviation of D than those of one-point optimal solutions. From these results, it is expected that the solutions with smaller standard deviation of L leads to the solutions with larger mean value and larger standard deviation of D. As for M_p, there exist the indistinct relations that the standard deviations of M_p decrease with the decrements in the absolute mean values of M_p in the one-optimal solutions and the robust optimal solutions obtained by using the

DFMOSS. These relations can be expected easily due to the same reason as the similar relations about L and D, too. However, it is different from the relations about L and D that there exist the solutions with almost the same absolute mean value of M_p and various standard deviations of M_p. Comparing the projections onto the standard deviation against the absolute mean value of M_p plane between the one-point optimal solutions and the robust optimal solutions obtained by using the DFMOSS, there exists little difference. These results indicate that the present robust aerodynamic design optimization using the DFMOSS can find the solutions with robust characteristic in aerodynamic performance parameters against wind variations, and can reveal qualitative trade-off information between the optimality and the robustness in aerodynamic performance parameters. In addition, these results show that the DFMOSS is an effective approach even in large-scale robust design optimizations with many design variables and objective functions.

Hereafter, three robust optimal solutions with almost the same optimality quality (mean value of L of about 545 N corresponding to 144 kgf on the Mars) and various robustness qualities (larger, medium and smaller standard deviations of L against the wind variations) obtained by using the DFMOSS, as shown in Fig. 5.8, are compared one another. Obviously, comparing these solutions in Figs. 5.8(a) and 5.8(b), it is shown that there exists the distinct relation that both the optimality (mean value) and the robustness (standard deviation) of D against the wind variations become worse when the robustness (standard deviation) of L against the wind variations becomes better at the constant optimality (mean value) of L. On the other hand, the distinct tendency about M_p can not be seen among these solutions in Fig. 5.8(c).

Figure 5.9 compares the wing planform configurations. In Fig. 5.9(a), all thee robust optimal solutions have the wing planform configurations with sweep-forward angle. These are because M_p around the center of gravity ($[x/c, y/c, z/c]_{CG} = [0.25, 0, 0]$) can be reduced. In addition, the wing configuration with large sweep-forward or sweep-back angle is infeasible due to the geometric constraint on the folded wing storability into the aeroshell. Therefore, in the present case, the feasible sweep angle does not have dominant effect on L, D, and those robustness against the wind variations. As for the dihedral angle, the distinct tendency can not be seen among the three robust optimal solutions in Fig. 5.9(b). In addition, the dihedral angle also does not have dominant effect on L, D, M_p and those robustness against the wind variations in the present case.

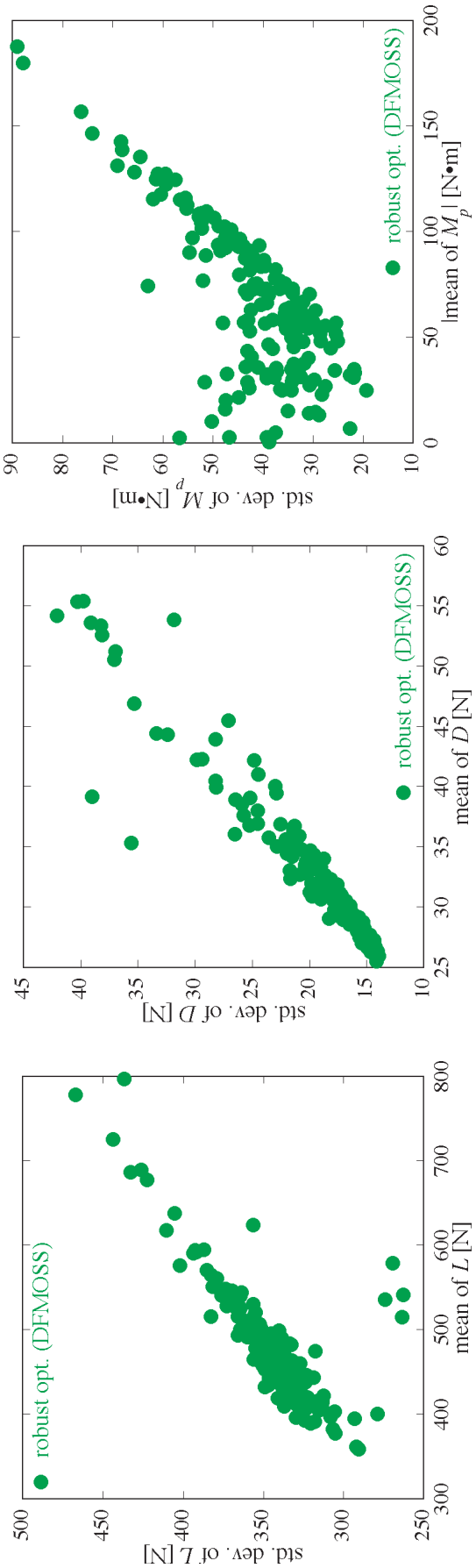


(a) Standard deviation against mean value of L.

(b) Standard deviation against mean value of D.

(c) Standard deviation against absolute mean value of M_p.

Figure 5.6 Optimal solution distributions obtained by the one-point optimization.

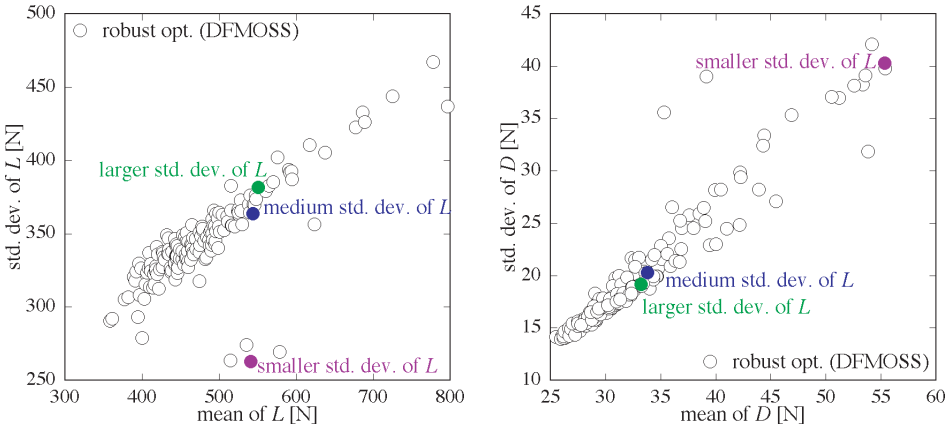


(a) Standard deviation against mean value of L.

(b) Standard deviation against mean value of D.

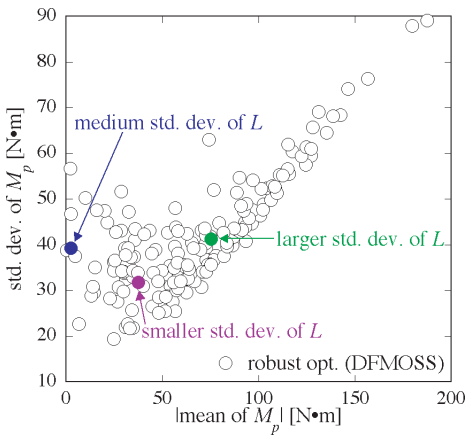
(c) Standard deviation against absolute mean value of M_p.

Figure 5.7 Robust solution distributions obtained by the robust optimization using DFMOSS.



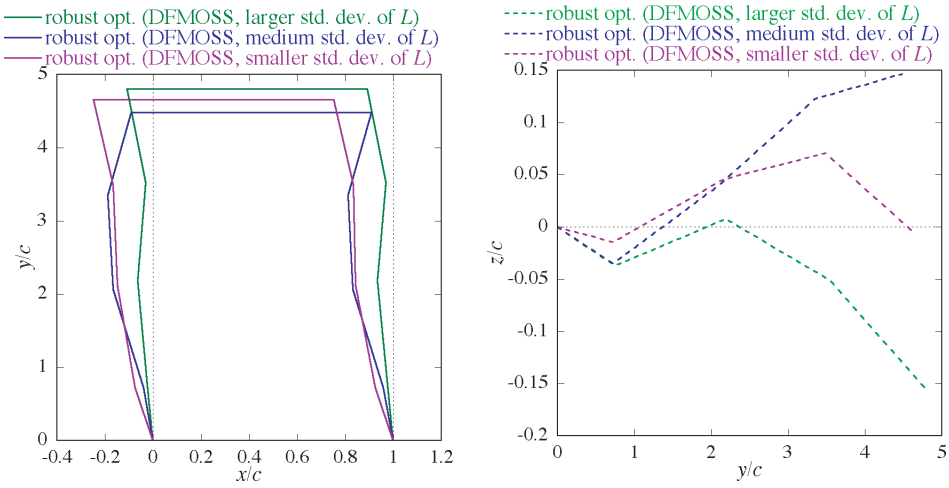
(a) Standard deviation against mean value of L.

(b) Standard deviation against mean value of D.



(c) Standard deviation against absolute mean value of M_p.

Figure 5.8 Compared three robust optimal solutions obtained by using the DFMOSS.



(a) Wing planform configuration on x–y plane.

(b) 25 % chord line on y–z plane.

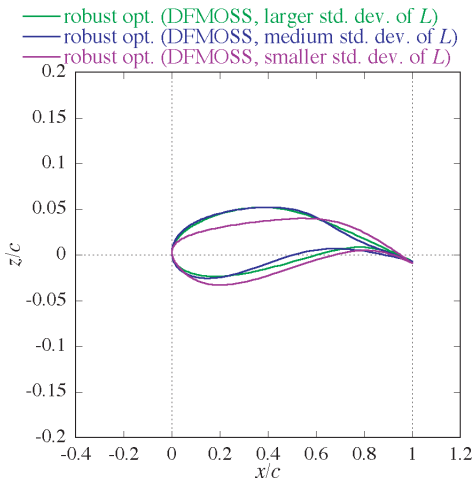
Figure 5.9 Wing planform configurations of three robust optimal solutions obtained by using DFMOSS.

Figure 5.10 compares the airfoil configurations at five sections (root, 1st fold, 2nd fold, 3rd fold and tip). The airfoil of all these robust optimal solutions have about 6∼8 % maximum thickness in those front parts (20∼30 % chord positions) to make a clearance of wing beam to stand the bending moment due to aerodynamic force. Comparing these robust optimal solutions, the thickness becomes slightly smaller as the standard deviation of L against the wind variations becomes smaller. In addition, the airfoil of the robust optimal solution with smaller standard deviation of L against the wind variations is like a supercritical airfoil, while the airfoils of the robust optimal solutions with medium and larger standard deviation of L against the wind variations look like normal ones. This is because the use of such supercritical-like airfoil can suppress the occurrence of strong shock waves and reduce the drastic increment in D with the increment in flight Mach number M_∞. As for the twist angle, the airfoils of all these robust optimal solutions have small twist-up angle at root, as shown in Fig. 5.10(a), to produce lift. At 2nd fold as shown in Fig. 5.10(c), the airfoils of all these robust solutions have larger twist-up angle than those roots. In addition, the solution with smaller standard deviation of L against the wind variations has the airfoil with larger twist-up angle locally at 2nd fold. At tip as shown in Fig. 5.10(e), the airfoils of all these robust optimal solutions have twist-down angle to decrease the effective angle of attack and avoid the tip stall with the

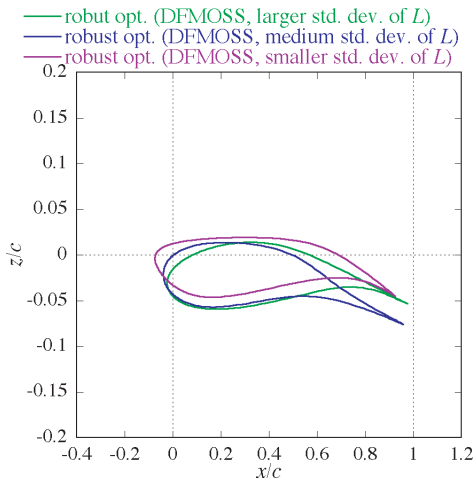
increment in angle of attack α.

Figures 5.11 and 5.12 show L and D histories against the dispersive flight condition parameters M_∞, α and β, respectively. As for L histories against M_∞ shown in Fig. 5.11(a), L increases monotonously with the increment in M_∞ due to the increment in dynamic pressure $\rho_\infty \left(M_\infty a_\infty\right)^2/2$ in all these robust optimal solutions. Qualitive tendencies of L–M_∞ are almost the same among these robust optimal solutions except that the increment in L against the increment in M_∞ becomes gentler as the standard deviation of L against the wind variations becomes smaller. On the other hand, as for L histories against α shown in Fig. 5.11(b) and those against β shown in Fig. 5.11(c), there exist distinct differences among these robust optimal solutions. In the robust optimal solution with smaller standard deviation of L against the wind variations, L increases monotonously with the increment in α, while the robust optimal solutions with medium and larger standard deviations of L against the wind variations have the drastic increments in L until $\alpha = 5.0$ [deg] and then stall at $\alpha = 8.0$ [deg]. In addition, L is kept large at any β in the robust optimal solution with smaller standard deviation of L, while L decreases monotonously with the increment in β in the robust optimal solutions with medium and larger standard deviations of L against the wind variations. From these results, it is shown that the robustness of L is mainly determined by the variations of L against α and β. As for D histories against M_∞ shown in Fig. 5.12(a) and those against α shown in Fig. 5.12(b), the robust optimal solution with smaller standard deviation of L against the wind variations has larger D at the design point $M_\infty = 0.4735$ and $\alpha = 2.0$ [deg], and also has more drastic increment in D with the increments in M_∞ and α than the robust optimal solutions with medium and larger standard deviations of L against the wind variations. On the other hand, as shown in Fig. 5.12(c), the solutions with smaller standard deviation of L against the wind variations has more drastic decrement in D with the increment in β than the robust optimal solution with medium and larger standard deviations of L against the wind variations.

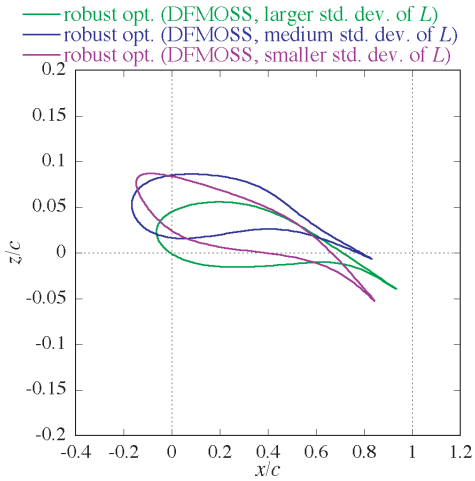
As described above, the robustness of L is mainly determined by the variations of L against α and β. Firstly, the variations of aerodynamic performance against α are discussed. Figures 5.13 and 5.14 show the local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different α (2.0, 5.0 and 8.0 [deg]) of the robust optimal solution with larger and smaller standard deviations of L against the wind variations obtained by using the DFMOSS, respectively. In the robust optimal solution with larger standard deviation of L



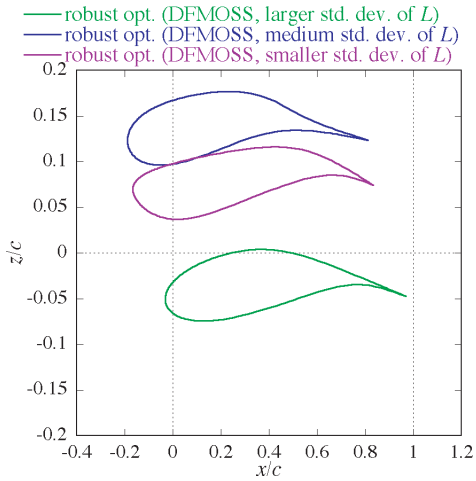
(a) Root.



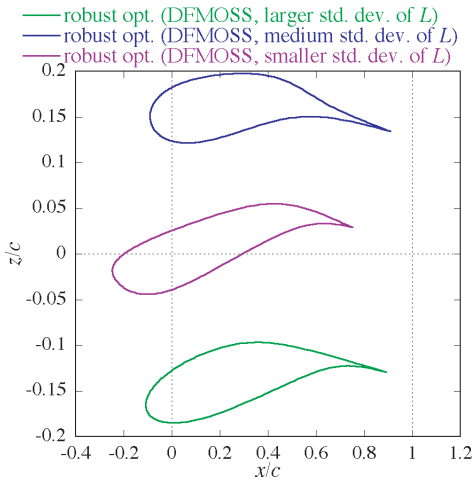
(b) 1st fold.



(c) 2nd fold.

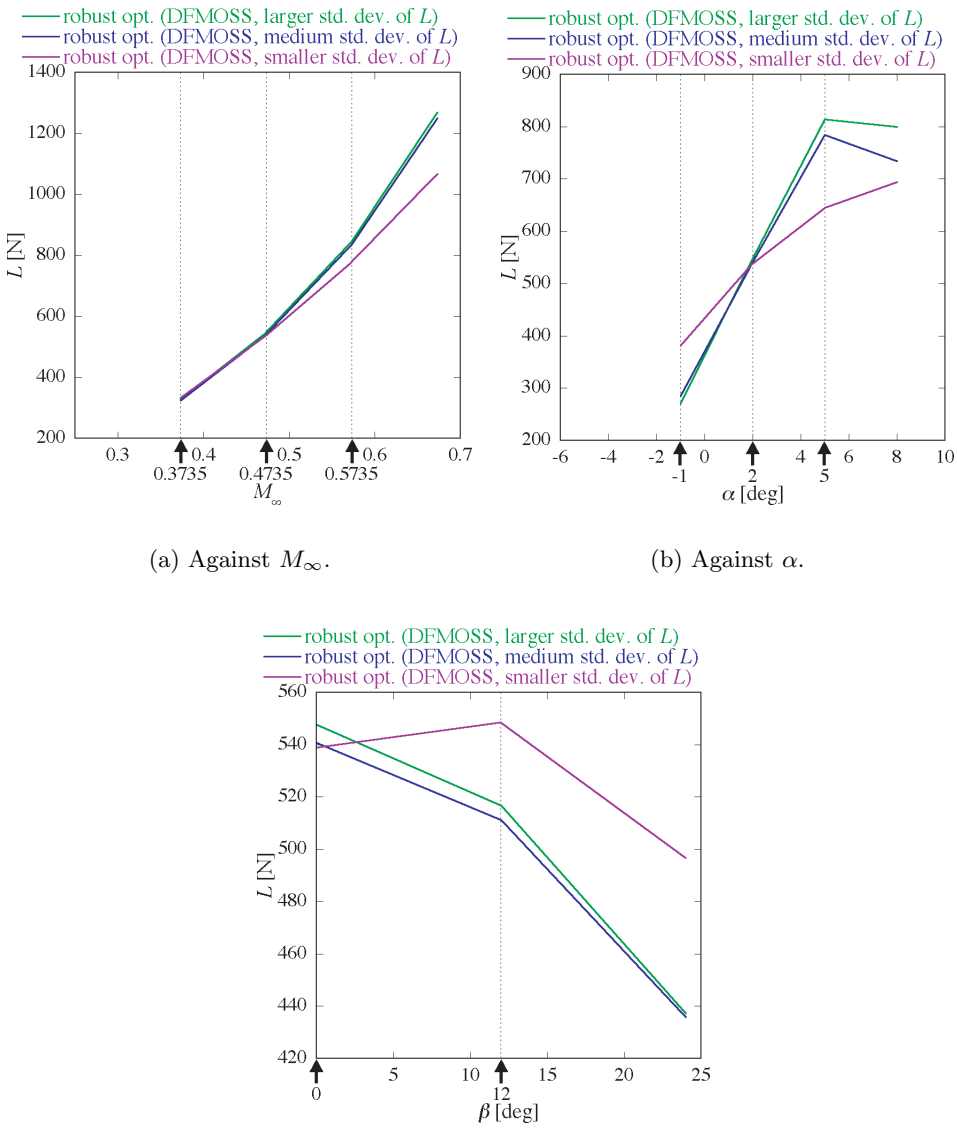


(d) 3rd fold.



(e) Tip.

Figure 5.10 Airfoil configurations of three robust optimal solutions obtained by using DFMOSS at five wing sections.

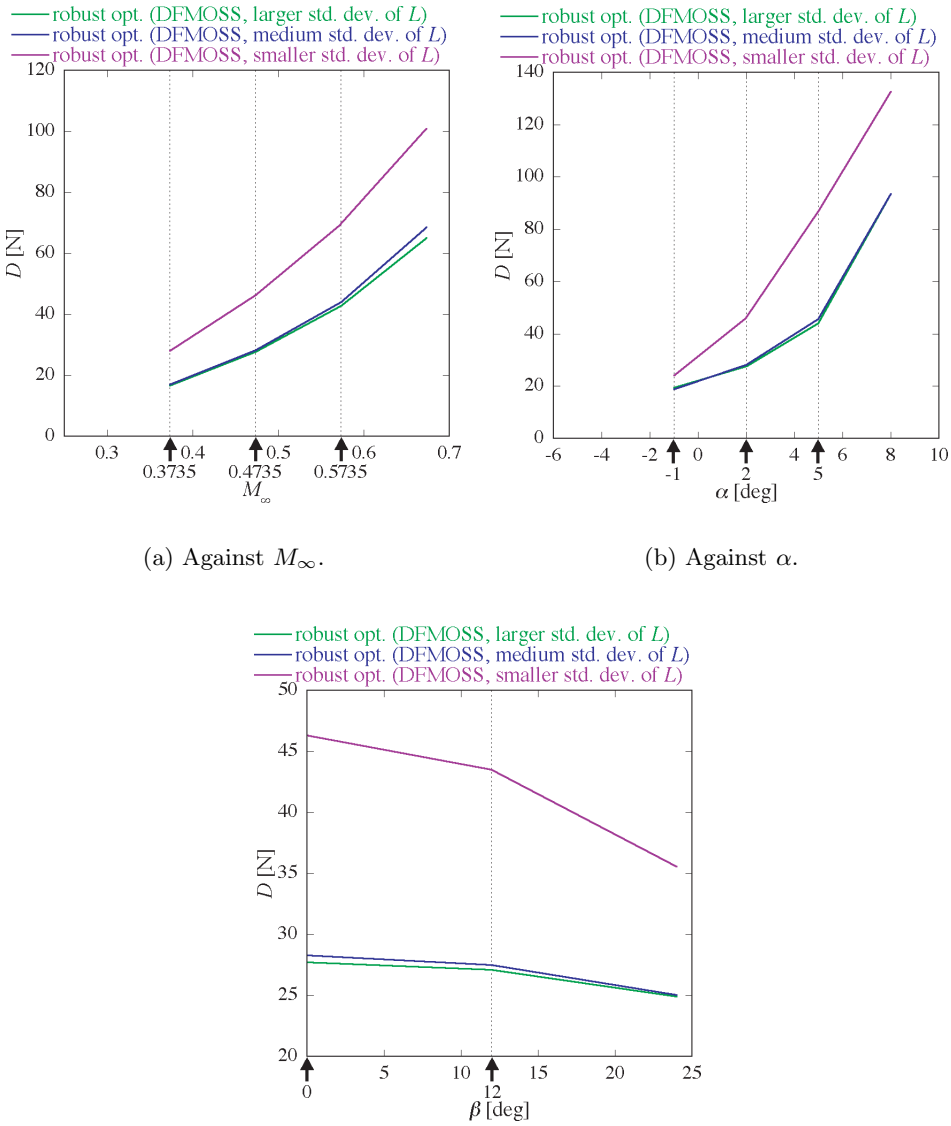


(a) Against M_∞.

(b) Against α.

(c) Against β.

Figure 5.11 L **histories against dispersive flight condition parameters of three robust optimal solutions obtained by using DFMOSS.**



(a) Against M_∞.

(b) Against α.

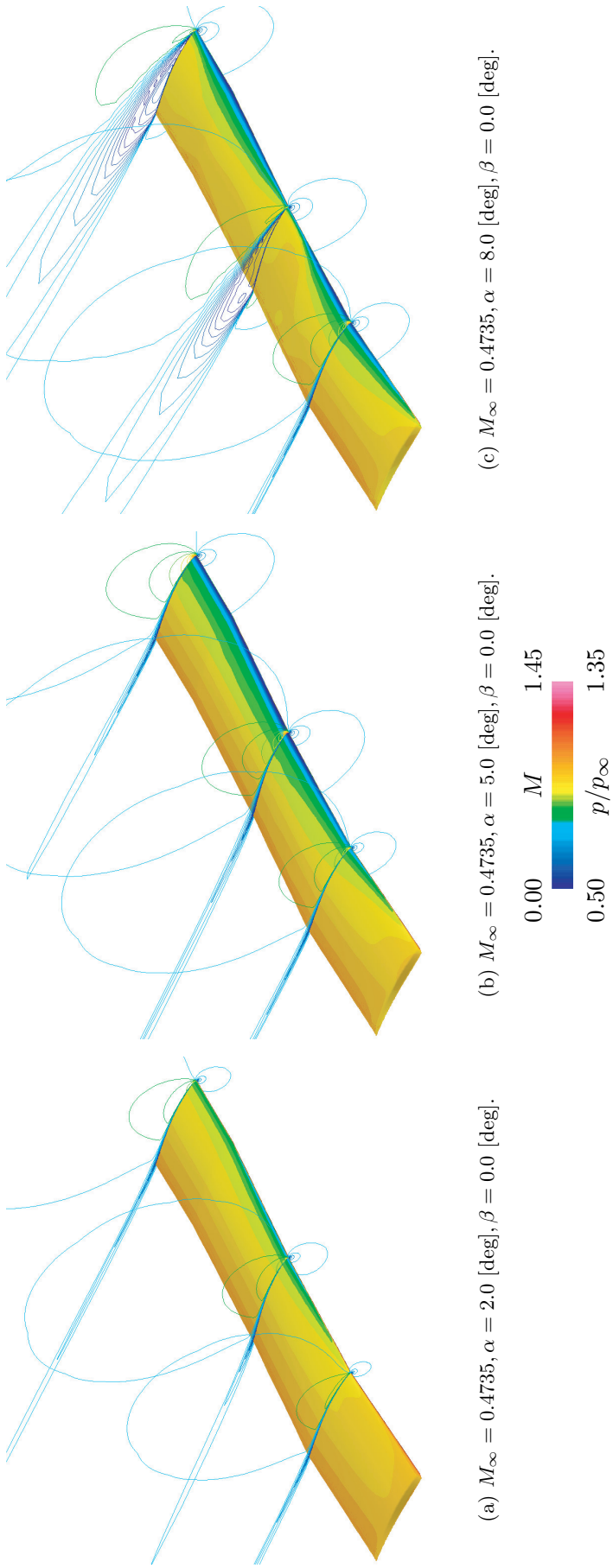
(c) Against β.

Figure 5.12 D **histories against dispersive flight condition parameters of three robust optimal solutions obtained by using DFMOSS.**

against the wind variations as shown in Fig. 5.13, the flow is almost attached over the whole region at α less than 5.0 deg. However, suddenly it begins to separate from the leading edges of the root and mid-span sections at $\alpha = 8.0$ [deg]. Over the upper surface at the sections where the leading-edge separation occurs, the flow can not be expanded sufficiently and the pressure level is kept higher compared to the attached flow. In the robust optimal solution with smaller standard deviation of L against the wind variations as shown in Fig. 5.14, on the other hand, the leading-edge separation already occurs at mid-span section at the design point $\alpha = 2.0$ [deg]. Then, the flow begins to separate also at the root section at α higher than 5.0 deg.

Figures 5.15 and 5.16 show the streamline patterns over the upper wing surface at three different α (2.0, 5.0 and 8.0 [deg]) of the robust optimal solution with larger and smaller standard deviations of L against the wind variations obtained by using the DFMOSS, respectively. In the robust optimal solution with larger standard deviation of L against the wind variations as shown in Fig. 5.15, the flow is fully attached over the whole upper surface at the design point $\alpha = 2.0$ [deg]. At $\alpha = 5.0$ [deg], the flow slightly separates from the rear part of the upper wing surface between the root and the mid-span sections, but leading-edge separations do not occur at any spanwise position. At $\alpha = 8.0$ [deg], suddenly vast leading-edge separation occurs over the upper wing surface between the root and the mid-span sections. In the robust optimal solution with smaller standard deviation of L against the wind variations as shown in Fig. 5.16, on the other hand, the leading-edge separation occurs over the upper surface of mid-span section at the design point $\alpha = 2.0$ [deg], and then the leading-edge separation region gradually spreads in the spanwise direction with the increment in α. Comparing these robust optimal solutions, the growth of leading-edge separation region with the increment in α of the robust optimal solution with smaller standard deviation of L looks weaker than that of the robust optimal solution with larger standard deviation of L.

Figure 5.17 shows the spanwise distributions of local L per unit span length at various α (2.0, 5.0 and 8.0 deg). The robust optimal solutions with larger and medium standard deviation of L against the wind variations have the ideal elliptic spanwise local L distributions which can realize large total L at α less than 5.0 deg. However, suddenly these distribution have dips between the root and the mid-span sections ($y/c = 0.0 \sim 3.0$) at $\alpha = 8.0$ [deg] which leads to the decrement in total L. These dips correspond to the sections where the leading-edge separations occur because the leading-edge separation involves higher pressure level over the upper

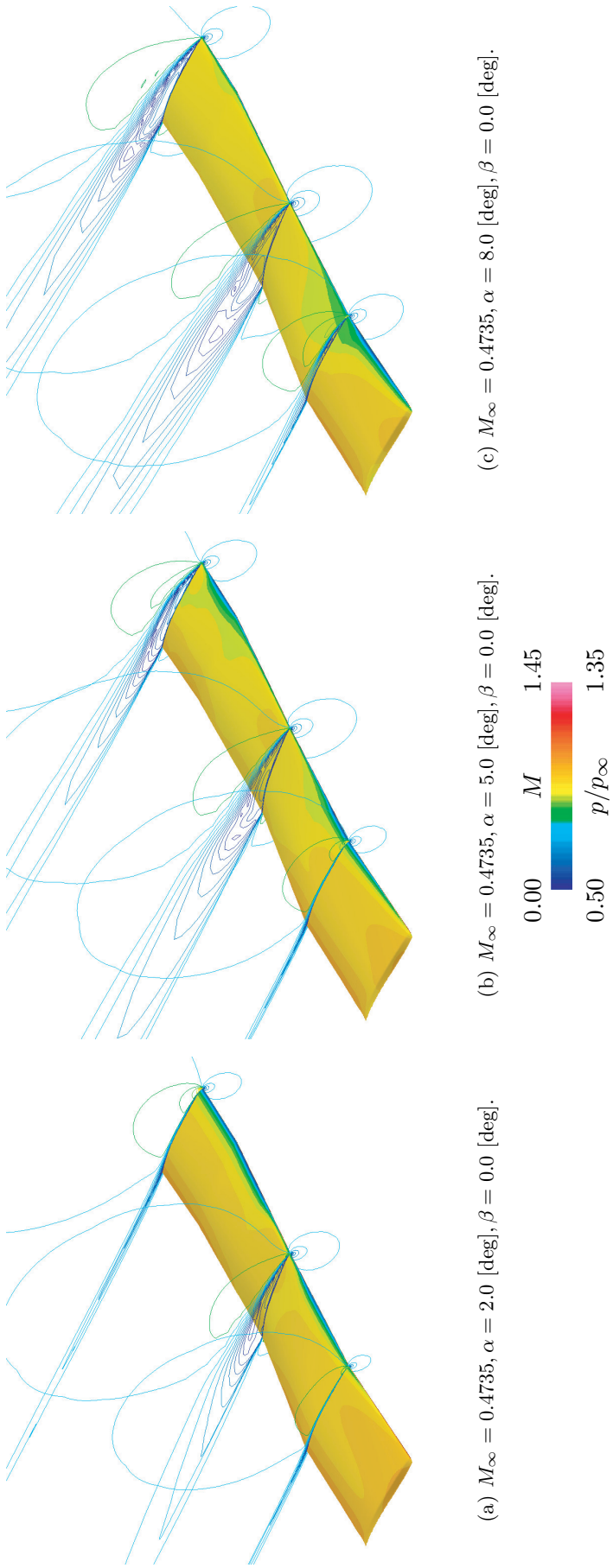


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg], $\beta = 0.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg], $\beta = 0.0$ [deg].

Figure 5.13 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different α of robust optimal solution with larger standard deviation of L (shown for only right half wings).

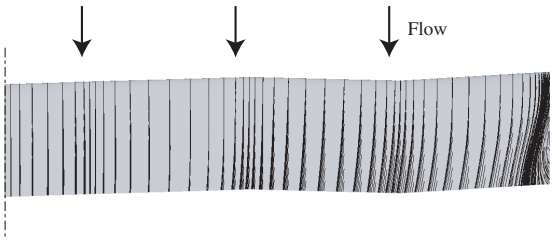


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].

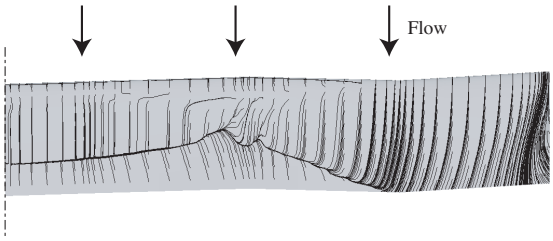
(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg], $\beta = 0.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg], $\beta = 0.0$ [deg].

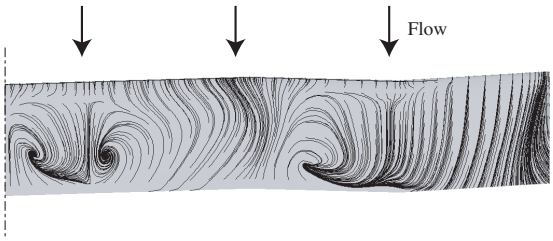
Figure 5.14 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different α of robust optimal solution with smaller standard deviation of L (shown for only right half wings).



(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].

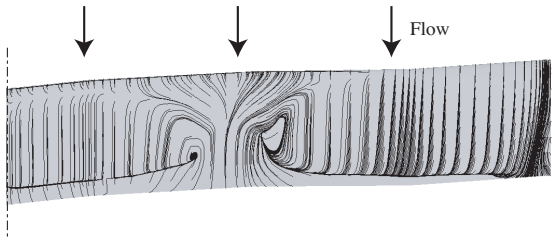


(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg], $\beta = 0.0$ [deg].

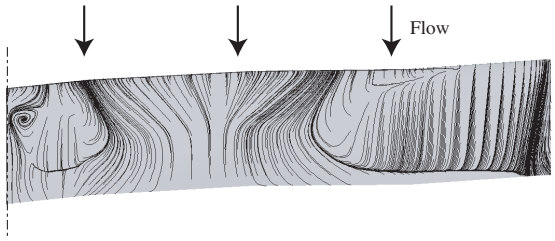


(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg], $\beta = 0.0$ [deg].

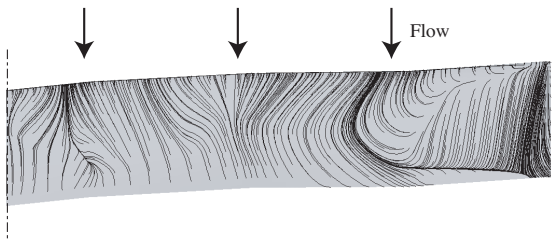
Figure 5.15 Streamline patterns over the upper wing surface at three different α of robust optimal solution with larger standard deviation of L (shown for only right half wings).



(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].



(b) $M_\infty = 0.4735, \alpha = 5.0$ [deg], $\beta = 0.0$ [deg].



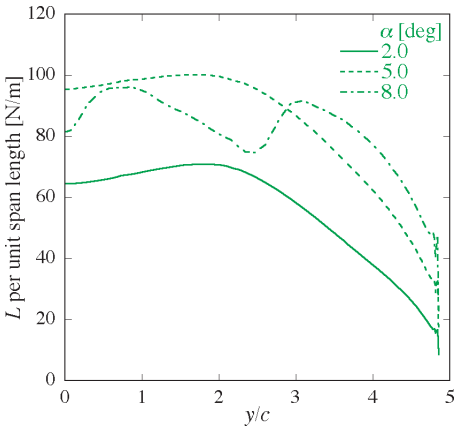
(c) $M_\infty = 0.4735, \alpha = 8.0$ [deg], $\beta = 0.0$ [deg].

Figure 5.16 Streamline patterns over the upper wing surface at three different α of robust optimal solution with smaller standard deviation of L (shown for only right half wings).

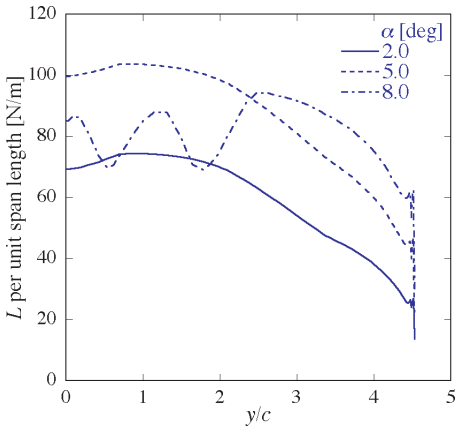
wing surface and eventually the smaller difference of pressure between the upper and lower surfaces, i.e., smaller local L compared to the attached flow. Such drastic changes of spanwise local L distributions due to the leading-edge separation lead to the stall at high α as shown in Fig. 5.11(b). On the other hand, the robust optimal solution with smaller standard deviation of L against the wind variations has the spanwise local L distribution with the dip at mid-span section ($y/c = 1.5 \sim 2.5$) even at the design point $\alpha = 2.0$ [deg], and this dip becomes wider with the increment in α. However, the change of spanwise local L distribution of the robust optimal solution with smaller standard deviation of L is smaller than those of the robust optimal solutions with larger and medium standard deviation of L. This is because the wing configuration of the robust optimal solution with smaller standard deviation of L has larger twist-up angle locally at the mid-span section (2nd fold), as shown in Fig. 5.10, and the leading-edge separation phenomenon becomes more stable against the increment in α than those of the robust optimal solutions with larger and medium standard deviations of L. Therefore, the robust optimal solution with smaller standard deviation of L against the wind variations can realize more robust characteristic of total L against the variation of α.

Next, the variations of aerodynamic performance against β are discussed. Figures 5.18 and 5.19 show the local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different β (0.0, 12.0 and 24.0 [deg]) of the robust optimal solution with larger and smaller standard deviations of L against the wind variations obtained by using the DFMOSS, respectively. In the robust optimal solution with larger standard deviation of L against the wind variations as shown in Fig. 5.18, the flow is almost attached over the whole region at any β. In the robust optimal solution with smaller standard deviation of L against the wind variations as shown in Fig. 5.19, on the other hand, the leading-edge separations occur at mid-span section at the design point $\beta = 0.0$ [deg], but these become weaker with the increment in β and then vanish at $\beta = 24.0$ [deg].

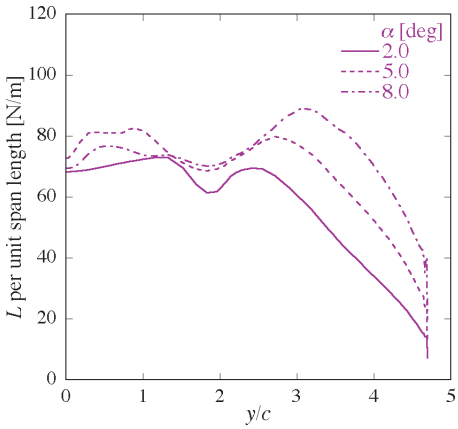
Figures 5.20 and 5.21 show the streamline patterns over the upper wing surface at three different β (0.0, 12.0 and 24.0 [deg]) of the robust optimal solution with larger and smaller standard deviations of L against the wind variations obtained by using the DFMOSS, respectively. In the robust optimal solution with larger standard deviation of L against the wind variations as shown in Fig. 5.20, the flow is fully attached over the whole upper surface at any β. In the robust optimal solution with smaller standard deviation of L against the wind variations as shown in Fig. 5.21,



(a) Robust optimal solution with larger standard deviation of L obtained by using DFMOSS.

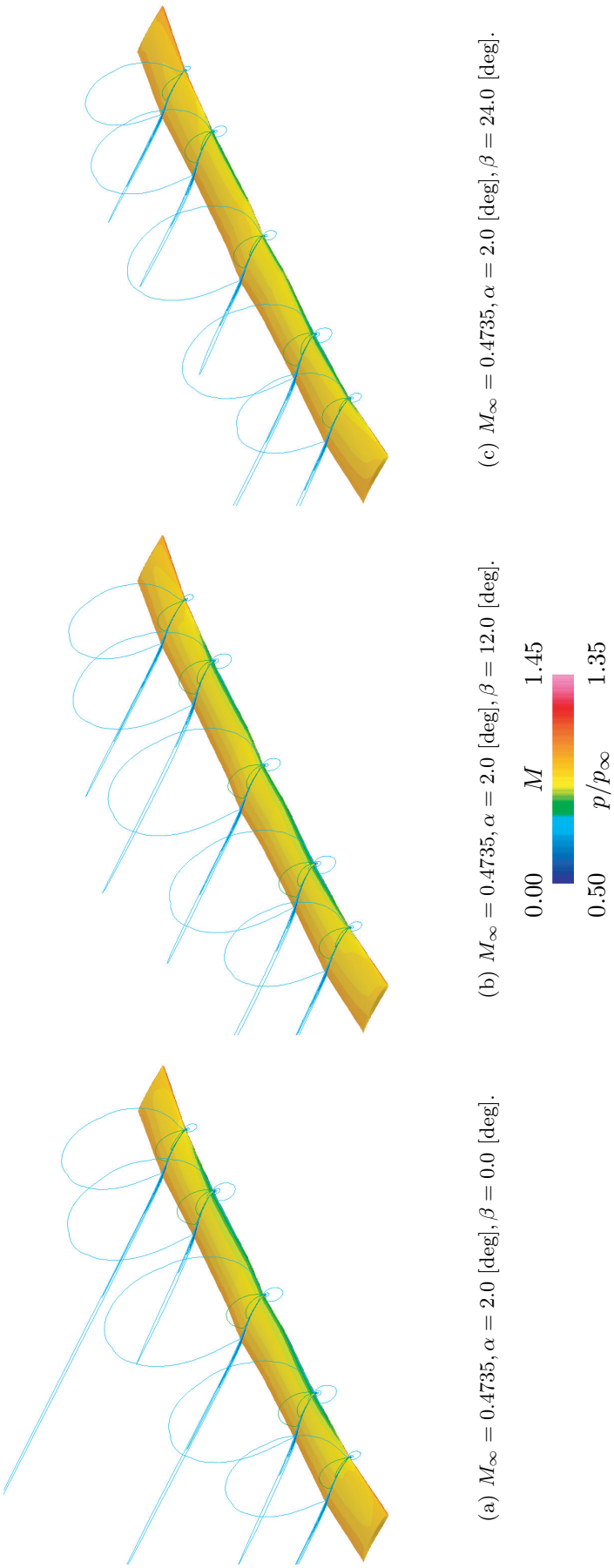


(b) Robust optimal solution with medium standard deviation of L obtained by using DFMOSS.



(c) Robust optimal solution with smaller standard deviation of L obtained by using DFMOSS.

Figure 5.17 Spanwise distributions of local L per unit span length at various α (shown for only right semi-spans).

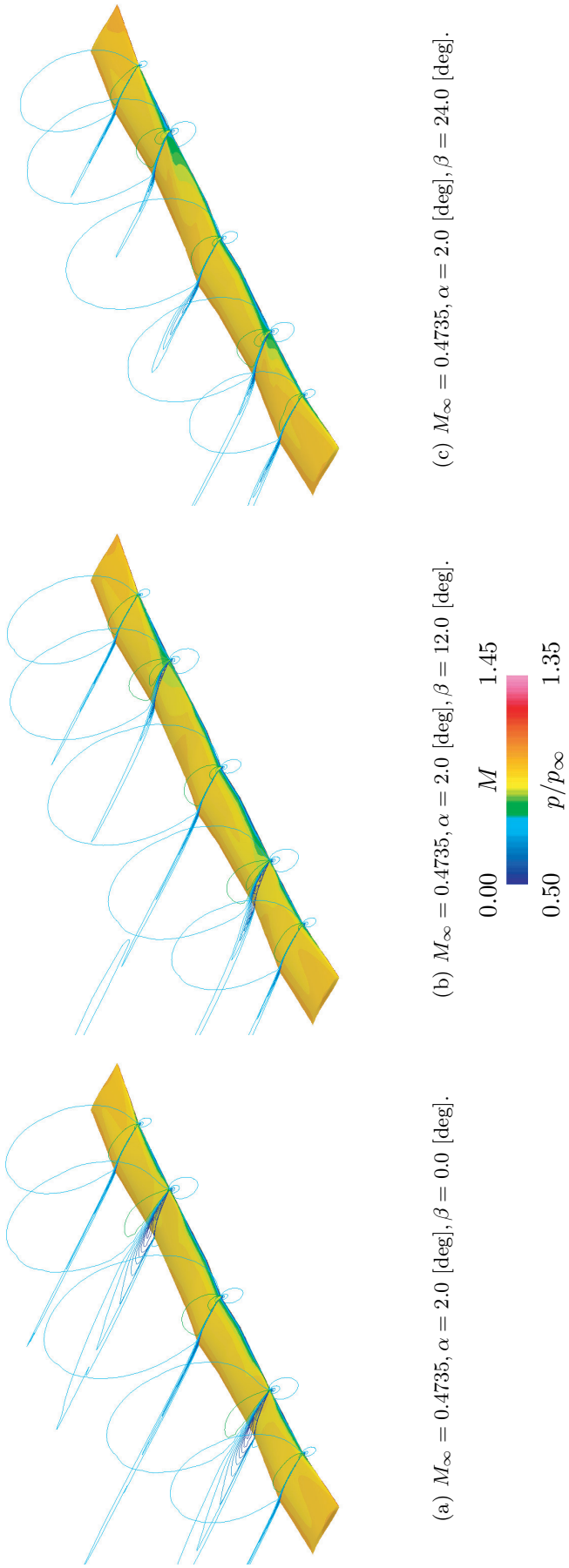


(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 12.0$ [deg].

(c) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 24.0$ [deg].

Figure 5.18 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different β of robust optimal solution with larger standard deviation of L (shown for full wings).



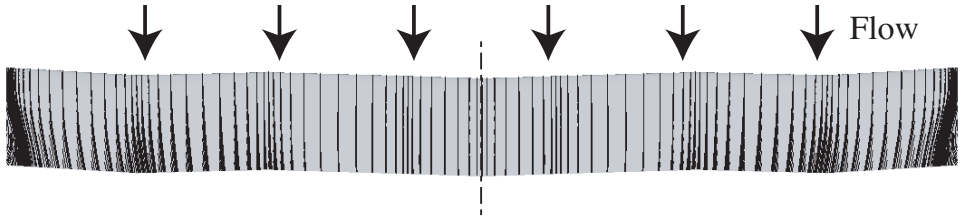
(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].

(b) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 12.0$ [deg].

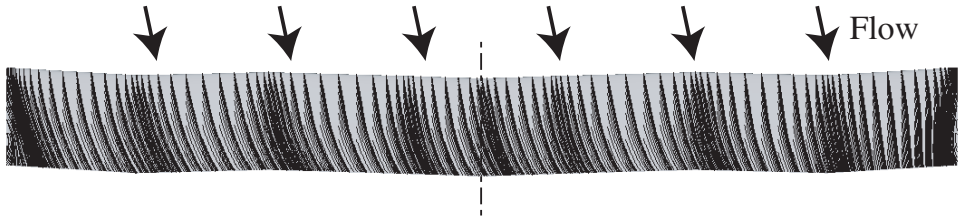
(c) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 24.0$ [deg].

Figure 5.19 Local Mach number contour plots around the wing and pressure contour plots over the wing surface at three different β of robust optimal solution with smaller standard deviation of L (shown for full wings).

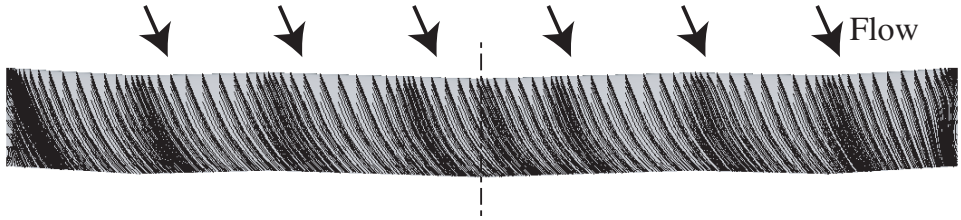
on the other hand, the leading-edge separation occurring over the upper surface of mid-span sections at the design point $\alpha = 2.0$ [deg] vanish with the increment in β.



(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].



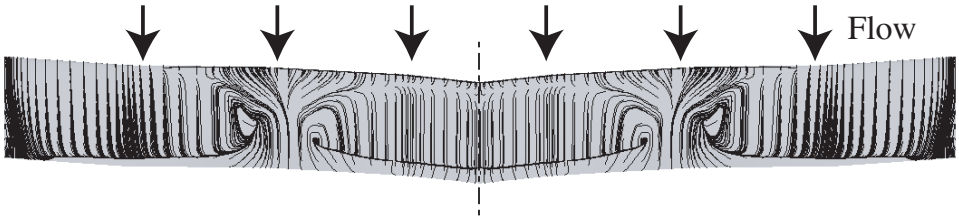
(b) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 12.0$ [deg].



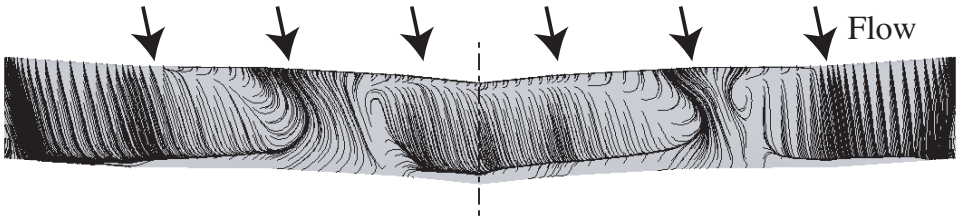
(c) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 24.0$ [deg].

Figure 5.20 Streamline patterns over the upper wing surface at three different β of robust optimal solution with larger standard deviation of L (shown for full wings).

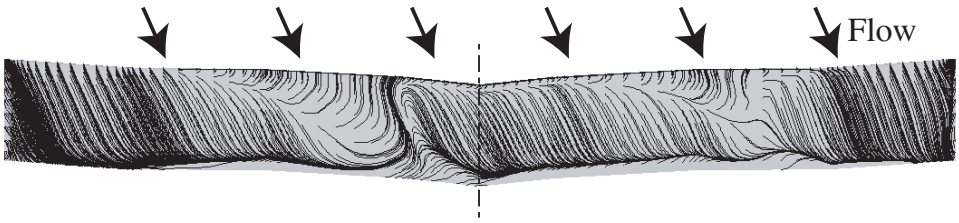
Figure 5.22 shows the spanwise distributions of local L per unit span length at various β (0.0, 12.0 and 24.0 deg). In the robust optimal solutions with larger and medium standard deviation of L against the wind variations, the local L decreases monotonously with the increment in β at all the spanwise positions and it leads to the monotonous decrement in total L against the increment in β as shown in Fig. 5.11(c). This is because the effective angle of attack becomes smaller with the increment in β. On the other hand, the robust optimal solution with smaller standard deviation of L against the wind variations has the spanwise local L distribution with the dips due



(a) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 0.0$ [deg].



(b) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 12.0$ [deg].



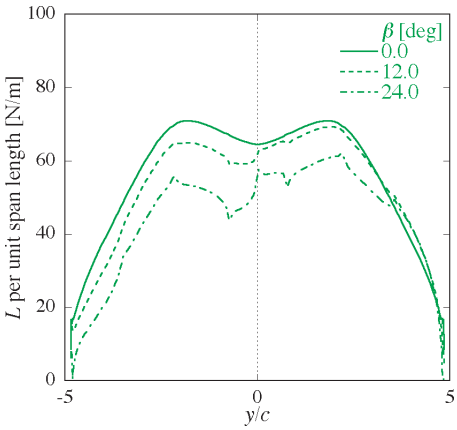
(c) $M_\infty = 0.4735, \alpha = 2.0$ [deg], $\beta = 24.0$ [deg].

Figure 5.21 Streamline patterns over the upper wing surface at three different β of robust optimal solution with smaller standard deviation of L (shown for full wings).

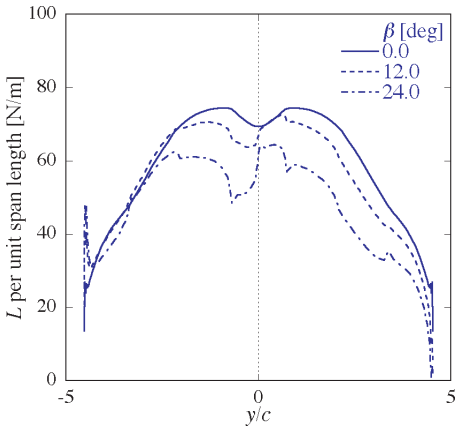
to the leading-edge separations at mid-span sections ($y/c = -2.5 \sim -1.5, 1.5 \sim 2.5$) at the design point $\beta = 0.0$ [deg]. However, these dips vanish and the local L at mid-span sections becomes larger with the increment in β, contrary to the robust optimal solutions with larger and medium standard deviations of L. This is because fortunately the locally large twist-up angle at mid-span sections (2nd folds) of the robust optimal solution with smaller standard deviation of L, as shown in Fig. 5.10, can realize the attached flow and large local L at the mid-span sections when β is high and the effective angle of attack is small, and eventually it leads to large total L even at high β. Therefore, the robust optimal solution with smaller standard deviation of L against the wind variations can realize more robust characteristic of total L against the variation of β.

Here, it should be remembered that, as shown in Fig. 5.8, the solution with better characteristic in terms of the robustness of L against the wind variations at constant optimality of L corresponds to the solutions with worse characteristics in terms of both the optimality and the robustness of D against the wind variations. As described above, the wing configuration of more robust solution in terms of L has larger twist-up angle locally at its mid-span section to stabilize the change of leading-edge separation phenomenon against the variation of α and to vanish the leading-edge separation at high β. However, such wing configuration with locally larger twist-up angle also leads to larger D at the twisted-up section, and worse characteristic in terms of the robustness of D against the wind variations. Such extreme solution with leading-edge separation seems to be undesirable and unrealistic to be used in real-world design. Therefore, more detailed investigation of the obtained trade-off information without such extreme solutions may be useful for more realistic designs of Mars exploratory airplane.

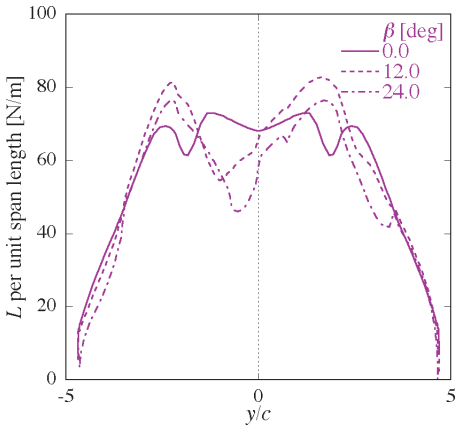
In addition, in the present study, the CFD simulation approach based on the Favre-averaged Navier-Stokes equations with the Baldwin-Lomax algebraic turbulence model is used for the reduction of computation time and resources. It is possible that the present CFD simulation does not have accuracy enough to simulate the flowfields involving large-scale separation phenomena. Therefore, it is required to investigate the accuracy of the present CFD simulation approach by comparing the obtained aerodynamic data with that obtained by more accurate approach such as a large eddy simulation (LES) and a direct numerical simulation (DNS) or corresponding experimental aerodynamic data. Such further investigations of the CFD simulation approach may lead to more realistic and reliable future designs of Mars



(a) Robust optimal solution with larger standard deviation of L obtained by using DFMOSS.



(b) Robust optimal solution with medium standard deviation of L obtained by using DFMOSS.



(c) Robust optimal solution with smaller standard deviation of L obtained by using DFMOSS.

Figure 5.22 Spanwise distributions of local L per unit span length at various β (shown for full spans).

exploratory airplane.

5.5 Summary

In this chapter, the robust aerodynamic design optimization of airfoil and wing planform configurations for Mars exploratory airplane considering the effects of wind variations was performed by using the DFMOSS and the CFD simulation. The present robust aerodynamic design optimization found the solutions with robust characteristic in aerodynamic performance parameters against wind variations, and revealed qualitative trade-off information between the optimality and the robustness in aerodynamic performance parameters. This result showed that the DFMOSS is an effective approach even in large-scale robust design optimizations with many design variables and objective functions.

Among the robust optimal solutions obtained in the present study, the solution with robust characteristic of lift against wind variations had the wing configuration with supercritical-like airfoil and larger twist-up angle locally at the mid-span section. Such wing configuration involved the leading-edge separation, which leaded to small local lift and large local drag, at the twisted-up section at the design point. Such extreme solution with leading-edge separation seems to be undesirable and unrealistic to be used in real-world design. Therefore, more detailed investigation of the obtained trade-off information without such extreme solutions may be useful for more realistic designs of Mars exploratory airplane. In addition, the present CFD simulation approach based on the Favre-averaged Navier-Stokes equations with the Baldwin-Lomax algebraic turbulence model did not have accuracy enough to simulate the flowfields involving large-scale separation phenomena. Therefore, further investigation of the CFD simulation approach for large-scale separation phenomena may lead to more realistic and more reliable future designs of Mars exploratory airplane.

Chapter 6

Concluding Remarks

In this dissertation, a new robust design optimization approach "design for multi-objective six sigma (DFMOSS)" has been developed to solve the drawbacks of a conventional robust optimization approach "design for six sigma (DFSS)" for more efficient and more useful robust design optimizations. Then, aerodynamic design optimizations of Mars exploratory airplane wing considering the effects of wind variations were realized by using the DFMOSS. Finally, realistic design information about the trade-off relation between the optimality and the robustness of aerodynamic performance of Mars exploratory airplane has been discussed based on the numerical results obtained by the present robust aerodynamic design optimizations.

In Chap. 3, the DFMOSS has been developed and applied to simple robust optimization problems to investigate efficiency and usefulness of the DFMOSS. These studies showed that the DFMOSS has some advantages over the DFSS. First, the DFMOSS did not require the advance specification of input parameters such as weighting factors and sigma level. Second, the DFMOSS obtained multiple robust optimal solutions effectively by only one case and revealed trade-off information between optimality and robustness. Third, the satisfied sigma level of each obtained robust optimal solution was evaluated easily and flexibly as post-processing in the robust optimizations. These results indicated clearly that the DFMOSS had more effective and more useful characteristics than the DFSS.

In Chap. 4, three robust aerodynamic design optimizations of airfoil configuration for Mars exploratory airplane considering the effects of wind variations were carried out by using the DFMOSS and the CFD simulation. In all cases, the robust aerodynamic design optimization using the DFMOSS revealed the trade-off relation between the optimality and the robustness in aerodynamic performance. In the first case considering the robustness of lift to drag ratio against the variation of flight

Mach number, it was shown that an airfoil configuration with smaller maximum camber can improve the robustness in lift to drag ratio against the variation of flight Mach number. This is because such airfoil can suppress the growth of shock wave, i.e., realize smaller increment in pressure drag (wave drag) against increment in flight Mach number. In the second case considering the robustness of pitching moment coefficient against the variation of flight Mach number, it was shown that an airfoil configuration with larger curvature in the front part can improve the robustness of pitching moment coefficient against the variation of flight Mach number. This is because such airfoil can suppress the backward movement of shock wave occurred over the upper surface of airfoil against increment in flight Mach number, and eventually it results in smaller change in pitching-down moment produced in the rear part of airfoil against increment in flight Mach number. In the third case considering the robustness of lift to drag ration against the variation of angle of attack, it was shown that an airfoil configuration with blunter leading edge can improve the robustness of lift to drag ratio against the variation of angle of attack. This is because such airfoil configuration can suppress the growth of the separation bubble generated near the leading edge against increment in angle of attack. It leads to gentler change of negative pressure level in the front part near the leading edge, i.e., smaller change of drag against increment in angle of attack, and eventually it results in smaller change of lift to drag ratio against the variation of angle of attack.

In Chap. 5, robust aerodynamic design optimization of airfoil and wing planform configurations for Mars exploratory airplane considering effects of wind variations was performed by using the DFMOSS and the CFD simulation. The present robust aerodynamic design optimization found the solutions with robust characteristic in aerodynamic performance parameters against wind variations, and revealed qualitative trade-off information between the optimality and the robustness in aerodynamic performance parameters. This result showed that the DFMOSS is an effective approach even in large-scale robust design optimizations with many design variables and objective functions. Among the robust optimal solutions obtained in the present study, the solution with robust characteristic of lift against wind variations had the wing configuration with supercritical-like airfoil and larger twist-up angle locally at the mid-span section. Such wing configuration involved the leading-edge separation, which leaded to small local lift and large local drag, at the twisted-up section at the design point. Such extreme solution with leading-edge separation seems to be undesirable and unrealistic to be used in real-world design. Therefore, more detailed

investigation of the obtained trade-off information without such extreme solutions may be useful for more realistic designs of Mars exploratory airplane. In addition, the present CFD simulation approach based on the Favre-averaged Navier-Stokes equations with the Baldwin-Lomax algebraic turbulence model did not have accuracy enough to simulate the flowfields involving large-scale separation phenomena. Therefore, further investigation of the CFD simulation approach for large-scale separation phenomena may lead to more realistic and more reliable future designs of Mars exploratory airplane.

Through the whole study of this dissertation, it became clear that the newly developed robust design optimization approach DFMOSS has capability enough to obtain trade-off information between optimality and robustness which is very useful and important design information in real-world designs, and this approach also is powerful design tools in the real-world aerodynamic design of Mars exploratory airplane wing. It is hoped that the robust design optimizations using the DFMOSS and the knowledge obtained in this dissertation contribute to further developments of the Mars exploratory airplanes and the future Mars exploration missions.

Bibliography

[1] URL: http://marsprogram.jpl.nasa.gov/mgs/.

[2] URL: http://www.esa.int/SPECIALS/Mars_Express/.

[3] URL: http://marsrovers.jpl.nasa.gov/home/.

[4] URL: http://www.isas.jaxa.jp/e/enterp/missions/hayabusa/inadex.shtml.

[5] URL: http://www.isas.jaxa.jp/e/enterp/missions/planet-c/index.shtml.

[6] URL: http://www.isas.jaxa.jp/e/enterp/missions/solar-b/index.shtml.

[7] URL: http://www.isas.jaxa.jp/e/enterp/missions/selene/index.shtml.

[8] URL: http://www.isas.jaxa.jp/e/enterp/missions/lunar-a/index.shtml.

[9] URL: http://www.isas.jaxa.jp/e/enterp/missions/bepi/index.shtml.

[10] URL: http://www.jaxa.jp/missions/plan/long_term/index_j.html, in Japanese.

[11] Hall, D. W., Parks, R. W., and Morris, S., "Airplane for Mars Exploration," Tech. rep., NASA Ames Research Center, Moffett Federal Airfield, California, May 1997, URL: http://www.redpeace.org/final_report.pdf.

[12] URL: http://marsairplane.larc.nasa.gov/.

[13] Guynn, M. D., Croom, M. A., Smith, S. C., Parks, R. W., and Gelhausen, P. A., "Evolution of a Mars Airplane Concept for the ARES Mars Scout Mission," AIAA Paper 2003–6578, September 2003.

[14] Sandford, S. P., Croom, M. A., Moses, R. W., Kuhl, C. A., Guynn, M. D., Braun, R. D., Levine, J. S., and Langford, J. S., "ARES and Beyond: Autonomous Aerial Platforms Provide a Unique Measurement Capability for Earth and Planetary Science," AIAA Paper 2003–6610, September 2003.

[15] Tanaka, Y., Okabe, Y., Suzuki, H., Nakamura, K., Kubo, D., Tokuhiro, M., and Rinoie, K., "Conceptual Design of Mars Airplane for Geographical Exploration," *Proceedings of the 36th JSASS Annual Meeting*, Tokyo, April 2005, pp. 61–64, in Japanese.

[16] URL: `http://www.enae.umd.edu/AGRC/Design00/MARV.html`.

[17] Tsuzuki, N., *A Study on a Miniature Rotary-Wing Vehicle for Mars Exploration: Its Feasibility and Aerodynamic Characteristics of the Rotor*, Doctoral Dissertation, University of Tokyo, Tokyo, 2005.

[18] URL: `http://avdil.gtri.gatech.edu/RCM/RCM/Entomopter/EntomopterProject.html`.

[19] Matsuda, Y., *Planetary Meteorology*, University of Tokyo Press, Tokyo, 2000, in Japanese.

[20] Liebeck, R. H., "A Class of Airfoils Designed for High Lift in Incompressible Flow," *Journal of Aircraft*, Vol. 10, No. 10, October 1973, pp. 610–617.

[21] Jones, R. T., *Wing Theory*, Princeton University Press, Princeton, New Jersey, 1990.

[22] Smith, M. D., Pearl, J. C., Conrath, B. J., and Christensen, P. R., "Thermal Emission Spectrometer Results: Mars Atmospheric Thermal Structure and Aerosol Distribution," *Journal of Geophysical Research*, Vol. 106, No. E10, October 2001, pp. 23929–23945.

[23] Hirota, I., *Global Meteorology*, University of Tokyo Press, Tokyo, 1992, in Japanese.

[24] Fujii, K., *Numerical Methods for Computational Fluid Dynamics*, University of Tokyo Press, Tokyo, 1994, in Japanese.

[25] Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Taylor & Francis, Washington, D.C., 2nd ed., 1997.

[26] Reuther, J. J., Jameson, A., Alonso, J. J., Rimlinger, M. J., and Saunders, D., "Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint For-

mulation and Parallel Computers, Part 1," *Journal of Aircraft*, Vol. 36, No. 1, January–February 1999, pp. 51–60.

[27] Reuther, J. J., Jameson, A., Alonso, J. J., Rimlinger, M. J., and Saunders, D., "Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers, Part 2," *Journal of Aircraft*, Vol. 36, No. 1, January–February 1999, pp. 61–74.

[28] Jameson, A., "Optimum Aerodynamic Design Using CFD and Control Theory," AIAA Paper 1995–1729, June 1995.

[29] Vanderplaats, G. N., *Numerical Optimization Techniques for Engineering Design: With Applications*, McGraw-Hill, Inc., New York, 1984.

[30] *Design Optimization Tools Users Manual*, Vanderplaats Research & Development, Inc., Colorado Springs, Colorado, 1999.

[31] Goldberg, D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1989.

[32] Oyama, A., *Wing Design Using Evolutionary Algorithms*, Doctoral Dissertation, Tohoku University, Sendai, March 2000.

[33] Deb, K., *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Ltd., Chichester, 2001.

[34] Sasaki, D., *Adaptive Range Multi-Objective Genetic Algorithms for Aerodynamic Design Problems*, Doctoral Dissertation, Tohoku University, Sendai, January 2004.

[35] Youn, B. D. and Choi, K. K., "Selecting Probabilistic Approaches for Reliability-Based Design Optimization," *AIAA Journal*, Vol. 42, No. 1, January 2004, pp. 124–131.

[36] *iSIGHT Reference Guide Version 7.1*, Engineous Software, Inc., 2002, pp. 220–233.

[37] Koch, P. N., Wujek, B., Golovidov, O., and Simpson, T. W., "Facilitating Probabilistic Multidisciplinary Design Optimization Using Kriging Approximation Models," AIAA Paper 2002–5415, September 2002.

[38] Shimoyama, K., Fujii, K., and Kobayashi, H., "Development of Realistic Optimization Method – Multi-Objective and Robust Optimization," AIAA Paper 2004–4475, August–September 2004.

[39] Deb, K. and Gupta, H., "Searching for Robust Pareto-Optimal Solutions in Multi-objective Optimization," *Evolutionary Multi-Criterion Optimization*, edited by C. A. Coello Coello, A. H. Aguirre, and E. Zitzler, Springer-Verlag, Berlin Heidelberg New York, 2005, pp. 150–164.

[40] Rai, M. M., "Robust Optimal Design With Differential Evolution," AIAA Paper 2004–4588, August–September 2004.

[41] Putko, M. M., Taylor III, A. C., Newman, P. A., and Green, L. L., "Approach for Input Uncertainty Propagation and Robust Design in CFD Using Sensitivity Derivatives," *Journal of Fluids Engineering, Transactions of the ASME*, Vol. 124, March 2002, pp. 60–69.

[42] Gumbert, C. R., Newman, P. A., and Hou, G. J.-W., "Effect of Random Geometric Uncertainty on the Computational Design of a 3-D Flexible Wing," AIAA Paper 2002–2806, June 2002.

[43] Kim, S., Ahn, J., and Kwon, J. H., "Reliability Based Wing Design Optimization Using Trust Region Framework," AIAA Paper 2004–4407, August–September 2004.

[44] Huyse, L., Padula, S. L., Lewis, R. M., and Li, W., "Probabilistic Approach to Free-Form Airfoil Shape Optimization Under Uncertainty," *AIAA Journal*, Vol. 40, No. 9, September 2002, pp. 1764–1772.

[45] Darwin, C., *On the Origin of Species by Means of Natural Selection*, Murray, London, 1859.

[46] Holland, J. H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan, 1975.

[47] Baker, J. E., "Reducing Bias and Inefficiency in the Selection Algorithm," *Proceedings of the 2nd International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1987, pp. 41–49.

[48] Eshelman, L. J. and Schaffer, J. D., "Real-coded Genetic Algorithms and Interval Schemata," *Foundations of Genetic Algorithms 2*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1993, pp. 187–202.

[49] Michalewicz, Z., *Genetic Algorithms + Data Structure = Evolution Programs*, Springer-Verlag, Berlin Heidelberg New York, 3rd, Revised and Extended ed., 1996.

[50] Eshelman, L. J., "The CHC Adaptive Search Algorithm: How to Have Safe When Engaging in Nontraditional Genetic Recombination," *Foundations of Genetic Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1991, pp. 265–283.

[51] Tsutsui, S. and Fujimoto, Y., "Forking Genetic Algorithms with Blocking and Shrinking modes (fGA)," *Proceedings of the 5th International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1993, pp. 206–213.

[52] De Jong, K. A., *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*, Doctoral Dissertation, University of Michigan, Ann Arbor, Michigan, 1975.

[53] Fonseca, C. M. and Fleming, P. J., "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization," *Proceedings of the 5th International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1993, pp. 416–423.

[54] Deb, K., "An Efficient Constraint-Handling Method for Genetic Algorithms," *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, No. 2–4, 2000, pp. 311–338.

[55] Coello Coello, C. A., "Constraint-Handling Using an Evolutionary Multiobjective Optimization Technique," *Civil Engineering and Environmental Systems*, Vol. 17, 2000, pp. 319–346.

[56] Oyama, A., Fujii, K., Shimoyama, K., and Liou, M.-S., "Pareto-Optimality-Based Constraint-Handling Technique and Its application to Compressor Design," AIAA Paper 2005–4983, June 2005.

[57] Oyama, A., Shimoyama, K., and Fujii, K., "New Constraint-Handling Method for Multi-Objective Multi-Constraint Evolutionary Optimization and Its Application to Space Plane Design," *EUROGEN 2005*, edited by R. Schilling, W. Hasse, J. Periaux, H. Baier, and G. Bugeda, FLM, Munich, September 2005.

[58] Runarsson, T. P. and Yao, X., "Stochastic Ranking for Constrained Evolutionary Optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 4, No. 3, 2000, pp. 284–294.

[59] Saliby, E., "Descriptive Sampling: A Better Approach to Monte Carlo Simulation," *Journal of the Operational Research Society*, Vol. 41, No. 12, 1990, pp. 1133–1142.

[60] Ramberg, J. S. and Schmeiser, B. W., "An Approximate Method for Generating Symmetric Random Variables," *Communications of the ACM*, Vol. 15, No. 11, November 1972, pp. 987–990.

[61] McKay, M. D., Beckman, R. J., and Conover, W. J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, May 1979, pp. 239–245.

[62] Liepmann, H. W. and Roshko, A., *Elements of Gasdynamics*, Dover Publications, Inc., Mineola, New York, 2001.

[63] Steger, J. L., "Implicit Finite-Difference Simulation of Flow about Arbitrary Two-Dimensional Geometries," *AIAA Journal*, Vol. 16, No. 7, July 1978, pp. 679–686.

[64] Pulliam, T. H. and Steger, J. L., "Implicit Finite-Difference Simulations of Three-Dimensional Compressible Flow," *AIAA Journal*, Vol. 18, No. 2, February 1980, pp. 159–167.

[65] Tamura, Y. and Fujii, K., "Conservation Law for Moving and Transformed Grids," AIAA Paper 93–3365, 1993.

[66] Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

[67] Anderson, W. K., Thomas, J. L., and van Leer, B., "Comparison of Finite Volume Flux Vector Spllittings for the Euler Equations," *AIAA Journal*, Vol. 24, No. 9, September 1986, pp. 1453–1460.

[68] Vatsa, V. N., Thomas, J. L., and Wedan, B. W., "Navier-Stokes Computations of Prolate Spheroids at Angle of Attack," AIAA Paper 87–2627, Augest 1987.

[69] Liou, M.-S. and Steffen, Jr., C. J., "A New Flux Splitting Scheme," *Journal of Computational Physics*, Vol. 107, 1993, pp. 23–29.

[70] Shima, E. and Jounouchi, T., "Role of CFD in Aeronautical Engineering (No. 14) –AUSM Type Upwind Schemes–," *Proceedings of the 14th NAL Symposium on Aircraft Computational Aerodynamics*, NAL SP–34, Tokyo, January 1997, pp. 7–12.

[71] van Leer, B., "Towards the Ultimate Conservation Difference Scheme. IV. A New Approach to Numerical Convection," *Journal of Computational Physics*, Vol. 23, 1977, pp. 276–299.

[72] van Leer, B., "Towards the Ultimate Conservation Difference Scheme. V. A Second-Order Sequel to Godunov's Method," *Journal of Computational Physics*, Vol. 32, 1979, pp. 101–136.

[73] Fujii, K. and Obayashi, S., "Practical Applications of New LU-ADI Scheme for the Three-Dimensional Navier-Stokes Computation of Transonic Viscous Flows," AIAA Paper 86–0513, January 1986.

[74] Beam, R. M. and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal*, Vol. 16, No. 4, April 1978, pp. 393–402.

[75] Pulliam, T. H. and Chaussee, D. S., "A Diagonal Form of an Implicit Approximate-Factorization Algorithm," *Journal of Computational Physics*, Vol. 39, 1981, pp. 347–363.

[76] Lombard, C. K., Bardina, J., Venkatapathy, E., and Oliger, J., "Multi-Dimensional Formulation of CSCM – An Upwind Flux Difference Eigenvector Split Method for the Compressible Navier-Stokes Equations," AIAA Paper 83–1895, July 1983.

[77] Yoon, S. and Jameson, A., "An LU-SSOR Scheme for the Euler and Navier-Stokes Equations," AIAA Paper 87–0600, January 1987.

[78] Obayashi, S. and Kuwahara, K., "An Approximate LU Factorization Method for the Compressible Navier-Stokes Equations," *Journal of Computational Physics*, Vol. 63, No. 1, 1986, pp. 157–167.

[79] MacCormack, R. W., "A Numerical Method for Solving the Equations of Compressible Viscous Flow," *AIAA Journal*, Vol. 20, No. 9, September 1982, pp. 1275–1281.

[80] Brandt, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Mathematics of Computation*, Vol. 31, No. 138, April 1977, pp. 333–390.

[81] Baldwin, B. S. and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78–257, January 1978.

[82] Cebeci, T., "Calculation of Compressible Turbulent Boundary Layers with Heat and Mass Transfer," AIAA Paper 71–741, June 1971.

[83] Deb, K., "Optimal Design of a Welded Beam via Genetic Algorithms," *AIAA Journal*, Vol. 29, No. 11, November 1991, pp. 2013–2015.

[84] Mezura-Montes, E., Coello Coello, C. A., and Landa-Becerra, R., "Engineering Optimization Using a Simple Evolutionary Algorithm," *Proceedings of the 15th International Conference on Tools with Artificial Intelligence*, Sacramento, California, November 2003, pp. 149–156.

[85] Oyama, A., Obayashi, S., Nakahashi, K., and Hirose, N., "Fractional Factorial Design of Genetic Coding for Aerodynamic Optimization," AIAA Paper 99–3298, June–July 1999.