SHAPE OPTIMIZATION OF CAMBERED AIRFOILS USING A GENETIC ALGORITHM AND A MULTIPOINT INVERSE METHOD

BY

KYLE TSAI

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Aerospace Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Adviser:

Professor Michael S. Selig

ABSTRACT

This thesis presents a process to optimize a cambered airfoil with the MATLAB genetic algorithm (GA) and a multipoint inverse method called PROFOIL. XFOIL was used to evaluate the aerodynamic performance of each airfoil. Data processing techniques and a custom penalty function were developed in order to overcome challenges in integrating these tools. The viability of this approach was assessed in three airfoil optimization studies. First, the optimizer was tuned using a parameterized study of various GA configurations for optimizing the $(C_l/C_d)_{max}$ for an 18% thick airfoil at the design conditions: $C_m = -0.063$ at $Re = 6.88 \times 10^6$ and $C_m = -0.030$ at $Re = 2.00 \times 10^6$. The first is a typical flow condition for a wind turbine at the $r/R = 0.75$ blade section location, and the second is identical to the requirements used in designing the Liebeck L1003. This tuned GA was used for the rest of the thesis. In the second study, the optimization of $(C_l/C_d)_{max}$ for an 18% thick airfoil with design $C_m = -0.060$ was conducted at $Re = 6.00 \times 10^6$, which is a typical condition for general aviation aircraft. It was observed that the optimized airfoil resembles the Liebeck L1003 airfoil, which was designed with a Stratford pressure recovery distribution. In the third study, a series of $C_{l_{max}}$ optimization runs was performed for varying pitching moments at $Re = 6.00 \times 10^6$, revealing final solutions that segregated into two types of airfoils that differ in camber. It was shown that the optimizer converged on reflexed airfoils for design coefficients of moment $C_m = 0.000$ through $C_m = -0.050$ and on aft-loaded airfoils for $C_m = -0.075$ through $C_m = -0.200$. In addition, both groupings of airfoils exhibited an increase in $C_{l_{max}}$ concomitant with increasing nose-down pitching moment. The results indicate that this approach can reproduce airfoils designed with central design philosophies using only a limited number of design inputs.

To Caroline, Cindy, and Eve.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my family for instilling in me the importance of education and the drive to work hard for the things I believe in. I would not have had the privilege of pursuing higher education were it not for the sacrifices you made for me. I would also like to thank Michael Rogers for your presence throughout my life. You taught me to see goodness in people. Tim, Irma, Tiffany, and Mike Mulligan, I feel so lucky to have my California family. Thank you for giving me a home away from home.

Professor Selig, you took a chance on me and let me run wild with my ideas. And when I flew too close to the sun, you pulled me back. I am deeply grateful for your guidance during my time in your lab and am honored to have worked with you.

Gavin Ananda, you've patiently helped me through countless research problems—thank you for always having an open door to your office. Giovanni Fiore, you were my partner for all things GA and PROFOIL related, and I am thankful we did this together. Or Dantsker, thank you for providing me the computing equipment I needed to complete my research. I also want to express gratitude to Amal Sahai, Ian Jessen, and Sara Timtim for their expertise on genetic algorithms.

To the members of the UIUC Applied Aerodynamics Group past and present: Daniel Uhlig, Rob Deters, Adam Ragheb, Gavin Ananda, Giovanni Fiore, Or Dantsker, and Brent Pomeroy, my journey through graduate school has been enriched by our interactions. To Guy Tal, thank you for giving me the opportunity to be an educator. And finally, to Yao Huang who supported me through the entire research process.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$A_{crossed}$	area of crossed region of airfoil
A_{total}	total enclosed area of airfoil
C	array representing the child individual
C_d	airfoil coefficient of drag
C_f	airfoil coefficient of friction
C_l	airfoil coefficient of lift
$C_{l_{max}}$	maximum airfoil coefficient of lift
C_l/C_d	airfoil glide ratio
$(C_l/C_d)_{max}$	maximum airfoil lift-to-drag ratio
C_m	airfoil coefficient of moment
f_{T1}	fitness of Tier 1 penalty
f_{T2}	fitness of Tier 2 penalty
f_{T2}	fitness of Tier 3 penalty
i_{LE}	index of the leading edge arc limit
K	upper surface recovery parameter
$\overline{K}$	lower surface recovery parameter
m	crossover point
n	crossover point
K_s	trailing-edge thickness parameter
n_{cross}	number of crossover children
$n_{crossings}$	number of crossed airfoil segments
n_{elite}	number of elite children
n_{mut}	number of mutation children

n_{panels}	number of coordinates in airfoil
n_{pop}	number of individuals in a population
n_{tour}	tournament size
P_1	array representing the first parent individual
P_2	array representing the second parent individual
r_{cross}	crossover fraction of population
r_{elite}	ratio of elite count to population
r_{heur}	heuristic ratio
r_{inter}	intermediate ratio
r_{rand}	random number in the interval (0,1)
Re	Reynolds number
Re_x	Reynolds number along a flat plate
r/R	non-dimensionalized blade section location
s	control points of an airfoil
t/c	thickness to chord ratio
v^*	design velocity level for a specified segment
α	angle of attack
α^*	design angle of attack referenced to zero-lift line
μ	mean
σ	standard deviation
τ_w	shear stress at wall
ϕ	arc limit
ϕ_{ISEG}	penultimate arc limit
ϕ_s	trailing-edge upper surface closure arc limit
$\overline{\phi}_s$	trailing-edge lower surface closure arc limit

LIST OF ABBREVIATIONS

ABCO	Aggregate-based Crossover Operator
DCO	Discrete Crossover Operator
FG	Final Generation
FOM	Figure of Merit
GA	Genetic Algorithm
MF	Maximum Fitness

CHAPTER 1

INTRODUCTION

Over the next twenty years, passenger air traffic is projected to grow by 4.9% annually and air cargo by 4.7% annually, and the demand for airframes will reach 43,560 by 2034, more than double the current number of commercial airplanes in the sky today [1]. One of the consequences of this dramatic growth is increased carbon footprint. In 2012, the aviation industry produced 689 million tons of CO_2 [2]. The ICAO estimates that by 2050, this number could grow to 4,531 million tons if aggressive environmental policies and fuel-saving technologies are not actively developed [3]. With increasing demand for air travel and environmental impact, the goal of improving the aerodynamic efficiency of an aircraft is as salient as ever.

1.1 Motivations

At the crux of the performance of an aircraft is its airfoil, the optimization of which can provide returns on its carbon footprint and operating cost over the course of the entire product lifecycle. There exist two fundamental approaches to airfoil design: the direct and the inverse method [4]. In the direct method, the designer starts with an airfoil geometry, which is analyzed to determine the velocity distribution from which its aerodynamic performance can be derived as shown in Fig. 1.1. Depending on the result, iteration continues until acceptable results are achieved. The disadvantage of the direct method is that searching for the optimum design is nearly accidental rather than a deliberate process. In other words, direct methods are wasteful in their navigation of the search space as there is high non-linearity between the shape of the geometry and the actual behavior of the flow [5]. The inverse design approach, on the other hand, prescribes the velocity distribution from the outset in

order to derive the airfoil shape as shown in Fig. 1.2. In this way, requirements and design goals, manifested in the velocity distribution, directly drive the design of the airfoil, and in doing so, lessen the computational effort in optimization.

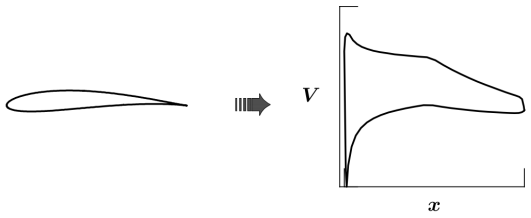

Figure 1.1: Direct method concept (taken from Ref. [4]).

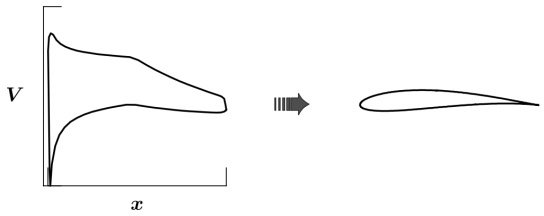

Figure 1.2: Inverse method concept (taken from Ref. [4]).

Realistically, for an airfoil to actually be useful, it must have adequate performance across multiple design objectives. For example, it may be necessary for an airfoil to operate at both high and low-lift conditions. In single point design, the airfoil would be designed for one objective and then reconciled against the other in post-design analysis. PROFOIL [4] is an inverse method that can simultaneously prescribe the velocity distribution of the upper surface at a high angle of attack and the lower surface at a low angle of attack. The multipoint design capability of PROFOIL enables the designer to meet all requirements in one pass instead of compromising across multiple attempts, which incidentally, also increases the computational efficiency of the approach.

If a multipoint inverse solver were to be coupled with a genetic algorithm (GA), the pairing could be a powerful way to automate airfoil design. GAs are a search heuristic that mimic biological evolution to solve optimization problems. They are particularly well-suited for the problem of airfoil design because they can traverse the highly nonlinear and discontinuous search spaces inherent in aerodynamics problems. Furthermore, because genetic algorithms

consider entire populations of candidate solutions instead of single candidates, they are less susceptible to being mired in local optima.

1.2 Previous Work

Inverse design methods coupled with genetic algorithms have actually already enjoyed success in designing optimized transonic, low Reynolds number, and wind turbine airfoils [5, 6, 7, 8]. Over the past two decades, three main approaches have arisen: target pressure distribution GA, hybrid GA, and the multipoint inverse GA.

1.2.1 Target Pressure Distribution GA

Given a target pressure distribution, genetic algorithms can be used to evolve an existing airfoil to achieve that performance. Similar to the direct method, the airfoil is geometrically parameterized and a flow solver evaluates its pressure distribution. Unlike the direct method in which the airfoil is evaluated for its aerodynamic performance, the fitness in this approach is calculated as a scaling of the difference between the target pressure distribution and that of the evaluated airfoil. The genetic algorithm iterates on the geometry of the airfoil until that difference is minimized.

Vicini and Quagliarella [9] used two fifth-order B-splines constituting of 22 design variables to parameterize the airfoil geometry. A full potential transonic flow solver was then used to evaluate the pressure distribution of the airfoil. The authors evolved a NACA 0012 airfoil into a CAST-10 airfoil at Mach 0.765 and a 0 degree angle of attack under eight GA configurations over 10 trials. The averaged results indicated that a properly tuned GA reaches convergence in half the number of generations as untuned GAs. The convergence rates of the results also showed that genetic algorithms, in general, are adept at exploring a large design space and rapidly finding a suboptimal solution, but refinement into a global optimum is computationally expensive.

Holst and Pulliam [10] used the PARSEC [11] method to parameterize the airfoil shape with 10 design variables. A 2D implicit flow solver ARC2D [12] was used to evaluate the

pressure distribution. They showed that given a target pressure distribution for an NACA 0012 airfoil at Mach 0.7 and 2 degrees angle of attack, a real-numbered GA can iterate on a population of random initial inputs until the target airfoil is reproduced. They also showed that by imposing a symmetry constraint that halves the number of design variables, convergence was achieved more rapidly.

There are drawbacks to the target distribution approach. As mentioned before, geometric parameterization can be computationally expensive depending on the number of design variables. On the other hand, too few design variables gives rise to the possibility that the search space may be too small. Furthermore, this approach presumes that the optimized pressure distribution of the target airfoil is known a priori, which is presumptuous and limiting as a design tool.

1.2.2 Hybrid GA

The hybrid approach addresses the design limitations of the target pressure distribution approach by allowing the designer to first optimize the target pressure distribution. Instead of being prescribed, the pressure distribution itself is geometrically parameterized by B-splines. The fitness is then directly evaluated using aerodynamic performance characteristics derived by the shape of the pressure distribution, which is optimized using a real-number genetic algorithm. An inverse method then solves for the airfoil shape geometry using the optimized target pressure distribution.

The hybrid approach was first performed by Obayashi and Takanashi [7]. The fitness evaluation was achieved using conventional methods of integrating the area under the pressure distribution curve. The authors used the inverse design code WinDes [8] to perform inverse design. The CFD package LANS2D [13] was then used to validate the results of the inverse design by assessing how close the resulting pressure distribution was from the target.

Two transonic optimization cases were performed for the design conditions of $M = 0.75$, $C_l = 0.5$, and a trailing-edge edge pressure coefficient of 0.15; and $M = 0.80$, $C_l = 0.7$, and trailing-edge pressure coefficient 0.10. It was shown that this process is capable of generating supercritical airfoils for both design conditions.

Though the hybrid approach is not strictly a direct method, the pressure distribution can be said to be directly designed. Vicini and Quaqliarella [9] call this methodology the hybrid approach because of its dualistic nature: the direct design of the pressure distribution followed by the inverse design of the airfoil shape. However, because of the reliance on geometric parameterization for the pressure distribution, it is also susceptible to being computationally expensive and limited by the geometric expression of the pressure distribution itself.

Another shortcoming is the lack of multipoint design capability. The nuances of airfoil design arise when incorporating off-design requirements, such as those of takeoff, landing, or engine-out conditions. In this approach, it is the prerogative of the user to translate design goals into pressure distributions that exhibit the required aerodynamic characteristics, which are usually not obvious. It is possible to perform single point design of an airfoil and assess its suitability for other design points in post-design analysis, but this can lead to a haphazard and wasteful search.

1.2.3 Multipoint Inverse GA

As the name suggests, multipoint inverse design is a method that builds in the capability to satisfy multiple design requirements simultaneously. Airfoils are parameterized by prescribing their target velocity distribution using a conformal mapping method [4]. This velocity distribution prescription is used as an input for the multipoint inverse method to generate an airfoil geometry. Fitness values for the genetic algorithm are obtained by using an inviscid/viscous flow solver to obtain the aerodynamic performance characteristics of each airfoil candidate.

Gardner and Selig [5] used a FORTRAN-based GA [14], an inverse-method airfoil design code PROFOIL, and an inviscid/viscous flow solver XFOIL [15], collectively called ProfoilGA [5]. The authors performed two studies: (1) a comparison of the effectiveness of this approach against that of a conventional direct design approach that uses Bezier curves for airfoil parameterization and (2) an attempt to optimize a cambered airfoil with known design specifications.

In the first study, the authors performed drag minimization for a 10% thick symmetric airfoil at $C_l = 0.92$ and $Re = 0.3 \times 10^6$. ProfoilGA and the direct design approaches were measured against the speed at which an airfoil more efficient than the Eppler E168 is found, and their ability to find the best airfoil in a given amount of time. The authors found that when properly tuned, ProfoilGA performed better than the direct design method according to both metrics. Its speediness and thoroughness in search are attributed to the PROFOIL boundary-layer-development iteration scheme, which allows it to have fewer design variables than direct geometry parameterization. In the second study, the authors used ProfoilGA to optimize a cambered airfoil with the same design specifications. They were able to achieve an airfoil that has a $(C_l/C_d)_{max}$ that is 1% greater than that of the SG6042, considered to be state-of-the-art.

Fiore and Selig [6] used a custom GA coupled with PROFOIL for multi-objective optimization. The authors explored the optimization of an 18% thick wind turbine airfoil at $Re = 5.84 \times 10^6$ for both aerodynamic efficiency, measured by $(C_l/C_d)_{max}$, and resilience to particle erosion. Similar to Gardner's study, they also compared the effectiveness of this approach against that of the direct design method using Bezier curves and found that the multipoint inverse design approach outperforms the direct design method in both the particle erosion figure of merit and in $(C_l/C_d)_{max}$.

In summary, the multipoint inverse design approach has the following advantages over the direct design method:

1) Multipoint inverse design provides the designer with a level of control of the pressure distribution such that multiple design conditions can be satisfied in the optimization loop rather than in the post-design process, saving valuable computational time.

2) PROFOIL requires fewer design variables than direct geometric parameterization, enabling this approach to search the design space more effectively and efficiently. Gardner and Fiore [5, 6] both demonstrate that under the same design specifications, this approach produces airfoils that outperform those produced by the direct design approach.

1.3 Research Objectives

Unlike the aforementioned multipoint inverse approaches, this thesis presents a process for optimizing cambered airfoils using the GA of the MATLAB Optimization Toolbox. Such an approach has the advantages of being rapidly developed, tested, and deployed. The goal of this thesis is to assess the viability of using the MATLAB GA coupled with PROFOIL by performing a series of optimizations for which $(C_l/C_d)_{max}$ or $C_{l_{max}}$ are maximized. Three studies are performed in this thesis: (1) the tuning of a GA coupled with a multipoint inverse method, (2) the optimization of an airfoil for $(C_l/C_d)_{max}$, and (3) $C_{l_{max}}$ optimization for a series of airfoils with varying design pitching moments.

CHAPTER 2

OVERVIEW OF TOOLS

This chapter provides background information on genetic algorithms, outlines the MATLAB GA framework, and introduces PROFOIL usage concepts.

2.1 GA Background

Genetic algorithms are a stochastic search technique inspired by natural evolution. GAs were invented by John Holland in the 1960s and further refined by Holland, his research group, and colleagues. Holland was not the first to study evolution-inspired algorithms, but his work pioneered the widespread application of genetic algorithms to solving practical problems and the establishment of the evolutionary computing field. His innovation was the concept of population-based evolution: the notion of iterating entire populations of potential solutions or individuals generation after generation [16]. In order to understand Holland's genetic algorithm as an analog of natural evolution, it is necessary to review the requisite biological concepts.

2.1.1 Biological Terminology

At the heart of this engine of evolution is the mimicry of Darwin's theory of natural selection as described by Mendelian laws [17]. Darwin's concept of survival of the fittest explains that because of diversity within a population, some individuals will invariably be fitter than others. The fitter individuals will tend to mate with each other and produce children that propagate their advantageous traits. Across millennia, this process leads to evolution. Darwin's theory of evolution provided a soft explanation of how these traits are inherited.

Building on this idea, Mendel refined the mechanism of inheritance to be how we understand it today: that traits are expressed by discontinuous chunks of genetic material (alleles), and these mix according to predetermined rules, which, over time, give rise to the diversity in populations that Darwin first observed.

All organisms are composed of cells that contain DNA stored in the form of thread-like structures called chromosomes, which are essentially chains of genes. Genes are short stretches of DNA that are the basic unit of heredity. Genes occupy specific locations along the chromosome called loci. A variant of a gene at a particular locus is called an allele. A simplified example is that of eye color. The gene for eye color has several different forms such as the allele for blue eyes or that for brown eyes. These diverse alleles translate directly into physical variations that make some organisms fitter than others.

Genetic algorithms are specifically concerned with the mechanism by which this diversity is generated: sexual reproduction. Most body cells contain two complete sets of chromosomes from each parent; whereas, the sex cells, being sperm and eggs, contain only one complete set, which is a result of a specialized type of cell division called meiosis. When the egg is fertilized by the sperm, the resultant offspring will, once again, comprise of two complete sets of chromosomes, a mixture of DNA contributed by each parent, and yet genetically distinct from either one. In other words, sexual reproduction relies on meiosis to facilitate genetic mixing while keeping the number of chromosomes the same from generation to generation.

In biology, the genetic differences between parents and children occur because of chromosomal recombination and mutation during meiosis and fertilization. In recombination, a pair of parent chromosomes pair, segregate, and reunite at the same loci, forming new and unique arrangements of alleles along the offspring chromosome, thus ensuring children are slightly different from their parents. During the creation of sex cells, DNA undergoes meiosis during which it is possible for DNA sequences to mutate or incur small errors in replication. Both chromosomal recombination and mutation are mechanisms that imbue genetic diversity throughout a population.

2.1.2 Genetic Algorithm Concepts

Abstracted, these biological ideas are the inspiration for the genetic algorithm. In Holland's scheme, each individual in a population is a candidate solution. Physically, they are chromosomes comprised of strings of inputs much as they would be chains of alleles in biological organisms. The solutions are to the optimization problem as the traits expressed by the alleles are to the problem of survival. In this section, we introduce biologically inspired GA concepts and terminology relevant to this thesis.

Genetic Algorithms (GA)

The GA is a heuristic search method used in optimization problems based on a framework whose main components are selection, crossover, mutation, and elitism operators. These GA operators determine the rules by which the search is conducted.

GA Operators

The concepts of natural selection, recombination, and mutation are the operators that drive the evolution of populations of candidate solutions generation after generation towards convergence. The selection operator mimics the laws of natural selection and determines which parents will produce children for the next generation. The recombination operator is most often referred to as crossover and it determines how parent chromosomes are recombined to form children. The mutation operator applies random changes to children chromosomes, introducing uncertainty and catalyzing evolution. There is always a possibility that children are less fit than their parents, so to prevent this, elitism employed used to preserve the fittest individuals in order to ensure that each successive generation does not decrease in average fitness [18]. These operators are described in fuller detail in Chapter 2.2.

GA Configuration

The GA configuration refers to the settings that define the operation of the algorithm. These include the operators and the parameters that define them. Because the approach defined

in this thesis incorporates an inverse method, the design inputs of the airfoil are also an important element of the configuration.

GA Tuning

Evolutionary computing (EC) practitioners agree that proper GA configuration is critical for performance [19]. However, these settings are not usually known beforehand and are, too often, haphazardly decided or dictated by convention. Therefore, the GA must first be tuned before any meaningful results are drawn from the optimization. The two components of the GA configuration that are investigated in the GA tuning study of this thesis are the crossover operator and number of design variables that define the shape representation of the airfoil.

Individual

An individual is a candidate solution for the search. Conceptually, it is a chromosome composed of a string of genes, which confer traits that provide advantage or disadvantage in the organism's survival. Physically, it is an array of input variables that solve the fitness function. In GA, the terms chromosome, individual, and solution are interchangeable.

Populations and Generations

A population is a set of individuals that constitute a generation. Physically, it is an array of candidate solutions. The same individual can appear more than once in a population. Fitter individuals of a population are encouraged to reproduce, while weaker individuals are discouraged from propagating their genetic traits.

Feasible vs. Infeasible Solutions

The definition of feasibility and infeasibility in the context of this study has been revised. Officially, feasible solutions are those that satisfy the constraints of the optimization function, and infeasible solutions are those that do not. In the case of a GA coupled with an inverse

method, this distinction is trivial as airfoils are generated based those very constraints. Real-world results, however, demonstrate that not every function call results in a successful evaluation of the fitness function. Through trial and error, the author has found that certain individuals will challenge the ability of PROFOIL or XFOIL to yield converged or physically realizable results. Such infeasible individuals give rise to exceptions, which can be thought of as constraint violations. In parallel, it is proposed that the definition of an infeasible solution be modified to refer to an individual whose fitness cannot be successfully evaluated by the fitness function.

Fitness

All individuals are attributed a fitness, which is physically evaluated by the fitness function. The fitness of an individual is the heuristic that informs the GA of the direction in which to search. A low fitness is desirable for minimization problems and a high fitness is desirable for maximization problems. The studies performed in this thesis are only concerned with maximization problems.

Feasible solutions are evaluated using the fitness function. Infeasible solutions are special cases that are evaluated using penalty functions and are attributed an undesirable fitness. Infeasibility is discussed in Chapters 2.1.3 and 3.4.

Fitness Landscape

The fitness landscape is the collective fitness of all individuals in a population. Conceptually, it can be thought of as the topology of solutions whose peaks and valleys represent each fitness. The diversity and the overall fitness of the landscape influence the convergence of the algorithm towards a final solution.

Figure of Merit (FOM)

The figure of merit (FOM) is the measure of the performance of a system. In this thesis, the FOMs used for GA tuning are the maximum fitness (MF) and final generation (FG)

of an optimization run. The maximum fitness is the fitness of the final solution. The final generation is equivalent to the number of generations the optimizer takes to converge on the final solution.

Search Quality

The search quality refers to the confidence that the globally optimal solution has been found based on the knowledge that each population is thoroughly diverse, the design space has been exhaustively searched, and the algorithm has satisfactorily reached convergence.

Diversity

Just as genetic diversity drives biological evolution, diversity in the context of genetic algorithms drives search quality by enabling a large design space to be explored. Diversity is measured by the average distance between individuals in a population. It is promoted by increasing the rate of mutation during reproduction.

Figure 2.1 visualizes two populations on an x-y axis. The population comprised of blue plus-signs has high diversity, while the population comprised of red diamonds has low diversity. The initial population of each optimization run is created automatically by MATLAB and is designed to be as uniformly diverse as possible within the constraints of the design space [20].

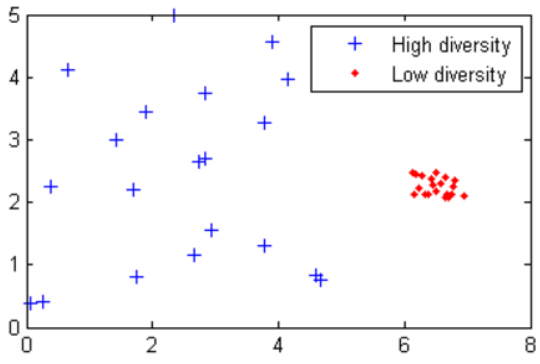

Figure 2.1: Visualization of a high diversity population vs. a low diversity population (taken from Ref. [20]).

Convergence

The GA is said to have converged on a final solution when the relative fitness of each individual no longer increases with each generation. Naturally, the higher performing regions of the search space will be smaller than the weaker areas. As each generation gets stronger, the population will become self-selecting until eventually only the top-performing individuals are consistently represented in the population. In other words, with each passing generation, the fitness landscape will become better as its composition of high-performing individuals increases. This trend represents an incidental decrease in diversity as the GA transitions to concentrating the search in a small area to find the global maximum.

Final Solution

The final solution of an optimization run refers to the best performing individual of the optimization run. The final solution will be the maximum fitness individual of the generation at which stopping criteria are met.

2.1.3 Penalty Function

The purpose of the penalty function is to evaluate a fitness for infeasible individuals. Technically, there are two basic types of penalty functions: exterior penalty functions and interior penalty functions. Exterior penalty functions penalize infeasible solutions and interior penalty functions penalize feasible solutions [21]. This study deals with exterior penalty functions, and for the sake of expediency, will refer to them simply as penalty functions.

The penalty function discourages the GA from searching for solutions in the infeasible solution space, which improves computational efficiency. The reader, may at this point, contest the point of a penalty function in the first place: Why not simply force the optimizer to search solely in the feasible region? There are two reasons why this is not done: practicality and due diligence.

Realistically, the airfoil design space is non-linear, so it may be difficult to elucidate the terms of its feasibility as inputs to the GA. Through experience, however, it is possible to

gain an intuition for what works and what does not. The Applied Aerodynamics Laboratory at UIUC, for example, is well-versed in the nuances of PROFOIL. But as with any form of human intuition, that understanding may be incomplete. And even if one were to be able to prescribe these constraints to the optimizer, the inverse methods and flow solvers used to analyze the inputs are unreliable and may not converge. In short, searching within the feasible region does not necessarily guarantee feasibility.

Even if feasible solutions were assured, it would still be folly to follow through. One may be led to believe that because infeasible solutions do not contribute to the bottom line of the fitness landscape, it would be advantageous to completely avoid them and preempt having to invest the upfront coding effort of dealing with these exceptions in the first place. The irony is that the best solutions often lie on the boundary of infeasibility [22] much like the fine line between genius and madness. Therefore, such indiscriminate avoidance will preclude a globally optimal solution [23, 24, 25].

Crafting an effective penalty function is an art and can be an academic exercise in it of itself. Too lax of a penalty function and the GA may spend too much time meandering through the infeasible design space. Too overzealous of a penalty may drive the algorithm away from the true optimum [26]. Therefore, the design of the penalty function is a critical tool for influencing search quality and convergence speed, both of which are inextricably related.

The main classes of (exterior) penalty functions are static, dynamic, and adaptive. The static penalty is straightforward in that it applies a constant penalization to infeasible individuals. A commonly implemented variation to the static function leverages a cost-to-repair metric, which penalizes according to the hypothetical cost of returning it a state such that it becomes feasible. In other words, it is a calculation of how severely the individual violates the constraint. The formulaic simplicity of the static penalty, however, is sometimes problematic for achieving a feasible final solution because the constant penalty allows for exploration of the infeasible design region through all stages of the search. The dynamic penalty remedies this conflict by introducing a scaling variable that increases the penalty with the progress of the search. In this way, highly infeasible solutions are allowed early on, but are discouraged later in order to confidently move the final solution into the feasible design space. While the

cost-to-repair and the progress of search are important considerations, static and dynamic penalties ignore the success of the search itself. Adaptive penalty functions actively guide the GA away from unattractive regions and towards attractive regions of the design space based on the success of previous generations [21].

There are obviously a myriad of options to incorporate in the design of a penalty function; however, their investigation is outside the scope of this study. In this thesis, the static penalty has been found to consistently converge on feasible final solutions. Chapter 3.4 describes the design of the penalty function in more detail.

2.2 MATLAB GA Framework

The algorithm of the MATLAB GA contains three basic steps: (1) the creation of a random initial population, (2) the evolution of each generation in an optimization loop, and (3) the termination of the optimization loop according to stopping criteria.

1. The MATLAB GA uses a creation function to generate a random initial population that is uniformly diverse.

2. While stopping criteria have not been met, the algorithm creates a sequence of new populations where the individuals of the current generation are used to create children of the next population according to the rules prescribed by selection, crossover, mutation, and elitism operators. The steps for creating the new population are outlined below:

   (a) Obtain the fitness of each individual in the current population using the fitness function and the penalty function.

   (b) Select the parents based on the fitness using the selection operator. Parents will yield elite, crossover, and mutation children using elitism, crossover, and mutation operators. These operators will be elaborated in this chapter.

   (c) Preserve a fraction of the parent population to be passed to the next generation as elite children using the elitism operator.

(d) Recombine a fraction of the parent population to produce crossover children using the crossover operator.

(e) Mutate the remaining parents population to produce mutant children using the mutation operator.

(f) Replace the current population with the elite, crossover, and mutant children to form the next generation.

3. Stop the algorithm when one of the stopping criteria have met and return the fittest individual of the last generation. This individual is the final solution, and its fitness is the maximum fitness.

2.2.1 Calling the GA

The GA is actually one of many optimization tools that can be called from MATLAB's command-line. In fact, there is even a GUI that can be called either by typing `optimtool` into the console and following the onscreen wizard for those less programming inclined. However, the author encourages manipulation of the GA from the command-line for maximum transparency and control. This section describes how to call and configure the MATLAB GA using the command-line. An example script can be found in Chapter 3.3 as Listing 3.1. The MATLAB GA is called using the following command:

Listing 2.1: Calling (single-objective) MATLAB GA.

```
ga(@fitnessfun, nvars, options);
```

Here, `@fitnessfun` is the handle that refers to the fitness function MATLAB script, `nvars` is the number of input variables, and `options` is a set of comma-separated parameters that set the operators and stopping criteria of the GA. The MATLAB Optimization Toolbox provides a myriad of different options to choose from when configuring the genetic algorithm, which are described in this chapter.

2.2.2 Reproduction Options

These options consist of the elite count and the crossover fraction. The elite count, n_{elite}, specifies the number of parent individuals that are to be passed directly to the next generation as elite children. The elite count may be a positive integer no less than 0 and no more than the size of the population n_{pop}, and is often calculated as a percentage of the population r_{elite}. The default is 5% of the population.

$$n_{elite} = ceil(r_{elite} * n_{pop}) \tag{2.1}$$

The elite count option can be set using the syntax:

Listing 2.2: Setting the elite count option.

```
options = gaoptimset('EliteCount',ceil(r_elite * n_pop))
```

The crossover fraction, r_{cross}, controls the proportion of the next generation, other than elite children, to be produced by the crossover operator. The default is 80% of the population.

$$n_{cross} = r_{cross} * n_{pop} \tag{2.2}$$

The crossover fraction option can be set using the syntax:

Listing 2.3: Setting the crossover fraction option.

```
options = gaoptimset('CrossoverFraction',r_cross)
```

2.2.3 Adaptive Feasible Mutation Operator

The remaining n_{mut} children are created using the mutation operator.

$$n_{mut} = n_{pop} - n_{elite} - n_{cross} \tag{2.3}$$

This thesis uses the adaptive feasible mutation, which is a scheme that guides the direction of search based on the success of the last generation. The operator chooses a direction and

step length that satisfies bounds and linear constraints. The adaptive feasible mutation option can be set using the syntax:

Listing 2.4: Setting the adaptive feasible mutation option.

```
options = gaoptimset('MutationFcn',{@mutationadaptfeasible})
```

2.2.4 Tournament Selection Operator

The rules for creating crossover children are controlled by the selection and crossover operators. The selection scheme used in this thesis is tournament selection. Tournament selection chooses parents from a population of individuals. Successive tournaments are held comprising of a user-defined number of individuals n_{tour}, where the individual with the best fitness is selected. Pairwise groupings of these selected individuals are passed to the crossover operator to produce children for the next generation. Tournaments are repeated as many times as necessary to create enough children as dictated by crossover fraction. A tournament size of 1 ensures purely random selection for all crossover parents. A tournament size of the entire population ensures that the individual with the best fitness is always chosen. All optimization runs performed in this thesis used the default of $n_{tour} = 2$. Tournament selection and tournament size can be set using the syntax:

Listing 2.5: Setting the tournament selection option.

```
options = gaoptimset('SelectionFcn',{@selectiontournament,n_tour})
```

2.2.5 Crossover Operators

Depending on the encoding scheme of the individual, the MATLAB framework contains two types of crossover operators: aggregate-based crossover operators (ABCO) and discrete crossover operators (DCO). ABCOs are used in real-coded genetic algorithms and produce children through a linear combination of two parent arrays [27]. DCOs are used in binary-coded genetic algorithms and produce children by splitting, swapping, and concatenating entries in parent arrays. MATLAB allows the user to select from three ABCOs: arithmetic,

heuristic, and intermediate crossover; and from three DCOs, scattered, single point and two point crossover. In this thesis, all six crossover operators are assessed for their performance during GA tuning, but only one is selected for the airfoil optimization studies.

In the following subsections, the variables P_1, P_2, and C refer to the arrays that describe the parent and child individuals. Variables r_{rand}, r_{heur}, and r_{inter}, refer to a uniformly distributed random number between 0 and 1, and user-defined ratios for the heuristic and crossover operators, respectively.

Arithmetic Crossover Operator

Arithmetic crossover creates children that are the weighted arithmetic mean of the two parents. The weighting is randomly generated by the GA.

$$C = r_{rand} * P_1 + (1 - r_{rand}) * P_2 \tag{2.4}$$

Arithmetic crossover is set using the syntax:

Listing 2.6: Setting the arithmetic crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossoverarithmetic});
```

Heuristic Crossover Operator

Heuristic crossover creates children on the line that intersects both parents, a small distance away from the better parent and in the direction away from the worser parent. This distance can be adjusted using the ratio r_{heur}. The default value is 1.2. Assuming that Parent 1 is larger than Parent 2,

$$C = P_2 + r_{heur} * (P_1 - P_2) \tag{2.5}$$

Heuristic crossover and heuristic ratio are set using the syntax:

Listing 2.7: Setting the heuristic crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossoverheuristic,r_heur});
```

Intermediate Crossover Operator

Intermediate crossover creates children as a weighted average of the parents. The weights can be specified using ratio r_{inter}, which can be a scalar or a row vector that is as long as the total number of input variables. The default is a row array of 1's.

$$C = P_1 + r_{rand} * r_{inter} * (P_2 - P_1) \tag{2.6}$$

If ratio r_{inter} is a scalar, all the children will lie on the line that intersects both parents. If it is a vector for which all elements are in the range [0,1], the children will be created within a hypercube defined by the parents on opposite corners. Intermediate crossover and intermediate ratio are set using the syntax:

Listing 2.8: Setting the intermediate crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossoverintermediate, r_inter
   });
```

Scattered Crossover Operator

Scattered crossover creates a random binary array that represents the loci at which parent chromosomes will be swapped to produce children. A value of 1 indicates genes from Parent 1 and 0 indicates genes from Parent 2. For example, if the binary array is `[11001000]` and the parents are represented as

```
P1 = [a b c d e f g h]
P2 = [1 2 3 4 5 6 7 8]
```

the operator returns child

```
C = [a b 3 4 e 6 7 8]
```

Scattered crossover is set using the syntax:

Listing 2.9: Setting the scattered crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossoverscattered});
```

Single Point Crossover Operator

Single point crossover produces children by swapping the genes of parents at crossover point n that is randomly chosen. The child is produced by concatenating the entries of Parent 1 up to and including n and the entries of Parent 2 after n.

For example, if the parents are represented as

```
P1 = [a b c d e f g h]
P2 = [1 2 3 4 5 6 7 8]
```

and the crossover point is $n = 3$, the operator returns child

```
C = [a b c 4 5 6 7 8]
```

Single point crossover is set using the syntax:

Listing 2.10: Setting the single point crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossoversinglepoint});
```

Two Point Crossover Operator

Two point crossover produces children using the same concept as single point crossover, but with two randomly chosen loci. If we call these crossover positions m and n, the operator selects array entries from positions 1 through m from the Parent 1, $m + 1$ through n from Parent 2, and entries greater than n from Parent 1. For example, if the crossover points are $m = 3$ and $n = 6$, and the parents are represented as

```
P1 = [a b c d e f g h]
P2 = [1 2 3 4 5 6 7 8]
```

then the operator returns child

```
C = [a b c 4 5 6 g h]
```

Two point crossover is set using the syntax:

Listing 2.11: Setting the two point crossover option.

```
options = gaoptimset('CrossoverFcn',{@crossovertwopoint});
```

Table 2.1: Table of Stopping Criteria Values

Parameter Name	Variable	Parameter Value
Max Generation Limit	Generations	200
Stall Generation Limit	StallGenLimit	50
Function Tolerance	TolFun	1e-6

2.2.6 Stopping Criteria

The stopping criteria define the conditions for determining convergence. Two criteria are used: a maximum generation limit and a stall generation limit. The maximum generation criteria is a limit set to the maximum number of iterations that the optimizer may run for. The stall generation limit stops the optimizer if the change in best fitness achieved is less than or equal to a parameter called function tolerance. The MATLAB Optimization Toolbox refers to these parameters as `Generations`, `StallGenLimit`, and `TolFun` [18]. Their values are described in Table 2.1:

Listing 2.12: Setting stopping criteria.

```
options = gaoptimset('Generations',200,'StallGenLimit',1e-6,'TolFun'
   ,50);
```

2.3 PROFOIL Usage Concepts

PROFOIL is a multipoint inverse method based on the theory of Eppler [28]. The theory of Eppler [29] uses conformal mapping to divide the airfoil into segments, mapped about a circle, for which the velocity is constant at a specified angle of attack relative to zero lift [30]. As opposed to the Jowkouski transformation in which various airfoil profiles can be achieved by scaling and translating a circle about the complex plane [31], this approach fixes the circle and directly controls the transformation in order to obtain different airfoil geometries [4]. This transform is determined by a piecewise curve of the function $\alpha^*(\phi)$, which controls the velocity distribution. Figure 2.2 shows the airfoil design process using PROFOIL.

PROFOIL extends Eppler's approach in capability. Whereas the theory of Eppler only considers airfoils with cusped trailing edges, PROFOIL also allows for the design of finite

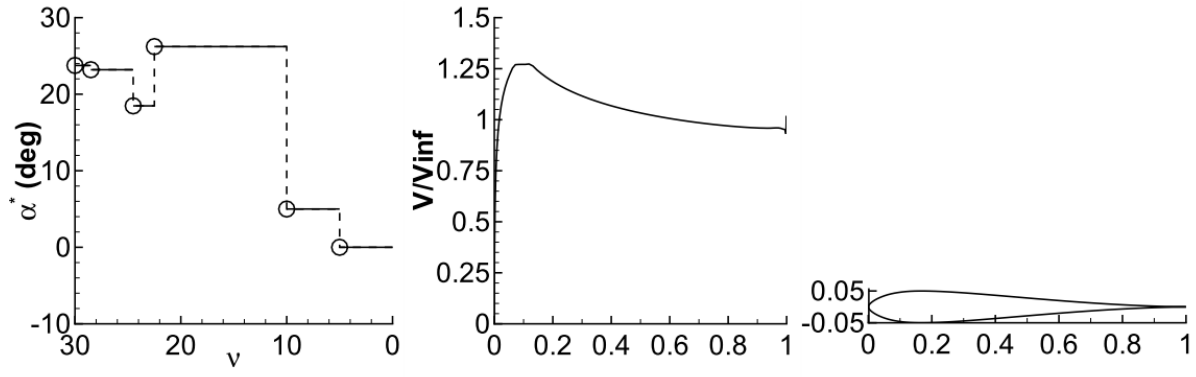

Figure 2.2: The inverse design process using PROFOIL (taken from Ref. [5]).

trailing-edge angle airfoils through an additional term in the conformal transformation that permits control of the recovery region of the velocity distribution. In addition, PROFOIL employs multidimensional Newton iteration to achieve non-constant velocity distributions for each segment, which allows the designer to control the boundary layer development of the airfoil, as well as its geometric design parameters, such as maximum thickness, camber, or pitching moment [4].

2.3.1 Definitions

This section describes the variables used to define the velocity distribution from which the airfoil geometry is derived. Figure 2.3 shows a 4 segment airfoil mapped from a circle. In this example, the airfoil is divided into four segments, the minimum number needed for this method. These segments are defined by dimensionless arc limits, ϕ, that circumscribe the circle in counter-clockwise order. Arc limits ϕ_1, ϕ_2, ϕ_3, and ϕ_4, are mapped to the points s_1, s_2, s_3, and s_4 on the airfoil. The arc limits are measured in 6 degree sectors; therefore, the circle arc limit begins at $\phi_0 = 0$ and ends at $\phi_4 = 60$. Each arc limit ϕ is measured from the point ϕ_0.

Each segment is associated with a design angle of attack α^* that is referenced to the zero-lift line. In this way, specifying α^* values allows the user to prescribe a constant velocity distribution of a segment for a given lift coefficient, assuming $C_l \approx 2\pi\alpha$ [32]. For example, if

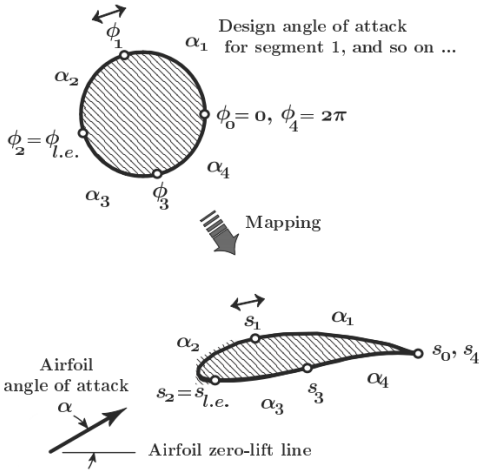

Figure 2.3: Mapping of 4 segment airfoil from a circle (taken from Ref. [4]).

the design angle of attack for the second segment, α_2^*, is 10 deg, when the airfoil is operated at 10 deg, the velocity distribution along the second segment will be constant. If α_3^* is 5 deg, the third segment would have a constant velocity at 5 deg.

2.3.2 Usage

PROFOIL is a keyword-based program that can be controlled by a script containing commands and design requirements. This script is the PROFOIL configuration file, and it must be named `profoil.in` in order for PROFOIL properly load it. Listing 2.13 is an example configuration file for a 4 segment airfoil that outlines the keyword conventions for the required inputs, and their corresponding parameters for correct PROFOIL usage. Highlighted in green are the input variables used in this method. The actual optimization studies used configuration files for 6, 7, and 8 segment airfoils, which would have more α^* input variables.

Listing 2.13: Profoil configuration file template and keyword conventions.

```
1  COORD    <n_panels>
2  FOIL     <φ1>       <α1*>
3  FOIL     <φ2>       <α2*>
4  FOIL     <φ3>       <α3*>
5  FOIL     <φ_ISEG>   <α*_ISEG>
6  PHIS     <φs>       <φ̄s>
7  REC      <K>        <K̄>
8  VLEV     <JSEG>     <v*>
9  ILE      <i_LE>
10 TOLSPEC 0.00001
11 ITERMAX 30
12 NEWT1G0 100  <Ks>     1    4
13 IDES
14 NEWT1G0 101  <Cm0>    4    1
15 IDES
16 NEWT1G0 102  <t/cmax>  6    300 .5
17 IDES
18 FINISH  100
19 ALFASP  2
20 4
21 10
22 VELDIST 50 60
23 *
```

Required Design Lines

The required design command lines include the `COORD`, `FOIL`, `PHI`, `REC`, and `VLEV` lines.

- **Line 1:** The `COORD-line` takes parameter n_{panels}, which indicates how many airfoil coordinates to generate. For the reasons outlined in Chapter 3.3.3, $n_{panel} = 240$ is recommended.

- **Lines 2–5:** The `FOIL-lines` take a pair of parameters: ϕ and α^*. The number of `FOIL-lines` determines how many segments define the airfoil, with four being the minimum. In the case of this study, airfoils are defined by six to eight segments. The arc limit of the last `FOIL` line, ϕ_{ISEG}, designates the number of sectors to divide the circle. For example, if ϕ_{ISEG} is 60, the circle would be discretized into 60 sectors of 6 degree arc lengths. For such an airfoil, starting from the trailing edge and on the upper side, $\phi = 0$ would correspond to the trailing edge, $\phi \approx 15$ would be near the middle of the upper surface, $\phi \approx 30$ would be near the leading edge, $\phi \approx 45$ would

be near the middle of the lower surface, and $\phi = 60$ would, again, correspond to the trailing edge.

- **Line 6:** The `PHIS-line` takes the parameters ϕ_s and $\overline{\phi}_s$, the trailing-edge upper surface closure arc limit and the trailing-edge lower surface closure arc limit, respectively. These parameters define the area at which the airfoil closes at the trailing edge. PROFOIL requires that the arc limits fall within the first and last segments of the airfoil, that is, ϕ_s must be greater than the first arc limit, ϕ_1, and $\overline{\phi}_s$ must be larger than the penultimate arc limit, ϕ_{ISEG-1}.

- **Line 7:** The `REC-line` takes K and $\overline{K}$, the upper and lower surface recovery parameters. These control the initial slope of the recovery region. Larger values indicate steeper slopes. Acceptable values for K and $\overline{K}$ are 0.02000.

- **Line 8:** The `VLEV-line` specifies the design velocity level v^* for the specified segment JSEG. A recommended value for v^* is 1.52728, which is used as a starting point.

- **Line 9:** The `ILE-line` specifies the index of the leading edge arc limit i_{LE}. For the 4 segment airfoil, $i_{LE} = 2$.

Newton Iteration Lines

The `NEWT1G0` lines prepare PROFOIL for multidimensional Newton iteration and the three digit codes that follow each `NEWT1G0` command specify which design parameter to iterate on. After each `NEWT1G0-line`, the `IDES-line` is used to solve the inverse equations.

- **Line 12:** `NEWT1G0 100` refers to the trailing-edge thickness parameter K_s. Without specifying this line, PROFOIL will likely produce crossed airfoils given an arbitrary set of α^*-ϕ input values. A small value of K_s yields a thickness of effectively 0, while a large value of 2 produces a thick trailing edge. K_s is recommended to be set to 0.350.

- **Line 14:** `NEWT1G0 101` specifies the pitching moment at zero-lift C_{m_0}.

- **Line 16:** `NEWT1G0 102` specifies the maximum thickness to chord ratio t/c_{max}.

- **Lines 10 and 11:** The `TOLSPEC` and `ITERMAX` lines specify the convergence tolerance and the maximum number of iterations. These lines are necessary for PROFOIL to achieve automatic convergence.

Output Data

The following lines control how PROFOIL outputs airfoil geometry and velocity distribution data.

- **Line 18:** The `FINISH 100-line` prompts PROFOIL to write the airfoil coordinates to a file called `profoil.xy`.
- **Lines 19–22:** The last lines of the configuration file control how PROFOIL outputs velocity distribution data. `ALFASP` takes an integer parameter indicating how many velocity distributions for a specific angle of attack to generate. The selected angles are specified on the subsequent lines. `VELDIST 50 60` prompts PROFOIL to write the velocity distribution data to file `FOR050.dat` and `profoil.vel`, respectively.
- **Line 23:** The asterisk denotes the end of the file and is important for loading the file correctly by PROFOIL.

2.3.3 Specification of Arc Limits

As discussed in Chapter 2.3.1, the airfoil segments are defined by arc limits, which act as control points on the geometry. Their arc limits can be found in Table 2.2. The italicized numbers indicate the arc limits that map to the lower surface of the airfoil.

The arc limits for the 6 segment airfoils designed in this thesis are referenced from the MAN085 airfoil of Listing 3.2. The PROFOIL User Guide, unfortunately, does not provide

Table 2.2: Table of Arc Limits for 6, 7, and 8 Segment Airfoils

Segments	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7	ϕ_8
6	15.5	19.5	25.5	32.2	*45.5*	*60.0*		
7	15.5	18.5	21.5	25.5	32.2	*45.5*	*60.0*	
8	15.5	18.5	21.5	24.5	27.5	32.2	*45.5*	*60.0*

7 and 8 segment airfoil examples, so these arc limits were invented. For the 7 and 8 segment arc limits, the first ϕ (15.5), that of the leading edge (32.2), and those of the lower surface (45.5 and 60.0) were appropriated from the 6 segment airfoil. Because the lower surface arc limits are identical, the additional degrees of freedom were awarded to the upper surface, the shape of which is much more influential in the aerodynamic performance of the airfoil than that of the lower surface.

2.3.4 Data Structure of an Individual

In addition to the design angles that correspond to an airfoil's arc limits, each individual is also described by its design pitching moment and design thickness to chord ratio. 6 segment airfoils are thus described by eight total design variables, 7 segment airfoils described by nine, and 8 segment airfoils described by ten. This information is encoded as an array of parameters as shown in Listing 2.14. The GA recognizes this array as a chromosome.

Listing 2.14: Data structure of individuals.

```
6 Segment Airfoil = [<C_m> <t/c> <α1*> <α2*> <α3*> <α4*> <α5*> <α6*>]
7 Segment Airfoil = [<C_m> <t/c> <α1*> <α2*> <α3*> <α4*> <α5*> <α6*> <α7*>]
8 Segment Airfoil = [<C_m> <t/c> <α1*> <α2*> <α3*> <α4*> <α5*> <α6*> <α7*> <α8*>]
```

2.3.5 Data Structure of a Population

The MATLAB GA reads in populations as a 2-D array of n_{pop} rows and $nvars$ columns. Assuming the individual arrays are horizontal vectors, the population array is a vertical concatenation of each individual.

CHAPTER 3

METHODOLOGY

This chapter describes the computational equipment, the input parameters of the three studies performed in this thesis, and the architecture of the GA method in these optimization trials. Flowcharts and code snippets are provided for explaining how data is stored and processed through the course of an optimization run. Also documented are defensive programming tactics and a custom penalty function that were developed in order overcome unforeseen challenges in the execution of the method.

3.1 Computational Equipment

Three PC workstations were used in this thesis. Each ran Windows 7 Professional on a quad-core Q6600 2.4 GHz processor with 8 GB of DDR2 RAM. Mathworks MATLAB 2014a Revision A, PROFOIL Version 3.2.0 , and XFOIL 6.99 were used for this investigation.

3.2 Input Parameters

Three studies were performed in this thesis. In the first study, GA tuning was performed by systematically varying the crossover operator and the number of segments that represent the airfoil. The best performing configuration was selected based on the figures of merit of maximum fitness and final generation. Using the tuned configuration, the second study optimized an airfoil for $(C_l/C_d)_{max}$. Using the same configuration, the third study optimized a series of airfoils for $C_{l_{max}}$ at various design pitching moments. These studies are referred to as Studies 1 through 3, respectively.

1. GA tuning was accomplished in a series of four cases, labeled Cases 1 through 4.

- In the first case, all six aggregate-based (ABCO) and discrete crossover options (DCO) were tested at Design Condition 1: $Cm = -0.063$ and $t/c = 18\%$ at $Re = 6.88 \times 10^6$ using a 6 segment airfoil representation scheme. These conditions reflect flow conditions for a wind turbine at the $r/R = 0.75$ blade section location [6].
- Using the same design goals, Case 2 explored the effect of using 7 and 8 segments airfoils for an ABCO option (heuristic crossover at heuristic ratio = 1.2) and for a DCO option (single point crossover).
- The third case, intended to be a more thorough version of Case 2 at a different design condition, analyzed the effect of varying the number of airfoil segments from 6 through 8 for all crossover options at Design Condition 2: $Cm = -0.030$ and $t/c = 18\%$ at $Re = 2.00 \times 10^6$. These requirements are identical to those used in designing the Liebeck L1003 airfoil [33].
- And lastly, the fourth case specifically investigated the effect of varying the heuristic ratio from 1.2 through 2.0, again, at the first design condition.

2. Analysis of the performance of each GA configuration concluded in the use of 6 segment airfoil representation and heuristic crossover at heuristic ratio = 1.8 for Studies 2 and 3. In Study 2, an airfoil was optimized for $(C_l/C_d)_{max}$ for the design goal of $Cm = -0.060$ and $t/c = 18\%$ at $Re = 6.00 \times 10^6$, a typical operating condition of general aviation aircraft.

3. In Study 3, nine 18% thick airfoils were maximized for $C_{l_{max}}$ for a series of increasingly negative design pitching moments from zero through -0.200 with 0.025 intervals at $Re = 6.00 \times 10^6$.

In total, 42 optimization runs were completed in the course of these three studies. Each of these optimization runs is referred to as a trial. Case 1 through 4 comprise of a total of 32 trials, spanning Trials 1 through 32. Study 2 was accomplished in one optimization run, and is referred to as Trial 33. Study 3 was accomplished in 9 optimization runs, spanning Trials 34 through 42. A detailed summary of the input parameters for these three studies can be found in Fig. 3.2.

Table 3.1: Summary of Mutation, Selection, Reproduction, and Stopping Criteria Parameters.

Option	Parameter
Adaptive Feasible Mutation	
Tournament Selection	$n_{tour} = 2$
Elitism Count	$r_{elite} = 0.05$
Crossover Fraction	$r_{cross} = 0.80$
Maximum Generation Limit	200
Stall Generation Limit	50
Function Tolerance	$1\times\ 10^{-6}$

All studies are performed with adaptive feasible mutation, tournament selection of tournament size $n_{tour} = 2$, identical reproduction options, and equal stopping criteria, summarized in the table below. The initial population of each optimization run is created by MATLAB's creation function. Furthermore, all airfoils are designed to have a thickness to chord ratio of 18%. The population size for each GA configuration was calculated as 15 times the number of airfoil segments. The MATLAB default is $15 \times nvars$ [20], but as the design variables C_m and t/c were fixed in these studies, it was decided to withhold including them in this calculation to minimize the population for speedier convergences. To minimize memory-related I/O complications, each workstation was restarted at the conclusion of each optimization run.

3.3 GA Architecture

The architecture of the GA can be understood as a nested loop: an inner loop within the body of an outer loop. This interaction is depicted in Fig. 3.1, which is a flowchart of GA operation with off-page connectors A and B that call out a portion that will be continued in Fig. 3.2. In this flowchart, the outer loop calls the inner loop, which executes to completion, and this repeats until the outer loop itself is finished. Here, the inner loop is the evaluation of the fitness landscape of the current generation by calling the fitness function `fitness.m` for every individual of the population.

The outer loop is the GA that uses selection, recombination, mutation, and elitism op-

Table 3.2: Summary of Input Parameters for Each Trial.

Trial	Segments	Crossover Operator	C_m	t/c	Re
Study 1 Case 1					
1	6	Arithmetic	-0.063	18%	6.88×10^6
2	6	Heuristic (1.2)	-0.063	18%	6.88×10^6
3	6	Intermediate [1]	-0.063	18%	6.88×10^6
4	6	Scattered	-0.063	18%	6.88×10^6
5	6	Single Point	-0.063	18%	6.88×10^6
6	6	Two Point	-0.063	18%	6.88×10^6
Study 1 Case 2					
7–8	7,8	Heuristic (1.2)	-0.063	18%	6.88×10^6
9–10	7,8	Single Point	-0.063	18%	6.88×10^6
Study 1 Case 3					
11–28	6,7,8	Arithmetic	-0.030	18%	2.00×10^6
	6,7,8	Heuristic (1.2)	-0.030	18%	2.00×10^6
	6,7,8	Intermediate [1]	-0.030	18%	2.00×10^6
	6,7,8	Scattered	-0.030	18%	2.00×10^6
	6,7,8	Single Point	-0.030	18%	2.00×10^6
	6,7,8	Two Point	-0.030	18%	2.00×10^6
Study 1 Case 4					
29	6	Heuristic (1.4)	-0.063	18%	6.88×10^6
30	6	Heuristic (1.6)	-0.063	18%	6.88×10^6
31	6	Heuristic (1.8)	-0.063	18%	6.88×10^6
32	6	Heuristic (2.0)	-0.063	18%	6.88×10^6
Study 2					
33	6	Heuristic (1.8)	-0.060	18%	6.00×10^6
Study 3					
34	6	Heuristic (1.8)	0.000	18%	6.00×10^6
35	6	Heuristic (1.8)	-0.025	18%	6.00×10^6
36	6	Heuristic (1.8)	-0.050	18%	6.00×10^6
37	6	Heuristic (1.8)	-0.075	18%	6.00×10^6
38	6	Heuristic (1.8)	-0.100	18%	6.00×10^6
39	6	Heuristic (1.8)	-0.125	18%	6.00×10^6
40	6	Heuristic (1.8)	-0.150	18%	6.00×10^6
41	6	Heuristic (1.8)	-0.175	18%	6.00×10^6
42	6	Heuristic (1.8)	-0.200	18%	6.00×10^6

erators to perform mating decisions on the current generation to create the individuals for the next generation. In this report, the term GA has been used to loosely to refer to the algorithm in a general fashion (as in the first sentence of this paragraph). Within the context of discussing architecture, however, the author will attempt to retain semantic precision by maintaining a distinction between the GA outer loop and the fitness function inner loop.

As shown in Listing 3.1, the outer loop GA is called using a script called `main.m`, which contains the parameters that dictate the behavior of the GA. These parameters include the GA options, design inputs, and the stopping criteria. While stopping criteria have not been met, the GA creates population after population of individuals. Each iteration of the outer loop is a generation of evolution.

The inner loop is the fitness function `fitness.m`, which is physically a toolchain that controls the sequential operation of PROFOIL and XFOIL. Each iteration of the inner loop yields a fitness evaluation that contributes to a growing fitness landscape. Within each generation, there will be as many fitness function calls as there are individuals of the population. In each function call, the toolchain attempts to realize these inputs into an actual airfoil geometry, which is then analyzed using a flow solver, yielding aerodynamic performance data. The fitness of an individual is reduced from this data and is passed back to the GA at the end of each iteration of the inner loop. A flowchart of the fitness function can be found in Fig. 3.2.

Both the GA and the fitness function are basic input-output routines that pass data back and forth at different cadences. This back and forth between the inner and outer loop is repeated until stopping criteria are met, whereupon the fittest individual of the last generation is the final converged solution of the optimization run.

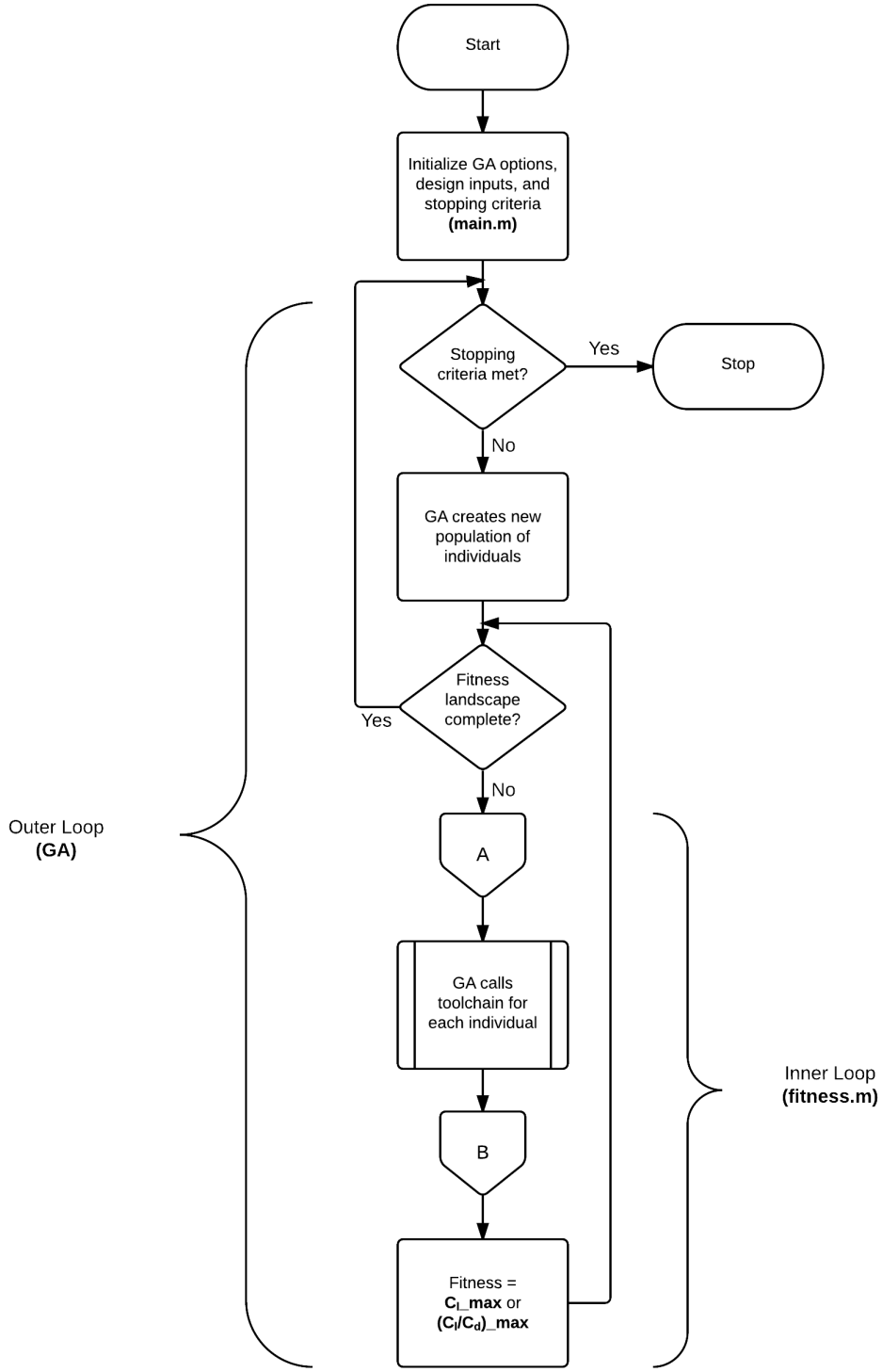

Figure 3.1: Flowchart of GA operation.

.

Listing 3.1: MATLAB script main.m.

```
%% INITIALIZE PARAMETERS

% Set function handle to the fitness function
FitnessFunction = @fitness;

% GA Configuration Parameters
AP      = 6;                                    % Number of alpha*-phi pairs
nvars   = AP + 2;                                % Total number of variables
n_pop   = 15 * AP;                      % Number of individuals in population

% Option Parameters
intermediate_ratio  = 1*ones(1,nvars);                  % Intermediate Ratio
heuristic_ratio     = 1.2;                                 % Heuristic Ratio
tournament_size     = 2;                                   % Tournament Size
crossover_fraction  = 0.8;                              % Crossover Fraction
elitism_fraction    = 0.05;                               % Elitism Fraction
gen_max             = 200;                             % Maximum Generations
gen_stall           = 50;                           % Maximum Stall Generations
tolerance           = 1e-6;                                % Function Tolerance

% Constraint Parameters
Cm_lb   = -0.03;     Cm_ub   = -0.03;                % Pitching moment bounds
tc_lb   = 0.18;      tc_ub   = 0.18;        % Thickness to chord ratio bounds
alfa_lb = -10;       alfa_ub = 15;                            % Alpha* bounds

%% SET GA OPTIONS
A   = [];                                   % No linear inequality constraints
b   = [];                                   % No linear inequality constraints
Aeq = [];                                     % No linear equality constraints
beq = [];                                     % No linear equality constraints
lb  = horzcat(Cm_lb,tc_lb,str2num(repmat(sprintf(' %d',alfa_lb),1,AP)));
ub  = horzcat(Cm_ub,tc_ub,str2num(repmat(sprintf(' %d',alfa_ub),1,AP)));

options = gaoptimset(...
'UseParallel',false,...                             % No Parallel Processing
'PopulationType','doubleVector',...                     % Specifies Encoding
'PopulationSize',n_pop,...                                % Population Size
'CrossoverFraction',crossover_fraction,...              % Crossover Fraction
'EliteCount',ceil(elitism_fraction * n_pop),...              % Elitism Count
'CrossoverFcn',{@crossoverheuristic,heuristic_ratio},...         % Crossover
'MutationFcn',{@mutationadaptfeasible},...                        % Mutation
'SelectionFcn',{@selectiontournament,tournament_size},...;       % Selection
'Generations',gen_max,...                           % Maximum Generation Limit
'StallGenLimit',gen_stall,...                         % Stall Generation Limit
'TolFun',tolerance);                                     % Function Tolerance

%% CALL GA
ga(FitnessFunction,nvars,A,b,Aeq,beq,lb,ub,[],options);


```

3.3.1 Toolchain

The toolchain itself is physically the fitness function `fitness.m` that `main.m` calls for as many feasible individuals that are evaluated in the entire optimization run. This toolchain consists of the sequential execution of PROFOIL followed by XFOIL. PROFOIL loads the design requirements from the `profoil.in` configuration file. After executing the multipoint inverse solver, PROFOIL outputs an airfoil geometry in the form of x-y coordinates into the `profoil.xy` airfoil coordinate file. XFOIL loads this coordinate file, performs a polar sweep from -10 to 15 deg at 0.25 degree increments, and then outputs a polar file `polar.dat`. The fitness of the individual is then calculated from this polar file. A flowchart of toolchain can be found in Fig. 3.2. The off-page connectors A and B indicate how this flowchart is integrated into the flowchart of GA operation in Fig. 3.1. Off-page connectors C and D refer to flowcharts of the penalty functions that are described in Figs. 3.4 and 3.5.

3.3.2 PROFOIL Configuration File

The configuration files of the airfoils designed in this thesis are based off of the example configuration files used in the PROFOIL User's Guide [4]. Parameters that are passed as inputs from MATLAB are highlighted in green. Listing 3.2 contains the contents of an example PROFOIL configuration file for a 6 segment airfoil.

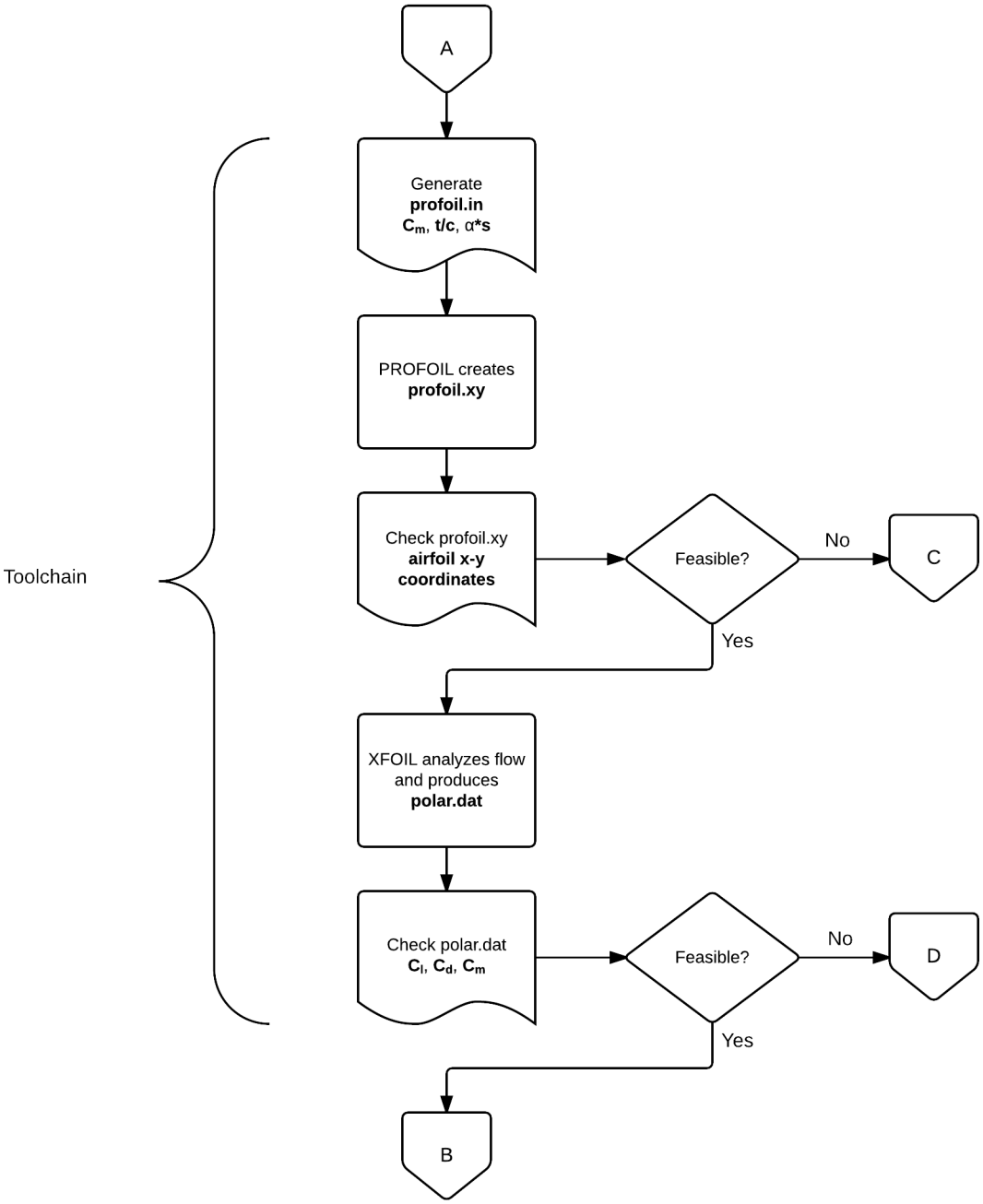

Figure 3.2: Flowchart of toolchain operation.

Listing 3.2: Example 6 segment PROFOIL configuration file MAN085.in [4].

```
# 085

AIRFOIL MAN085
COORD 60
FOIL    15.50000    10.00000    1
FOIL    19.50000     6.00000    2
FOIL    25.50000     8.00000    3
FOIL    32.20000    10.00000    4
FOIL    45.50000     4.00000    5
FOIL    60.00000     4.00000    6
PHIS     3.50000    56.50000
REC       .02000       .02000
VLEV     1           1.52728
ILE      4

TOLSPEC 0.00001
ITERMAX 15

# Specify Ks and iterate on LE arc limit
NEWT1GO  100 0.350 1 4
IDES

# Specify moment and iterate on velocity level
NEWT1GO  101 -0.15  4 1
IDES

# Specify t/c and iterate alpha* +/-
NEWT1GO  102 .1500 6 300 .5
IDES

FINISH 100
ALFASP 7
2
4
6
8
10
12
14

VELDIST 60

DUMP
AIRFOIL MAN085
*


```

In this example, the number of coordinates is set to 60 and $i_{ILE} = 4$. The arc limits for this example are set to 15.5, 19.5, 25.5, 32.2, 45.5, and 60, and their corresponding α^*

angles are 10, 6, 8, 10, 4, and 4 deg. The geometric design inputs are set to $K_s = 0.35$, $C_m = -0.15$, and $t/c = 0.15$. The aforementioned recommended values for the parameters ϕ_s, $\overline{\phi}_s$, K, $\overline{K}$, and v^* are referenced from the `PHIS-line`, `REC-line`, and `VLEV-line` of this example. The velocity distributions for angles of attack 2, 4, 6, 8, 10, 12, and 14 deg relative to the zero-lift line are written into file `profoil.vel`.

3.3.3 Defensive Programming Tactics

In a perfect world, the operating environment is stable, the optimization loop does not break, and the numerical methods contained therein always produce reasonable results. Reality, unfortunately, is rarely so accommodating. The nature of repetitive input/output operations, the quirks of numerical computation, and the general Windows environment necessitate robust measures to ensure that the GA optimization loop is not prematurely interrupted, that the code does not break under uncontrolled circumstances, and that the data generated by the process is reliable and accurate. These measures can be described as defensive programming tactics and they are characterized as: (1) Operations Continuity, (2) Preserving Data Integrity, and (3) Toolchain Exception Handling.

Operations Continuity

The following are measures that were taken to ensure that the optimizer was not interrupted mid-operation.

Disable Windows Update

By default, the Windows operating system enables automatic Windows Updating, which can cause a computer to restart regardless of active processes. It is recommended that the user disable this feature.

Disable sleep power mode

The default power setting of most operating systems allow for a sleep mode after a certain period of inactivity. Through experience, it has been found that a running MATLAB script is not enough to constitute the activity necessary to prevent the onset of sleeping. It is recommended that the user disable this power saving feature.

Disable error reporting dialog box

It is possible for XFOIL or PROFOIL to crash, triggering a pop-up dialog box that interrupts the optimization run. The dialog box will contain this wording [34]:

```
[Window Title]
Program.exe

[Content]
Program.exe has stopped working.
A problem caused the program to stop working correctly.
Please close the program.

[Close the program] [Debug the program]
```

It is recommended that the user disable error reporting by editing the registry by setting the DontShowUI registry value to 1, which can be found by expanding the following path:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\Windows Error Reporting\DontShowUI
```

Disable optimization tool plots

The Optimization Tool contains automated plotting functions for the GA. Should any be selected, the specified plot will appear at the indicated plot interval. Unfortunately, a dialog box will also appear with the following messages:

```
[Close dialog]
YES will stop the solver (if running) and close the figure.
NO will cancel this request.
```

Be sure to de-select all plot functions from the Optimization Tool when running the optimizer. When running from the command line, simply avoid any references to plotting functions as they are disabled by default.

Close all open files after each toolchain call

Because multiple files need to be opened and read through the course of each fitness function call, the physical limit of open files can be exceeded through the course of an optimization run. Should this happen, the following error message will appear in the console and break the optimizer:

```
Error using fprintf
Invalid file identifier.  Use fopen to generate a valid file identifier.
```

It is important to close all opened files at the end of each function. One can achieve this using the `fclose()` function in Listing 3.3.

Listing 3.3: Syntax for fclose().

```
fclose(fileID); % Close specific file
fclose('all');  % Close all open files
```

Deploy XFOIL as a standalone process

The most typical way to execute Windows programs in MATLAB is to use the `dos` function, which executes MS-DOS commands in a DOS shell. For example, the user can call XFOIL and load an airfoil coordinate text file by typing:

```
dos('xfoil.exe < profoil.xy');
```

Unfortunately, the stability of XFOIL can be capricious and when it invariably fails to converge on a result, the program can hang. MATLAB only continues after the command

finishes execution, so should XFOIL hang, the toolchain can be stalled. At this point, terminating XFOIL using the `quit` command is usually not possible. To prevent interruption to the optimization run, the author suggests that XFOIL be deployed as a standalone system process by using the following code [35]:

```
arg = {'cmd', '/c','xfoil.exe < coordinates.txt','>','nul'};
PB = java.lang.ProcessBuilder(arg);
PB.start;
```

The beauty of running XFOIL in this way is that it can be easily killed using the following MATLAB system command:

Listing 3.4: Syntax for killing the XFOIL system task.

```
dos('taskkill /f /im xfoil.exe')
```

In fact, the author advises that the user kill XFOIL after each function call as a hard policy in order to ensure that hanging instances do not bottleneck the resources of the computer.

Preserving Data Integrity

This section provides suggestions for ensuring that the data generated throughout the optimizer is accurate and reliable. The reader may wonder why, for a black-box scheme optimizer, would the data involved in the interstitial steps of its operation warrant so much effort. The reason is because for genetic algorithms, the consequences of inaccurate fitness evaluations are serious. The GA makes decisions on whether to propagate the traits of an individual to the next generation solely based on its fitness. If the reported fitness is lower than the true fitness, the selection operator will discourage the individual's chances of passing its genetic material to subsequent generations. A perfectly feasible portion of the design space could be lost and the global optimum may never be realized. If the reported fitness is higher than the true fitness, the selection operator will encourage the propagation of individuals that have erroneously inflated performance, skewing all future generations. It is recommended that the user impose the following measures to ensure the accuracy of each fitness evaluation.

Panel airfoil geometry to at least 240 coordinates

As mentioned before, XFOIL can underestimate drag, and consequently, inflate the actual $(C_l/C_d)_{max}$ of an airfoil. There are times, however, that XFOIL yields unreasonably high airfoil glide ratios because of excessively low drag estimations that still pass the drag lower bound check. Gardner also observed such individuals during ProfoilGA operation [5]. These aberrant individuals have a tendency to dominate the normal population in fitness. Vigilance must be exercised in excising such individuals and making sure their traits do not propagate. In most situations, paneling the airfoil to at least 240 coordinates will resolve this issue.

XFOIL gives the user the ability to re-panel any airfoil geometry by smoothly repaneling the airfoil to any panel count the user desires. As the number of panels increases, XFOIL trades ease of convergence and computation time with fidelity. For most airfoils, there is little difference in the polar data generated for different panel counts. The author has observed, however, that for certain airfoils, the difference between 240 panels and 120 panels, is an order of magnitude in drag coefficient. Though suspicious, a very high $(C_l/C_d)_{max}$ in it of itself is not grounds for removal. However, when the $(C_l/C_d)_{max}$ decreases by a magnitude simply by repaneling to a finer mesh, it is clear that these results are erroneous.

Unfortunately, the author has not been able to discern any correlation of the physical characteristics of the airfoil to XFOIL's proclivity to produce a falsely optimistic airfoils—this phenomenon can only be attributed to a quirk of numerical computation. In short, there is an inverse proportion between the number of coordinates that describe the airfoil geometry versus XFOIL's tendency to overestimate $(C_l/C_d)_{max}$. Through trial and error, it has been found that paneling airfoils to 240 coordinates ensures reasonable drag estimates, while not imposing unreasonable computation effort.

Start XFOIL polar accumulation at 0 degrees

When performing polar accumulation in XFOIL, it is important to start at a reasonable angle of attack before progressing to more extreme angles of attack. For example, if the goal is to accumulate polar data in the angle of attack range -20 to 20 deg at 0.5 deg intervals, it may tempting to perform this in one sweep from 20 through -20 deg using

the XFOIL command `ASEQ -20 20 0.5`. However, for many feasible airfoils, XFOIL will not be able to converge at the first angle of attack of -20 deg, and the airfoil individual would be prematurely considered infeasible and penalized. It is recommended that polar accumulation be initated at 0 deg and to first sweep up into positive angles of attack before sweeping down. These XFOIL commands can be implemented using the sequential XFOIL commands of `ASEQ 0 20 0.5` and `ASEQ 0 -20 -0.5`.

Delete polar file after each fitness function call

It is recommended that the polar file be deleted after each fitness function call. The polar file should not be left to be overwritten in successive fitness function calls. Inherent to XFOIL's polar accumulation is its ability to repeatedly open, write, and close the polar file. When XFOIL executes successfully, the polar file will close normally. However, should XFOIL encounter an exception and its process be forcibly killed, the polar file will not successfully close and remain open. If the existing polar is not deleted, XFOIL may continue writing to that open polar in subsequent successfully converged fitness function calls. The polar file will accrue the results of all converged successive airfoils, rendering the fitness evaluation for each individual incorrect. Using unique polar filenames is another solution that will easily preempt this polar accrual situation, but at the cost of increased disk storage.

Sample length of polar to determine if XFOIL has hung

The matter of actually determining if XFOIL has actually become a hung process is not obvious. There are two symptomatic ways of determining if XFOIL is no longer responding: (1) the polar has ceased accumulation and (2) there is no activity for a longer than usual period of time. It may be tempting to simply kill the XFOIL process after a fixed amount of time, but this practice is discouraged because it is computationally inefficient and may lead to data integrity problems. For this solution, the best case scenario is that the user overestimates the amount of time that XFOIL should complete its analysis in order for the true fitness of the individual to be consistently evaluated correctly. This practice is computationally costly because of the recurring overhead in the safety margin. The worst case

scenario is that the user underestimates the time that XFOIL takes to converge and accidentally curtails polar accumulation, leading to an inaccurate evaluation of the fitness. The latter outcome is actually quite difficult to avoid because there are always edge case airfoils for which XFOIL takes an inordinate amount of time to reach convergence. Furthermore, XFOIL's speed varies from computer to computer. If the user's intuition about XFOIL convergence time is honed on one system, that intuition will be inaccurate for another system. Therefore, it is recommended that the user kill XFOIL after determining if the polar has actually ceased accumulation rather than after a fixed period of time.

Listing 3.6 contains a MATLAB implementation of how to sample the polar file. It is composed of a while-loop that breaks under the condition that the change in polar length is less than one line for a two second interval. Within the while loop, the length of the polar is sampled twice with a pause in between each sample. When this loop breaks, the XFOIL task is killed using the command previously provided in Listing 3.4.

Avoid race conditions when sampling the polar

A race condition is a situation in which the timing of sequential operations cannot be guaranteed, giving rise to a bug. Of all the temporary files created through the course of a function call, the author has found that only the polar file requires a pause to ensure that there is enough time for it to be created before it is checked for exceptions. The author has found that a Windows 7 Bootcamped Macbook Pro running a 2.4 GHz second generation Core i5 CPU with 4 GB of RAM and a 5400 RPM harddrive takes approximately 0.5 sec to reliably create the polar file. Other systems will vary in performance.

Check for drag lower bound violation

XFOIL provides fairly reasonable results for the coefficients of lift of most airfoils, but it is known that XFOIL tends to underestimate coefficients of drag. It is recommended that the user implement a physical reality check on these results. The Blasius solution for the laminar flow over a flat plate provides a solution for the friction coefficient along a flat plate surface as a function of its Reynolds number in incompressible flow [31].

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho U^2} = \frac{0.664}{\sqrt{Re_x}} \tag{3.1}$$

Integrating it across the surface of the flat plate, we arrive at the coefficient of drag.

$$C_d = \frac{1}{A}\int C_f \, dA = \frac{1}{l}\int_{x=0}^{l} \frac{0.664}{\sqrt{Re_x}} dx = \frac{1.328}{\sqrt{Re_l}} \tag{3.2}$$

This calculation of drag occurs at idealized conditions and sets a lower bound that the $C_{d_{min}}$ of a polar should always exceed.

Toolchain Exception Handling

In the programming world, exceptions are errors—anomalous conditions that require special processing. Toolchain exceptions are errors that occur during the execution of PROFOIL and XFOIL and are the edge cases that the user must account for to ensure successful GA operation. The optimizer does not know a priori which individuals will yield an error; therefore, it is important to incorporate robust exception handling measures to ensure uninterrupted operation. These measures are basically checks performed on the outputs of PROFOIL and XFOIL execution: `profoil.in` and `polar.dat`. Each individual that passes through the toolchain will exhibit one or more of the following scenarios outlined below. These exceptions are listed in the order in which their handling should be performed. They will be prescribed a disadvantageous fitness according to the rules described later in Chapter 3.4.

PROFOIL Output Cases

Normal PROFOIL operation is defined as the existence of a `profoil.xy` output file that contains real values that form an uncrossed airfoil. However, the user must account for three other possible outcomes after PROFOIL executes. Ideally, PROFOIL outputs an uncrossed airfoil geometry described by coordinates with real numbers. Should PROFOIL not converge, it can either produce an airfoil coordinate file that contains a series of NaNs or it may fail to yield an output file at all. When PROFOIL does converge, it can still produce geometry that contains self-intersections or crossings, which are not realizable. PROFOIL does not

inherently check if the coordinates form self-intersections, allowing for the possibility of crossed airfoils to be produced. These crossed airfoils are illustrated in Fig. 3.3.

- **Exception 1:** profoil.xy does not exist; penalize and return.
- **Exception 2:** profoil.xy exists, but contains NaN values; penalize and return.
- **Exception 3:** profoil.xy exists, and contains real values forming a crossed airfoil; penalize and return.

The existence of the `profoil.xy` file is checked using a try-catch statement. Real number coordinates are checked using a simple if-statement. The number of segments is calculated using a self-intersection algorithm called Fast and Robust Self-Intersections [36]. These checks can be seen implemented in Listing 3.5. A flowchart of these PROFOIL output scenarios can be found in Fig. 3.4. In this flowchart, the off-page connector C is used to indicate where this location continues from the toolchain flowchart in Fig. 3.2.

XFOIL Output Cases

Normal XFOIL operation is defined as the existence of a `polar.dat` output file with 5 deg or more of polar accumulation, containing the true $(C_l/C_d)_{max}$ or $C_{l_{max}}$ at the specified flow condition. All drag values must exceed the drag lower bound. However, there are several exceptions that can occur. Many times XFOIL does not converge, whereupon either no polar file is generated or XFOIL produces a polar that is empty. If XFOIL does converge and produce a polar file, it is possible that only a contrived range of angles of attack is analyzed; therefore, it is recommended that five degrees be the minimum polar span for evaluating the fitness of an airfoil. Lastly, because XFOIL tends to underestimate the drag of an airfoil, a drag lower bound check is implemented.

- **Exception 4:** polar.dat does not exist; penalize and return.
- **Exception 5:** polar.dat exists, but contains no values; penalize and return.

- **Exception 6:** polar.dat exists, but contains fewer than 5 deg of accumulated data; penalize and return.
- **Exception 7:** polar.dat exists and contains 5 deg or more of data, but does not pass the drag lower bound check; penalize and return.

Through the course of sampling the polar.dat to assess if XFOIL has hung, the existence of the polar itself is checked. If the polar file exists, its length can be checked using the `size()` function in MATLAB, which returns the dimensions of an array. The drag lower bound check was previously described. These checks can be seen implemented in Listing 3.6. A flowchart of these XFOIL output scenarios can be found in Fig. 3.5. In this flowchart, the off-page connector D is used to indicate where this location continues from the toolchain flowchart in Fig. 3.2.

3.4 Penalty Function Design

As mentioned in Chapter 2.1.3, the static penalty has been found to be adequate in handling the seven exceptions outlined in Chapter 3.3.3. However, because these exceptions differ in downstream impact to the GA, the magnitude of each penalty should appropriately reflect their severity. This section describes a rationale for a tiered ranking of these penalties according to their contribution to a high quality search space. Because the penalized fitnesses of infeasible individuals must be also ranked against the non-penalized aerodynamic fitnesses of feasible individuals, this section also discusses a fitness schema that reconciles this need within the way that MATLAB physically sorts various numeric types.

3.4.1 Ranking Infeasibility

Different exceptions warrant different penalizations. Since the purpose of the penalty function is to encourage searching the infeasible solution space given that infeasible solutions can impart advantageous traits through recombination, each penalty is judged by the potential of the exception to contribute genetic value. This sub-section explores the differences

in genetic value of each type of exception and presents a three-tiered system, outlined in Table 3.3, for categorizing their relative severities.

Tier 1 exceptions have high genetic value because they are almost feasible. Tier 2 exceptions have comparatively lower genetic value because they are more infeasible. Tier 3 exceptions, unlike Tier 1 and Tier 2, confer no genetic value. The design space of Tier 1 is the most desirable and should be lightly penalized. The middle tier should be more aggressively penalized in order to encourage the direction of search towards Tier 1 feasibility. The bottom tier should be completely avoided and is awarded the most severe penalization.

All exceptions except for Exception 3 belong in Tier 3 because they are situations in which toolchain outputs yields unusable data. These infeasible design regions should be avoided because no heuristic information can be gained from them; therefore, they have no genetic value. Exception 3 contributes to the quality of the search because it is the only exception that yields airfoil coordinates, albeit, ones with self-intersections. Tiers 1 and 2 separate airfoils with one self-intersection versus those with many. An airfoil with five self-intersections is more severely exceptional than an airfoil with two self-intersections, and so it would stand to reason that the GA should prefer the solution that generates only two. Therefore, the penalty function should aggressively penalize crossed airfoils in order of increasing self-intersections. Individuals with one self-intersection should occupy the first, least penalized tier. All other crossed individuals with two or more crossings should occupy Tier 2.

Hypothetically, it is conceivable to create a ranking scheme that increases in penalty according to the order in which the exceptions occur with the rationale being that the later the exception is caught, the higher the computational impact. An important point to keep in mind, however, is that the implementation of the penalty function should be consistent with its purpose, which is to improve search quality through the judicious inclusion of the infeasible search space, not simply to encourage speedy resolutions to exceptions.

Table 3.3: Table Describing Three-Tiered Hierarchy of Penalties.

Tier	Genetic Value	Relative Penalty
1	High	Low
2	Low	Medium
3	None	High

3.4.2 Sorting in MATLAB

By default, MATLAB represents floating point numbers as double-precision numeric types [18], which are constructed according to the IEEE 754 standard. This standard includes non-real quantities such as NaN and Inf, which stand for "not a number" and "positive infinity," respectively. A simple sort of sample values reveals how MATLAB sorts these non-real values:

```
>> sort([-2,1,2,inf,-inf,0,-1,nan],'ascend')


ans =
-Inf    -2    -1     0     1     2   Inf   NaN
```

Because the MATLAB GA is inherently a minimizer, negative values are fitter than positive values. Three convenient domain subsets emerge in order of decreasing fitness: negative real numbers, positive real numbers, and NaNs. Because MATLAB ranks NaN lower than positive infinity, we can prescribe Tier 1 and Tier 2 penalties according to any numerical calculation and know that if a Tier 3 penalty is prescribed NaN, it will always be ranked lowest among them.

3.4.3 Establishing a Fitness Schema

We take advantage of these demarcations to create a schema for describing the whole range of possible fitnesses by reconciling the 3-tier penalty hierarchy within the framework of how MATLAB physically ranks numeric values as shown in Table 3.4. The exceptions associated with each tier are also included for clarity.

Table 3.4: Table Describing Fitness Schema.

Tier	Exception	Domain	Numeric Range
	No exceptions; Feasible Solutions	Negative Real	$[-1000, 0]$
1	3: Airfoil with only 1 self-intersection	Positive Real	(0,100)
2	3: Airfoil more than 1 self-intersections	Positive Real	$[100, \infty]$
3	1,2,4,5,6,7	Non-real	NaN

Negative Real Number Domain

All feasible solutions are prescribed in the negative real domain. Conveniently, the $(C_l/C_d)_{max}$ and $C_{l_{max}}$ for positive lift production are physically positive entities. Since the GA minimizes, this value is negated in order to ensure correct ranking.

Positive Real Number Domain

Tiers 1 and 2 can be wholly captured by the positive real domain. This situation is particularly fitting because crossed airfoils can be physically ranked against each other. An important consideration is to ensure that the penalization scheme exists reasonably on the same scale as the feasible solution. Because the feasible fitness values are limited to the hundreds magnitude and do not exceed 400, Tier 1 penalties are designed to span the range 0 to 100.

Non-Real Domain

Tier 3 exceptions appropriately occupy the NaN domain as they cannot be compared to each other. For example, it would be nonsensical to prescribe a preference between a solution for which the polar file does not exist and a solution for which the PROFOIL input file is missing.

3.4.4 Crafting the Penalty Function

This section describes how the penalty of each exception is actually calculated. A MATLAB package called `selfintersect` [36] is used to return a list of segments and intersection

points from which the Tier 1 and Tier 2 penalties are calculated. Listing 3.7 documents the code used to perform this calculation. The crossed airfoils that receive Tier 1 and Tier 2 penalties are visualized in Fig. 3.3. A flowchart of all penalties are represented in Figs. 3.4 and 3.5. Finally, the actual implementation of the exception handling measures discussed in this chapter and their respective penalizations are shown in Listings 3.5 and 3.6.

Tier 1 Penalty

The Tier 1 penalty leverages the concept of cost-to-repair, which was described in Chapter 2.1.3. In such a scheme, the infeasible individual is penalized according to how much it would cost to "repair" it such that it becomes feasible. In other words, it is a measure of how much the constraint is violated. The cost-to-repair for crossed airfoils with only one intersection can be said to be the fraction of the area that is crossed.

A crossed airfoil containing only one self-intersection will contain two closed geometric loops. Airfoil coordinate tradition dictates that the first coordinate begins at the trailing edge and traces the upper surface first. Therefore, all points past the intersection form the crossed loop. If the coordinates associated with these two loops can be separated, the `polyarea()` function can be used to determine the area of each closed loop. In order to scale this penalty to a value between 0 and 100, we find the ratio of its crossed area $A_{crossed}$ to total area A_{total} and then multiply it by 100 to find the percentage. This percentage is the Tier 1 penalty fitness f_{T1}.

$$f_{T1} = \frac{A_{crossed}}{A_{total}} \times 100 \tag{3.3}$$

Tier 2 Penalty

Tier 2 exceptions have two or more self-intersections. It would be difficult to leverage the cost-to-repair concept because calculating the crossed area with two or more crossings is not obvious. However, since the penalization should scale aggressively according to the number of self-intersections, an exponential function is used to scale the penalty based on the number

of crossings $n_{crossings}$ calculated by the `selfintersections` package. Tier 2 penalties will have a lower bound of 100, and an upper bound of infinity; however, in practice, values of 100 million or more have been rarely observed.

$$f_{T2} = (n_{crossings})^{10} \tag{3.4}$$

Tier 3 Penalty

As discussed before, Tier 3 exceptions will have a flat penalty of NaN.

$$f_{T3} = NaN \tag{3.5}$$

Listing 3.5: MATLAB script implementing PROFOIL output exception handling.

```
% Load 'profoil.in' and run PROFOIL.
run_profoil('profoil.in');

% Try to read in x-y coordinates from 'profoil.xy'.
try
    [x,y] = read_coordinates('profoil.xy');
% Handle Exception 1: if profoil.xy does not exist, penalize and return.
catch
    fitness = NaN;
    return;
end

% Handle Exception 2: if any coordinate is NaN, penalize and return.
if sum(isnan(x))  sum(isnan(y))
    fitness = NaN;
    return;
end

% Handle Exception 3: if the number of crossings exceeds 0, penalize and
return.
if n_crossings > 0
    % Call function to calculate penalty for crossed airfoil.
    [fitness,~,~,~] = crossingpenalty(x,y);
    return;
end

```

Listing 3.6: MATLAB script implementing XFOIL output exception handling.

```
% Load 'profoil.xy' and run XFOIL.
run_xfoil('profoil.xy');

% Determine if XFOIL is running and that the polar file has been created.
[response, cmdout] = system('tasklist /fi "imagename eq xfoil.exe"');
if exist(file_polar, 'file') == 2 && sum(strfind(cmdout,'xfoil.exe')) > 0
    polar_diff = 1;
    while pacc_diff > 0
        pause(3); % Pause to wait for polar file to be created
        length1 = numel(textread('polar.dat','%1c%*[^\n]')); % Read length
        pause(2); % Pause for XFOIL to calculate more data points
        length2 = numel(textread('polar.dat','%1c%*[^\n]')); % Read length
        polar_diff = length2 - length1; % Calculate difference in file length
    end
    dos('taskkill /f /im xfoil.exe')    % Kill XFOIL
else
% Handle Exception 4: polar.dat does not exist, penalize and return.
    fitness = NaN;
    return;
end

% Call function to read in data from polar file into polar matrix.
[polar] = read_polar(file_polar);

% Handle Exception 5: if polar.dat exists, but is empty, penalize and return.
if size(polar,1) == 0
    fitness = NaN;
    return;
end

% Extract polar data into usable vectors
alfa = polar(:,1); Cl = polar(:,2); Cd = polar(:,3); Cm = polar(:,5);

% Handle Exception 6: if polar is less than 5 degrees, penalize and return.
alfa_span = max(alfa) - min(alfa);
if alfa_span < 5
    fitness = NaN;
    return;
end

% Multiply by 2 for upper and lower surface of flat plate.
Cd_blasius = 2 * (1.328/(Re)^0.5);
% Handle Exception 7: if lower drag bound is violated, penalize and return.
if sum(Cd < Cd_blasius) > 0
    fitness = NaN'
    return;
end

% Multiply the aerodynamic fitness by -1 to accomodate the minimizer.
fitness(1) = -1 * max(Cl./Cd);
fitness(2) = -1 * max(Cl);

```

Listing 3.7: MATLAB script crossingpenalty.m.

```
function [penalty,n_segments,x0,y0] = crossingpenalty(x,y)
% selfintersect.m finds a list of segments and intersection points x0,y0.
[x0,y0,segments] = selfintersect(x,y);
tolerance = 0.00001;

% Removes false intersections at trailing edge
ind_fp = find(abs(x0 - 1) < tolerance);
if abs(y0(ind_fp)) < tolerance
segments(ind_fp,:) = [];
x0(ind_fp) = [];
y0(ind_fp) = [];
end

% 1 intersection
if size(segments,1) == 1
cross1 = segments(1,1); cross2 = segments(1,2);
x_pos = x(cross1:cross2);
y_pos = y(cross1:cross2);
x_neg = horzcat(x(cross2:end),x(1:cross1));
y_neg = horzcat(y(cross2:end),y(1:cross1));
pos_area = polyarea(x_pos,y_pos);
neg_area = polyarea(x_neg,y_neg);
tot_area = pos_area + neg_area;
penalty = (neg_area/tot_area) * 100;

% Remove the penalty if area is virtually 0.
if abs(penalty) < tolerance
n_segments = 0;                    % set to be uncrossed
else
n_segments = size(segments,1);     % set as crossed
end
end

% More than 1 intersection
if size(segments,1) > 1
penalty = (10^size(segments,1));
n_segments = size(segments,1);
end

% 0 intersections
if size(segments,1) == 0
penalty = 0;
n_segments = 0;
end

end


```

3.4.5 Real-World Performance

In this study, the author has found that the penalty function is instrumental to search speed. If one were to analyze the allocation of computing time of one loop of the toolchain using the MATLAB Profiler, the user will find that, by and large, XFOIL execution time dominates by an order of a magnitude over everything else.

Feasible solutions, by virtue of the viability of their geometry, will quickly and quietly be realized by XFOIL. Infeasible solutions, on the other hand, will kick and scream into the night—pushing XFOIL to the maximum number of iterations only to cause XFOIL to not converge and hang. The author has found that at 100 iterations on a 2.4 GHz second generation i5 processor, this difference can be 30 to 45 sec per toolchain call. In short, it is expensive for the optimizer to attempt to push an infeasible solution through XFOIL. By discouraging such results, the penalty function becomes a critical tool for efficiently searching the non-linear design space of an inverse solver.

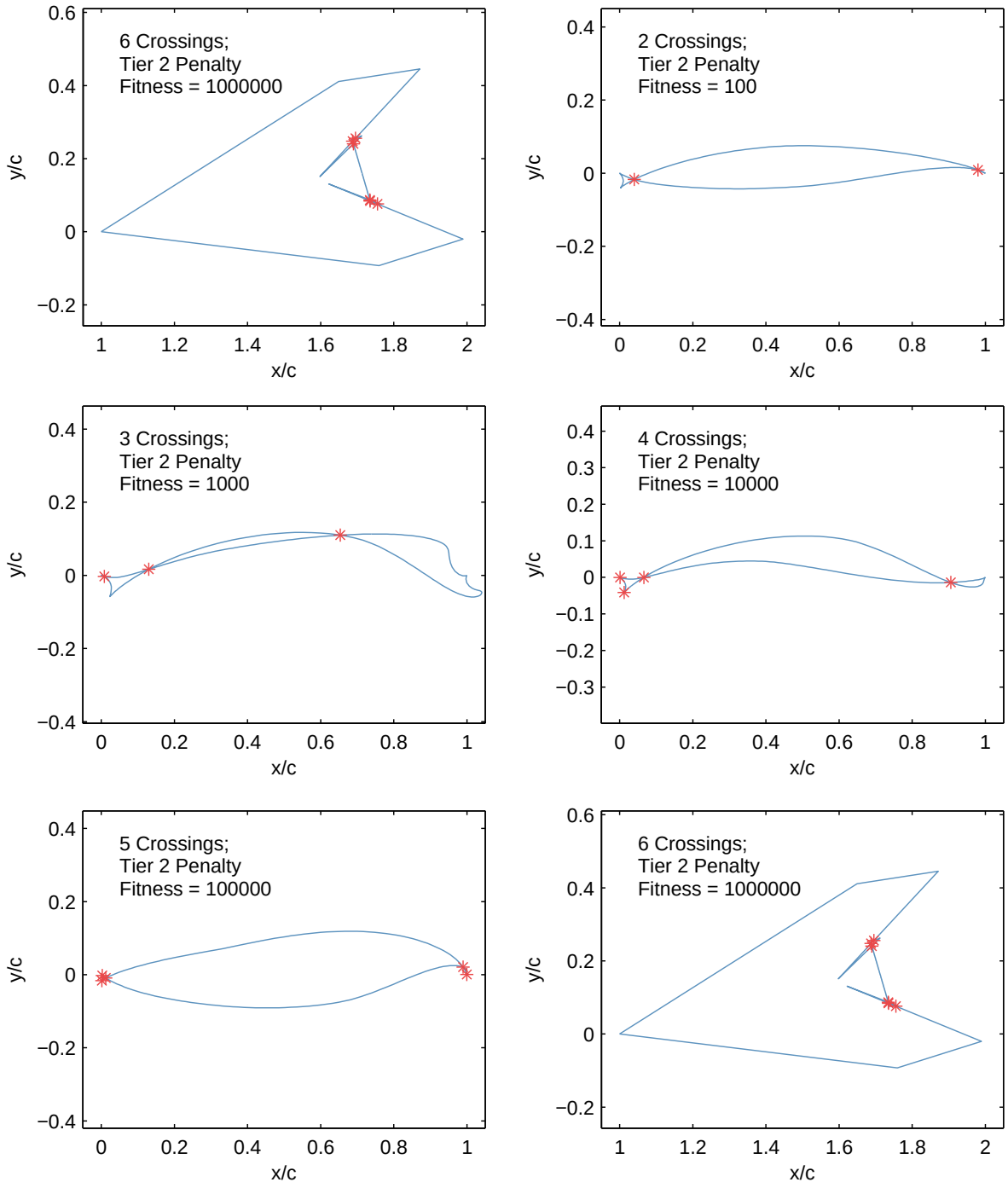

Figure 3.3: Examples of crossed airfoils and their respective penalties.

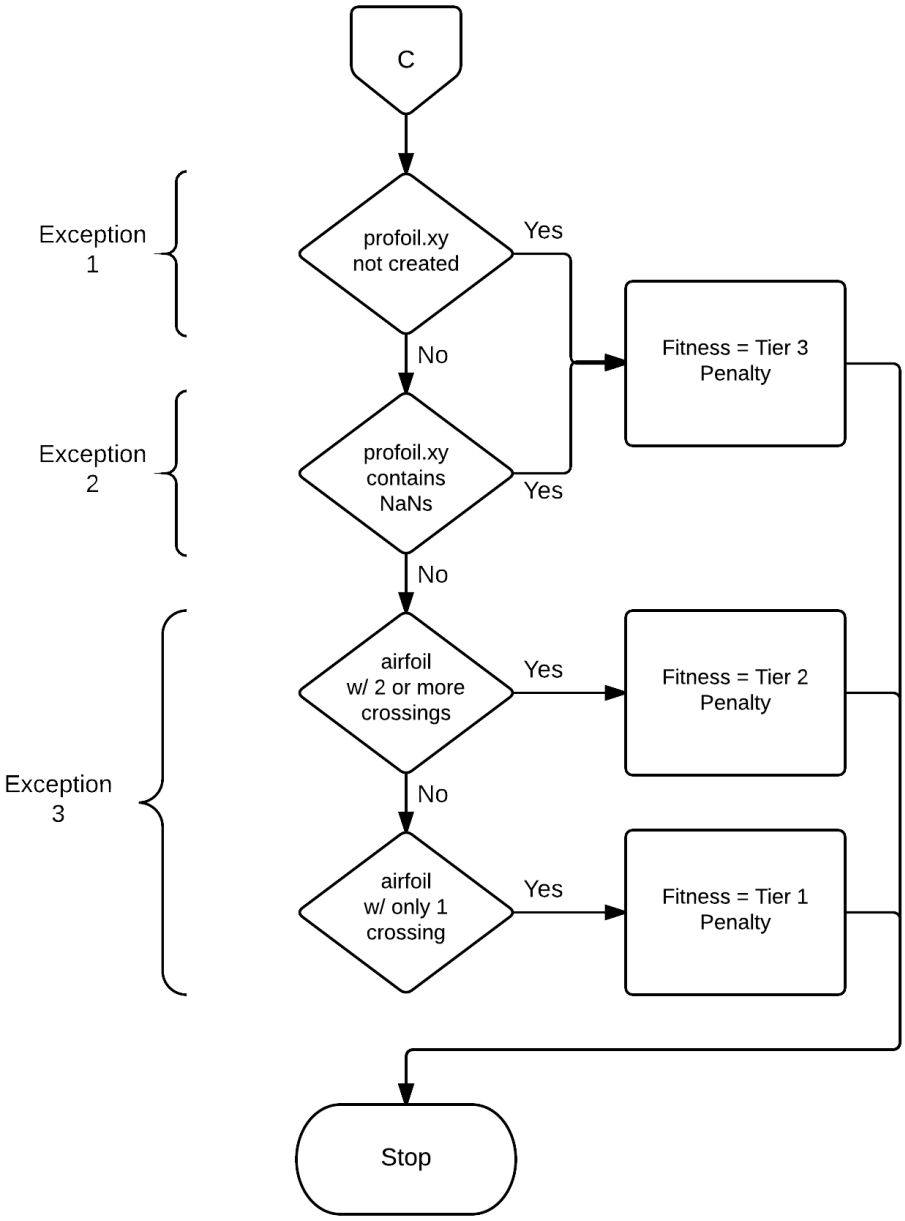

Figure 3.4: Flowchart that describes PROFOIL output exception handling and penalizations.

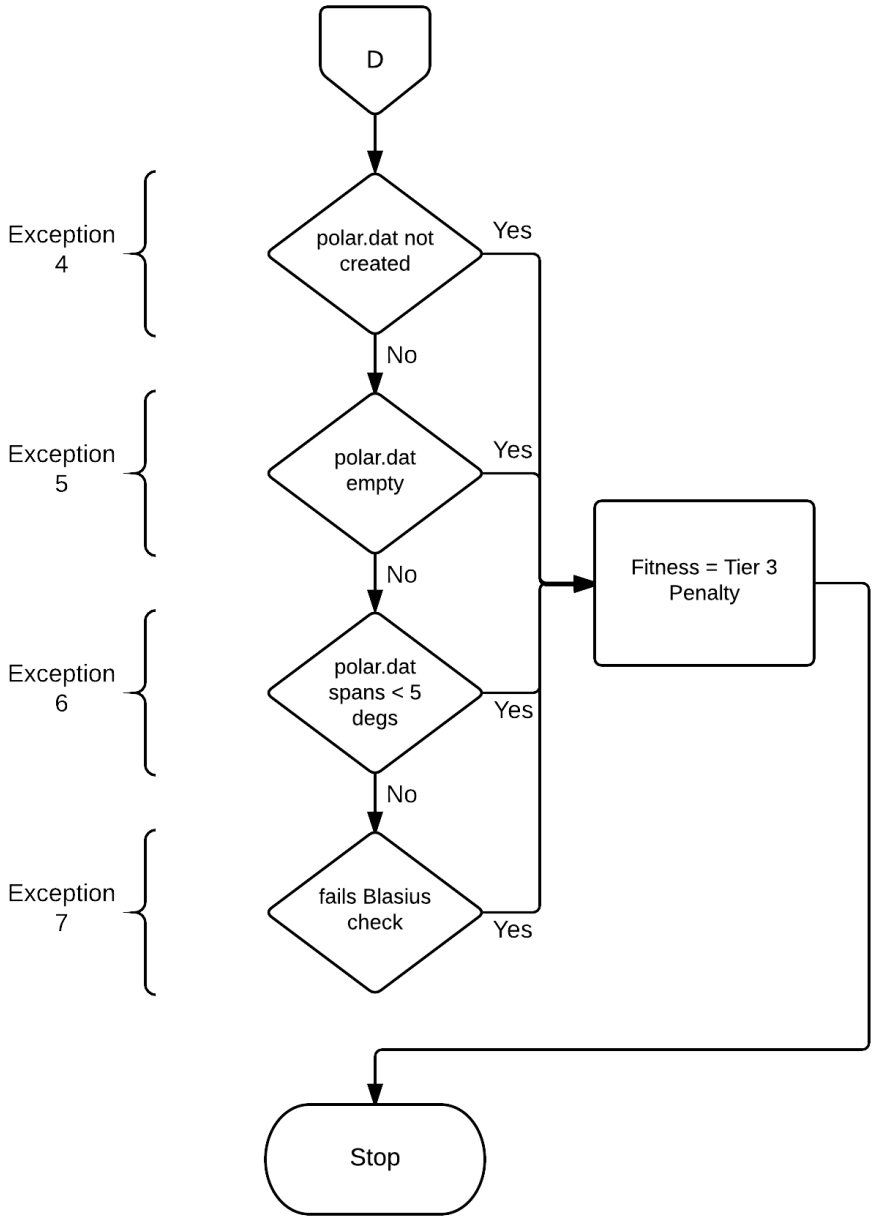

Figure 3.5: Flowchart that describes XFOIL output exception handling and penalizations.

CHAPTER 4

RESULTS AND DISCUSSION

This chapter discusses the results of the three studies conducted in this thesis: GA tuning, $(C_l/C_d)_{max}$ optimization, and $C_{l_{max}}$ optimization at varying design pitching moments. The figures of merit used in comparing GA configurations are the maximum fitness (MF) at convergence and the final generation (FG). Because the fitness itself is the search heuristic, the maximum achieved fitness directly illustrates how effective a particular GA configuration is at navigating the search space. In other words, the maximum fitness measures the raw performance of the GA. On the other hand, because this process is meant as a design tool, the GA must also provide a solution in a reasonable amount of time. Therefore, the final generation is a quantification of computational effort and a reflection of its real-world practicality. Discussion of the results of Study 2 and Study 3 focus on the aerodynamic aspects of the final solutions. Aerodynamic data are presented in the form of velocity distributions, lift curves, and drag polars.

4.1 Study 1: GA Tuning Results

Study 1 consisted of 32 optimization runs of varying airfoil segments and crossover types performed at two different design conditions: $C_m = -0.063$ and $t/c = 18\%$ at $Re = 6.88 \times 10^6$; and $C_m = -0.030$ and $t/c = 18\%$ at $Re = 2.00 \times 10^6$, which are referred to as Design Condition 1 and Design Condition 2 in this chapter. Trials 1–10 and 29–32 were performed at Design Condition 1 and Trials 11–28 were performed at Design Condition 2.

Table 4.1 shows the mean maximum fitness (μ) and standard deviation (σ) in maximum fitness for the trials of each design condition. The mean maximum fitness for the 14 trials conducted at Design Condition 1 was 328.88 and the standard deviation was 1.24. The

Table 4.1: Mean and Standard Deviation of the Fitness for Each Design Condition for Study 1.

Trials	Design Condition	Maximum Fitness	
		μ	σ
1–10, 29–32	(Design Condition 1)	328.88	1.24
11–28	(Design Condition 2)	210.80	0.46

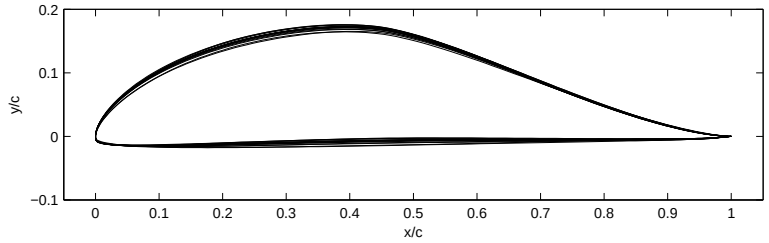

Figure 4.1: Final solution airfoil overlays of Trials 1–10 and 29–32 for Design Condition 1: $C_m = -0.063$ and $t/c = 18\%$ at $Re = 6.88 \times 10^6$.

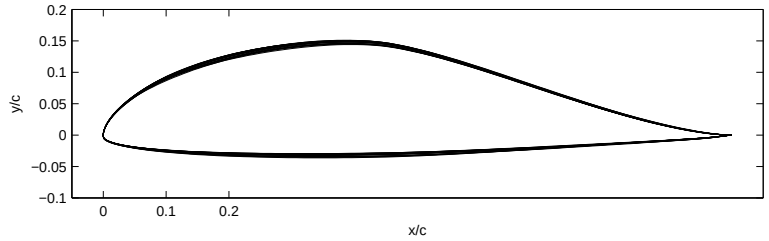

Figure 4.2: Final solution airfoil overlays of Trials 11–28 for Design Condition 2: $C_m = -0.030$ and $t/c = 18\%$ at $Re = 2.00 \times 10^6$.

mean maximum fitness for the 18 trials conducted at Design Condition 2 was 210.80 and the standard deviation was 0.46. There is almost no statistical spread in the final solutions of each design condition.

The final solution for each design condition shown in Figs. 4.1 and 4.2 show that nearly the same airfoil is realized at each design condition. Such strong agreement across all trials demonstrate that the MATLAB Optimization Toolbox is capable of reliably producing self-consistent results despite differences in GA configuration.

Table 4.2: Summary of Results for Each Trial in Case 1 of Study 1 at Design Condition 1.

Segments	Crossover Option	MF	FG	Time Elapsed
6	Arithmetic	327.99	138	72.6 hrs
6	Heuristic (1.2)	329.27	122	57.9 hrs
6	Intermediate [1]	327.84	121	73.2 hrs
6	Scattered	329.13	90	53.8 hrs
6	Single Point	328.48	97	58.0 hrs
6	Two Point	327.62	81	47.7 hrs

4.1.1 Case 1

Case 1 is a comparison of each aggregate-based crossover operator (ABCO) and discrete crossover operator (DCO) at Design Condition 1 using 6 segment airfoils. The results are summarized in Table 4.2. In Case 1, the configuration that attains the highest maximum fitness is that of heuristic crossover, which yields a final solution with maximum fitness of $C_l/C_{d_{max}} = 329.27$ after 122 generations. The fastest converging configuration is that of two point crossover, which yields a final solution with a maximum fitness of $C_l/C_{d_{max}} = 327.62$ after 81 generations. In fact, all DCO options in Case 1 converge must faster than ABCO options. All trials completed within 2 to 3 days of computing time.

4.1.2 Case 2

Case 2 varied the number of airfoil segments for heuristic crossover at heuristic ratio = 1.2 and single point crossover. Optimization runs were made for 7 and 8 segment airfoils in Trials 7 through 10. For comparison, the results for 6 segments are also shown in Table 4.3. As the number of segments increased, heuristic crossover at heuristic ratio = 1.2 consistently performed well in terms of maximum fitness, while the single point crossover seemed to perform poorly past 7 segments. In fact, the highest fitnesses for both crossover options were achieved at 7 segments.

Figure 4.3 shows the progression of maximum fitness through the optimization run and indicates the generation at which convergence begins. The right-side up triangle indicates the generation at which convergence begins for the heuristic crossover trials and the upside-down triangle is the generation at which convergence begins for the single point trials. At 6

Table 4.3: Summary of Results for Each Trial in Case 2 of Study 1 at Design Condition 1.

	Heuristic (1.2)		Single Point	
Segments	MF	FG	MF	FG
6	329.27	122	328.48	97
7	329.89	140	329.35	92
8	329.27	91	326.18	147

and 7 segments, single point converges faster than heuristic crossover at heuristic ratio = 1.2, but prematurely. At 8 segments, single point crossover not only takes longer to converge, but its final solution is worse compared to that generated by the heuristic option. As the airfoil representation scheme increases in complexity, single point crossover performs poorly in maximum fitness and in computational time.

4.1.3 Case 3

Because only two crossover options were actually observed in Case 2, a more thorough optimization at a different design condition was performed. In Case 3, all crossover options were tested using 6, 7, and 8 segment airfoils at Design Condition 2. The results can be found in Table 4.4. When comparing the spread of the performances of the airfoils designed in this case compared to those designed at Design Condition 1 across the other three cases, there was even more agreement between the final solutions at this design point. This slight disparity is attributed to the fact that the trials performed at Design Condition 1 across the other three cases were heavily skewed towards the heuristic crossover type, while the trials performed at Design Condition 2 were uniformly distributed across all six crossover options.

In Table 4.4, five out of the six crossover options achieved the highest maximum fitness with the 7 segment airfoil, with the only exception being scattered crossover. The average maximum fitness for all crossover options at 7 segments at Design Condition 1 was 211.16, while the maximum fitness averages at 6 and 8 segments were 210.56 and 210.73, respectively. Furthermore, it would appear that at Design Condition 2, DCO options converge faster than ABCO options for all segments.

With the information gathered from Cases 1, 2, and 3, it seems that two general conclusions

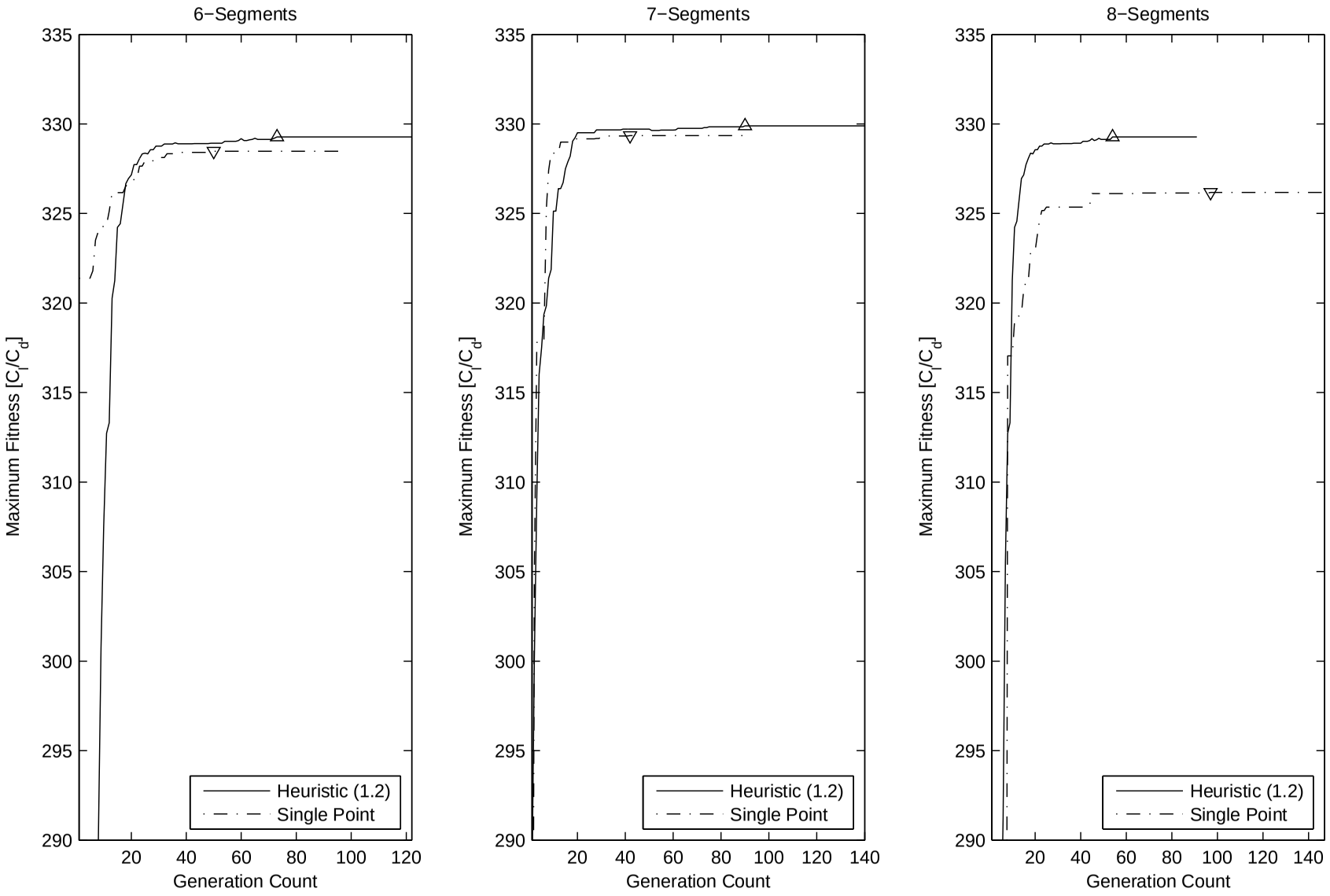

Figure 4.3: The maximum fitness per generation for heuristic crossover at heuristic ratio = 1.2 versus single point crossover at 6, 7, and 8 segments.

Table 4.4: Summary of Results for Each Trial in Case 3 of Study 1 at Design Condition 2.

	6 Segments		**7 Segments**		**8 Segments**	
Crossover Option	MF	FG	MF	FG	MF	FG
Arithmetic	210.50	124	210.88	93	210.17	89
Heuristic (1.2)	210.34	92	211.16	160	211.09	134
Intermediate [1]	210.24	109	210.84	89	210.21	119
Scattered	210.76	80	211.42	47	211.73	68
Single Point	210.73	77	211.40	97	210.47	58
Two Point	210.76	93	211.28	66	210.49	93
Mean MF	*210.56*		*211.16*		*210.73*	

can be made: 7 segment airfoils achieve the highest maximum fitness and discrete crossover operators achieve convergence faster than aggregate-based crossover operators for 6, 7, and 8 segments. At Design Condition 2, heuristic crossover at heuristic ratio = 1.2 performs better than single point crossover in terms of maximum fitness at 8 segments, but not at 6 or 7 segments as it had at Design Condition 1. When considering the trials for the other crossover options at Design Condition 2, the three highest achieved maximum fitnesses for any given number of segments are from the scattered, single point, and two point crossover options—all discrete operators. This data suggests that the tuning for achieving the highest maximum fitness possible is design condition dependent.

4.1.4 Case 4

In determining a tuned GA configuration for Studies 2 and 3, it was decided that because they are to be performed at $C_m = -0.060$ and $Re = 6 \times 10^6$, which is close to Design Condition 1, heuristic crossover merited additional exploration. Case 4 explored various heuristic ratios at 6 segments in order assess the potential of the heuristic crossover. Trials 30 through 33 were performed for heuristic ratios 1.4 through 2.0 at 0.2 intervals at Design Condition 1. The results are shown in Table 4.5.

Figure 4.4 is a scatter plot of the maximum fitness and final generation of each Design Condition 1 trial. The filled data points represent the optimal solutions or the Pareto set. In multi-objective optimization, there may be multiple optimal solutions that are non-dominated, defined to be the set of solutions for which at least one objective is optimized,

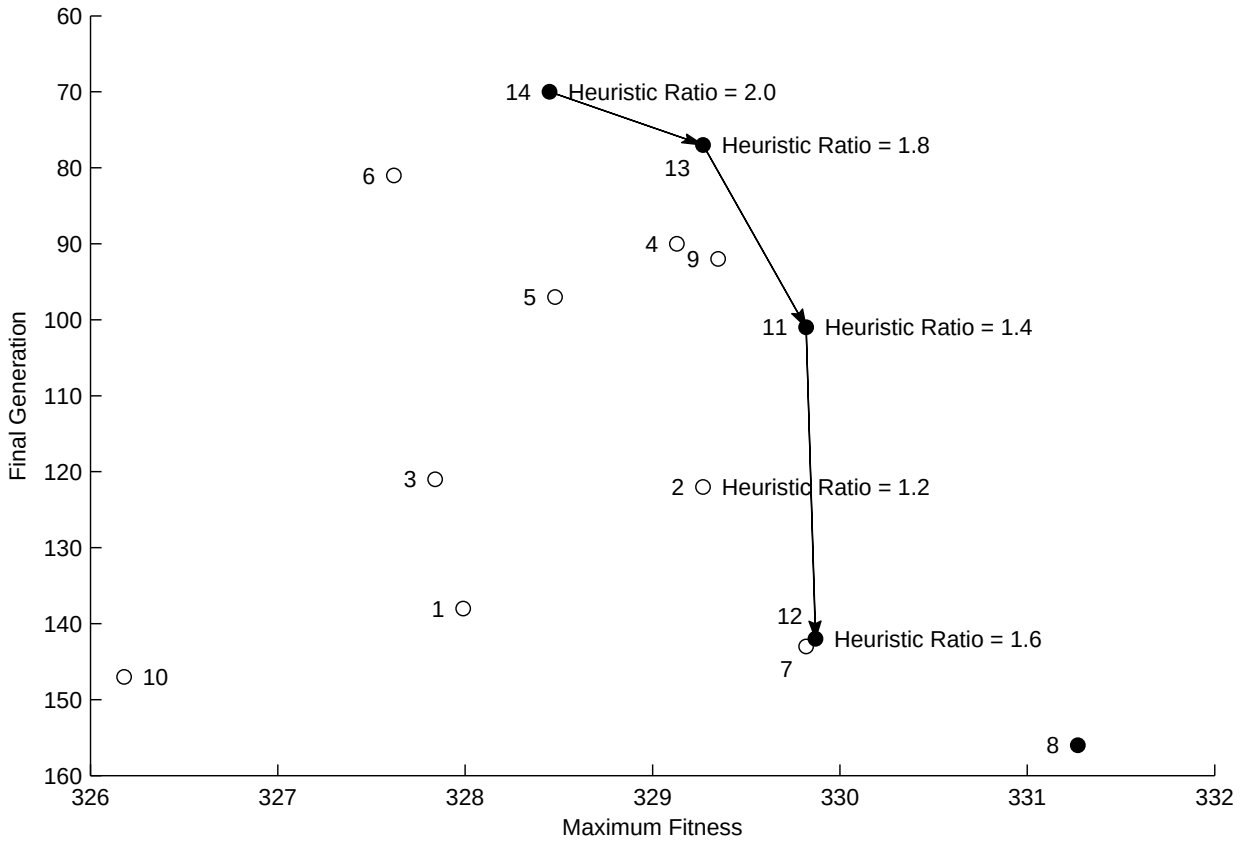

Figure 4.4: Final generation versus maximum fitness scatter plot of all Design Condition 1 trials.

while all objectives are constant [37]. This concept is best explained by using the example of two solutions, individuals A and B, where A has a higher maximum fitness and a lower final generation than B. In this case, individual A clearly *dominates* B in both figures of merit. Individual A is said to be non-dominated if it has a higher maximum fitness, but a higher final generation than B. In other words, individual A has better raw performance, but is also less computationally efficient than B. Any non-dominated solution is not definitively better than any other non-dominated solution, which makes each member of the Pareto set an optimal solution. Figure 4.4 is a scatter plot of the final generation and maximum fitness of all Design Condition 1 trials. Each data point is labeled with the trial number and the Pareto set is indicated by filled circles. It is clear that the range of heuristic ratios 1.4 through 2.0 at 6 segments heavily represent the optimal configuration among all 14 trials.

The data points for heuristic ratio 1.6 and 2.0 represent the two extremes in maximum

Table 4.5: Summary of Results for Each Trial in Case 4 of Study 1 at Design Condition 1.

Trial	Segments	Crossover Option	$(C_l/C_d)_{max}$	FG
2	6	Heuristic (1.2)	329.27	122
30	6	Heuristic (1.4)	329.82	101
31	6	Heuristic (1.6)	329.87	142
32	6	Heuristic (1.8)	329.27	77
33	6	Heuristic (2.0)	328.45	70

performance and computational efficiency. The most appropriate ratio has a balance of both characteristics, which means it lies somewhere along the frontier between these two points, which is visualized by the arrows in Fig. 4.4 connecting each adjacent optimal solution data point in the direction of increasing final generation number and increasing maximum fitness. The slopes of these lines represent the computational cost of increasing maximum fitness. Steep slopes represent a high computational cost for little gain in maximum fitness, while a shallow slope represents low computational cost for high gain in maximum fitness, which is advantageous. According to this rationale, heuristic ratio = 1.8 is the best choice because the arrow that connects heuristic ratios 2.0 and 1.8 is the shallowest. Therefore, for design conditions close to that of Design Condition 1, heuristic crossover with heuristic ratio = 1.8 yields the best balance between maximum fitness and computational time.

It was decided that the 6 segment representation combined with this crossover option was adequate for the purposes of attaining strong maximum fitnesses within a reasonable amount of computational time. Therefore, the tuned GA configuration used in Studies 2 and 3 is heuristic crossover at heuristic ratio = 1.8 at 6 segments. Incidentally, at 77 generations, the tuned GA configuration converges in nearly half the time as the Case 1 and Case 2 trials, which is consistent with the results of Vicini and Quagliarella [9].

4.2 Study 2: Optimization for $(C_l/C_d)_{max}$

In Study 2, Trial 33 was an optimization of $(C_l/C_d)_{max}$ conducted at the design condition: $C_m = -0.060$ and $t/c = 18\%$ at $Re = 6.00 \times 10^6$. The final generation achieved a maximum fitness of 315.08 after 79 generations. Figure 4.6 shows its pressure distribution at the $(C_l/C_d)_{max}$ angle of attack of 10.00 deg. Figure 4.5 shows that this slightly reflexed airfoil reaches a maximum C_l of 1.92 at 10.75 deg after which it abruptly stalls.

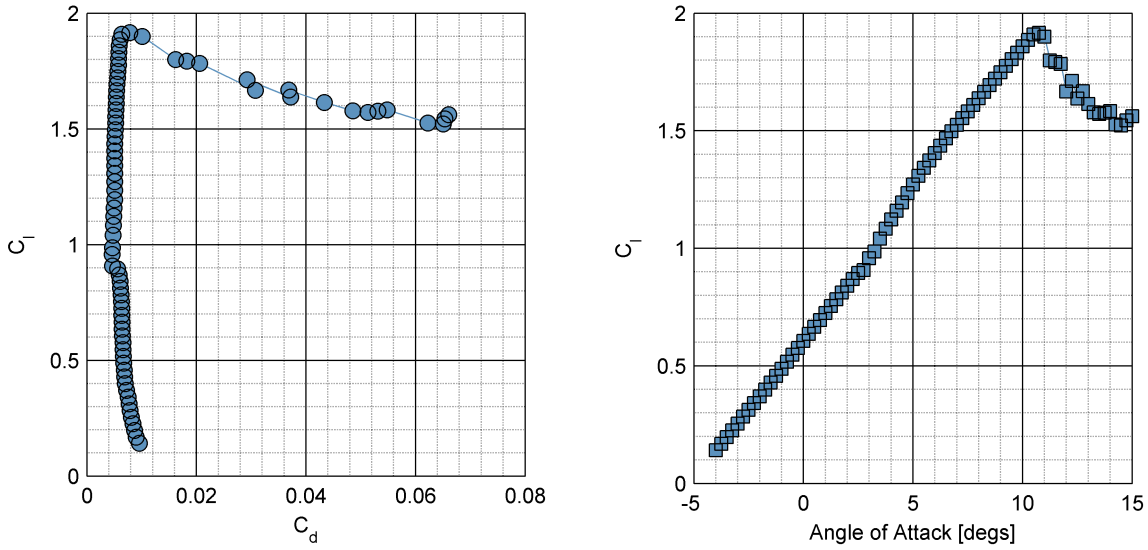

Figure 4.5: Trial 33 drag polar and lift curve computed from -4 to 15 deg in XFOIL.

Incidentally, the final solution airfoil of Trial 33 resembles the Liebeck L1003, which is an 18% thick airfoil optimized for maximum lift at $Re = 2.00 \times 10^6$ shown in Fig. 4.7. Liebeck's design philosophy was to use the inverse method to derive a geometry that would yield laminar rooftop followed by a Stratford distribution [38] in the upper surface pressure recovery region at a specified design angle of attack. Because the boundary layer is on the verge of separation at all points along this segment, $C_f = 0$ along this length, which should allow the airfoil to attain the maximum $C_{l_{max}}$ for a specified design condition. Inspection of the final solution velocity distribution in Fig. 4.6 indicates that it is also similar to that of the Liebeck L1003, exhibiting a flat rooftop followed by a curve similar to that of the Stratford distribution.

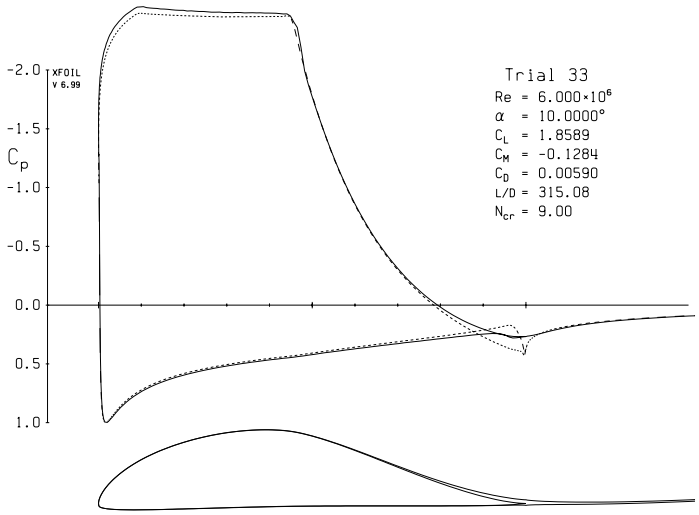

Figure 4.6: Trial 33 final solution geometry and velocity distribution at 10.00 deg plotted in XFOIL.

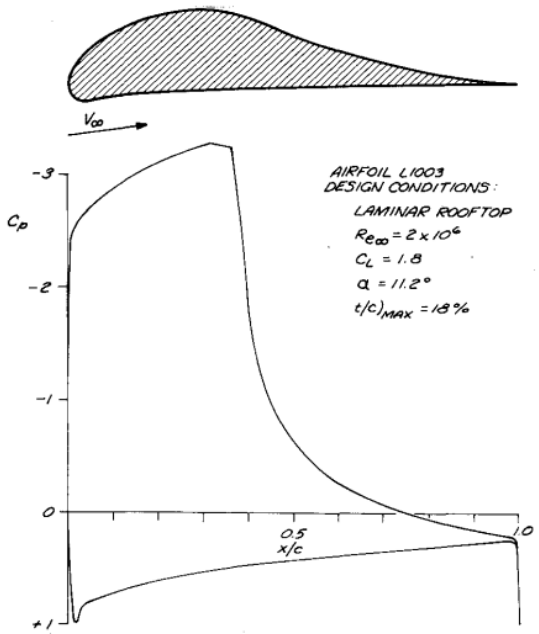

Figure 4.7: Liebeck L1003 and its pressure distribution at the design angle of attack of 11.2 deg (taken from Ref. [33]).

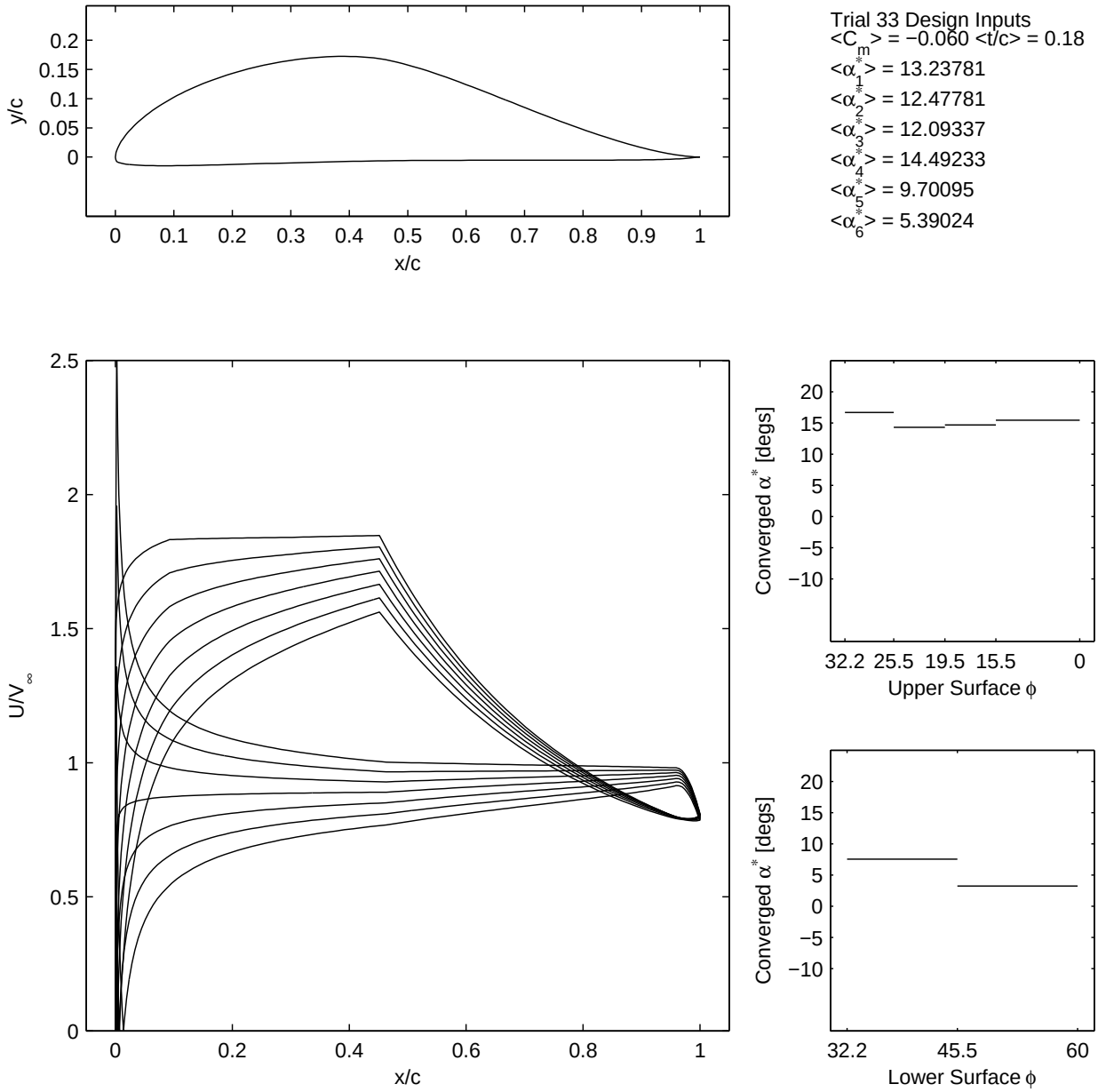

Figure 4.8: Trial 33 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

4.3 Study 3: Optimization for $C_{l_{max}}$ at Varying Pitching Moments

In Study 3, nine 18% thick airfoils were optimized for $C_{l_{max}}$ in Trials 34 through 42. Each optimization run was performed at $Re = 6.00 \times 10^6$ for pitching moments that increased in negativity from $C_m = 0.000$ through $C_m = -0.020$ at intervals of 0.025. Their results are summarized in Table 4.6.

Table 4.6: Summary of Results for Each Trial in Study 3.

Trial No.	Segments	Crossover Operator	C_m	MF	FG	α_{max}
Study 3						
34	6	Heuristic (1.8)	0.000	2.1048	122	12.75
35	6	Heuristic (1.8)	−0.025	2.1154	114	15.00
36	6	Heuristic (1.8)	−0.050	2.1333	141	15.00
37	6	Heuristic (1.8)	−0.075	2.0448	107	17.25
38	6	Heuristic (1.8)	−0.100	2.1258	161	17.25
39	6	Heuristic (1.8)	−0.125	2.1729	147	16.50
40	6	Heuristic (1.8)	−0.150	2.2497	116	16.75
41	6	Heuristic (1.8)	−0.175	2.2587	200	15.75
42	6	Heuristic (1.8)	−0.200	2.2618	63	15.00

Figure 4.9 is a 3-by-3 matrix of plots that overlay the geometries of the initial generation with that of the final generation for each trial. The arrangements of the plots are such that the optimization runs increase in nose-down pitching moment from top to bottom and, column-wise, from left to right. The leftmost column of plots depicts overlays for the $C_m = 0.000$, -0.025, and -0.050 optimization runs (Trials 34–36). The center column depicts overlays for the $C_m = -0.075$, -0.100, and -0.125 optimization runs (Trials 37–39). And the rightmost column depicts overlays for the $C_m = -0.150$, -0.175, and -0.200 optimization runs (Trials 40–42). It can be seen that the final generation geometries can be categorized as either a reflexed or aft-loaded airfoil.

The reflexed airfoils of range $C_m = 0.000$ to -0.050 have a concave camber line, while the aft-loaded airfoils of the more aggressive range, $C_m = -0.075$ to 0.200, have a convex camber line. The mean cambers of these airfoils can be more clearly seen in Figs. 4.10 and 4.11, which depict overlays of final solution geometries and their respective mean camber lines, calculated as the average of the upper surface and lower surface y-coordinates at each x-

position. Inspection of the design velocity distributions of each optimization run confirm this segregation in airfoil type as Figs. 4.15–4.17 show the characteristic negative lift regions at the trailing edge of the reflexed airfoils, while Figs. 4.18–4.23 show positive lift production towards the aft of the airfoil.

The overlays of the aft-loaded airfoils in 4.11 show a clear relationship between design pitching moment and camber: as the design pitching moment varies in the direction of increasing negativity, the mean camber line of the airfoil becomes more pronounced. At the same time, the maximum coefficient of lift steadily increases from 2.04 to 2.26. A comparison of their lift curves in Figs. 4.13 and 4.14 shows that as design pitching moment decreases, the entire lift curve moves up and left. This trend is consistent with Gopalarathnam and Selig's studies of low-speed natural laminar flow airfoils [39], which are also aft-loaded by design.

Analysis of the reflexed airfoils designed for pitching moments from zero to $C_m = -0.050$ reveals the relationship between maximum lift production, camber, and pitching moment. The reflexed airfoils exhibit large leading edge camber with minor reflexed trailing edge camber. This characteristic is consistent with the results of Thin Airfoil Theory (TAT) in the prediction of the zero-lift pitching moment of NACA 4-series airfoils as a function of their maximum camber and chordwise position of maximum camber [40]. Figure 4.12 shows that camber near the leading edge has a much smaller effect on pitching moment compared to camber near the trailing edge. This result makes sense because camber positioned near the aft has a longer moment arm with which to exert mechanical advantage at the aerodynamic center. In fact, according to Hepperle, it is possible to manipulate the camber near the trailing edge to achieve nearly any desired coefficient of moment [41].

The GA optimizer takes advantage of this fact to realize high-lift airfoils with near neutral pitching moments. The addition of the slight reflexed camber near the trailing edge significantly reduces its pitching moment, while incurring only minor penalties in maximum lift and drag [40]. Thus, the reflexed airfoil is the best compromise for achieving high-lift with low coefficient of moment, and the convergence to this airfoil type demonstrates that the GA method presented in this thesis can effectively search this region of the inverse design space.

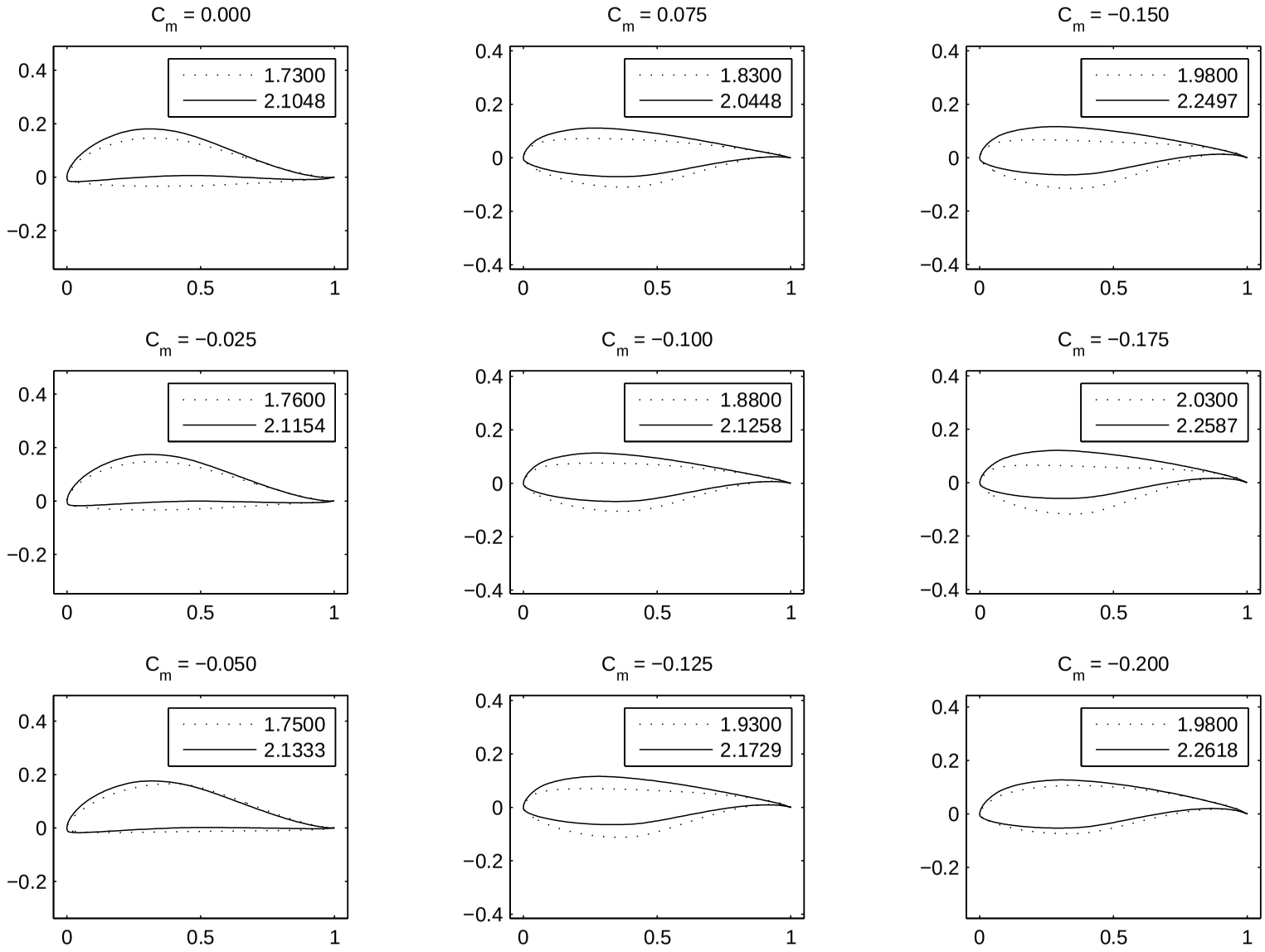

Figure 4.9: Solution evolution comparing the maximum fitness geometry of the final generation (solid) versus that of the first generation (dotted).

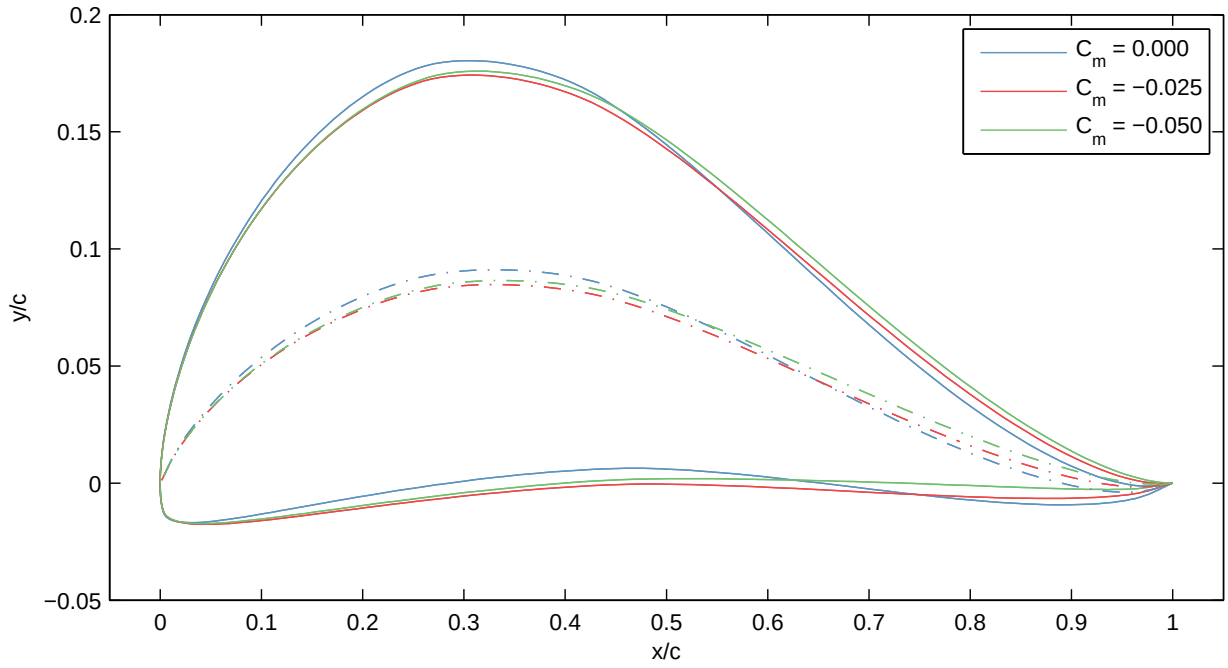

Figure 4.10: Airfoil overlays of final solution geometries designed at coefficients of moment: $C_m = 0.000$, -0.025, and -0.050.

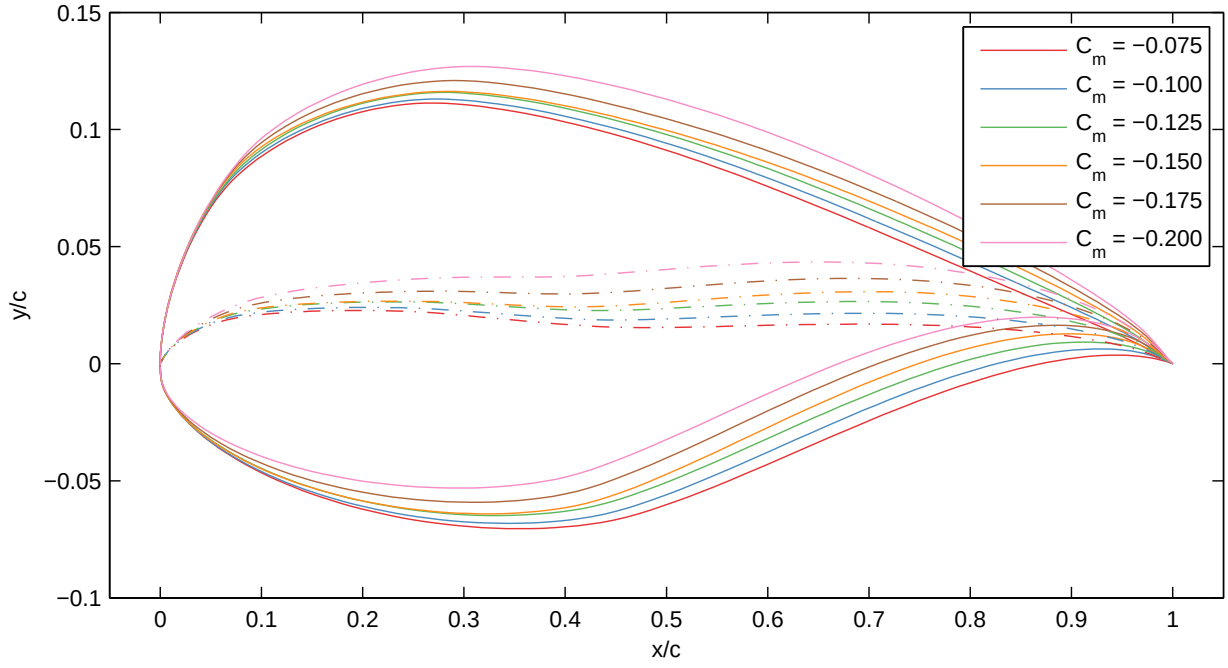

Figure 4.11: Airfoil overlays of final solution geometries designed at coefficients of moment: $C_m = -0.075$, -0.100, -0.125, -0.150, -0.175, and -0.200.

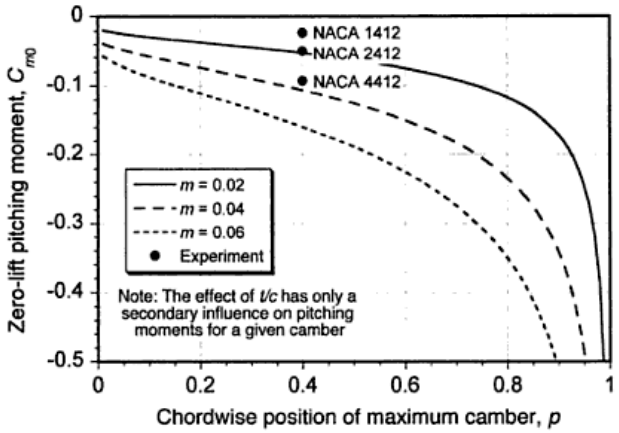

Figure 4.12: The effect of the chordwise position of maximum camber on the pitching moment of NACA 4-series airfoils (taken from Ref. [40]).

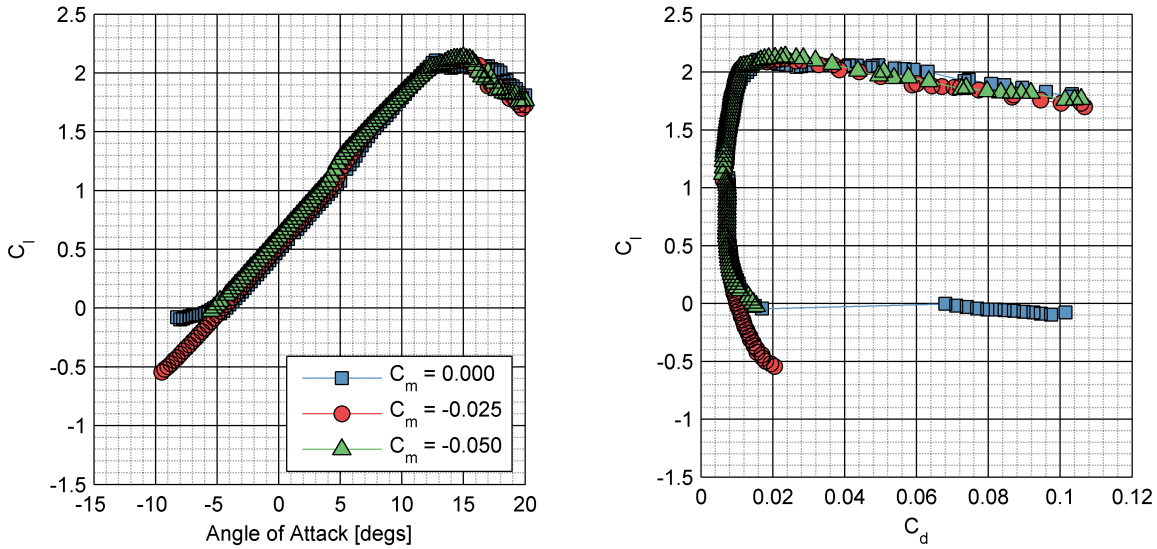

Figure 4.13: Drag polar and lift curve for $C_m = 0.000$, -0.025, and -0.050 computed from -10 to 20 deg in XFOIL.

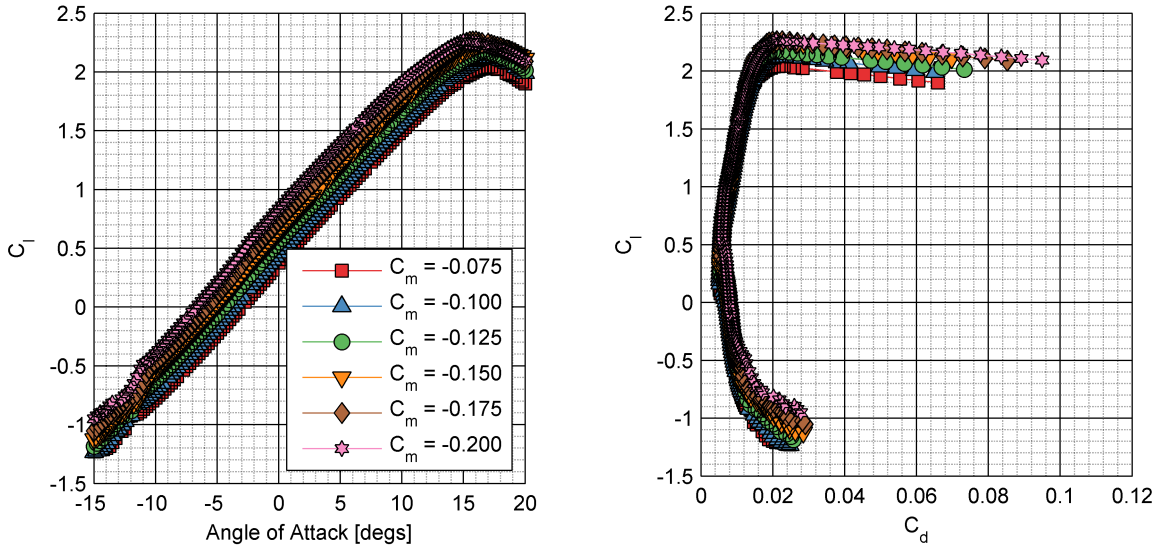

Figure 4.14: Drag polar and lift curve for $C_m = -0.075$, -0.100, -0.125, -0.150, -0.175, and -0.200 computed from -15 to 20 deg in XFOIL.

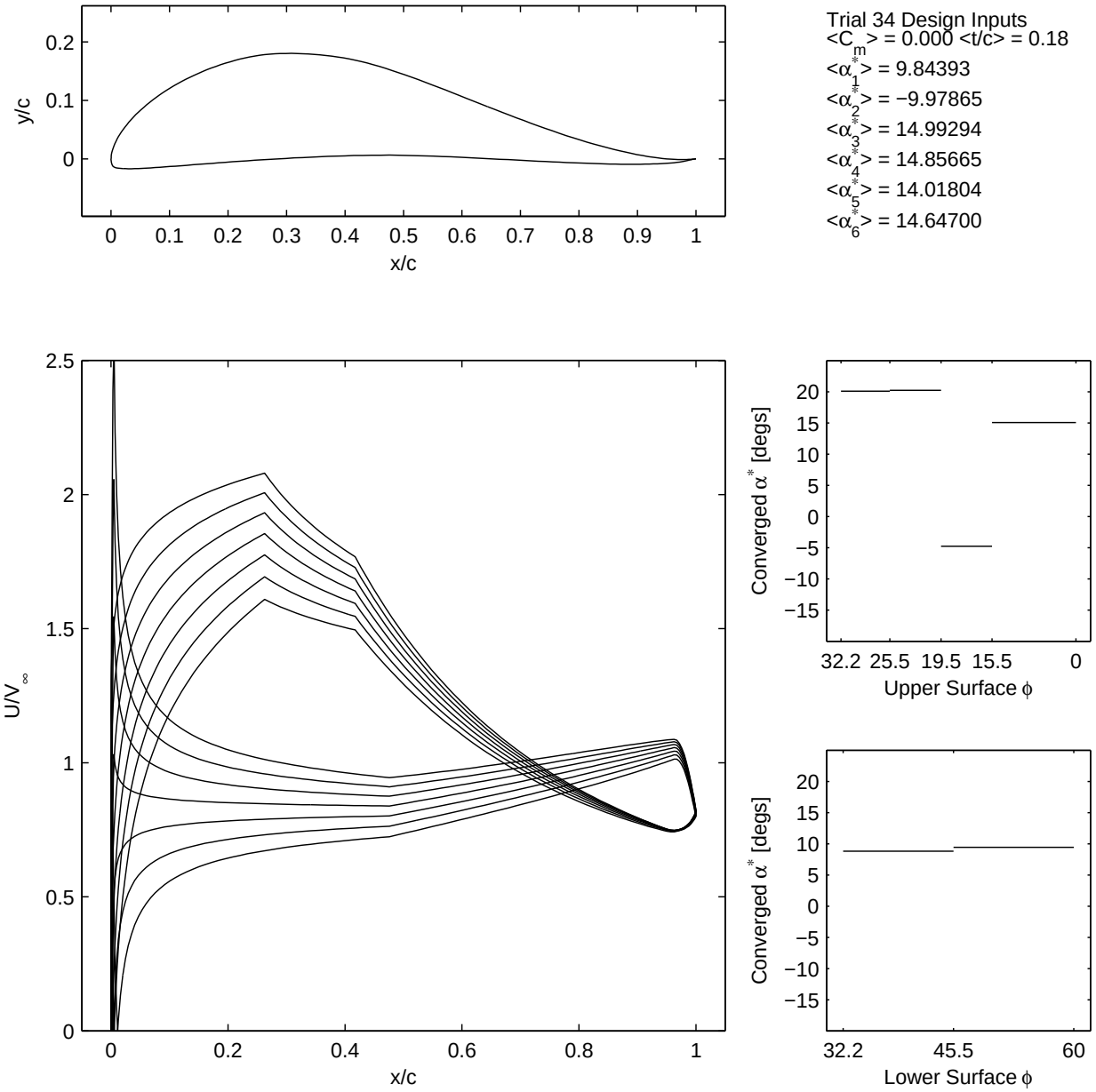

Figure 4.15: Trial 34 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

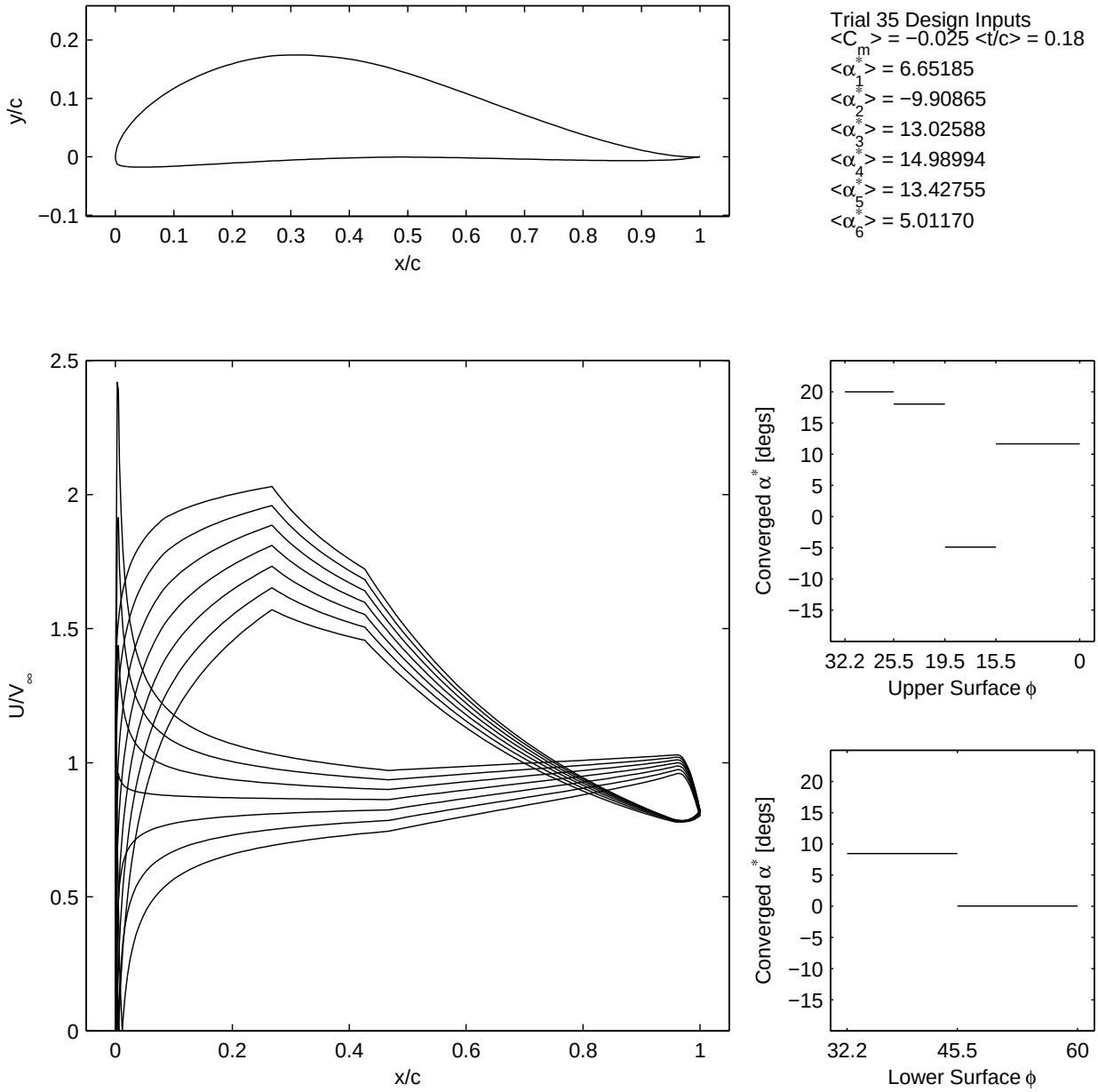

Figure 4.16: Trial 35 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

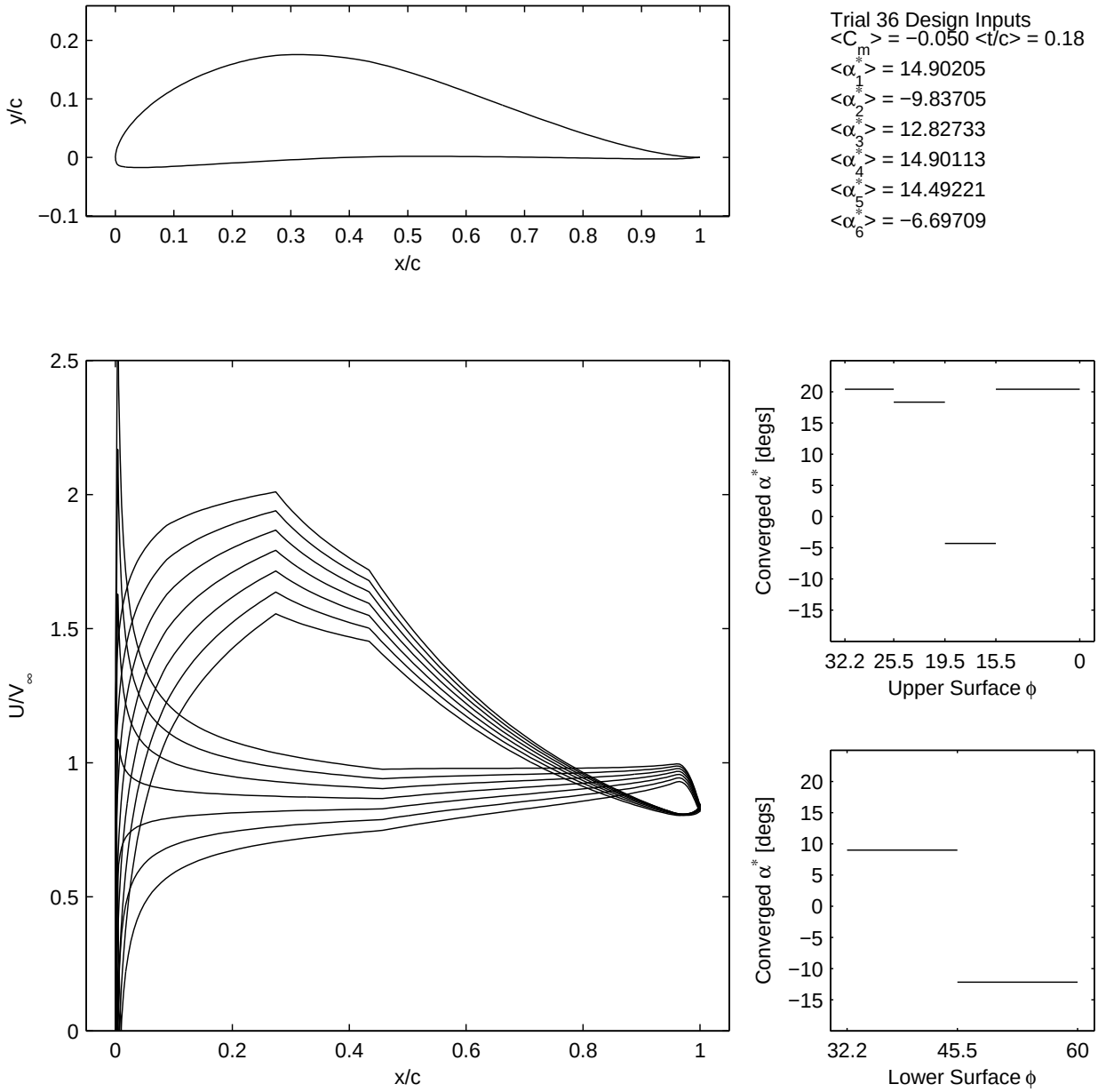

Figure 4.17: Trial 36 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

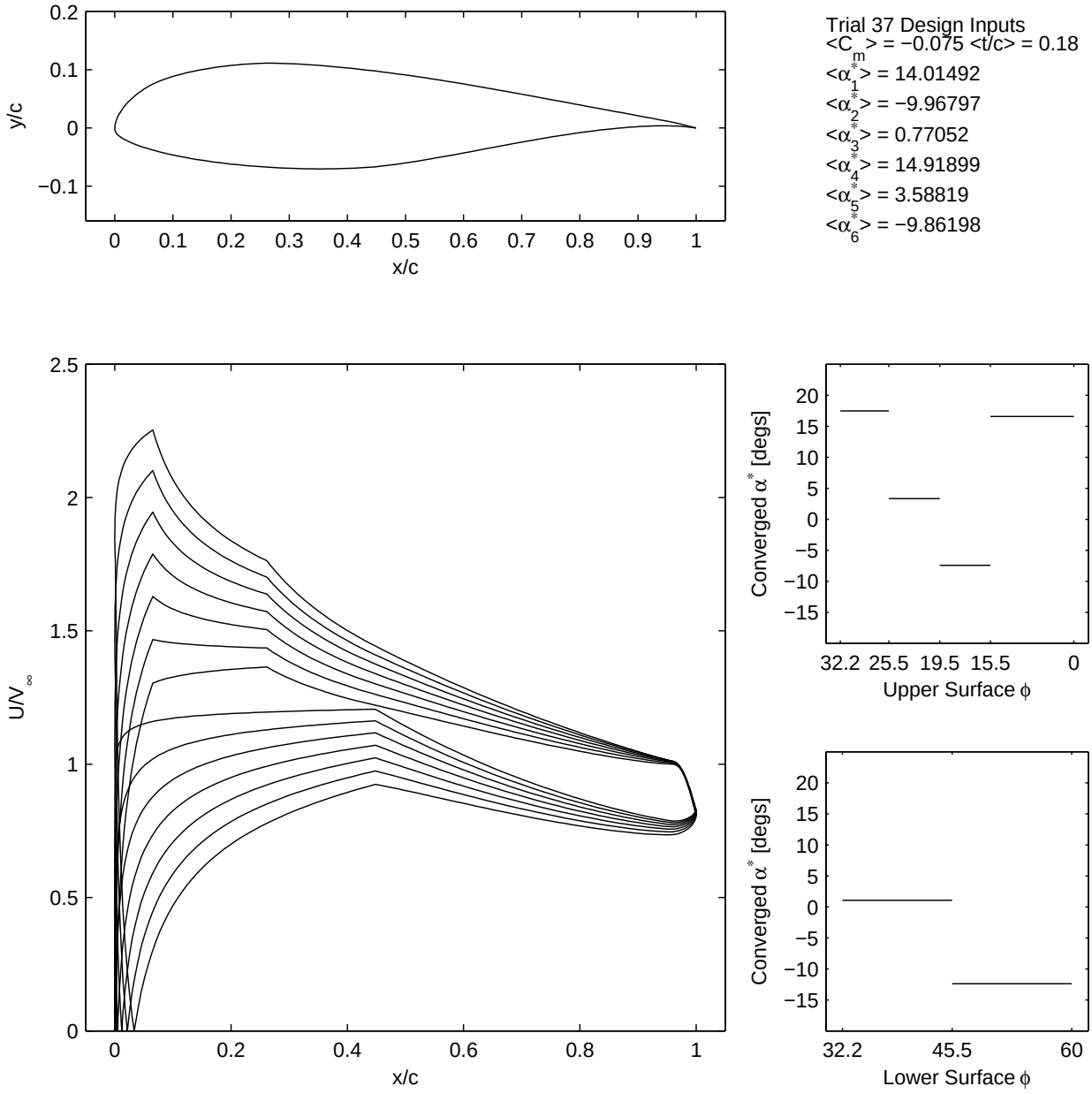

Figure 4.18: Trial 37 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

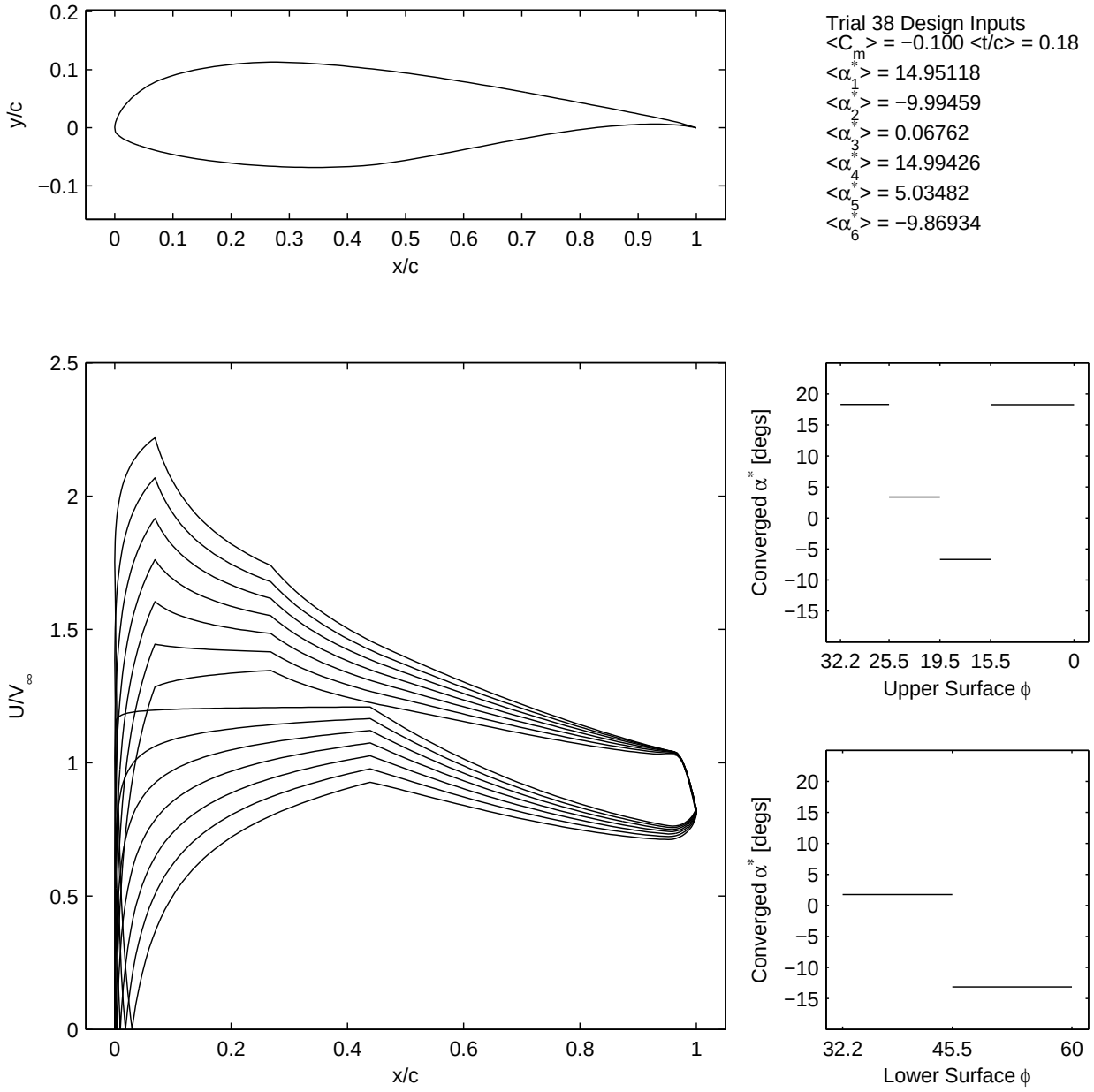

Figure 4.19: Trial 38 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

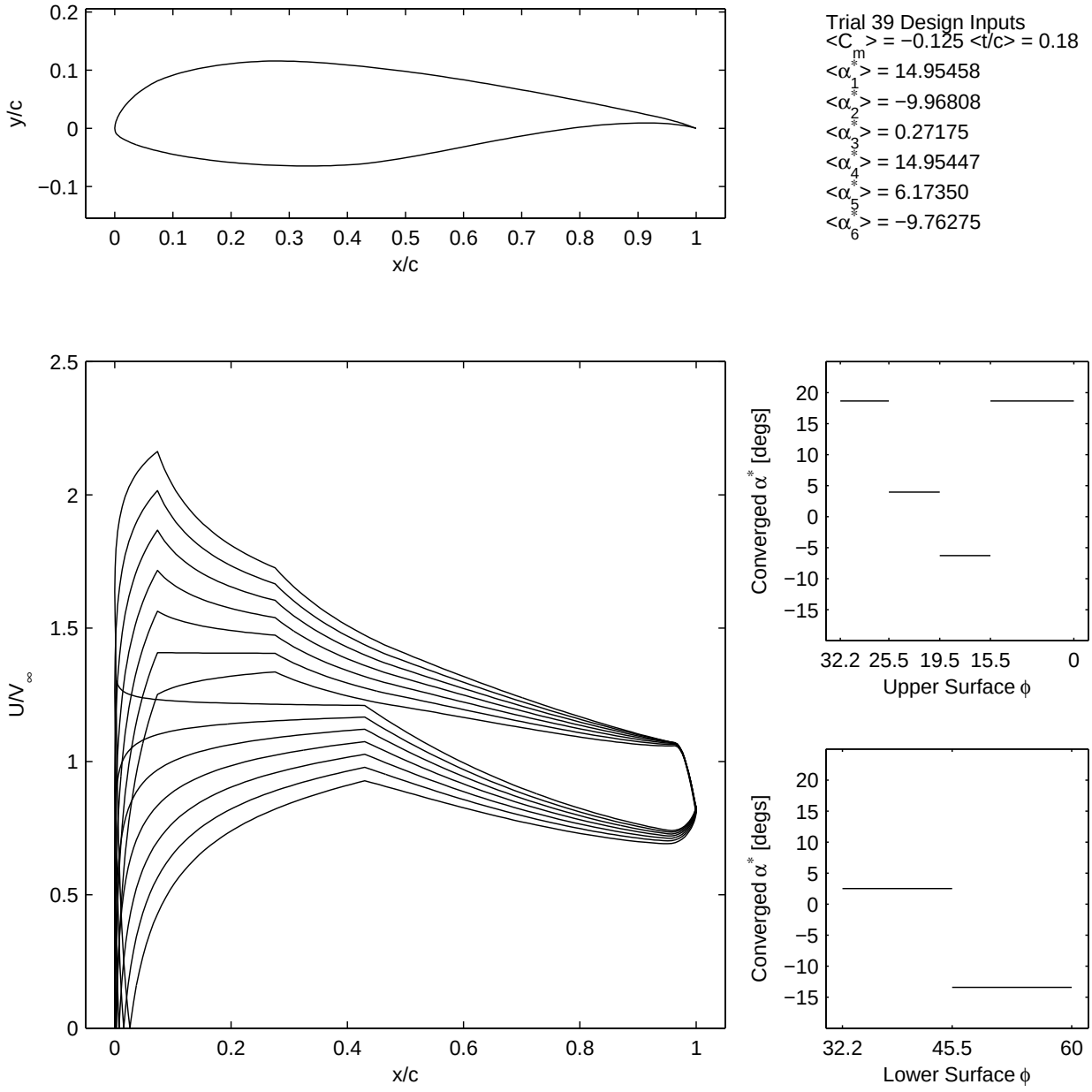

Figure 4.20: Trial 39 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

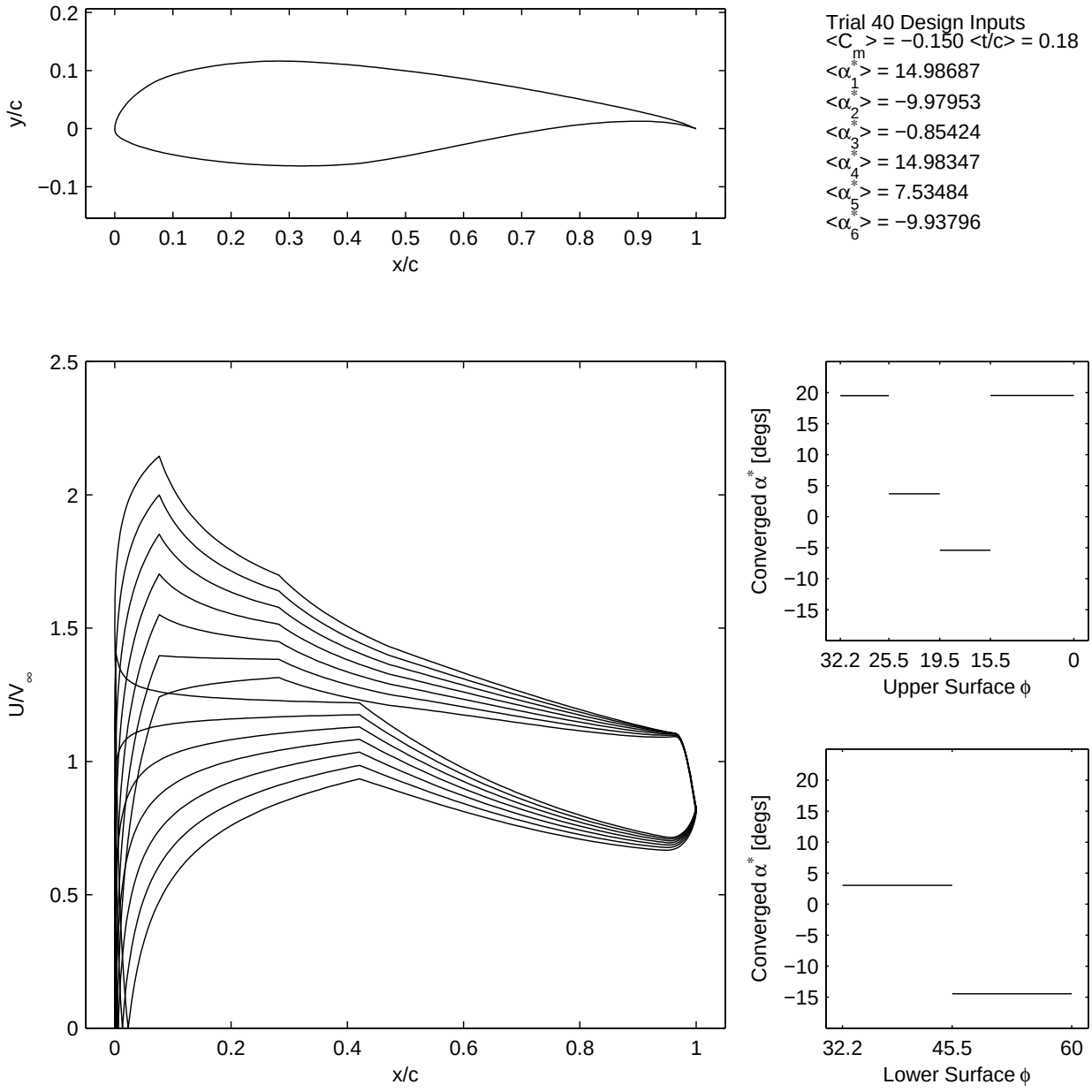

Figure 4.21: Trial 40 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

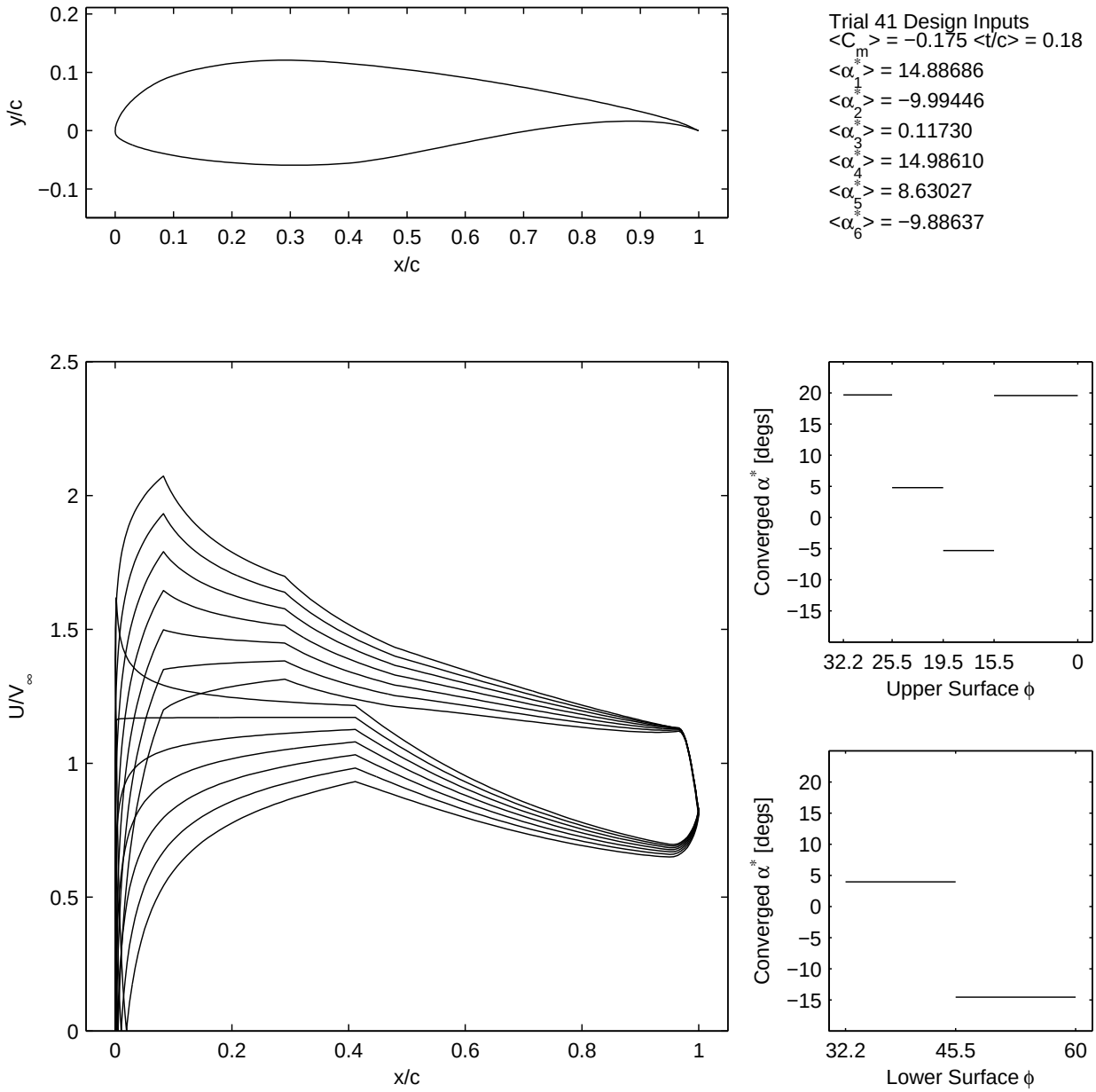

Figure 4.22: Trial 41 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

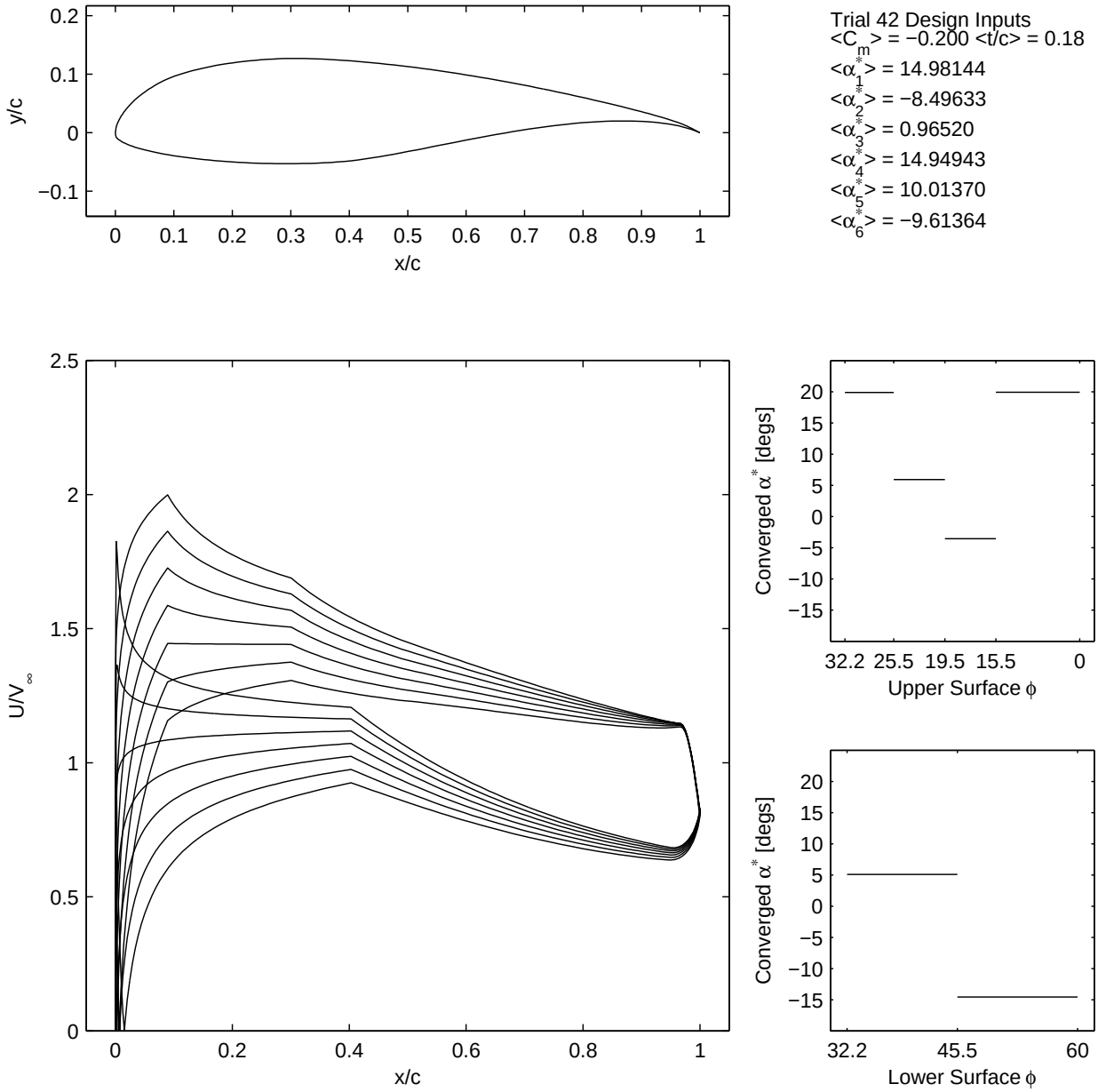

Figure 4.23: Trial 42 final solution design inputs, velocity distribution, and α^*-ϕ distribution.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this thesis, a process of optimizing cambered airfoils using a multipoint inverse method and a GA was implemented using MATLAB GA, PROFOIL, and XFOIL. Instrumental to its operation were various data processing measures that dealt with the prevention of optimizer interruptions, the handling of data processing exceptions, and the preservation of the integrity of the data. A static penalty function was designed to properly assign disadvantaged fitness values to infeasible individuals that could not be normally processed by the toolchain.

A total of 42 optimization trials across three studies achieved a tuned GA for a certain design condition, an airfoil was maximized for $(C_l/C_d)_{max}$, and a series of airfoils were maximized for $C_{l_{max}}$ at increasingly negative design coefficients of moment. GA tuning revealed that the heuristic crossover with heuristic ratio = 1.8 in combination with a 6 segment airfoil representation yields a good balance of maximum fitness and convergence speed for the design condition $C_m = -0.063$ and $t/c = 18\%$ at $Re = 6.88 \times 10^6$. In the process of varying the GA configuration, it was also found that the GA was able to consistently converge on the same airfoil despite different GA configurations. It was also shown that the 7 segment airfoil representation achieves the highest overall maximum fitness. In the second study, the airfoil maximized for $(C_l/C_d)_{max}$ at the design condition $C_m = -0.060$ and $t/c = 18\%$ at $Re = 6.00 \times 10^6$ achieved a $(C_l/C_d)_{max}$ of 315.08 after 79 generations. The airfoil shape and velocity distribution resembled that of the Liebeck L1003. For the third study, the series of airfoils optimized for $C_{l_{max}}$ at the design condition $t/c = 18\%$ and $Re = 6 \times 10^6$ showed that as the design coefficient of moment decreases, there is a trend

of increasing $C_{l_{max}}$, which is consistent with the results of Gopalarathnam and Selig. The results also demonstrate convergence towards reflexed airfoils for design pitching moments $C_m = 0.000$ through $C_m = -0.050$ and aft-loaded airfoils for design pitching moments $C_m - 0.075$ through $C_m = -0.200$. Analysis of the final solution geometries at each design pitching moment reveals that the optimizer successfully takes advantage of high camber to achieve high lift at aggressive nose-down pitching moments and reflexed camber for achieving high lift at more neutral pitching moments.

In conclusion, the GA coupled with a multipoint inverse method can be a powerful design tool for optimizing cambered airfoils. The convergence to an airfoil resembling the Liebeck L1003 in the second study, which was purposefully designed to sustain a Stratford pressure recovery distribution, and the convergence to a reflexed or aft-loaded airfoil depending on design pitching moment in the third study, shows that this approach can reproduce airfoils that were designed with central design philosophies using only C_m, t/c, and α^* design values.

However simple this process may be, it is important to note that without the designer, any black box airfoil optimizer is still insufficient in capturing the complexity of the real-world design process. Thus, this thesis makes no attempt to do so—its scope modest by design, but ambitious in the author's hope that this approach proves useful in aiding the designer to make decisions that will contribute to sustainable aviation for the decades to come.

5.2 Future Work

A feature that would have significantly increased computational performance is the parallelization of the fitness function call. Genetic algorithms are "embarrassingly parallel" in that they do not require communication or dependency between the potentially parallel tasks of evaluating the fitness of each individual. The method described in this thesis was implemented in a serial fashion, which does not take advantage of the multi-core processor prevalent in modern-day workstations. Implementing this feature would realize immediate computational efficiency gains.

A possible research direction is the application of this approach towards solar aircraft airfoil design. Advances in photovoltaic cells and high density battery technology have enabled

solar powered aircraft to achieve ultra-long endurances. It may be possible to augment their performance by performing multi-objective optimization for aerodynamic efficiency and solar energy absorption at specific latitudes or flight patterns.

REFERENCES

[1] Boeing. Current Market Outlook. Dec. 2015.

[2] ATAG. Reducing Emissions From Aviation Through Carbon-Neutral Growth from 2020. July 2013.

[3] David S. Lee. Aviation Greenhouse Gas Emissions. In *ICAO Environmental Report 2010*, pages 42–46. ICAO.

[4] Michael S. Selig. PROFOIL: A Multipoint Inverse Airfoil Design Method User's Guide. Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, Apr. 2014.

[5] Ben A. Gardner and Michael S. Selig. Airfoil Design Using a Genetic Algorithm and an Inverse Method. AIAA Paper 2003-0043, Reno, NV, Jan. 2003.

[6] Giovanni Fiore and Michael S. Selig. Optimization of Wind Turbine Airfoils Subject to Particle Erosion. AIAA Paper 2015-3393, Dallas, TX, June 2015.

[7] Shigeru Obayashi and Susumu Takanashi. Genetic Optimization of Target Pressure Distributions for Inverse Design Methods. *AIAA Journal*, 34(5):881—886, May 1996.

[8] Susumu Takanashi. Transonic Wing and Airfoil Design Using Inverse Code WINDES. *Special Publication of National Aerospace Laboratory SP-5*, NAL SP-5. Japan Aerospace Exploration Agency, Tokyo, Japan, Nov. 1985.

[9] Alessandro Vicini and Domenico Quagliarella. Inverse and Direct Airfoil Design Using a Multiobjective Genetic Algorithm. *AIAA Journal*, 35(9):1499–1505, Sep. 1997.

[10] Terry L. Holst and Thomas H. Pulliam. Aerodynamic Shape Optimization Using a Real-Number-Encoded Genetic Algorithm. AIAA Paper 2001-2473, Anaheim, CA, June 2001.

[11] Helmut Sobieczky. Parametric Airfoils and Wings. *Notes on Numerical Fluid Mechanics*, 68:71–88, 1998.

[12] Thomas. H. Pulliam. ARC2D–EFFICIENT SOLUTION METHODS FOR THE NAVIER-STOKES EQUATIONS (CRAY VERSION). Feb. 1999.

[13] Kisa Matsushima, Shigeru Obayashi, and Kozo Fujii. Navier-Stokes Computations of Transonic Flow Using the LU-ADI Method. AIAA Paper 87-0421, Jan. 1987.

[14] David L. Carroll. FORTRAN GENETIC ALGORITHM (GA) DRIVER. `http://www.cuaerospace.com/Technology/GeneticAlgorithm/GADriverFreeVersion.aspx`, Jan. 1994.

[15] Mark Drela. XFOIL Subsonic Airfoil Development System. `http://web.mit.edu/drela/Public/web/xfoil/`, Dec. 2013.

[16] Mitchell Melanie. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA, 1996.

[17] John H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA, 1992.

[18] MATLAB. *version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, MA, 2010.

[19] Agoston E. Eiben and James. E. Smith. *Introduction to Evolutionary Computing*. Springer-Verlag, Berlin, Germany, 2003.

[20] MATLAB. Genetic Algorithm Options. `http://www.mathworks.com/help/gads/genetic-algorithm-options.html`.

[21] Alice E. Smith and David W. Coit. *Handbook of Evolutionary Computation*. Institute of Physics Publishing and Oxford University Press, Bristol, U.K., 1997.

[22] Wojciech W. Siedlecki and Jack Sklansky. Constrained Genetic Optimization via Dynamic Reward-penalty Balancing and Its Use in Pattern Recognition. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 141–150, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.

[23] Edward J. Anderson and Michael C. Ferris. Genetic Algorithms for Combinatorial Optimization: The Assemble Line Balancing Problem. *ORSA Journal on Computing*, 6(2):161–173, 1994.

[24] David W. Coit, Alice E. Smith, and David M. Tate. Adaptive Penalty Methods For Genetic Optimization Of Constrained Combinatorial Problems. *INFORMS Journal on Computing*, 8:173–182, 1996.

[25] Zbigniew Michalewicz. Genetic Algorithms, Numerical Optimization, and Constraints. pages 151–158. Morgan Kaufmann Publishers Inc., 1995.

[26] Zbigniew Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, New York, NY, 1994.

[27] Francisco Herrera, Manuel Lozano, and Ana M. Sánchez. A Taxonomy for the Crossover Operator for Real-Coded Genetic Algorithms: An Experimental Study. *International Journal of Intelligent Systems*, 18, 2003.

[28] Michael S. Selig and Mark D. Maughmer. Multipoint Inverse Airfoil Design Method Based on Conformal Mapping. *AIAA Journal*, 30(5), May 1992.

[29] Richard Eppler. Direkte Berrechnung von Tragflügelprofilen aus der Druckverteilung. Ingenieur-Archive 25(1):32–57, 1957. Translated as "Direct Calculation of Airfoils from Pressure Distribution," NASA TT F-15, 417, Mar. 1974.

[30] Richard Eppler. *Airfoil Design and Data*. Springer-Verlag, New York, 1990.

[31] John D. Anderson. *Fundamentals of Aerodynamics*. Aeronautical and Aerospace Engineering Series. McGraw-Hill, 2nd edition, 2001.

[32] Barnes W. McCormick. *Aerodynamics, Aeronautics, and Flight Mechanics*. Wiley, New York, NY, 2nd edition, 1994.

[33] Robert H. Liebeck. Design of Subsonic Airfoils for High Lift. *Journal of Aircraft*, 15(9):547–561, Sep. 1978.

[34] Alejandro C. Magencio. How to disable the pop up that Windows shows when an app crashes. `http://blogs.msdn.com/b/alejacma/archive/2011/02/18/how-to-disable-the-pop-up-that-windows-shows-when-an-app-crashes.aspx`, Feb. 2011.

[35] Rafael Oliveira. XFOIL–MATLAB Interface. `http://www.mathworks.com/matlabcentral/fileexchange/30478-xfoil-matlab-interface`, Feb. 2011.

[36] Antoni J. Canós. Fast and Robust Self-Intersections. `http://www.mathworks.com/matlabcentral/fileexchange/13351-fast-and-robust-self-intersections/content/selfintersect.m`, Dec. 2006.

[37] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A Fast and Elitist Multi-Objective Genetic Algorithm-NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), Apr. 2002.

[38] B. S. Stratford. The Prediction of Separation of the Turbulent Boundary Layer. *Journal of Fluid Mechanics*, 5(01):1–16, 1 1959.

[39] Ashok Gopalarathnam and Michael S. Selig. Low-Speed Natural-Laminar-Flow Airfoils: Case Study in Inverse Airfoil Design. *Journal of Aircraft*, 38(1):57–63, Jan.–Feb. 2001.

[40] J. Gordon Leishman. *Principles of Helicopter Aerodynamics*. Cambridge Aerospace Series. Cambridge University Press, New York, NY, 2nd edition, 2006.

[41] Martin Hepperle. Reflex and Moment Coefficient. `http://www.mh-aerotools.de/airfoils/nf_3.htm`, 2008.

APPENDIX A

PROFOIL CONFIGURATION FILES FOR TRIALS 33–42

Listing A.1: Trial 33 PROFOIL Configuration File.

```
1     TRIAL 33
2     COORD 240
3     FOIL 15.500000 13.237814 1
4     FOIL 19.500000 12.477810 2
5     FOIL 25.500000 12.093370 3
6     FOIL 32.200000 14.492335 4
7     FOIL 45.500000 9.700953 5
8     FOIL 60.000000 5.390243 6
9     PHIS 3.500000 56.500000
10    REC  0.020000 0.020000
11    VLEV 1 1.527280
12    ILE  4
13    TOLSPEC 0.00001
14    ITERMAX 30
15    NEWT1G0 100 0.350 1 4
16    IDES
17    NEWT1G0 101 -0.060000 4 1
18    IDES
19    NEWT1G0 102 0.180000 6 300 .5
20    IDES
21    FINISH 100
22    ALFASP 7
23    2
24    4
25    6
26    8
27    10
28    12
29    14
30    VELDIST 50 60
31    DUMP
32    AIRFOIL GA
33    *
34
```

Listing A.2: Trial 34 PROFOIL Configuration File.

```
1     TRIAL 34
2     COORD 240
3     FOIL 15.500000 9.843931 1
4     FOIL 19.500000 -9.978654 2
5     FOIL 25.500000 14.992944 3
6     FOIL 32.200000 14.856654 4
7     FOIL 45.500000 14.018041 5
8     FOIL 60.000000 14.647002 6
9     PHIS 3.500000 56.500000
10    REC  0.020000 0.020000
11    VLEV 1 1.527280
12    ILE  4
13    TOLSPEC 0.00001
14    ITERMAX 30
15    NEWT1G0 100 0.350 1 4
16    IDES
17    NEWT1G0 101 0.000000 4 1
18    IDES
19    NEWT1G0 102 0.180000 6 300 .5
20    IDES
21    FINISH 100
22    ALFASP 7
23    2
24    4
25    6
26    8
27    10
28    12
29    14
30    VELDIST 50 60
31    DUMP
32    AIRFOIL GA
33    *
34
35
```

Listing A.3: Trial 35 PROFOIL Configuration File.

```
TRIAL 35
COORD 240
FOIL 15.500000 6.651850 1
FOIL 19.500000 -9.908647 2
FOIL 25.500000 13.025875 3
FOIL 32.200000 14.989939 4
FOIL 45.500000 13.427552 5
FOIL 60.000000 5.011704 6
PHIS 3.500000 56.500000
REC  0.020000 0.020000
VLEV 1 1.527280
ILE  4
TOLSPEC 0.00001
ITERMAX 30
NEWT1G0 100 0.350 1 4
IDES
NEWT1G0 101 -0.025000 4 1
IDES
NEWT1G0 102 0.180000 6 300 .5
IDES
FINISH 100
ALFASP 7
2
4
6
8
10
12
14
VELDIST 50 60
DUMP
AIRFOIL GA
*


```

Listing A.4: Trial 36 PROFOIL Configuration File.

```
1     TRIAL 36
2     COORD 240
3     FOIL 15.500000 14.902052 1
4     FOIL 19.500000 -9.837049 2
5     FOIL 25.500000 12.827329 3
6     FOIL 32.200000 14.901132 4
7     FOIL 45.500000 14.492207 5
8     FOIL 60.000000 -6.697094 6
9     PHIS 3.500000 56.500000
10    REC  0.020000 0.020000
11    VLEV 1 1.527280
12    ILE  4
13    TOLSPEC 0.00001
14    ITERMAX 30
15    NEWT1G0 100 0.350 1 4
16    IDES
17    NEWT1G0 101 -0.050000 4 1
18    IDES
19    NEWT1G0 102 0.180000 6 300 .5
20    IDES
21    FINISH 100
22    ALFASP 7
23    2
24    4
25    6
26    8
27    10
28    12
29    14
30    VELDIST 50 60
31    DUMP
32    AIRFOIL GA
33    *
34
35
```

Listing A.5: Trial 37 PROFOIL Configuration File.

```
TRIAL 37
COORD 240
FOIL 15.500000 14.014918 1
FOIL 19.500000 -9.967975 2
FOIL 25.500000 0.770516 3
FOIL 32.200000 14.918989 4
FOIL 45.500000 3.588191 5
FOIL 60.000000 -9.861979 6
PHIS 3.500000 56.500000
REC  0.020000 0.020000
VLEV 1 1.527280
ILE  4
TOLSPEC 0.00001
ITERMAX 30
NEWT1G0 100 0.350 1 4
IDES
NEWT1G0 101 -0.075000 4 1
IDES
NEWT1G0 102 0.180000 6 300 .5
IDES
FINISH 100
ALFASP 7
2
4
6
8
10
12
14
VELDIST 50 60
DUMP
AIRFOIL GA
*


```

Listing A.6: Trial 38 PROFOIL Configuration File.

```
TRIAL 38
COORD 240
FOIL 15.500000 14.951185 1
FOIL 19.500000 -9.994586 2
FOIL 25.500000 0.067615 3
FOIL 32.200000 14.994256 4
FOIL 45.500000 5.034816 5
FOIL 60.000000 -9.869344 6
PHIS 3.500000 56.500000
REC  0.020000 0.020000
VLEV 1 1.527280
ILE  4
TOLSPEC 0.00001
ITERMAX 30
NEWT1G0 100 0.350 1 4
IDES
NEWT1G0 101 -0.100000 4 1
IDES
NEWT1G0 102 0.180000 6 300 .5
IDES
FINISH 100
ALFASP 7
2
4
6
8
10
12
14
VELDIST 50 60
DUMP
AIRFOIL GA
*


```

Listing A.7: Trial 39 PROFOIL Configuration File.

```
TRIAL 39
COORD 240
FOIL 15.500000 14.954582 1
FOIL 19.500000 -9.968077 2
FOIL 25.500000 0.271751 3
FOIL 32.200000 14.954474 4
FOIL 45.500000 6.173503 5
FOIL 60.000000 -9.762745 6
PHIS 3.500000 56.500000
REC  0.020000 0.020000
VLEV 1 1.527280
ILE  4
TOLSPEC 0.00001
ITERMAX 30
NEWT1G0 100 0.350 1 4
IDES
NEWT1G0 101 -0.125000 4 1
IDES
NEWT1G0 102 0.180000 6 300 .5
IDES
FINISH 100
ALFASP 7
2
4
6
8
10
12
14
VELDIST 50 60
DUMP
AIRFOIL GA
*


```

Listing A.8: Trial 40 PROFOIL Configuration File.

```
1      TRIAL 40
2      COORD 240
3      FOIL 15.500000 14.886864 1
4      FOIL 19.500000 -9.994456 2
5      FOIL 25.500000 0.117304 3
6      FOIL 32.200000 14.986095 4
7      FOIL 45.500000 8.630271 5
8      FOIL 60.000000 -9.886373 6
9      PHIS 3.500000 56.500000
10     REC  0.020000 0.020000
11     VLEV 1 1.527280
12     ILE  4
13     TOLSPEC 0.00001
14     ITERMAX 30
15     NEWT1G0 100 0.350 1 4
16     IDES
17     NEWT1G0 101 -0.175000 4 1
18     IDES
19     NEWT1G0 102 0.180000 6 300 .5
20     IDES
21     FINISH 100
22     ALFASP 7
23     2
24     4
25     6
26     8
27     10
28     12
29     14
30     VELDIST 50 60
31     DUMP
32     AIRFOIL GA
33     *
34
35
```

Listing A.9: Trial 41 PROFOIL Configuration File.

```
TRIAL 41
COORD 240
FOIL 15.500000 14.926893 1
FOIL 19.500000 -9.969060 2
FOIL 25.500000 0.089597 3
FOIL 32.200000 14.984141 4
FOIL 45.500000 8.638252 5
FOIL 60.000000 -9.847158 6
PHIS 3.500000 56.500000
REC  0.020000 0.020000
VLEV 1 1.527280
ILE  4
TOLSPEC 0.00001
ITERMAX 30
NEWT1G0 100 0.350 1 4
IDES
NEWT1G0 101 -0.175000 4 1
IDES
NEWT1G0 102 0.180000 6 300 .5
IDES
FINISH 100
ALFASP 7
2
4
6
8
10
12
14
VELDIST 50 60
DUMP
AIRFOIL GA
*


```

Listing A.10: Trial 42 PROFOIL Configuration File.

```
1       TRIAL 42
2       COORD 240
3       FOIL 15.500000 14.981439 1
4       FOIL 19.500000 -8.496330 2
5       FOIL 25.500000 0.965199 3
6       FOIL 32.200000 14.949429 4
7       FOIL 45.500000 10.013697 5
8       FOIL 60.000000 -9.613640 6
9       PHIS 3.500000 56.500000
10      REC  0.020000 0.020000
11      VLEV 1 1.527280
12      ILE  4
13      TOLSPEC 0.00001
14      ITERMAX 30
15      NEWT1G0 100 0.350 1 4
16      IDES
17      NEWT1G0 101 -0.200000 4 1
18      IDES
19      NEWT1G0 102 0.180000 6 300 .5
20      IDES
21      FINISH 100
22      ALFASP 7
23      2
24      4
25      6
26      8
27      10
28      12
29      14
30      VELDIST 50 60
31      DUMP
32      AIRFOIL GA
33      *
34
35
```

APPENDIX B

POLAR FILES FOR TRIALS 33–42

Listing B.1: Trial 33 XFOIL Polar File.

```
1
2     XFOIL         Version 6.99
3
4     Calculated polar for: TRIAL 33
5
6     1 1 Reynolds number fixed          Mach number fixed
7
8     xtrf =   1.000 (top)        1.000 (bottom)
9     Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11    alpha    CL         CD       CDp       CM      Top_Xtr  Bot_Xtr
12    ------ -------- --------- --------- -------- -------- --------
13    0.000   0.6061   0.00648   0.00162  -0.0813   0.4964   0.0060
14    0.250   0.6357   0.00643   0.00158  -0.0818   0.4962   0.0063
15    0.500   0.6654   0.00640   0.00156  -0.0823   0.4960   0.0067
16    0.750   0.6951   0.00637   0.00154  -0.0829   0.4958   0.0069
17    1.000   0.7247   0.00633   0.00152  -0.0834   0.4955   0.0092
18    1.250   0.7542   0.00629   0.00151  -0.0840   0.4953   0.0172
19    1.500   0.7835   0.00623   0.00151  -0.0845   0.4951   0.0368
20    1.750   0.8124   0.00613   0.00153  -0.0849   0.4949   0.0812
21    2.000   0.8411   0.00603   0.00155  -0.0854   0.4946   0.1424
22    2.250   0.8689   0.00586   0.00159  -0.0857   0.4944   0.2475
23    2.500   0.8952   0.00558   0.00166  -0.0858   0.4942   0.4157
24    2.750   0.9075   0.00466   0.00176  -0.0831   0.4939   0.8621
25    3.000   0.9571   0.00459   0.00197  -0.0880   0.4936   0.9885
26    3.250   0.9862   0.00463   0.00200  -0.0885   0.4934   0.9889
27    3.500   1.0415   0.00475   0.00209  -0.0948   0.4930   0.9921
28    3.750   1.0836   0.00482   0.00215  -0.0983   0.4925   0.9932
29    4.000   1.1228   0.00488   0.00220  -0.1011   0.4921   0.9942
30    4.250   1.1592   0.00496   0.00227  -0.1033   0.4916   0.9954
31    4.500   1.1943   0.00504   0.00235  -0.1053   0.4911   0.9966
32    4.750   1.2344   0.00506   0.00236  -0.1083   0.4908   0.9971
33    5.000   1.2719   0.00506   0.00236  -0.1108   0.4907   0.9977
34    5.250   1.3082   0.00506   0.00236  -0.1130   0.4904   0.9984
35    5.500   1.3420   0.00505   0.00236  -0.1147   0.4898   0.9989
36    5.750   1.3737   0.00505   0.00236  -0.1160   0.4890   0.9992
37    6.000   1.4049   0.00505   0.00236  -0.1171   0.4878   0.9994
38    6.250   1.4361   0.00506   0.00238  -0.1183   0.4865   0.9997
39    6.500   1.4672   0.00510   0.00241  -0.1195   0.4850   0.9999
40    6.750   1.4972   0.00519   0.00248  -0.1205   0.4829   1.0000
41    7.000   1.5253   0.00524   0.00254  -0.1211   0.4819   1.0000
42    7.250   1.5538   0.00526   0.00258  -0.1218   0.4812   1.0000
43    7.500   1.5820   0.00529   0.00262  -0.1224   0.4803   1.0000
44    7.750   1.6100   0.00533   0.00268  -0.1230   0.4796   1.0000
45
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
8.000	1.6381	0.00537	0.00273	-0.1236	0.4788	1.0000
8.250	1.6659	0.00543	0.00280	-0.1242	0.4779	1.0000
8.500	1.6938	0.00549	0.00287	-0.1248	0.4769	1.0000
8.750	1.7214	0.00556	0.00296	-0.1254	0.4758	1.0000
9.000	1.7489	0.00565	0.00305	-0.1260	0.4745	1.0000
9.250	1.7763	0.00574	0.00315	-0.1266	0.4734	1.0000
9.500	1.8040	0.00577	0.00322	-0.1272	0.4725	1.0000
9.750	1.8316	0.00583	0.00329	-0.1278	0.4710	1.0000
10.000	1.8589	0.00590	0.00338	-0.1284	0.4686	1.0000
10.250	1.8855	0.00603	0.00350	-0.1289	0.4641	1.0000
10.500	1.9093	0.00637	0.00377	-0.1291	0.4463	1.0000
10.750	1.9161	0.00784	0.00498	-0.1272	0.4016	1.0000
11.000	1.8999	0.01010	0.00703	-0.1224	0.3589	1.0000
11.250	1.7995	0.01618	0.01302	-0.1082	0.3241	1.0000
11.500	1.7926	0.01825	0.01511	-0.1055	0.3185	1.0000
11.750	1.7835	0.02056	0.01744	-0.1028	0.3135	1.0000
12.000	1.6668	0.03074	0.02754	-0.0919	0.2771	1.0000
12.250	1.7123	0.02926	0.02615	-0.0937	0.2890	1.0000
12.500	1.6373	0.03719	0.03400	-0.0870	0.2611	1.0000
12.750	1.6677	0.03682	0.03371	-0.0877	0.2693	1.0000
13.000	1.6140	0.04334	0.04013	-0.0830	0.2427	1.0000
13.250	1.5785	0.04853	0.04519	-0.0796	0.2170	1.0000
13.500	1.5717	0.05132	0.04794	-0.0781	0.2062	1.0000
13.750	1.5771	0.05310	0.04972	-0.0773	0.2007	1.0000
14.000	1.5831	0.05480	0.05143	-0.0766	0.1979	1.0000
14.250	1.5263	0.06224	0.05851	-0.0726	0.1420	1.0000
14.500	1.5216	0.06504	0.06122	-0.0715	0.1262	1.0000
14.750	1.5439	0.06532	0.06159	-0.0718	0.1360	1.0000
15.000	1.5614	0.06602	0.06236	-0.0718	0.1413	1.0000
0.000	0.6061	0.00648	0.00162	-0.0813	0.4964	0.0060
-0.250	0.5766	0.00656	0.00170	-0.0809	0.4966	0.0056
-0.500	0.5469	0.00660	0.00174	-0.0804	0.4968	0.0056
-0.750	0.5173	0.00667	0.00180	-0.0799	0.4970	0.0055
-1.000	0.4876	0.00672	0.00185	-0.0794	0.4972	0.0053
-1.250	0.4580	0.00679	0.00191	-0.0790	0.4974	0.0052
-1.500	0.4285	0.00688	0.00199	-0.0786	0.4975	0.0046
-1.750	0.3989	0.00697	0.00208	-0.0781	0.4977	0.0046
-2.000	0.3702	0.00720	0.00234	-0.0779	0.4979	0.0044
-2.250	0.3420	0.00752	0.00272	-0.0779	0.4981	0.0043
-2.500	0.3128	0.00770	0.00292	-0.0776	0.4983	0.0043
-2.750	0.2838	0.00789	0.00313	-0.0773	0.4985	0.0043
-3.000	0.2547	0.00810	0.00337	-0.0770	0.4986	0.0042
-3.250	0.2261	0.00838	0.00368	-0.0768	0.4988	0.0042
-3.500	0.1975	0.00871	0.00405	-0.0767	0.4990	0.0042
-3.750	0.1689	0.00901	0.00439	-0.0765	0.4992	0.0042
-4.000	0.1413	0.00956	0.00504	-0.0766	0.4993	0.0042

Listing B.2: Trial 34 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 34
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.4964   0.00805   0.00308  -0.0268   0.3815   0.0080
14      0.250   0.5261   0.00792   0.00290  -0.0271   0.3773   0.0080
15      0.500   0.5561   0.00786   0.00282  -0.0274   0.3732   0.0080
16      0.750   0.5859   0.00777   0.00271  -0.0278   0.3701   0.0080
17      1.250   0.6452   0.00762   0.00250  -0.0285   0.3633   0.0081
18      1.500   0.6750   0.00759   0.00245  -0.0288   0.3607   0.0081
19      1.750   0.7048   0.00758   0.00241  -0.0292   0.3575   0.0082
20      2.000   0.7346   0.00758   0.00240  -0.0297   0.3544   0.0082
21      2.250   0.7644   0.00757   0.00238  -0.0301   0.3527   0.0083
22      2.500   0.7942   0.00758   0.00239  -0.0305   0.3505   0.0085
23      2.750   0.8239   0.00763   0.00243  -0.0310   0.3479   0.0089
24      3.000   0.8536   0.00767   0.00247  -0.0315   0.3451   0.0101
25      3.250   0.8833   0.00769   0.00250  -0.0320   0.3439   0.0113
26      3.500   0.9128   0.00771   0.00255  -0.0325   0.3424   0.0175
27      3.750   0.9422   0.00775   0.00261  -0.0330   0.3406   0.0288
28      4.000   0.9714   0.00779   0.00270  -0.0335   0.3386   0.0520
29      4.250   1.0001   0.00781   0.00280  -0.0339   0.3360   0.0966
30      4.500   1.0285   0.00778   0.00288  -0.0343   0.3351   0.1579
31      4.750   1.0563   0.00771   0.00298  -0.0345   0.3342   0.2530
32      5.000   1.0827   0.00761   0.00312  -0.0346   0.3330   0.3948
33      5.500   1.1840   0.00703   0.00378  -0.0452   0.3300   0.9860
34      5.750   1.2221   0.00725   0.00397  -0.0477   0.3282   0.9878
35      6.000   1.2549   0.00747   0.00418  -0.0492   0.3263   0.9896
36      6.250   1.2975   0.00756   0.00426  -0.0526   0.3258   0.9900
37      6.500   1.3426   0.00766   0.00435  -0.0567   0.3252   0.9905
38      6.750   1.3845   0.00776   0.00444  -0.0600   0.3244   0.9910
39      7.000   1.4242   0.00787   0.00454  -0.0630   0.3235   0.9917
40      7.250   1.4632   0.00799   0.00465  -0.0658   0.3226   0.9925
41      7.500   1.4982   0.00813   0.00478  -0.0679   0.3215   0.9933
42      7.750   1.5296   0.00830   0.00494  -0.0692   0.3203   0.9938
43      8.000   1.5596   0.00847   0.00511  -0.0703   0.3190   0.9943
44      8.250   1.5882   0.00866   0.00529  -0.0711   0.3178   0.9946
45      8.500   1.6162   0.00885   0.00548  -0.0718   0.3169   0.9949
46      8.750   1.6428   0.00906   0.00570  -0.0722   0.3166   0.9959
47      9.000   1.6728   0.00916   0.00582  -0.0733   0.3163   0.9960
48      9.250   1.7027   0.00928   0.00595  -0.0744   0.3160   0.9961
49      9.500   1.7325   0.00942   0.00610  -0.0756   0.3156   0.9962
50      9.750   1.7627   0.00956   0.00625  -0.0769   0.3152   0.9963
51      10.000   1.7930   0.00973   0.00643  -0.0783   0.3146   0.9964
52      10.250   1.8231   0.00993   0.00664  -0.0797   0.3140   0.9966
53      10.500   1.8530   0.01015   0.00687  -0.0811   0.3134   0.9967
54      10.750   1.8822   0.01040   0.00713  -0.0825   0.3128   0.9969
55      11.000   1.9112   0.01069   0.00745  -0.0839   0.3122   0.9971
56      11.250   1.9410   0.01115   0.00791  -0.0859   0.3114   0.9973
57      11.500   1.9714   0.01194   0.00873  -0.0889   0.3099   0.9976
58      11.750   1.9996   0.01289   0.00970  -0.0916   0.3084   0.9981
59      12.000   2.0260   0.01372   0.01055  -0.0937   0.3077   0.9986
60      12.250   2.0524   0.01433   0.01119  -0.0954   0.3073   0.9991
61      12.500   2.0837   0.01486   0.01175  -0.0980   0.3067   0.9998
62      12.750   2.1048   0.01570   0.01262  -0.0991   0.3059   1.0000
63
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.0853	0.01758	0.01456	-0.0944	0.3050	1.0000
13.250	2.0776	0.01939	0.01640	-0.0918	0.3038	1.0000
13.500	2.0684	0.02151	0.01856	-0.0895	0.3022	1.0000
13.750	2.0602	0.02369	0.02077	-0.0875	0.3008	1.0000
14.000	2.0434	0.02662	0.02375	-0.0852	0.2988	1.0000
14.250	2.0511	0.02786	0.02502	-0.0845	0.2983	1.0000
14.500	2.0566	0.02929	0.02650	-0.0838	0.2975	1.0000
14.750	2.0593	0.03102	0.02827	-0.0830	0.2967	1.0000
15.000	2.0614	0.03278	0.03007	-0.0822	0.2953	1.0000
15.250	2.0581	0.03502	0.03233	-0.0813	0.2934	1.0000
15.500	2.0450	0.03812	0.03545	-0.0799	0.2899	1.0000
15.750	2.0473	0.03999	0.03736	-0.0793	0.2898	1.0000
16.000	2.0576	0.04115	0.03855	-0.0790	0.2875	1.0000
16.250	2.0549	0.04344	0.04088	-0.0783	0.2855	1.0000
16.500	2.0350	0.04725	0.04466	-0.0769	0.2760	1.0000
16.750	2.0524	0.04780	0.04529	-0.0769	0.2817	1.0000
17.000	2.0579	0.04940	0.04690	-0.0765	0.2786	1.0000
17.250	2.0314	0.05389	0.05135	-0.0751	0.2664	1.0000
17.500	2.0240	0.05674	0.05422	-0.0744	0.2627	1.0000
17.750	2.0207	0.05920	0.05668	-0.0739	0.2579	1.0000
18.000	2.0027	0.06313	0.06060	-0.0730	0.2500	1.0000
18.250	1.9228	0.07335	0.07067	-0.0707	0.2225	1.0000
18.500	1.9338	0.07457	0.07193	-0.0708	0.2232	1.0000
18.750	1.8967	0.08083	0.07814	-0.0699	0.2086	1.0000
19.000	1.8862	0.08441	0.08173	-0.0696	0.2033	1.0000
19.250	1.8629	0.08943	0.08674	-0.0692	0.1942	1.0000
19.500	1.8272	0.09601	0.09328	-0.0689	0.1810	1.0000
19.750	1.7874	0.10330	0.10054	-0.0687	0.1683	1.0000
20.000	1.8087	0.10330	0.10061	-0.0690	0.1734	1.0000
0.000	0.4964	0.00805	0.00308	-0.0268	0.3815	0.0080
-0.250	0.4662	0.00808	0.00312	-0.0264	0.3856	0.0079
-0.500	0.4370	0.00837	0.00349	-0.0262	0.3906	0.0080
-0.750	0.4068	0.00845	0.00360	-0.0259	0.3952	0.0079
-1.000	0.3766	0.00851	0.00370	-0.0255	0.4007	0.0079
-1.250	0.3469	0.00873	0.00399	-0.0253	0.4065	0.0079
-1.500	0.3171	0.00890	0.00422	-0.0250	0.4119	0.0079
-1.750	0.2873	0.00912	0.00450	-0.0248	0.4184	0.0079
-2.000	0.2580	0.00943	0.00489	-0.0247	0.4227	0.0079
-2.250	0.2284	0.00966	0.00518	-0.0246	0.4273	0.0078
-2.500	0.1993	0.01002	0.00561	-0.0246	0.4299	0.0078
-2.750	0.1705	0.01044	0.00611	-0.0246	0.4321	0.0078
-3.000	0.1411	0.01069	0.00642	-0.0246	0.4349	0.0077
-3.250	0.1125	0.01113	0.00693	-0.0246	0.4371	0.0077
-3.500	0.0847	0.01173	0.00764	-0.0249	0.4386	0.0077
-3.750	0.0567	0.01225	0.00824	-0.0251	0.4397	0.0076
-4.000	0.0302	0.01314	0.00926	-0.0255	0.4403	0.0076
-4.250	0.0030	0.01384	0.01008	-0.0258	0.4421	0.0075
-4.500	-0.0224	0.01500	0.01139	-0.0263	0.4438	0.0075
-4.750	-0.0445	0.01699	0.01363	-0.0273	0.4451	0.0073
-5.250	-0.0030	0.06800	0.06582	-0.0093	0.4436	0.0068
-5.500	-0.0182	0.07108	0.06892	-0.0063	0.4448	0.0068
-5.750	-0.0318	0.07402	0.07186	-0.0034	0.4458	0.0067
-6.000	-0.0439	0.07682	0.07467	-0.0007	0.4468	0.0067
-6.250	-0.0523	0.07973	0.07757	0.0019	0.4475	0.0067
-6.500	-0.0531	0.08204	0.07986	0.0030	0.4483	0.0067
-6.750	-0.0576	0.08451	0.08233	0.0046	0.4488	0.0066
-7.000	-0.0640	0.08709	0.08489	0.0061	0.4493	0.0066
-7.250	-0.0709	0.08984	0.08762	0.0078	0.4497	0.0066
-7.500	-0.0784	0.09254	0.09032	0.0095	0.4500	0.0066
-7.750	-0.0890	0.09463	0.09240	0.0108	0.4503	0.0065
-8.000	-0.0971	0.09749	0.09525	0.0126	0.4513	0.0065
-8.250	-0.0779	0.10149	0.09933	0.0140	0.4508	0.0065

Listing B.3: Trial 35 XFOIL Polar File.

```
 XFOIL         Version 6.99

 Calculated polar for: TRIAL 35

 1 1 Reynolds number fixed          Mach number fixed

 xtrf =   1.000 (top)        1.000 (bottom)
 Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000

 alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
 ------ -------- --------- --------- -------- -------- --------
 0.000   0.5331   0.00741   0.00221  -0.0449   0.3771   0.0077
 0.250   0.5625   0.00736   0.00213  -0.0452   0.3733   0.0078
 0.500   0.5922   0.00734   0.00209  -0.0455   0.3697   0.0079
 0.750   0.6218   0.00731   0.00205  -0.0458   0.3673   0.0081
 1.000   0.6514   0.00731   0.00205  -0.0462   0.3642   0.0083
 1.250   0.6810   0.00734   0.00207  -0.0465   0.3606   0.0086
 1.500   0.7107   0.00733   0.00206  -0.0469   0.3590   0.0090
 1.750   0.7404   0.00734   0.00207  -0.0473   0.3567   0.0092
 2.000   0.7699   0.00736   0.00209  -0.0476   0.3542   0.0103
 2.250   0.7994   0.00740   0.00212  -0.0480   0.3511   0.0108
 2.500   0.8289   0.00741   0.00214  -0.0484   0.3499   0.0125
 2.750   0.8583   0.00741   0.00217  -0.0488   0.3484   0.0203
 3.000   0.8875   0.00741   0.00221  -0.0492   0.3466   0.0377
 3.250   0.9164   0.00742   0.00228  -0.0496   0.3446   0.0667
 3.500   0.9449   0.00740   0.00236  -0.0499   0.3422   0.1190
 3.750   0.9727   0.00731   0.00244  -0.0501   0.3408   0.2101
 4.000   1.0005   0.00722   0.00252  -0.0503   0.3399   0.3055
 4.250   1.0278   0.00712   0.00262  -0.0504   0.3388   0.4140
 4.500   1.0627   0.00618   0.00294  -0.0524   0.3375   0.9876
 4.750   1.1167   0.00638   0.00310  -0.0582   0.3359   0.9908
 5.000   1.1687   0.00653   0.00322  -0.0636   0.3339   0.9922
 5.250   1.2107   0.00667   0.00333  -0.0669   0.3318   0.9935
 5.500   1.2475   0.00677   0.00342  -0.0691   0.3312   0.9949
 5.750   1.2830   0.00687   0.00351  -0.0710   0.3306   0.9961
 6.000   1.3237   0.00691   0.00354  -0.0740   0.3298   0.9966
 6.250   1.3647   0.00697   0.00359  -0.0772   0.3289   0.9973
 6.500   1.4031   0.00705   0.00365  -0.0798   0.3279   0.9981
 6.750   1.4393   0.00713   0.00373  -0.0819   0.3269   0.9989
 7.000   1.4717   0.00724   0.00383  -0.0833   0.3257   0.9993
 7.250   1.5024   0.00737   0.00394  -0.0844   0.3244   0.9996
 7.500   1.5324   0.00749   0.00406  -0.0853   0.3232   0.9997
 7.750   1.5624   0.00763   0.00419  -0.0863   0.3221   0.9999
 8.000   1.5927   0.00771   0.00428  -0.0873   0.3218   1.0000
 8.250   1.6206   0.00781   0.00439  -0.0878   0.3215   1.0000
 8.500   1.6483   0.00792   0.00451  -0.0883   0.3211   1.0000
 8.750   1.6758   0.00803   0.00463  -0.0887   0.3207   1.0000
 9.000   1.7032   0.00815   0.00476  -0.0892   0.3201   1.0000
 9.250   1.7303   0.00829   0.00491  -0.0897   0.3193   1.0000
 9.500   1.7572   0.00844   0.00506  -0.0901   0.3184   1.0000
 9.750   1.7838   0.00861   0.00524  -0.0906   0.3172   1.0000
 10.000   1.8100   0.00882   0.00544  -0.0911   0.3156   1.0000
 10.250   1.8358   0.00905   0.00567  -0.0915   0.3139   1.0000
 10.500   1.8617   0.00925   0.00588  -0.0919   0.3126   1.0000
 10.750   1.8880   0.00938   0.00603  -0.0923   0.3119   1.0000
 11.000   1.9138   0.00954   0.00620  -0.0927   0.3111   1.0000
 11.250   1.9389   0.00975   0.00642  -0.0930   0.3098   1.0000
 11.500   1.9633   0.00999   0.00668  -0.0933   0.3086   1.0000
 11.750   1.9869   0.01028   0.00697  -0.0936   0.3074   1.0000
 12.000   2.0096   0.01060   0.00731  -0.0937   0.3063   1.0000
 12.250   2.0311   0.01100   0.00773  -0.0939   0.3050   1.0000
 12.500   2.0509   0.01175   0.00850  -0.0945   0.3036   1.0000
 12.750   2.0701   0.01239   0.00918  -0.0949   0.3033   1.0000
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.0719	0.01333	0.01017	-0.0924	0.3029	1.0000
13.250	2.0784	0.01418	0.01105	-0.0907	0.3016	1.0000
13.500	2.0832	0.01535	0.01226	-0.0892	0.2993	1.0000
13.750	2.0952	0.01619	0.01314	-0.0885	0.3001	1.0000
14.000	2.0884	0.01830	0.01525	-0.0865	0.2951	1.0000
14.250	2.1053	0.01887	0.01588	-0.0862	0.2975	1.0000
14.500	2.1046	0.02061	0.01765	-0.0847	0.2942	1.0000
14.750	2.1118	0.02184	0.01891	-0.0839	0.2930	1.0000
15.000	2.1154	0.02333	0.02044	-0.0828	0.2912	1.0000
15.250	2.0979	0.02637	0.02348	-0.0806	0.2839	1.0000
15.500	2.0995	0.02811	0.02525	-0.0796	0.2824	1.0000
15.750	2.0644	0.03276	0.02988	-0.0768	0.2700	1.0000
16.000	2.0194	0.03846	0.03554	-0.0739	0.2549	1.0000
16.250	2.0646	0.03677	0.03394	-0.0752	0.2650	1.0000
16.500	2.0039	0.04399	0.04110	-0.0719	0.2466	1.0000
16.750	1.9603	0.04997	0.04702	-0.0696	0.2304	1.0000
17.000	1.8901	0.05873	0.05567	-0.0668	0.2048	1.0000
17.250	1.8995	0.06013	0.05712	-0.0667	0.2064	1.0000
17.500	1.8810	0.06425	0.06123	-0.0658	0.1976	1.0000
17.750	1.8759	0.06707	0.06406	-0.0654	0.1936	1.0000
18.000	1.8664	0.07043	0.06743	-0.0649	0.1885	1.0000
18.250	1.8720	0.07224	0.06927	-0.0648	0.1878	1.0000
18.500	1.8461	0.07734	0.07435	-0.0640	0.1775	1.0000
18.750	1.7841	0.08659	0.08351	-0.0629	0.1558	1.0000
19.000	1.8027	0.08701	0.08400	-0.0631	0.1603	1.0000
19.250	1.7589	0.09457	0.09152	-0.0626	0.1460	1.0000
19.500	1.7326	0.10028	0.09721	-0.0625	0.1358	1.0000
19.750	1.7007	0.10674	0.10364	-0.0625	0.1237	1.0000
20.000	1.7303	0.10576	0.10276	-0.0628	0.1329	1.0000
-0.250	0.5037	0.00751	0.00233	-0.0447	0.3799	0.0076
-0.500	0.4745	0.00762	0.00250	-0.0446	0.3847	0.0075
-0.750	0.4450	0.00770	0.00260	-0.0443	0.3887	0.0075
-1.000	0.4154	0.00776	0.00270	-0.0441	0.3934	0.0075
-1.250	0.3856	0.00782	0.00278	-0.0438	0.3982	0.0075
-1.500	0.3558	0.00786	0.00285	-0.0436	0.4035	0.0075
-1.750	0.3263	0.00798	0.00300	-0.0434	0.4086	0.0075
-2.000	0.2967	0.00808	0.00315	-0.0432	0.4152	0.0074
-2.250	0.2670	0.00817	0.00328	-0.0429	0.4214	0.0074
-2.500	0.2372	0.00827	0.00342	-0.0427	0.4278	0.0074
-2.750	0.2078	0.00842	0.00363	-0.0426	0.4337	0.0074
-3.000	0.1782	0.00860	0.00383	-0.0425	0.4373	0.0073
-3.250	0.1488	0.00880	0.00407	-0.0424	0.4407	0.0073
-3.500	0.1194	0.00898	0.00430	-0.0423	0.4440	0.0072
-3.750	0.0899	0.00917	0.00451	-0.0422	0.4463	0.0072
-4.000	0.0610	0.00950	0.00489	-0.0423	0.4476	0.0072
-4.250	0.0321	0.00981	0.00525	-0.0423	0.4497	0.0071
-4.500	0.0024	0.00993	0.00539	-0.0422	0.4521	0.0071
-4.750	-0.0265	0.01023	0.00574	-0.0423	0.4540	0.0070
-5.000	-0.0549	0.01062	0.00620	-0.0424	0.4555	0.0070
-5.250	-0.0834	0.01100	0.00663	-0.0425	0.4567	0.0070
-5.500	-0.1125	0.01123	0.00688	-0.0426	0.4575	0.0069
-5.750	-0.1409	0.01162	0.00732	-0.0427	0.4583	0.0068
-6.000	-0.1695	0.01194	0.00770	-0.0428	0.4603	0.0068
-6.250	-0.1982	0.01222	0.00802	-0.0429	0.4620	0.0067
-6.500	-0.2260	0.01270	0.00857	-0.0431	0.4634	0.0067
-6.750	-0.2541	0.01309	0.00901	-0.0433	0.4646	0.0067
-7.000	-0.2817	0.01357	0.00955	-0.0436	0.4657	0.0066
-7.250	-0.3112	0.01366	0.00963	-0.0436	0.4666	0.0066
-7.500	-0.3368	0.01454	0.01063	-0.0441	0.4673	0.0066
-7.750	-0.3655	0.01477	0.01086	-0.0442	0.4679	0.0065
-8.000	-0.3917	0.01548	0.01165	-0.0446	0.4683	0.0065
-8.250	-0.4210	0.01556	0.01172	-0.0446	0.4701	0.0064
-8.500	-0.4435	0.01698	0.01333	-0.0453	0.4716	0.0064
-8.750	-0.4721	0.01719	0.01354	-0.0455	0.4730	0.0064
-9.000	-0.4977	0.01796	0.01438	-0.0458	0.4742	0.0063
-9.250	-0.5197	0.01943	0.01601	-0.0464	0.4754	0.0063
-9.500	-0.5437	0.02051	0.01717	-0.0468	0.4763	0.0063

Listing B.4: Trial 36 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 36
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.6099   0.00722   0.00203  -0.0664   0.3844   0.0075
14      0.250   0.6398   0.00720   0.00202  -0.0668   0.3817   0.0078
15      0.500   0.6696   0.00722   0.00203  -0.0672   0.3782   0.0082
16      0.750   0.6993   0.00724   0.00203  -0.0676   0.3741   0.0083
17      1.000   0.7292   0.00724   0.00203  -0.0679   0.3721   0.0084
18      1.250   0.7588   0.00724   0.00202  -0.0683   0.3694   0.0087
19      1.500   0.7884   0.00727   0.00203  -0.0687   0.3661   0.0092
20      1.750   0.8180   0.00728   0.00205  -0.0691   0.3638   0.0096
21      2.000   0.8477   0.00729   0.00206  -0.0695   0.3621   0.0101
22      2.250   0.8773   0.00731   0.00209  -0.0699   0.3601   0.0118
23      2.500   0.9067   0.00735   0.00213  -0.0704   0.3577   0.0173
24      2.750   0.9360   0.00738   0.00219  -0.0708   0.3549   0.0279
25      3.000   0.9653   0.00736   0.00223  -0.0712   0.3538   0.0507
26      3.250   0.9943   0.00732   0.00228  -0.0716   0.3525   0.0972
27      3.500   1.0228   0.00724   0.00236  -0.0719   0.3511   0.1820
28      3.750   1.0508   0.00716   0.00246  -0.0722   0.3494   0.2856
29      4.000   1.0780   0.00705   0.00258  -0.0724   0.3476   0.4239
30      4.250   1.1195   0.00615   0.00287  -0.0758   0.3450   0.9922
31      4.500   1.1756   0.00625   0.00295  -0.0821   0.3442   0.9964
32      4.750   1.2297   0.00631   0.00298  -0.0879   0.3433   0.9982
33      5.000   1.2730   0.00637   0.00302  -0.0914   0.3422   0.9994
34      5.250   1.3101   0.00645   0.00308  -0.0937   0.3409   1.0000
35      5.500   1.3384   0.00656   0.00318  -0.0941   0.3397   1.0000
36      5.750   1.3668   0.00667   0.00328  -0.0945   0.3383   1.0000
37      6.000   1.3950   0.00680   0.00339  -0.0950   0.3366   1.0000
38      6.250   1.4231   0.00692   0.00350  -0.0954   0.3353   1.0000
39      6.500   1.4514   0.00700   0.00359  -0.0958   0.3350   1.0000
40      6.750   1.4795   0.00709   0.00368  -0.0962   0.3345   1.0000
41      7.000   1.5076   0.00718   0.00378  -0.0967   0.3339   1.0000
42      7.250   1.5355   0.00729   0.00388  -0.0971   0.3332   1.0000
43      7.500   1.5633   0.00740   0.00399  -0.0975   0.3325   1.0000
44      7.750   1.5909   0.00752   0.00411  -0.0980   0.3317   1.0000
45      8.000   1.6183   0.00765   0.00425  -0.0984   0.3307   1.0000
46      8.250   1.6454   0.00780   0.00439  -0.0988   0.3298   1.0000
47      8.500   1.6724   0.00795   0.00454  -0.0992   0.3287   1.0000
48      8.750   1.6990   0.00812   0.00471  -0.0995   0.3273   1.0000
49      9.000   1.7254   0.00830   0.00489  -0.0999   0.3259   1.0000
50      9.250   1.7524   0.00841   0.00501  -0.1003   0.3255   1.0000
51      9.500   1.7791   0.00852   0.00514  -0.1007   0.3250   1.0000
52      9.750   1.8058   0.00864   0.00527  -0.1011   0.3242   1.0000
53      10.000   1.8319   0.00880   0.00544  -0.1014   0.3230   1.0000
54      10.250   1.8574   0.00899   0.00563  -0.1017   0.3215   1.0000
55      10.500   1.8821   0.00922   0.00586  -0.1019   0.3198   1.0000
56      10.750   1.9061   0.00948   0.00612  -0.1021   0.3177   1.0000
57      11.000   1.9301   0.00971   0.00636  -0.1022   0.3163   1.0000
58      11.250   1.9546   0.00988   0.00655  -0.1024   0.3159   1.0000
59      11.500   1.9785   0.01007   0.00677  -0.1026   0.3153   1.0000
60      11.750   2.0015   0.01030   0.00701  -0.1026   0.3146   1.0000
61      12.000   2.0233   0.01058   0.00732  -0.1026   0.3136   1.0000
62      12.250   2.0433   0.01104   0.00780  -0.1026   0.3126   1.0000
63      12.500   2.0503   0.01195   0.00875  -0.1012   0.3117   1.0000
64      12.750   2.0547   0.01289   0.00972  -0.0992   0.3106   1.0000
65
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.0650	0.01380	0.01066	-0.0982	0.3096	1.0000
13.250	2.0726	0.01500	0.01187	-0.0972	0.3072	1.0000
13.500	2.0853	0.01590	0.01281	-0.0967	0.3072	1.0000
13.750	2.1010	0.01661	0.01356	-0.0964	0.3059	1.0000
14.000	2.1110	0.01772	0.01470	-0.0958	0.3040	1.0000
14.250	2.1202	0.01891	0.01592	-0.0952	0.3031	1.0000
14.500	2.1209	0.02069	0.01772	-0.0940	0.3001	1.0000
14.750	2.1237	0.02235	0.01940	-0.0931	0.2961	1.0000
15.000	2.1333	0.02353	0.02063	-0.0926	0.2970	1.0000
15.250	2.1179	0.02657	0.02367	-0.0906	0.2895	1.0000
15.500	2.1180	0.02849	0.02561	-0.0897	0.2859	1.0000
15.750	2.1007	0.03186	0.02898	-0.0879	0.2771	1.0000
16.000	2.0683	0.03654	0.03364	-0.0855	0.2658	1.0000
16.250	2.0098	0.04363	0.04064	-0.0821	0.2461	1.0000
16.500	1.9684	0.04954	0.04651	-0.0799	0.2312	1.0000
16.750	1.9471	0.05374	0.05070	-0.0786	0.2215	1.0000
17.000	2.0034	0.05089	0.04798	-0.0803	0.2363	1.0000
17.250	1.9534	0.05781	0.05483	-0.0781	0.2176	1.0000
17.500	1.9190	0.06352	0.06048	-0.0767	0.2029	1.0000
17.750	1.8540	0.07256	0.06941	-0.0747	0.1784	1.0000
18.000	1.8655	0.07383	0.07072	-0.0748	0.1797	1.0000
18.250	1.8346	0.07966	0.07652	-0.0740	0.1664	1.0000
18.500	1.8238	0.08346	0.08033	-0.0738	0.1614	1.0000
18.750	1.8205	0.08640	0.08330	-0.0737	0.1590	1.0000
19.000	1.8194	0.08918	0.08611	-0.0738	0.1566	1.0000
19.250	1.8190	0.09183	0.08880	-0.0739	0.1560	1.0000
19.500	1.7600	0.10139	0.09828	-0.0734	0.1349	1.0000
19.750	1.7611	0.10399	0.10090	-0.0737	0.1333	1.0000
20.000	1.7681	0.10584	0.10280	-0.0740	0.1348	1.0000
-0.250	0.5802	0.00721	0.00204	-0.0661	0.3890	0.0073
-0.500	0.5508	0.00733	0.00220	-0.0659	0.3927	0.0069
-0.750	0.5214	0.00743	0.00235	-0.0657	0.3969	0.0068
-1.000	0.4916	0.00747	0.00241	-0.0654	0.4015	0.0068
-1.250	0.4619	0.00754	0.00251	-0.0651	0.4062	0.0068
-1.500	0.4322	0.00760	0.00261	-0.0648	0.4115	0.0068
-1.750	0.4026	0.00770	0.00275	-0.0646	0.4169	0.0068
-2.000	0.3729	0.00780	0.00290	-0.0644	0.4225	0.0067
-2.250	0.3434	0.00792	0.00307	-0.0641	0.4292	0.0067
-2.500	0.3138	0.00809	0.00328	-0.0640	0.4348	0.0067
-2.750	0.2844	0.00829	0.00355	-0.0638	0.4404	0.0067
-3.000	0.2551	0.00856	0.00387	-0.0637	0.4439	0.0067
-3.250	0.2258	0.00881	0.00418	-0.0636	0.4480	0.0067
-3.500	0.1969	0.00919	0.00463	-0.0636	0.4508	0.0067
-3.750	0.1680	0.00958	0.00509	-0.0636	0.4523	0.0067
-4.000	0.1398	0.01016	0.00578	-0.0637	0.4550	0.0067
-4.250	0.1128	0.01105	0.00684	-0.0639	0.4574	0.0067
-4.750	0.0601	0.01322	0.00934	-0.0645	0.4608	0.0069
-5.000	0.0321	0.01380	0.01000	-0.0646	0.4618	0.0070
-5.250	0.0046	0.01449	0.01078	-0.0647	0.4626	0.0070
-5.500	-0.0223	0.01539	0.01180	-0.0648	0.4646	0.0070

Listing B.5: Trial 37 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 37
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL         CD       CDp       CM      Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.3767   0.00551   0.00110  -0.0688   0.3115   0.5145
14      0.250   0.4067   0.00554   0.00113  -0.0691   0.3094   0.5163
15      0.500   0.4367   0.00559   0.00115  -0.0693   0.3064   0.5176
16      1.000   0.4965   0.00568   0.00121  -0.0698   0.3003   0.5209
17      1.250   0.5264   0.00571   0.00124  -0.0701   0.2977   0.5228
18      1.500   0.5562   0.00577   0.00128  -0.0704   0.2939   0.5243
19      1.750   0.5859   0.00582   0.00132  -0.0706   0.2897   0.5254
20      2.000   0.6157   0.00587   0.00135  -0.0709   0.2861   0.5265
21      2.250   0.6450   0.00596   0.00141  -0.0710   0.2792   0.5274
22      2.500   0.6743   0.00605   0.00146  -0.0712   0.2716   0.5282
23      2.750   0.7029   0.00620   0.00154  -0.0713   0.2576   0.5289
24      3.000   0.7310   0.00640   0.00166  -0.0713   0.2406   0.5296
25      3.250   0.7591   0.00659   0.00177  -0.0713   0.2259   0.5301
26      3.500   0.7873   0.00677   0.00189  -0.0714   0.2121   0.5311
27      3.750   0.8153   0.00696   0.00201  -0.0714   0.1993   0.5331
28      4.000   0.8435   0.00713   0.00214  -0.0714   0.1876   0.5350
29      4.250   0.8717   0.00730   0.00226  -0.0715   0.1776   0.5366
30      4.750   0.9281   0.00764   0.00250  -0.0716   0.1607   0.5394
31      5.000   0.9563   0.00779   0.00262  -0.0716   0.1538   0.5406
32      5.250   0.9844   0.00795   0.00275  -0.0717   0.1472   0.5416
33      5.500   1.0123   0.00813   0.00288  -0.0717   0.1406   0.5424
34      5.750   1.0405   0.00827   0.00300  -0.0718   0.1361   0.5443
35      6.000   1.0686   0.00843   0.00314  -0.0718   0.1313   0.5468
36      6.250   1.0965   0.00858   0.00327  -0.0719   0.1272   0.5489
37      6.500   1.1245   0.00873   0.00341  -0.0719   0.1233   0.5507
38      6.750   1.1524   0.00888   0.00354  -0.0720   0.1203   0.5525
39      7.000   1.1799   0.00906   0.00369  -0.0720   0.1162   0.5539
40      7.500   1.2355   0.00935   0.00397  -0.0720   0.1121   0.5583
41      7.750   1.2628   0.00953   0.00414  -0.0720   0.1088   0.5611
42      8.000   1.2906   0.00965   0.00427  -0.0721   0.1078   0.5635
43      8.250   1.3180   0.00981   0.00442  -0.0721   0.1062   0.5656
44      8.500   1.3449   0.00999   0.00459  -0.0720   0.1042   0.5673
45      8.750   1.3718   0.01018   0.00477  -0.0719   0.1017   0.5706
46      9.000   1.3992   0.01031   0.00492  -0.0719   0.1011   0.5739
47      9.250   1.4263   0.01046   0.00508  -0.0719   0.1003   0.5768
48      9.500   1.4530   0.01063   0.00525  -0.0718   0.0992   0.5792
49      9.750   1.4795   0.01082   0.00544  -0.0717   0.0981   0.5823
50      10.000   1.5056   0.01101   0.00564  -0.0715   0.0966   0.5863
51      10.250   1.5315   0.01122   0.00585  -0.0713   0.0950   0.5899
52      10.500   1.5580   0.01138   0.00602  -0.0712   0.0948   0.5927
53      10.750   1.5843   0.01154   0.00620  -0.0711   0.0944   0.5973
54      11.000   1.6103   0.01171   0.00640  -0.0710   0.0940   0.6018
55      11.250   1.6359   0.01189   0.00660  -0.0707   0.0935   0.6052
56      11.500   1.6613   0.01209   0.00681  -0.0705   0.0929   0.6100
57      11.750   1.6862   0.01230   0.00703  -0.0702   0.0923   0.6148
58      12.000   1.7107   0.01253   0.00727  -0.0698   0.0915   0.6186
59      12.250   1.7347   0.01277   0.00753  -0.0694   0.0906   0.6242
60      12.500   1.7578   0.01305   0.00781  -0.0688   0.0891   0.6293
61      12.750   1.7813   0.01328   0.00807  -0.0683   0.0885   0.6338
62
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	1.8053	0.01348	0.00830	-0.0679	0.0883	0.6401
13.250	1.8289	0.01368	0.00852	-0.0674	0.0876	0.6449
13.500	1.8515	0.01391	0.00879	-0.0668	0.0870	0.6516
13.750	1.8725	0.01420	0.00908	-0.0660	0.0857	0.6573
14.000	1.8901	0.01452	0.00942	-0.0645	0.0843	0.6641
14.250	1.9045	0.01489	0.00979	-0.0625	0.0826	0.6703
14.500	1.9227	0.01514	0.01008	-0.0612	0.0822	0.6777
15.000	1.9555	0.01581	0.01082	-0.0581	0.0804	0.6930
15.250	1.9688	0.01629	0.01131	-0.0563	0.0789	0.7008
15.500	1.9803	0.01686	0.01191	-0.0543	0.0769	0.7092
15.750	1.9953	0.01732	0.01241	-0.0529	0.0765	0.7191
16.000	2.0092	0.01785	0.01299	-0.0514	0.0758	0.7293
16.250	2.0200	0.01856	0.01373	-0.0497	0.0741	0.7403
16.500	2.0273	0.01951	0.01471	-0.0478	0.0718	0.7525
16.750	2.0384	0.02031	0.01557	-0.0465	0.0709	0.7670
17.000	2.0421	0.02161	0.01690	-0.0447	0.0675	0.7823
17.250	2.0448	0.02309	0.01842	-0.0431	0.0642	0.8012
17.500	2.0308	0.02589	0.02125	-0.0408	0.0575	0.8243
17.750	2.0247	0.02837	0.02382	-0.0395	0.0538	0.8600
18.500	1.9918	0.03785	0.03363	-0.0370	0.0441	1.0000
18.750	1.9798	0.04169	0.03752	-0.0370	0.0418	1.0000
19.000	1.9701	0.04545	0.04134	-0.0374	0.0400	1.0000
19.250	1.9552	0.05001	0.04595	-0.0381	0.0379	1.0000
19.500	1.9345	0.05545	0.05146	-0.0393	0.0352	1.0000
19.750	1.9182	0.06055	0.05662	-0.0406	0.0334	1.0000
20.000	1.9008	0.06599	0.06213	-0.0423	0.0317	1.0000
-0.500	0.3168	0.00543	0.00105	-0.0683	0.3192	0.5089
-0.750	0.2867	0.00540	0.00104	-0.0680	0.3233	0.5060
-1.000	0.2569	0.00534	0.00102	-0.0678	0.3295	0.5043
-1.250	0.2269	0.00530	0.00100	-0.0675	0.3343	0.5012
-1.500	0.1969	0.00527	0.00098	-0.0673	0.3400	0.4965
-1.750	0.1669	0.00522	0.00097	-0.0670	0.3456	0.4928
-2.000	0.1368	0.00520	0.00096	-0.0667	0.3510	0.4861
-2.250	0.1068	0.00518	0.00095	-0.0665	0.3570	0.4763
-2.500	0.0763	0.00548	0.00103	-0.0658	0.3621	0.4044
-2.750	0.0458	0.00606	0.00126	-0.0649	0.3678	0.2860
-3.250	-0.0142	0.00666	0.00153	-0.0637	0.3786	0.1702
-3.500	-0.0440	0.00692	0.00166	-0.0632	0.3834	0.1283
-3.750	-0.0735	0.00712	0.00177	-0.0629	0.3908	0.0964
-4.000	-0.1031	0.00732	0.00190	-0.0625	0.3961	0.0707
-4.250	-0.1323	0.00748	0.00201	-0.0622	0.4044	0.0522
-4.500	-0.1616	0.00762	0.00212	-0.0620	0.4119	0.0392
-4.750	-0.1908	0.00775	0.00223	-0.0617	0.4200	0.0304
-5.000	-0.2199	0.00786	0.00234	-0.0615	0.4299	0.0239
-5.250	-0.2489	0.00797	0.00245	-0.0614	0.4402	0.0194
-5.500	-0.2778	0.00807	0.00257	-0.0612	0.4510	0.0162
-5.750	-0.3065	0.00819	0.00270	-0.0611	0.4629	0.0132
-6.000	-0.3351	0.00829	0.00282	-0.0610	0.4758	0.0113
-6.250	-0.3636	0.00840	0.00296	-0.0609	0.4899	0.0097
-6.500	-0.3921	0.00851	0.00310	-0.0608	0.5035	0.0085
-6.750	-0.4203	0.00863	0.00325	-0.0607	0.5184	0.0075
-7.000	-0.4484	0.00876	0.00341	-0.0607	0.5325	0.0070
-7.250	-0.4765	0.00889	0.00357	-0.0606	0.5458	0.0067
-7.500	-0.5046	0.00902	0.00373	-0.0606	0.5590	0.0064
-7.750	-0.5324	0.00917	0.00391	-0.0606	0.5721	0.0062
-8.250	-0.5873	0.00951	0.00430	-0.0607	0.5978	0.0057
-8.500	-0.6143	0.00970	0.00452	-0.0608	0.6104	0.0054
-8.750	-0.6413	0.00987	0.00473	-0.0609	0.6232	0.0054
-9.000	-0.6680	0.01006	0.00495	-0.0610	0.6360	0.0053
-9.250	-0.6943	0.01025	0.00517	-0.0612	0.6488	0.0052
-9.500	-0.7203	0.01047	0.00543	-0.0614	0.6616	0.0051
-9.750	-0.7459	0.01069	0.00568	-0.0617	0.6746	0.0051

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-10.000	-0.7709	0.01093	0.00596	-0.0620	0.6879	0.0049
-10.250	-0.7957	0.01117	0.00623	-0.0624	0.7010	0.0049
-10.500	-0.8196	0.01142	0.00653	-0.0629	0.7178	0.0049
-10.750	-0.8424	0.01171	0.00686	-0.0635	0.7351	0.0048
-11.000	-0.8654	0.01196	0.00716	-0.0640	0.7509	0.0048
-11.250	-0.8868	0.01228	0.00753	-0.0648	0.7676	0.0048
-11.500	-0.9057	0.01265	0.00796	-0.0658	0.7863	0.0046
-12.000	-0.9224	0.01375	0.00926	-0.0704	0.8450	0.0046
-12.250	-0.9300	0.01407	0.00976	-0.0732	0.9029	0.0047
-12.500	-1.0324	0.01479	0.01068	-0.0577	0.9811	0.0045
-12.750	-1.0541	0.01559	0.01150	-0.0573	0.9914	0.0044
-13.250	-1.1132	0.01717	0.01310	-0.0535	0.9975	0.0044
-13.500	-1.1504	0.01807	0.01400	-0.0500	0.9990	0.0044
-13.750	-1.1825	0.01895	0.01490	-0.0475	0.9997	0.0043
-14.000	-1.1980	0.02007	0.01605	-0.0475	1.0000	0.0043
-14.250	-1.2086	0.02094	0.01694	-0.0488	1.0000	0.0043
-14.500	-1.2113	0.02242	0.01848	-0.0503	1.0000	0.0042
-14.750	-1.2213	0.02347	0.01955	-0.0514	1.0000	0.0042
-15.000	-1.2282	0.02476	0.02089	-0.0525	1.0000	0.0042

Listing B.6: Trial 38 XFOIL Polar File.

```
 XFOIL         Version 6.99

 Calculated polar for: TRIAL 38

 1 1 Reynolds number fixed          Mach number fixed

 xtrf =   1.000 (top)        1.000 (bottom)
 Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000

 alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
------ -------- --------- --------- -------- -------- --------
 0.000   0.4636   0.00563   0.00119  -0.0918   0.3139   0.5039
 0.250   0.4934   0.00569   0.00122  -0.0920   0.3095   0.5055
 0.500   0.5234   0.00573   0.00125  -0.0923   0.3073   0.5066
 0.750   0.5531   0.00579   0.00128  -0.0925   0.3035   0.5073
 1.000   0.5828   0.00586   0.00132  -0.0927   0.2983   0.5085
 1.250   0.6127   0.00590   0.00135  -0.0929   0.2949   0.5104
 1.500   0.6420   0.00598   0.00140  -0.0931   0.2874   0.5120
 1.750   0.6714   0.00606   0.00145  -0.0933   0.2797   0.5131
 2.000   0.7001   0.00620   0.00153  -0.0934   0.2661   0.5142
 2.250   0.7284   0.00638   0.00163  -0.0934   0.2496   0.5152
 2.500   0.7565   0.00657   0.00174  -0.0934   0.2339   0.5161
 2.750   0.7848   0.00674   0.00184  -0.0934   0.2206   0.5169
 3.000   0.8130   0.00692   0.00196  -0.0934   0.2078   0.5178
 3.250   0.8411   0.00710   0.00207  -0.0934   0.1960   0.5185
 3.500   0.8693   0.00727   0.00219  -0.0934   0.1856   0.5191
 3.750   0.8975   0.00743   0.00230  -0.0934   0.1764   0.5196
 4.000   0.9259   0.00758   0.00241  -0.0934   0.1682   0.5210
 4.250   0.9543   0.00773   0.00252  -0.0935   0.1609   0.5230
 4.500   0.9824   0.00789   0.00265  -0.0935   0.1537   0.5247
 4.750   1.0106   0.00805   0.00277  -0.0935   0.1475   0.5262
 5.000   1.0389   0.00819   0.00288  -0.0936   0.1427   0.5276
 5.250   1.0669   0.00835   0.00302  -0.0936   0.1369   0.5289
 5.500   1.0951   0.00849   0.00313  -0.0937   0.1335   0.5301
 5.750   1.1229   0.00866   0.00327  -0.0936   0.1287   0.5311
 6.000   1.1511   0.00879   0.00339  -0.0937   0.1262   0.5318
 6.250   1.1788   0.00896   0.00353  -0.0937   0.1225   0.5335
 6.500   1.2069   0.00909   0.00366  -0.0937   0.1199   0.5362
 6.750   1.2348   0.00923   0.00379  -0.0937   0.1177   0.5383
 7.000   1.2623   0.00940   0.00394  -0.0937   0.1148   0.5403
 7.250   1.2900   0.00954   0.00408  -0.0937   0.1128   0.5420
 7.500   1.3177   0.00968   0.00422  -0.0937   0.1115   0.5435
 7.750   1.3451   0.00984   0.00437  -0.0936   0.1097   0.5446
 8.000   1.3722   0.01002   0.00454  -0.0936   0.1074   0.5477
 8.250   1.3995   0.01017   0.00469  -0.0935   0.1058   0.5506
 8.500   1.4269   0.01031   0.00484  -0.0935   0.1051   0.5531
 8.750   1.4540   0.01046   0.00500  -0.0934   0.1041   0.5553
 9.000   1.4808   0.01063   0.00516  -0.0933   0.1030   0.5570
 9.250   1.5073   0.01082   0.00535  -0.0932   0.1017   0.5602
 9.500   1.5337   0.01101   0.00554  -0.0930   0.1002   0.5638
 9.750   1.5600   0.01119   0.00573  -0.0928   0.0991   0.5670
10.000   1.5866   0.01134   0.00590  -0.0927   0.0988   0.5695
10.250   1.6130   0.01150   0.00607  -0.0925   0.0985   0.5728
10.500   1.6392   0.01166   0.00626  -0.0924   0.0980   0.5770
10.750   1.6651   0.01184   0.00645  -0.0922   0.0975   0.5804
11.000   1.6906   0.01204   0.00665  -0.0919   0.0969   0.5831
11.250   1.7159   0.01224   0.00687  -0.0916   0.0963   0.5876
11.500   1.7408   0.01245   0.00709  -0.0912   0.0956   0.5920
11.750   1.7652   0.01268   0.00733  -0.0908   0.0948   0.5955
12.000   1.7891   0.01293   0.00759  -0.0903   0.0938   0.5999
12.250   1.8126   0.01319   0.00787  -0.0897   0.0927   0.6049
12.500   1.8370   0.01337   0.00808  -0.0893   0.0925   0.6090
12.750   1.8611   0.01357   0.00830  -0.0889   0.0922   0.6142
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	1.8847	0.01378	0.00854	-0.0884	0.0917	0.6196
13.500	1.9291	0.01428	0.00908	-0.0869	0.0899	0.6303
13.750	1.9481	0.01457	0.00938	-0.0856	0.0889	0.6354
14.000	1.9639	0.01490	0.00973	-0.0837	0.0875	0.6415
14.250	1.9805	0.01522	0.01007	-0.0821	0.0865	0.6475
14.500	1.9990	0.01549	0.01037	-0.0807	0.0861	0.6537
14.750	2.0166	0.01579	0.01071	-0.0793	0.0854	0.6605
15.000	2.0326	0.01617	0.01111	-0.0777	0.0843	0.6674
15.250	2.0468	0.01664	0.01160	-0.0759	0.0830	0.6745
15.500	2.0591	0.01721	0.01219	-0.0740	0.0810	0.6823
15.750	2.0744	0.01766	0.01268	-0.0725	0.0804	0.6900
16.000	2.0889	0.01818	0.01324	-0.0710	0.0795	0.6995
16.250	2.1012	0.01882	0.01392	-0.0693	0.0782	0.7088
16.500	2.1082	0.01978	0.01488	-0.0672	0.0753	0.7186
16.750	2.1194	0.02056	0.01571	-0.0657	0.0739	0.7289
17.000	2.1239	0.02179	0.01696	-0.0637	0.0708	0.7398
17.250	2.1258	0.02328	0.01847	-0.0617	0.0669	0.7518
17.500	2.1155	0.02575	0.02095	-0.0592	0.0608	0.7642
17.750	2.1130	0.02788	0.02313	-0.0577	0.0576	0.7787
18.000	2.1019	0.03089	0.02619	-0.0562	0.0531	0.7952
18.250	2.0888	0.03435	0.02971	-0.0551	0.0488	0.8146
18.500	2.0760	0.03806	0.03350	-0.0545	0.0452	0.8405
18.750	2.0710	0.04119	0.03679	-0.0545	0.0433	0.9012
19.250	2.0393	0.04981	0.04565	-0.0545	0.0385	1.0000
19.500	2.0269	0.05429	0.05020	-0.0553	0.0367	1.0000
19.750	2.0094	0.05955	0.05553	-0.0565	0.0347	1.0000
20.000	1.9910	0.06515	0.06119	-0.0579	0.0328	1.0000
0.000	0.4636	0.00563	0.00119	-0.0918	0.3139	0.5039
-0.250	0.4337	0.00559	0.00116	-0.0916	0.3174	0.5016
-0.500	0.4036	0.00556	0.00114	-0.0913	0.3200	0.4987
-0.750	0.3735	0.00554	0.00112	-0.0910	0.3226	0.4954
-1.000	0.3436	0.00548	0.00110	-0.0908	0.3273	0.4940
-1.250	0.3137	0.00543	0.00108	-0.0906	0.3316	0.4916
-1.500	0.2837	0.00539	0.00106	-0.0903	0.3373	0.4880
-1.750	0.2537	0.00536	0.00104	-0.0901	0.3436	0.4833
-2.000	0.2238	0.00530	0.00102	-0.0899	0.3506	0.4780
-2.250	0.1937	0.00529	0.00101	-0.0896	0.3563	0.4690
-2.500	0.1632	0.00538	0.00102	-0.0891	0.3642	0.4292
-3.000	0.0996	0.00626	0.00137	-0.0868	0.3786	0.2438
-3.250	0.0687	0.00654	0.00150	-0.0861	0.3854	0.1870
-3.500	0.0380	0.00681	0.00163	-0.0854	0.3921	0.1371
-3.750	0.0077	0.00703	0.00176	-0.0848	0.3994	0.1008
-4.000	-0.0224	0.00723	0.00187	-0.0843	0.4057	0.0738
-4.250	-0.0523	0.00739	0.00199	-0.0839	0.4141	0.0539
-4.500	-0.0821	0.00753	0.00210	-0.0835	0.4216	0.0403
-4.750	-0.1117	0.00766	0.00221	-0.0832	0.4303	0.0306
-5.000	-0.1411	0.00776	0.00231	-0.0829	0.4402	0.0242
-5.250	-0.1705	0.00786	0.00242	-0.0827	0.4503	0.0196
-5.500	-0.1998	0.00797	0.00254	-0.0824	0.4613	0.0159
-5.750	-0.2289	0.00808	0.00266	-0.0822	0.4734	0.0131
-6.000	-0.2578	0.00817	0.00278	-0.0820	0.4868	0.0114
-6.250	-0.2866	0.00828	0.00292	-0.0819	0.5012	0.0094
-6.500	-0.3152	0.00839	0.00306	-0.0817	0.5161	0.0084
-6.750	-0.3438	0.00850	0.00320	-0.0816	0.5310	0.0076
-7.000	-0.3722	0.00862	0.00336	-0.0815	0.5466	0.0070
-7.250	-0.4004	0.00874	0.00352	-0.0814	0.5636	0.0067
-7.500	-0.4285	0.00885	0.00368	-0.0814	0.5795	0.0065
-7.750	-0.4565	0.00899	0.00385	-0.0813	0.5942	0.0064
-8.000	-0.4844	0.00914	0.00403	-0.0813	0.6087	0.0062
-8.250	-0.5120	0.00929	0.00422	-0.0812	0.6229	0.0060
-8.500	-0.5395	0.00946	0.00443	-0.0812	0.6369	0.0059
-8.750	-0.5664	0.00968	0.00468	-0.0813	0.6509	0.0055

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-9.000	-0.5933	0.00987	0.00491	-0.0813	0.6646	0.0055
-9.500	-0.6461	0.01027	0.00539	-0.0816	0.6930	0.0053
-9.750	-0.6717	0.01050	0.00566	-0.0818	0.7078	0.0053
-10.000	-0.6971	0.01071	0.00591	-0.0820	0.7230	0.0053
-10.250	-0.7216	0.01095	0.00621	-0.0824	0.7410	0.0052
-10.500	-0.7453	0.01122	0.00652	-0.0828	0.7597	0.0051
-10.750	-0.7685	0.01147	0.00683	-0.0833	0.7773	0.0051
-11.000	-0.7902	0.01178	0.00720	-0.0840	0.7989	0.0049
-11.250	-0.8100	0.01213	0.00763	-0.0847	0.8230	0.0050
-11.500	-0.8262	0.01257	0.00818	-0.0857	0.8543	0.0049
-11.750	-0.8256	0.01294	0.00886	-0.0896	0.9456	0.0048
-12.000	-0.9201	0.01352	0.00948	-0.0760	0.9652	0.0048
-12.500	-0.9684	0.01498	0.01096	-0.0745	0.9889	0.0048
-12.750	-1.0000	0.01583	0.01182	-0.0721	0.9908	0.0045
-13.000	-1.0312	0.01653	0.01252	-0.0701	0.9928	0.0047
-13.250	-1.0517	0.01752	0.01352	-0.0694	0.9949	0.0045
-13.500	-1.0903	0.01849	0.01451	-0.0655	0.9963	0.0045
-13.750	-1.1264	0.01953	0.01557	-0.0618	0.9974	0.0044
-14.000	-1.1578	0.02011	0.01613	-0.0599	0.9981	0.0045
-14.250	-1.1822	0.02143	0.01748	-0.0578	0.9990	0.0044
-14.500	-1.2050	0.02262	0.01870	-0.0562	0.9997	0.0043
-14.750	-1.2212	0.02387	0.01999	-0.0556	1.0000	0.0043
-15.000	-1.2332	0.02472	0.02084	-0.0566	1.0000	0.0044

Listing B.7: Trial 39 XFOIL Polar File.

```
 XFOIL         Version 6.99

 Calculated polar for: TRIAL 39

 1 1 Reynolds number fixed          Mach number fixed

 xtrf =   1.000 (top)        1.000 (bottom)
 Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000

 alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
------ -------- --------- --------- -------- -------- --------
 0.000   0.5543   0.00577   0.00129  -0.1150   0.3184   0.4941
 0.250   0.5840   0.00583   0.00132  -0.1152   0.3141   0.4959
 0.500   0.6139   0.00587   0.00135  -0.1154   0.3116   0.4970
 0.750   0.6435   0.00594   0.00139  -0.1156   0.3074   0.4979
 1.000   0.6729   0.00601   0.00143  -0.1158   0.3023   0.4985
 1.250   0.7025   0.00607   0.00147  -0.1160   0.2979   0.4989
 1.500   0.7318   0.00616   0.00152  -0.1161   0.2908   0.5001
 1.750   0.7608   0.00627   0.00158  -0.1163   0.2804   0.5016
 2.000   0.7892   0.00642   0.00167  -0.1163   0.2653   0.5030
 2.250   0.8171   0.00662   0.00179  -0.1162   0.2489   0.5042
 2.500   0.8453   0.00679   0.00189  -0.1162   0.2353   0.5052
 2.750   0.8734   0.00696   0.00201  -0.1162   0.2220   0.5063
 3.000   0.9014   0.00715   0.00213  -0.1162   0.2098   0.5073
 3.250   0.9295   0.00732   0.00224  -0.1161   0.1989   0.5083
 3.500   0.9576   0.00749   0.00236  -0.1161   0.1891   0.5092
 3.750   0.9857   0.00765   0.00247  -0.1161   0.1806   0.5099
 4.000   1.0138   0.00781   0.00259  -0.1161   0.1726   0.5106
 4.250   1.0418   0.00798   0.00271  -0.1161   0.1653   0.5111
 4.500   1.0697   0.00815   0.00284  -0.1160   0.1576   0.5122
 4.750   1.0978   0.00830   0.00296  -0.1161   0.1522   0.5142
 5.000   1.1259   0.00844   0.00308  -0.1161   0.1471   0.5161
 5.250   1.1538   0.00861   0.00322  -0.1161   0.1420   0.5177
 5.500   1.1819   0.00874   0.00333  -0.1161   0.1384   0.5192
 5.750   1.2096   0.00890   0.00347  -0.1161   0.1345   0.5206
 6.000   1.2375   0.00905   0.00360  -0.1160   0.1311   0.5218
 6.250   1.2653   0.00919   0.00373  -0.1160   0.1285   0.5228
 6.500   1.2926   0.00937   0.00388  -0.1159   0.1250   0.5236
 6.750   1.3204   0.00950   0.00401  -0.1159   0.1229   0.5257
 7.000   1.3481   0.00964   0.00415  -0.1159   0.1211   0.5284
 7.250   1.3754   0.00981   0.00430  -0.1158   0.1189   0.5306
 7.500   1.4025   0.00998   0.00446  -0.1157   0.1160   0.5326
 7.750   1.4301   0.01011   0.00460  -0.1157   0.1153   0.5343
 8.000   1.4573   0.01026   0.00475  -0.1156   0.1141   0.5357
 8.500   1.5109   0.01061   0.00508  -0.1153   0.1109   0.5409
 8.750   1.5374   0.01079   0.00527  -0.1152   0.1089   0.5436
 9.000   1.5644   0.01093   0.00542  -0.1151   0.1084   0.5461
 9.250   1.5911   0.01108   0.00558  -0.1149   0.1079   0.5482
 9.750   1.6438   0.01142   0.00594  -0.1146   0.1064   0.5545
10.000   1.6697   0.01160   0.00613  -0.1143   0.1055   0.5579
10.250   1.6952   0.01181   0.00634  -0.1140   0.1043   0.5607
10.500   1.7203   0.01203   0.00656  -0.1136   0.1030   0.5632
10.750   1.7454   0.01224   0.00679  -0.1133   0.1018   0.5678
11.000   1.7709   0.01242   0.00698  -0.1130   0.1016   0.5716
11.250   1.7962   0.01259   0.00718  -0.1126   0.1014   0.5746
11.500   1.8212   0.01278   0.00738  -0.1123   0.1011   0.5787
11.750   1.8459   0.01296   0.00760  -0.1119   0.1007   0.5833
12.000   1.8702   0.01317   0.00782  -0.1114   0.1002   0.5872
12.250   1.8938   0.01339   0.00806  -0.1109   0.0994   0.5912
12.500   1.9168   0.01364   0.00832  -0.1102   0.0985   0.5965
12.750   1.9389   0.01392   0.00861  -0.1094   0.0973   0.6007
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	1.9601	0.01423	0.00893	-0.1084	0.0958	0.6056
13.250	1.9808	0.01446	0.00919	-0.1074	0.0950	0.6111
13.500	2.0003	0.01467	0.00942	-0.1061	0.0945	0.6153
13.750	2.0190	0.01492	0.00970	-0.1047	0.0937	0.6217
14.000	2.0362	0.01524	0.01004	-0.1030	0.0925	0.6270
14.250	2.0521	0.01562	0.01043	-0.1013	0.0909	0.6328
14.500	2.0671	0.01606	0.01088	-0.0994	0.0891	0.6390
14.750	2.0851	0.01638	0.01124	-0.0981	0.0885	0.6449
15.000	2.1020	0.01675	0.01165	-0.0966	0.0876	0.6519
15.250	2.1166	0.01724	0.01215	-0.0949	0.0860	0.6584
15.500	2.1285	0.01786	0.01279	-0.0928	0.0838	0.6657
15.750	2.1426	0.01840	0.01337	-0.0912	0.0825	0.6731
16.000	2.1563	0.01899	0.01398	-0.0896	0.0814	0.6807
16.250	2.1651	0.01985	0.01486	-0.0874	0.0786	0.6897
16.500	2.1729	0.02081	0.01585	-0.0854	0.0754	0.6984
16.750	2.1696	0.02249	0.01751	-0.0824	0.0690	0.7066
17.000	2.1623	0.02459	0.01961	-0.0794	0.0630	0.7163
17.250	2.1553	0.02688	0.02193	-0.0770	0.0579	0.7258
17.500	2.1470	0.02950	0.02457	-0.0749	0.0532	0.7355
17.750	2.1402	0.03222	0.02734	-0.0734	0.0497	0.7461
18.000	2.1302	0.03545	0.03061	-0.0722	0.0460	0.7574
18.250	2.1183	0.03910	0.03431	-0.0714	0.0424	0.7692
18.750	2.0940	0.04720	0.04256	-0.0711	0.0369	0.7973
19.000	2.0837	0.05138	0.04682	-0.0715	0.0350	0.8153
19.250	2.0667	0.05657	0.05211	-0.0724	0.0326	0.8356
19.500	2.0522	0.06171	0.05737	-0.0735	0.0310	0.8646
19.750	2.0341	0.06701	0.06293	-0.0746	0.0292	1.0000
20.000	2.0132	0.07326	0.06927	-0.0764	0.0273	1.0000
0.000	0.5543	0.00577	0.00129	-0.1150	0.3184	0.4941
-0.250	0.5245	0.00572	0.00126	-0.1148	0.3222	0.4919
-0.500	0.4944	0.00570	0.00123	-0.1145	0.3250	0.4886
-0.750	0.4644	0.00566	0.00121	-0.1142	0.3287	0.4865
-1.000	0.4347	0.00560	0.00118	-0.1140	0.3325	0.4850
-1.250	0.4047	0.00556	0.00116	-0.1138	0.3354	0.4825
-1.500	0.3745	0.00553	0.00114	-0.1135	0.3385	0.4790
-1.750	0.3447	0.00548	0.00112	-0.1133	0.3454	0.4747
-2.250	0.2848	0.00540	0.00108	-0.1128	0.3580	0.4617
-2.500	0.2542	0.00543	0.00108	-0.1124	0.3637	0.4401
-2.750	0.2205	0.00589	0.00124	-0.1108	0.3714	0.3286
-3.000	0.1876	0.00629	0.00140	-0.1094	0.3784	0.2425
-3.250	0.1558	0.00656	0.00152	-0.1084	0.3853	0.1859
-3.500	0.1245	0.00678	0.00163	-0.1076	0.3931	0.1409
-4.000	0.0624	0.00721	0.00188	-0.1062	0.4068	0.0706
-4.250	0.0317	0.00738	0.00199	-0.1056	0.4125	0.0506
-4.500	0.0016	0.00750	0.00209	-0.1051	0.4206	0.0379
-5.000	-0.0586	0.00774	0.00230	-0.1043	0.4360	0.0224
-5.250	-0.0883	0.00783	0.00241	-0.1039	0.4456	0.0179
-5.500	-0.1180	0.00794	0.00252	-0.1036	0.4552	0.0144
-5.750	-0.1476	0.00804	0.00264	-0.1033	0.4657	0.0121
-6.000	-0.1770	0.00815	0.00277	-0.1030	0.4783	0.0097
-6.250	-0.2061	0.00825	0.00290	-0.1027	0.4927	0.0082
-6.500	-0.2350	0.00833	0.00303	-0.1025	0.5097	0.0076
-6.750	-0.2639	0.00843	0.00317	-0.1023	0.5261	0.0073
-7.000	-0.2926	0.00854	0.00333	-0.1022	0.5434	0.0068
-7.250	-0.3211	0.00864	0.00348	-0.1020	0.5612	0.0066
-7.500	-0.3495	0.00875	0.00363	-0.1019	0.5784	0.0065
-7.750	-0.3778	0.00887	0.00381	-0.1017	0.5959	0.0063
-8.000	-0.4058	0.00900	0.00398	-0.1016	0.6133	0.0061
-8.250	-0.4337	0.00916	0.00418	-0.1015	0.6300	0.0060
-8.500	-0.4615	0.00933	0.00439	-0.1014	0.6460	0.0058
-8.750	-0.4887	0.00952	0.00464	-0.1014	0.6628	0.0056

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-9.000	-0.5157	0.00973	0.00490	-0.1014	0.6794	0.0055
-9.250	-0.5425	0.00993	0.00515	-0.1014	0.6957	0.0054
-9.500	-0.5689	0.01014	0.00540	-0.1014	0.7122	0.0054
-9.750	-0.5949	0.01033	0.00564	-0.1015	0.7296	0.0054
-10.000	-0.6203	0.01054	0.00590	-0.1017	0.7498	0.0053
-10.250	-0.6450	0.01076	0.00619	-0.1020	0.7686	0.0052
-10.500	-0.6687	0.01102	0.00651	-0.1024	0.7901	0.0051
-10.750	-0.6910	0.01128	0.00685	-0.1029	0.8145	0.0051
-11.000	-0.7117	0.01160	0.00726	-0.1035	0.8420	0.0050
-11.250	-0.7271	0.01206	0.00787	-0.1046	0.8822	0.0051
-11.500	-0.7713	0.01236	0.00842	-0.1004	0.9563	0.0049
-11.750	-0.8394	0.01301	0.00907	-0.0915	0.9619	0.0048
-12.000	-0.8830	0.01375	0.00982	-0.0872	0.9739	0.0048
-12.500	-0.9233	0.01526	0.01134	-0.0870	0.9851	0.0047
-12.750	-0.9287	0.01608	0.01219	-0.0894	0.9890	0.0047
-13.000	-0.9608	0.01703	0.01315	-0.0867	0.9899	0.0046
-13.250	-0.9926	0.01789	0.01402	-0.0841	0.9910	0.0045
-13.500	-1.0257	0.01879	0.01492	-0.0813	0.9925	0.0046
-13.750	-1.0508	0.01994	0.01610	-0.0794	0.9943	0.0045
-14.000	-1.0711	0.02119	0.01739	-0.0782	0.9958	0.0044
-14.250	-1.1127	0.02218	0.01837	-0.0735	0.9965	0.0044
-14.500	-1.1363	0.02381	0.02007	-0.0707	0.9974	0.0043
-14.750	-1.1581	0.02507	0.02136	-0.0690	0.9982	0.0043
-15.000	-1.1856	0.02576	0.02202	-0.0675	0.9988	0.0043

Listing B.8: Trial 40 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 40
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL        CD       CDp       CM      Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.7299   0.00613   0.00155  -0.1603   0.3248   0.4730
14      0.250   0.7595   0.00619   0.00159  -0.1605   0.3208   0.4738
15      0.500   0.7886   0.00628   0.00164  -0.1606   0.3144   0.4743
16      0.750   0.8181   0.00634   0.00168  -0.1608   0.3097   0.4750
17      1.000   0.8472   0.00644   0.00174  -0.1609   0.3013   0.4765
18      1.250   0.8758   0.00657   0.00181  -0.1609   0.2882   0.4778
19      1.500   0.9036   0.00677   0.00192  -0.1608   0.2714   0.4790
20      1.750   0.9317   0.00695   0.00203  -0.1608   0.2566   0.4802
21      2.000   0.9595   0.00714   0.00215  -0.1607   0.2418   0.4811
22      2.250   0.9873   0.00733   0.00227  -0.1606   0.2285   0.4820
23      2.750   1.0432   0.00768   0.00250  -0.1604   0.2065   0.4836
24      3.000   1.0712   0.00785   0.00262  -0.1604   0.1979   0.4843
25      3.250   1.0993   0.00800   0.00273  -0.1604   0.1902   0.4850
26      3.500   1.1272   0.00816   0.00284  -0.1603   0.1828   0.4856
27      3.750   1.1550   0.00833   0.00297  -0.1602   0.1753   0.4861
28      4.000   1.1829   0.00848   0.00308  -0.1601   0.1698   0.4866
29      4.250   1.2108   0.00863   0.00320  -0.1601   0.1644   0.4872
30      4.500   1.2386   0.00879   0.00333  -0.1600   0.1590   0.4891
31      4.750   1.2667   0.00893   0.00345  -0.1600   0.1549   0.4909
32      5.000   1.2944   0.00909   0.00359  -0.1600   0.1506   0.4924
33      5.250   1.3222   0.00923   0.00372  -0.1599   0.1468   0.4938
34      5.500   1.3499   0.00937   0.00384  -0.1598   0.1440   0.4951
35      5.750   1.3771   0.00955   0.00399  -0.1597   0.1400   0.4964
36      6.000   1.4048   0.00968   0.00412  -0.1596   0.1379   0.4975
37      6.250   1.4322   0.00983   0.00425  -0.1595   0.1358   0.4985
38      6.500   1.4591   0.01001   0.00441  -0.1593   0.1331   0.4991
39      6.750   1.4863   0.01016   0.00456  -0.1592   0.1304   0.5012
40      7.000   1.5138   0.01029   0.00469  -0.1591   0.1294   0.5038
41      7.250   1.5410   0.01044   0.00484  -0.1590   0.1278   0.5058
42      7.500   1.5677   0.01061   0.00501  -0.1588   0.1260   0.5077
43      7.750   1.5941   0.01080   0.00518  -0.1586   0.1237   0.5094
44      8.000   1.6208   0.01096   0.00534  -0.1584   0.1224   0.5108
45      8.250   1.6475   0.01110   0.00549  -0.1582   0.1218   0.5118
46      8.500   1.6741   0.01125   0.00565  -0.1580   0.1210   0.5148
47      8.750   1.7004   0.01142   0.00583  -0.1577   0.1199   0.5178
48      9.000   1.7263   0.01161   0.00602  -0.1574   0.1187   0.5203
49      9.250   1.7518   0.01181   0.00622  -0.1570   0.1172   0.5225
50      9.500   1.7770   0.01202   0.00642  -0.1566   0.1156   0.5242
51      9.750   1.8025   0.01220   0.00662  -0.1563   0.1149   0.5268
52      10.000   1.8283   0.01236   0.00680  -0.1559   0.1146   0.5306
53      10.250   1.8537   0.01253   0.00699  -0.1556   0.1143   0.5337
54      10.500   1.8788   0.01271   0.00718  -0.1551   0.1139   0.5363
55      10.750   1.9034   0.01290   0.00739  -0.1546   0.1134   0.5383
56      11.000   1.9279   0.01310   0.00761  -0.1541   0.1127   0.5430
57      11.250   1.9517   0.01332   0.00784  -0.1535   0.1119   0.5467
58      11.500   1.9748   0.01357   0.00810  -0.1528   0.1108   0.5497
59      11.750   1.9972   0.01384   0.00837  -0.1519   0.1094   0.5529
60      12.000   2.0179   0.01412   0.00867  -0.1507   0.1078   0.5577
61      12.250   2.0386   0.01432   0.00889  -0.1495   0.1075   0.5616
62      12.500   2.0587   0.01453   0.00913  -0.1482   0.1071   0.5645
63      12.750   2.0789   0.01476   0.00939  -0.1470   0.1065   0.5700
64
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.0983	0.01504	0.00969	-0.1456	0.1055	0.5746
13.250	2.1163	0.01537	0.01003	-0.1440	0.1043	0.5780
13.500	2.1337	0.01576	0.01043	-0.1424	0.1024	0.5839
13.750	2.1508	0.01616	0.01085	-0.1407	0.1008	0.5891
14.000	2.1703	0.01645	0.01117	-0.1395	0.1003	0.5940
14.250	2.1883	0.01681	0.01156	-0.1381	0.0990	0.6004
14.500	2.2033	0.01732	0.01208	-0.1362	0.0967	0.6047
14.750	2.2178	0.01788	0.01265	-0.1343	0.0942	0.6114
15.000	2.2342	0.01833	0.01313	-0.1328	0.0930	0.6169
15.250	2.2467	0.01900	0.01381	-0.1307	0.0900	0.6231
15.500	2.2550	0.01990	0.01470	-0.1282	0.0848	0.6289
15.750	2.2587	0.02110	0.01588	-0.1251	0.0788	0.6345
16.000	2.2578	0.02263	0.01740	-0.1217	0.0723	0.6406
16.250	2.2508	0.02464	0.01939	-0.1180	0.0645	0.6454
16.500	2.2445	0.02678	0.02153	-0.1147	0.0585	0.6518
17.000	2.2392	0.03102	0.02582	-0.1100	0.0507	0.6644
17.250	2.2287	0.03404	0.02886	-0.1076	0.0458	0.6697
17.500	2.2203	0.03714	0.03199	-0.1059	0.0419	0.6765
17.750	2.2130	0.04034	0.03524	-0.1047	0.0389	0.6824
18.000	2.2060	0.04373	0.03868	-0.1038	0.0363	0.6900
18.250	2.1965	0.04757	0.04257	-0.1033	0.0339	0.6962
18.500	2.1860	0.05176	0.04683	-0.1031	0.0319	0.7034
18.750	2.1703	0.05678	0.05192	-0.1033	0.0294	0.7096
19.000	2.1589	0.06149	0.05671	-0.1038	0.0278	0.7167
19.250	2.1418	0.06710	0.06239	-0.1047	0.0259	0.7225
19.500	2.1253	0.07285	0.06823	-0.1059	0.0246	0.7300
19.750	2.1071	0.07897	0.07445	-0.1074	0.0231	0.7378
20.000	2.0887	0.08525	0.08082	-0.1091	0.0220	0.7422
0.000	0.7299	0.00613	0.00155	-0.1603	0.3248	0.4730
-0.250	0.7004	0.00607	0.00152	-0.1601	0.3285	0.4718
-0.500	0.6709	0.00600	0.00148	-0.1600	0.3330	0.4702
-0.750	0.6410	0.00596	0.00145	-0.1598	0.3362	0.4679
-1.000	0.6109	0.00593	0.00142	-0.1595	0.3386	0.4649
-1.250	0.5811	0.00587	0.00139	-0.1593	0.3430	0.4622
-1.500	0.5515	0.00582	0.00136	-0.1591	0.3467	0.4607
-1.750	0.5215	0.00578	0.00134	-0.1589	0.3491	0.4579
-2.000	0.4914	0.00574	0.00132	-0.1586	0.3536	0.4534
-2.250	0.4615	0.00569	0.00129	-0.1584	0.3582	0.4491
-2.750	0.4007	0.00566	0.00126	-0.1576	0.3697	0.4208
-3.000	0.3614	0.00621	0.00145	-0.1547	0.3768	0.2856
-3.250	0.3264	0.00651	0.00157	-0.1529	0.3846	0.2122
-3.750	0.2585	0.00701	0.00182	-0.1500	0.3989	0.1077
-4.000	0.2259	0.00719	0.00193	-0.1489	0.4082	0.0752
-4.250	0.1938	0.00734	0.00203	-0.1480	0.4164	0.0533
-4.500	0.1621	0.00747	0.00213	-0.1472	0.4237	0.0381
-4.750	0.1309	0.00757	0.00223	-0.1465	0.4320	0.0286
-5.000	0.0997	0.00769	0.00233	-0.1458	0.4380	0.0219
-5.250	0.0688	0.00778	0.00244	-0.1452	0.4467	0.0173
-5.500	0.0380	0.00788	0.00254	-0.1446	0.4551	0.0142
-6.000	-0.0232	0.00808	0.00277	-0.1435	0.4748	0.0101
-6.250	-0.0536	0.00817	0.00290	-0.1430	0.4866	0.0088
-6.500	-0.0838	0.00826	0.00303	-0.1425	0.4995	0.0082
-6.750	-0.1138	0.00836	0.00316	-0.1420	0.5149	0.0075
-7.000	-0.1435	0.00846	0.00332	-0.1416	0.5339	0.0069
-7.250	-0.1730	0.00854	0.00347	-0.1412	0.5554	0.0067
-7.500	-0.2020	0.00862	0.00362	-0.1409	0.5794	0.0066
-7.750	-0.2310	0.00871	0.00378	-0.1405	0.6042	0.0064
-8.000	-0.2598	0.00881	0.00397	-0.1402	0.6286	0.0063
-8.250	-0.2886	0.00895	0.00417	-0.1398	0.6521	0.0058
-8.500	-0.3171	0.00909	0.00438	-0.1395	0.6753	0.0059
-8.750	-0.3455	0.00928	0.00465	-0.1391	0.6983	0.0057

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-9.000	-0.3734	0.00950	0.00495	-0.1388	0.7206	0.0055
-9.250	-0.4006	0.00964	0.00516	-0.1387	0.7459	0.0054
-9.500	-0.4274	0.00982	0.00542	-0.1385	0.7698	0.0054
-9.750	-0.4531	0.01006	0.00575	-0.1385	0.7962	0.0054
-10.000	-0.4778	0.01024	0.00601	-0.1387	0.8223	0.0053
-10.250	-0.5006	0.01056	0.00645	-0.1388	0.8539	0.0053
-10.500	-0.5186	0.01095	0.00700	-0.1395	0.8956	0.0053
-10.750	-0.5643	0.01135	0.00758	-0.1348	0.9421	0.0053
-11.000	-0.6376	0.01176	0.00799	-0.1252	0.9510	0.0050
-11.250	-0.6944	0.01243	0.00867	-0.1184	0.9559	0.0049
-11.500	-0.7364	0.01318	0.00943	-0.1143	0.9610	0.0050
-11.750	-0.7683	0.01399	0.01024	-0.1120	0.9655	0.0047
-12.250	-0.8040	0.01557	0.01185	-0.1125	0.9770	0.0048
-12.500	-0.8294	0.01660	0.01290	-0.1108	0.9786	0.0046
-12.750	-0.8311	0.01737	0.01368	-0.1141	0.9831	0.0047
-13.000	-0.8611	0.01826	0.01458	-0.1117	0.9839	0.0046
-13.250	-0.8876	0.01915	0.01548	-0.1100	0.9848	0.0048
-13.500	-0.9134	0.02027	0.01661	-0.1079	0.9864	0.0046
-13.750	-0.9191	0.02160	0.01799	-0.1091	0.9896	0.0044
-14.000	-0.9471	0.02268	0.01908	-0.1066	0.9900	0.0044
-14.250	-0.9827	0.02388	0.02030	-0.1025	0.9905	0.0044
-14.500	-1.0075	0.02557	0.02204	-0.0992	0.9912	0.0044
-14.750	-1.0261	0.02820	0.02479	-0.0949	0.9925	0.0042
-15.000	-1.0530	0.02901	0.02557	-0.0933	0.9933	0.0042

Listing B.9: Trial 41 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 41
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.7299   0.00613   0.00155  -0.1603   0.3248   0.4730
14      0.250   0.7595   0.00619   0.00159  -0.1605   0.3208   0.4738
15      0.500   0.7886   0.00628   0.00164  -0.1606   0.3144   0.4743
16      0.750   0.8181   0.00634   0.00168  -0.1608   0.3097   0.4750
17      1.000   0.8472   0.00644   0.00174  -0.1609   0.3013   0.4765
18      1.250   0.8758   0.00657   0.00181  -0.1609   0.2882   0.4778
19      1.500   0.9036   0.00677   0.00192  -0.1608   0.2714   0.4790
20      1.750   0.9317   0.00695   0.00203  -0.1608   0.2566   0.4802
21      2.000   0.9595   0.00714   0.00215  -0.1607   0.2418   0.4811
22      2.250   0.9873   0.00733   0.00227  -0.1606   0.2285   0.4820
23      2.750   1.0432   0.00768   0.00250  -0.1604   0.2065   0.4836
24      3.000   1.0712   0.00785   0.00262  -0.1604   0.1979   0.4843
25      3.250   1.0993   0.00800   0.00273  -0.1604   0.1902   0.4850
26      3.500   1.1272   0.00816   0.00284  -0.1603   0.1828   0.4856
27      3.750   1.1550   0.00833   0.00297  -0.1602   0.1753   0.4861
28      4.000   1.1829   0.00848   0.00308  -0.1601   0.1698   0.4866
29      4.250   1.2108   0.00863   0.00320  -0.1601   0.1644   0.4872
30      4.500   1.2386   0.00879   0.00333  -0.1600   0.1590   0.4891
31      4.750   1.2667   0.00893   0.00345  -0.1600   0.1549   0.4909
32      5.000   1.2944   0.00909   0.00359  -0.1600   0.1506   0.4924
33      5.250   1.3222   0.00923   0.00372  -0.1599   0.1468   0.4938
34      5.500   1.3499   0.00937   0.00384  -0.1598   0.1440   0.4951
35      5.750   1.3771   0.00955   0.00399  -0.1597   0.1400   0.4964
36      6.000   1.4048   0.00968   0.00412  -0.1596   0.1379   0.4975
37      6.250   1.4322   0.00983   0.00425  -0.1595   0.1358   0.4985
38      6.500   1.4591   0.01001   0.00441  -0.1593   0.1331   0.4991
39      6.750   1.4863   0.01016   0.00456  -0.1592   0.1304   0.5012
40      7.000   1.5138   0.01029   0.00469  -0.1591   0.1294   0.5038
41      7.250   1.5410   0.01044   0.00484  -0.1590   0.1278   0.5058
42      7.500   1.5677   0.01061   0.00501  -0.1588   0.1260   0.5077
43      7.750   1.5941   0.01080   0.00518  -0.1586   0.1237   0.5094
44      8.000   1.6208   0.01096   0.00534  -0.1584   0.1224   0.5108
45      8.250   1.6475   0.01110   0.00549  -0.1582   0.1218   0.5118
46      8.500   1.6741   0.01125   0.00565  -0.1580   0.1210   0.5148
47      8.750   1.7004   0.01142   0.00583  -0.1577   0.1199   0.5178
48      9.000   1.7263   0.01161   0.00602  -0.1574   0.1187   0.5203
49      9.250   1.7518   0.01181   0.00622  -0.1570   0.1172   0.5225
50      9.500   1.7770   0.01202   0.00642  -0.1566   0.1156   0.5242
51      9.750   1.8025   0.01220   0.00662  -0.1563   0.1149   0.5268
52      10.000   1.8283   0.01236   0.00680  -0.1559   0.1146   0.5306
53      10.250   1.8537   0.01253   0.00699  -0.1556   0.1143   0.5337
54      10.500   1.8788   0.01271   0.00718  -0.1551   0.1139   0.5363
55      10.750   1.9034   0.01290   0.00739  -0.1546   0.1134   0.5383
56      11.000   1.9279   0.01310   0.00761  -0.1541   0.1127   0.5430
57      11.250   1.9517   0.01332   0.00784  -0.1535   0.1119   0.5467
58      11.500   1.9748   0.01357   0.00810  -0.1528   0.1108   0.5497
59      11.750   1.9972   0.01384   0.00837  -0.1519   0.1094   0.5529
60      12.000   2.0179   0.01412   0.00867  -0.1507   0.1078   0.5577
61      12.250   2.0386   0.01432   0.00889  -0.1495   0.1075   0.5616
62      12.500   2.0587   0.01453   0.00913  -0.1482   0.1071   0.5645
63      12.750   2.0789   0.01476   0.00939  -0.1470   0.1065   0.5700
64
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.0983	0.01504	0.00969	-0.1456	0.1055	0.5746
13.250	2.1163	0.01537	0.01003	-0.1440	0.1043	0.5780
13.500	2.1337	0.01576	0.01043	-0.1424	0.1024	0.5839
13.750	2.1508	0.01616	0.01085	-0.1407	0.1008	0.5891
14.000	2.1703	0.01645	0.01117	-0.1395	0.1003	0.5940
14.250	2.1883	0.01681	0.01156	-0.1381	0.0990	0.6004
14.500	2.2033	0.01732	0.01208	-0.1362	0.0967	0.6047
14.750	2.2178	0.01788	0.01265	-0.1343	0.0942	0.6114
15.000	2.2342	0.01833	0.01313	-0.1328	0.0930	0.6169
15.250	2.2467	0.01900	0.01381	-0.1307	0.0900	0.6231
15.500	2.2550	0.01990	0.01470	-0.1282	0.0848	0.6289
15.750	2.2587	0.02110	0.01588	-0.1251	0.0788	0.6345
16.000	2.2578	0.02263	0.01740	-0.1217	0.0723	0.6406
16.250	2.2508	0.02464	0.01939	-0.1180	0.0645	0.6454
16.500	2.2445	0.02678	0.02153	-0.1147	0.0585	0.6518
17.000	2.2392	0.03102	0.02582	-0.1100	0.0507	0.6644
17.250	2.2287	0.03404	0.02886	-0.1076	0.0458	0.6697
17.500	2.2203	0.03714	0.03199	-0.1059	0.0419	0.6765
17.750	2.2130	0.04034	0.03524	-0.1047	0.0389	0.6824
18.000	2.2060	0.04373	0.03868	-0.1038	0.0363	0.6900
18.250	2.1965	0.04757	0.04257	-0.1033	0.0339	0.6962
18.500	2.1860	0.05176	0.04683	-0.1031	0.0319	0.7034
18.750	2.1703	0.05678	0.05192	-0.1033	0.0294	0.7096
19.000	2.1589	0.06149	0.05671	-0.1038	0.0278	0.7167
19.250	2.1418	0.06710	0.06239	-0.1047	0.0259	0.7225
19.500	2.1253	0.07285	0.06823	-0.1059	0.0246	0.7300
19.750	2.1071	0.07897	0.07445	-0.1074	0.0231	0.7378
20.000	2.0887	0.08525	0.08082	-0.1091	0.0220	0.7422
0.000	0.7299	0.00613	0.00155	-0.1603	0.3248	0.4730
-0.250	0.7004	0.00607	0.00152	-0.1601	0.3285	0.4718
-0.500	0.6709	0.00600	0.00148	-0.1600	0.3330	0.4702
-0.750	0.6410	0.00596	0.00145	-0.1598	0.3362	0.4679
-1.000	0.6109	0.00593	0.00142	-0.1595	0.3386	0.4649
-1.250	0.5811	0.00587	0.00139	-0.1593	0.3430	0.4622
-1.500	0.5515	0.00582	0.00136	-0.1591	0.3467	0.4607
-1.750	0.5215	0.00578	0.00134	-0.1589	0.3491	0.4579
-2.000	0.4914	0.00574	0.00132	-0.1586	0.3536	0.4534
-2.250	0.4615	0.00569	0.00129	-0.1584	0.3582	0.4491
-2.750	0.4007	0.00566	0.00126	-0.1576	0.3697	0.4208
-3.000	0.3614	0.00621	0.00145	-0.1547	0.3768	0.2856
-3.250	0.3264	0.00651	0.00157	-0.1529	0.3846	0.2122
-3.750	0.2585	0.00701	0.00182	-0.1500	0.3989	0.1077
-4.000	0.2259	0.00719	0.00193	-0.1489	0.4082	0.0752
-4.250	0.1938	0.00734	0.00203	-0.1480	0.4164	0.0533
-4.500	0.1621	0.00747	0.00213	-0.1472	0.4237	0.0381
-4.750	0.1309	0.00757	0.00223	-0.1465	0.4320	0.0286
-5.000	0.0997	0.00769	0.00233	-0.1458	0.4380	0.0219
-5.250	0.0688	0.00778	0.00244	-0.1452	0.4467	0.0173
-5.500	0.0380	0.00788	0.00254	-0.1446	0.4551	0.0142
-6.000	-0.0232	0.00808	0.00277	-0.1435	0.4748	0.0101
-6.250	-0.0536	0.00817	0.00290	-0.1430	0.4866	0.0088
-6.500	-0.0838	0.00826	0.00303	-0.1425	0.4995	0.0082
-6.750	-0.1138	0.00836	0.00316	-0.1420	0.5149	0.0075
-7.000	-0.1435	0.00846	0.00332	-0.1416	0.5339	0.0069
-7.250	-0.1730	0.00854	0.00347	-0.1412	0.5554	0.0067
-7.500	-0.2020	0.00862	0.00362	-0.1409	0.5794	0.0066
-7.750	-0.2310	0.00871	0.00378	-0.1405	0.6042	0.0064
-8.000	-0.2598	0.00881	0.00397	-0.1402	0.6286	0.0063
-8.250	-0.2886	0.00895	0.00417	-0.1398	0.6521	0.0058
-8.500	-0.3171	0.00909	0.00438	-0.1395	0.6753	0.0059
-8.750	-0.3455	0.00928	0.00465	-0.1391	0.6983	0.0057

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-9.000	-0.3734	0.00950	0.00495	-0.1388	0.7206	0.0055
-9.250	-0.4006	0.00964	0.00516	-0.1387	0.7459	0.0054
-9.500	-0.4274	0.00982	0.00542	-0.1385	0.7698	0.0054
-9.750	-0.4531	0.01006	0.00575	-0.1385	0.7962	0.0054
-10.000	-0.4778	0.01024	0.00601	-0.1387	0.8223	0.0053
-10.250	-0.5006	0.01056	0.00645	-0.1388	0.8539	0.0053
-10.500	-0.5186	0.01095	0.00700	-0.1395	0.8956	0.0053
-10.750	-0.5643	0.01135	0.00758	-0.1348	0.9421	0.0053
-11.000	-0.6376	0.01176	0.00799	-0.1252	0.9510	0.0050
-11.250	-0.6944	0.01243	0.00867	-0.1184	0.9559	0.0049
-11.500	-0.7364	0.01318	0.00943	-0.1143	0.9610	0.0050
-11.750	-0.7683	0.01399	0.01024	-0.1120	0.9655	0.0047
-12.250	-0.8040	0.01557	0.01185	-0.1125	0.9770	0.0048
-12.500	-0.8294	0.01660	0.01290	-0.1108	0.9786	0.0046
-12.750	-0.8311	0.01737	0.01368	-0.1141	0.9831	0.0047
-13.000	-0.8611	0.01826	0.01458	-0.1117	0.9839	0.0046
-13.250	-0.8876	0.01915	0.01548	-0.1100	0.9848	0.0048
-13.500	-0.9134	0.02027	0.01661	-0.1079	0.9864	0.0046
-13.750	-0.9191	0.02160	0.01799	-0.1091	0.9896	0.0044
-14.000	-0.9471	0.02268	0.01908	-0.1066	0.9900	0.0044
-14.250	-0.9827	0.02388	0.02030	-0.1025	0.9905	0.0044
-14.500	-1.0075	0.02557	0.02204	-0.0992	0.9912	0.0044
-14.750	-1.0261	0.02820	0.02479	-0.0949	0.9925	0.0042
-15.000	-1.0530	0.02901	0.02557	-0.0933	0.9933	0.0042

Listing B.10: Trial 42 XFOIL Polar File.

```
1
2       XFOIL         Version 6.99
3
4       Calculated polar for: TRIAL 42
5
6       1 1 Reynolds number fixed          Mach number fixed
7
8       xtrf =   1.000 (top)        1.000 (bottom)
9       Mach =   0.000     Re =     6.000 e 6     Ncrit =   9.000
10
11      alpha    CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
12      ------ -------- --------- --------- -------- -------- --------
13      0.000   0.8316   0.00627   0.00170  -0.1847   0.3433   0.4654
14      0.250   0.8609   0.00634   0.00175  -0.1849   0.3388   0.4665
15      0.500   0.8902   0.00641   0.00179  -0.1851   0.3347   0.4673
16      0.750   0.9190   0.00651   0.00185  -0.1851   0.3282   0.4678
17      1.000   0.9482   0.00658   0.00190  -0.1853   0.3234   0.4683
18      1.250   0.9770   0.00668   0.00196  -0.1854   0.3154   0.4698
19      1.500   1.0053   0.00682   0.00204  -0.1854   0.3039   0.4712
20      1.750   1.0329   0.00701   0.00216  -0.1853   0.2877   0.4726
21      2.000   1.0602   0.00722   0.00228  -0.1852   0.2717   0.4738
22      2.250   1.0877   0.00741   0.00241  -0.1851   0.2579   0.4748
23      2.500   1.1151   0.00760   0.00254  -0.1849   0.2451   0.4758
24      2.750   1.1426   0.00779   0.00266  -0.1848   0.2339   0.4767
25      3.000   1.1701   0.00797   0.00279  -0.1847   0.2237   0.4775
26      3.250   1.1975   0.00814   0.00291  -0.1846   0.2145   0.4783
27      3.500   1.2249   0.00832   0.00304  -0.1844   0.2060   0.4790
28      3.750   1.2521   0.00851   0.00318  -0.1843   0.1975   0.4796
29      4.000   1.2794   0.00868   0.00331  -0.1841   0.1909   0.4801
30      4.250   1.3069   0.00883   0.00343  -0.1840   0.1848   0.4808
31      4.500   1.3340   0.00902   0.00358  -0.1839   0.1778   0.4828
32      4.750   1.3617   0.00916   0.00370  -0.1838   0.1737   0.4847
33      5.000   1.3887   0.00934   0.00385  -0.1837   0.1679   0.4863
34      5.250   1.4161   0.00948   0.00398  -0.1836   0.1646   0.4878
35      5.500   1.4427   0.00967   0.00414  -0.1834   0.1596   0.4892
36      5.750   1.4700   0.00981   0.00427  -0.1833   0.1570   0.4905
37      6.000   1.4967   0.00998   0.00442  -0.1830   0.1537   0.4916
38      6.250   1.5231   0.01017   0.00458  -0.1828   0.1496   0.4924
39      6.750   1.5766   0.01047   0.00488  -0.1824   0.1455   0.4965
40      7.000   1.6028   0.01066   0.00505  -0.1822   0.1423   0.4990
41      7.250   1.6292   0.01081   0.00521  -0.1819   0.1404   0.5011
42      7.500   1.6556   0.01097   0.00536  -0.1817   0.1391   0.5030
43      7.750   1.6815   0.01114   0.00553  -0.1814   0.1375   0.5046
44      8.000   1.7068   0.01134   0.00572  -0.1810   0.1354   0.5057
45      8.250   1.7322   0.01154   0.00592  -0.1806   0.1327   0.5090
46      8.500   1.7581   0.01169   0.00608  -0.1803   0.1320   0.5120
47      8.750   1.7836   0.01185   0.00626  -0.1799   0.1312   0.5146
48      9.000   1.8087   0.01203   0.00644  -0.1795   0.1302   0.5168
49      9.250   1.8332   0.01223   0.00664  -0.1789   0.1291   0.5185
50      9.500   1.8575   0.01244   0.00686  -0.1784   0.1278   0.5222
51      9.750   1.8813   0.01267   0.00709  -0.1777   0.1261   0.5257
52      10.000   1.9046   0.01290   0.00733  -0.1770   0.1245   0.5287
53      10.250   1.9269   0.01307   0.00752  -0.1760   0.1242   0.5311
54      10.500   1.9485   0.01325   0.00773  -0.1750   0.1238   0.5345
55      10.750   1.9698   0.01345   0.00795  -0.1738   0.1231   0.5387
56      11.000   1.9906   0.01368   0.00820  -0.1726   0.1221   0.5422
57      11.250   2.0109   0.01394   0.00847  -0.1714   0.1210   0.5448
58      11.500   2.0302   0.01425   0.00878  -0.1700   0.1194   0.5495
59      11.750   2.0493   0.01458   0.00912  -0.1685   0.1176   0.5539
60      12.000   2.0694   0.01486   0.00942  -0.1673   0.1165   0.5574
61      12.250   2.0898   0.01512   0.00971  -0.1662   0.1160   0.5619
62      12.500   2.1097   0.01541   0.01003  -0.1650   0.1151   0.5669
63      12.750   2.1284   0.01576   0.01040  -0.1636   0.1138   0.5709
64
```

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
13.000	2.1456	0.01619	0.01083	-0.1619	0.1120	0.5759
13.250	2.1617	0.01668	0.01133	-0.1602	0.1093	0.5811
13.500	2.1805	0.01702	0.01171	-0.1589	0.1086	0.5852
13.750	2.1984	0.01743	0.01215	-0.1575	0.1075	0.5916
14.000	2.2136	0.01797	0.01270	-0.1558	0.1053	0.5970
14.250	2.2267	0.01864	0.01337	-0.1538	0.1018	0.6026
14.500	2.2400	0.01930	0.01404	-0.1518	0.0986	0.6092
14.750	2.2511	0.02010	0.01485	-0.1497	0.0953	0.6147
15.000	2.2618	0.02096	0.01572	-0.1475	0.0915	0.6216
15.250	2.2544	0.02293	0.01762	-0.1432	0.0796	0.6260
15.500	2.2515	0.02476	0.01943	-0.1398	0.0725	0.6322
15.750	2.2498	0.02663	0.02130	-0.1368	0.0664	0.6372
16.000	2.2475	0.02867	0.02335	-0.1341	0.0609	0.6437
16.250	2.2413	0.03115	0.02583	-0.1313	0.0549	0.6492
16.750	2.2321	0.03639	0.03111	-0.1271	0.0461	0.6615
17.000	2.2243	0.03957	0.03432	-0.1254	0.0416	0.6677
17.250	2.2195	0.04263	0.03741	-0.1241	0.0386	0.6742
17.500	2.2093	0.04643	0.04125	-0.1229	0.0347	0.6806
17.750	2.2065	0.04957	0.04445	-0.1223	0.0329	0.6877
18.000	2.1952	0.05386	0.04879	-0.1217	0.0298	0.6945
18.250	2.1871	0.05791	0.05290	-0.1215	0.0279	0.7013
18.500	2.1755	0.06258	0.05763	-0.1215	0.0258	0.7086
18.750	2.1640	0.06739	0.06252	-0.1218	0.0240	0.7150
19.000	2.1521	0.07239	0.06759	-0.1224	0.0224	0.7225
19.250	2.1377	0.07786	0.07314	-0.1232	0.0208	0.7290
19.500	2.1219	0.08367	0.07902	-0.1243	0.0192	0.7365
19.750	2.1088	0.08917	0.08462	-0.1256	0.0181	0.7435
20.000	2.0939	0.09501	0.09054	-0.1271	0.0170	0.7515
0.000	0.8316	0.00627	0.00170	-0.1847	0.3434	0.4654
-0.250	0.8024	0.00620	0.00166	-0.1846	0.3478	0.4638
-0.500	0.7727	0.00615	0.00162	-0.1844	0.3508	0.4616
-0.750	0.7430	0.00610	0.00158	-0.1841	0.3551	0.4586
-1.000	0.7133	0.00604	0.00155	-0.1839	0.3592	0.4557
-1.250	0.6838	0.00599	0.00152	-0.1837	0.3622	0.4542
-1.500	0.6542	0.00593	0.00149	-0.1835	0.3667	0.4514
-1.750	0.6244	0.00588	0.00145	-0.1832	0.3712	0.4474
-2.000	0.5944	0.00584	0.00143	-0.1830	0.3748	0.4426
-2.250	0.5644	0.00580	0.00140	-0.1826	0.3808	0.4331
-2.500	0.5331	0.00582	0.00138	-0.1820	0.3866	0.4013
-2.750	0.4948	0.00620	0.00150	-0.1793	0.3963	0.2885
-3.000	0.4579	0.00653	0.00162	-0.1771	0.4058	0.1993
-3.250	0.4235	0.00674	0.00171	-0.1755	0.4146	0.1438
-3.500	0.3902	0.00690	0.00179	-0.1742	0.4243	0.1025
-3.750	0.3573	0.00704	0.00188	-0.1731	0.4353	0.0698
-4.000	0.3252	0.00714	0.00196	-0.1721	0.4462	0.0486
-4.250	0.2936	0.00723	0.00204	-0.1713	0.4569	0.0344
-4.500	0.2622	0.00731	0.00212	-0.1706	0.4670	0.0251
-4.750	0.2308	0.00740	0.00220	-0.1699	0.4762	0.0187
-5.000	0.1999	0.00747	0.00229	-0.1692	0.4867	0.0146
-5.250	0.1688	0.00754	0.00238	-0.1685	0.4973	0.0114
-5.500	0.1380	0.00761	0.00248	-0.1679	0.5088	0.0092
-5.750	0.1073	0.00768	0.00259	-0.1673	0.5213	0.0082
-6.000	0.0768	0.00775	0.00269	-0.1668	0.5354	0.0076
-6.250	0.0464	0.00781	0.00280	-0.1663	0.5508	0.0073
-6.500	0.0161	0.00789	0.00293	-0.1657	0.5672	0.0069
-6.750	-0.0143	0.00798	0.00309	-0.1651	0.5845	0.0064
-7.000	-0.0445	0.00807	0.00322	-0.1646	0.6010	0.0063
-7.500	-0.1047	0.00827	0.00353	-0.1635	0.6326	0.0060
-7.750	-0.1348	0.00839	0.00370	-0.1629	0.6485	0.0059
-8.000	-0.1647	0.00852	0.00388	-0.1624	0.6643	0.0059
-8.250	-0.1948	0.00868	0.00408	-0.1617	0.6795	0.0056
-8.500	-0.2248	0.00887	0.00433	-0.1611	0.6946	0.0055
-8.750	-0.2542	0.00904	0.00455	-0.1605	0.7101	0.0055

Continued on next page.

alpha	CL	CD	CDp	CM	Top_Xtr	Bot_Xtr
-9.000	-0.2836	0.00927	0.00484	-0.1599	0.7260	0.0053
-9.250	-0.3128	0.00952	0.00515	-0.1592	0.7416	0.0052
-9.500	-0.3416	0.00980	0.00549	-0.1586	0.7576	0.0051
-9.750	-0.3696	0.01013	0.00590	-0.1579	0.7753	0.0051
-10.000	-0.3969	0.01041	0.00624	-0.1574	0.7925	0.0050
-10.250	-0.4247	0.01094	0.00687	-0.1562	0.8129	0.0050
-10.500	-0.4504	0.01134	0.00736	-0.1555	0.8371	0.0050
-10.750	-0.4732	0.01173	0.00783	-0.1553	0.8616	0.0050
-11.000	-0.4882	0.01229	0.00855	-0.1562	0.8980	0.0050
-11.250	-0.5291	0.01262	0.00901	-0.1526	0.9351	0.0049
-11.500	-0.5982	0.01342	0.00987	-0.1429	0.9474	0.0049
-11.750	-0.6506	0.01410	0.01057	-0.1368	0.9527	0.0049
-12.000	-0.7005	0.01481	0.01129	-0.1312	0.9568	0.0049
-12.250	-0.7374	0.01546	0.01194	-0.1283	0.9614	0.0048
-12.500	-0.7579	0.01671	0.01324	-0.1271	0.9684	0.0049
-12.750	-0.7815	0.01716	0.01366	-0.1270	0.9722	0.0047
-13.500	-0.8173	0.02023	0.01679	-0.1291	0.9841	0.0047
-13.750	-0.8417	0.02182	0.01845	-0.1260	0.9849	0.0048
-14.000	-0.8741	0.02279	0.01941	-0.1230	0.9856	0.0047
-14.250	-0.9034	0.02357	0.02019	-0.1210	0.9864	0.0046
-14.500	-0.8983	0.02599	0.02272	-0.1215	0.9900	0.0048
-14.750	-0.9264	0.02681	0.02353	-0.1195	0.9902	0.0047
-15.000	-0.9519	0.02777	0.02449	-0.1175	0.9905	0.0046

APPENDIX C

FINAL SOLUTION COORDINATES FOR TRIALS 33–42

Table C.1: Trial 33 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.690 75	0.088 50	0.273 02	0.160 89
0.999 76	−0.000 00	0.679 19	0.093 07	0.263 47	0.159 05
0.999 03	0.000 00	0.667 65	0.097 64	0.254 00	0.157 08
0.997 81	0.000 02	0.656 15	0.102 20	0.244 63	0.155 00
0.996 11	0.000 07	0.644 67	0.106 72	0.235 36	0.152 80
0.993 90	0.000 16	0.633 25	0.111 20	0.226 19	0.150 50
0.991 21	0.000 30	0.621 87	0.115 63	0.217 14	0.148 09
0.988 01	0.000 51	0.610 56	0.119 99	0.208 21	0.145 58
0.984 33	0.000 79	0.599 31	0.124 26	0.199 40	0.142 97
0.980 16	0.001 17	0.588 13	0.128 45	0.190 73	0.140 26
0.975 50	0.001 66	0.577 04	0.132 53	0.182 19	0.137 46
0.970 38	0.002 27	0.566 02	0.136 49	0.173 79	0.134 58
0.964 79	0.003 02	0.555 10	0.140 31	0.165 55	0.131 61
0.958 76	0.003 92	0.544 27	0.144 00	0.157 46	0.128 57
0.952 30	0.004 98	0.533 54	0.147 52	0.149 53	0.125 44
0.945 43	0.006 20	0.522 91	0.150 88	0.141 76	0.122 25
0.938 16	0.007 60	0.512 38	0.154 05	0.134 16	0.118 98
0.930 51	0.009 16	0.501 96	0.157 03	0.126 74	0.115 66
0.922 51	0.010 91	0.491 66	0.159 80	0.119 51	0.112 27
0.914 16	0.012 83	0.481 47	0.162 34	0.112 45	0.108 82
0.905 48	0.014 94	0.471 40	0.164 63	0.105 59	0.105 32
0.896 49	0.017 23	0.461 45	0.166 65	0.098 93	0.101 77
0.887 22	0.019 69	0.451 63	0.168 32	0.092 47	0.098 17
0.877 67	0.022 34	0.441 82	0.169 61	0.086 22	0.094 52
0.867 86	0.025 16	0.431 92	0.170 59	0.080 16	0.090 81
0.857 82	0.028 15	0.421 98	0.171 32	0.074 30	0.087 07
0.847 57	0.031 32	0.411 99	0.171 83	0.068 65	0.083 30
0.837 11	0.034 64	0.401 97	0.172 13	0.063 20	0.079 49
0.826 48	0.038 12	0.391 94	0.172 24	0.057 95	0.075 67
0.815 68	0.041 74	0.381 89	0.172 16	0.052 92	0.071 84
0.804 74	0.045 51	0.371 84	0.171 91	0.048 11	0.067 99
0.793 67	0.049 40	0.361 81	0.171 48	0.043 51	0.064 15
0.782 48	0.053 42	0.351 78	0.170 90	0.039 13	0.060 31
0.771 20	0.057 54	0.341 78	0.170 15	0.034 97	0.056 48
0.759 84	0.061 77	0.331 81	0.169 25	0.031 04	0.052 66
0.748 41	0.066 08	0.321 87	0.168 20	0.027 33	0.048 87
0.736 93	0.070 46	0.311 98	0.167 01	0.023 86	0.045 11
0.725 41	0.074 91	0.302 15	0.165 68	0.020 61	0.041 38
0.713 87	0.079 41	0.292 37	0.164 21	0.017 59	0.037 70
0.702 31	0.083 94	0.282 66	0.162 62	0.014 81	0.034 07

Continued on next page.

x	y	x	y	x	y
0.012 26	0.030 49	0.175 12	−0.013 32	0.712 49	−0.005 48
0.009 95	0.026 98	0.186 24	−0.013 04	0.725 16	−0.005 48
0.007 88	0.023 55	0.197 61	−0.012 74	0.737 65	−0.005 47
0.006 05	0.020 19	0.209 23	−0.012 42	0.749 94	−0.005 47
0.004 45	0.016 93	0.221 08	−0.012 10	0.762 02	−0.005 46
0.003 10	0.013 77	0.233 17	−0.011 76	0.773 89	−0.005 45
0.001 99	0.010 72	0.245 46	−0.011 42	0.785 53	−0.005 43
0.001 12	0.007 80	0.257 97	−0.011 07	0.796 94	−0.005 41
0.000 50	0.005 01	0.270 67	−0.010 71	0.808 11	−0.005 39
0.000 13	0.002 38	0.283 56	−0.010 35	0.819 03	−0.005 36
0.000 00	−0.000 08	0.296 63	−0.009 99	0.829 69	−0.005 32
0.000 12	−0.002 35	0.309 86	−0.009 64	0.840 09	−0.005 28
0.000 51	−0.004 39	0.323 25	−0.009 28	0.850 21	−0.005 23
0.001 15	−0.006 14	0.336 79	−0.008 93	0.860 06	−0.005 17
0.002 18	−0.007 49	0.350 46	−0.008 58	0.869 61	−0.005 10
0.003 76	−0.008 52	0.364 25	−0.008 25	0.878 87	−0.005 02
0.005 89	−0.009 45	0.378 16	−0.007 92	0.887 83	−0.004 94
0.008 51	−0.010 28	0.392 17	−0.007 61	0.896 48	−0.004 84
0.011 58	−0.011 02	0.406 26	−0.007 31	0.904 81	−0.004 73
0.015 08	−0.011 69	0.420 44	−0.007 02	0.912 81	−0.004 61
0.019 01	−0.012 28	0.434 69	−0.006 76	0.920 49	−0.004 47
0.023 35	−0.012 80	0.448 99	−0.006 52	0.927 84	−0.004 32
0.028 10	−0.013 25	0.463 34	−0.006 31	0.934 85	−0.004 16
0.033 25	−0.013 64	0.477 72	−0.006 15	0.941 50	−0.003 98
0.038 79	−0.013 97	0.492 08	−0.006 02	0.947 81	−0.003 78
0.044 71	−0.014 24	0.506 42	−0.005 91	0.953 76	−0.003 56
0.051 02	−0.014 46	0.520 74	−0.005 82	0.959 35	−0.003 31
0.057 69	−0.014 62	0.535 02	−0.005 75	0.964 58	−0.003 03
0.064 74	−0.014 74	0.549 25	−0.005 68	0.969 46	−0.002 70
0.072 14	−0.014 81	0.563 42	−0.005 63	0.974 01	−0.002 34
0.079 89	−0.014 84	0.577 52	−0.005 59	0.978 23	−0.001 96
0.087 99	−0.014 83	0.591 55	−0.005 56	0.982 11	−0.001 58
0.096 42	−0.014 79	0.605 48	−0.005 54	0.985 67	−0.001 22
0.105 19	−0.014 70	0.619 31	−0.005 52	0.988 87	−0.000 89
0.114 27	−0.014 59	0.633 03	−0.005 51	0.991 71	−0.000 61
0.123 68	−0.014 44	0.646 63	−0.005 50	0.994 17	−0.000 38
0.133 38	−0.014 26	0.660 10	−0.005 49	0.996 23	−0.000 21
0.143 39	−0.014 06	0.673 43	−0.005 49	0.997 86	−0.000 09
0.153 69	−0.013 84	0.686 62	−0.005 48	0.999 04	−0.000 03
0.164 27	−0.013 59	0.699 64	−0.005 48	0.999 76	−0.000 00

Table C.2: Trial 34 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.668 35	0.079 76	0.245 33	0.175 74
0.999 76	−0.000 03	0.656 07	0.084 55	0.236 78	0.174 15
0.999 05	−0.000 11	0.643 82	0.089 36	0.228 30	0.172 37
0.997 84	−0.000 23	0.631 61	0.094 18	0.219 90	0.170 41
0.996 14	−0.000 38	0.619 45	0.098 99	0.211 58	0.168 28
0.993 93	−0.000 56	0.607 37	0.103 78	0.203 36	0.165 99
0.991 19	−0.000 76	0.595 35	0.108 53	0.195 23	0.163 55
0.987 92	−0.000 95	0.583 41	0.113 23	0.187 20	0.160 96
0.984 11	−0.001 11	0.571 56	0.117 87	0.179 28	0.158 24
0.979 74	−0.001 23	0.559 80	0.122 43	0.171 48	0.155 39
0.974 82	−0.001 27	0.548 13	0.126 90	0.163 79	0.152 41
0.969 36	−0.001 20	0.536 57	0.131 26	0.156 23	0.149 31
0.963 36	−0.001 00	0.525 12	0.135 52	0.148 80	0.146 10
0.956 84	−0.000 63	0.513 78	0.139 64	0.141 51	0.142 79
0.949 85	−0.000 08	0.502 56	0.143 63	0.134 35	0.139 37
0.942 40	0.000 67	0.491 46	0.147 46	0.127 34	0.135 86
0.934 53	0.001 61	0.480 48	0.151 13	0.120 47	0.132 26
0.926 25	0.002 75	0.469 63	0.154 63	0.113 77	0.128 57
0.917 57	0.004 09	0.458 90	0.157 93	0.107 21	0.124 81
0.908 52	0.005 63	0.448 31	0.161 04	0.100 82	0.120 97
0.899 13	0.007 37	0.437 85	0.163 92	0.094 60	0.117 06
0.889 40	0.009 33	0.427 52	0.166 56	0.088 55	0.113 10
0.879 36	0.011 49	0.417 33	0.168 91	0.082 67	0.109 07
0.869 03	0.013 86	0.407 19	0.170 96	0.076 97	0.105 00
0.858 43	0.016 43	0.397 05	0.172 75	0.071 45	0.100 88
0.847 58	0.019 21	0.386 93	0.174 33	0.066 11	0.096 72
0.836 51	0.022 18	0.376 84	0.175 71	0.060 97	0.092 53
0.825 23	0.025 34	0.366 80	0.176 90	0.056 01	0.088 31
0.813 76	0.028 69	0.356 82	0.177 92	0.051 26	0.084 07
0.802 13	0.032 22	0.346 91	0.178 76	0.046 69	0.079 81
0.790 35	0.035 91	0.337 08	0.179 42	0.042 33	0.075 54
0.778 45	0.039 77	0.327 35	0.179 91	0.038 17	0.071 27
0.766 43	0.043 77	0.317 71	0.180 23	0.034 22	0.067 00
0.754 32	0.047 92	0.308 19	0.180 37	0.030 48	0.062 75
0.742 14	0.052 19	0.298 78	0.180 34	0.026 94	0.058 50
0.729 90	0.056 57	0.289 51	0.180 12	0.023 62	0.054 28
0.717 62	0.061 06	0.280 39	0.179 71	0.020 51	0.050 09
0.705 31	0.065 63	0.271 42	0.179 10	0.017 61	0.045 93
0.692 99	0.070 28	0.262 62	0.178 25	0.014 93	0.041 82
0.680 67	0.075 00	0.253 95	0.177 13	0.012 47	0.037 75

Continued on next page.

x	y	x	y	x	y
0.010 23	0.033 75	0.172 37	−0.007 68	0.730 62	−0.004 04
0.008 21	0.029 80	0.183 87	−0.006 81	0.743 01	−0.004 66
0.006 41	0.025 94	0.195 67	−0.005 94	0.755 16	−0.005 24
0.004 83	0.022 15	0.207 74	−0.005 07	0.767 07	−0.005 80
0.003 47	0.018 45	0.220 08	−0.004 19	0.778 73	−0.006 32
0.002 34	0.014 86	0.232 68	−0.003 33	0.790 13	−0.006 80
0.001 43	0.011 38	0.245 53	−0.002 47	0.801 28	−0.007 24
0.000 74	0.008 01	0.258 62	−0.001 62	0.812 16	−0.007 65
0.000 28	0.004 78	0.271 94	−0.000 79	0.822 77	−0.008 01
0.000 04	0.001 70	0.285 47	0.000 02	0.833 11	−0.008 33
0.000 02	−0.001 22	0.299 20	0.000 80	0.843 16	−0.008 60
0.000 23	−0.003 96	0.313 14	0.001 55	0.852 94	−0.008 83
0.000 66	−0.006 51	0.327 25	0.002 27	0.862 42	−0.009 01
0.001 32	−0.008 82	0.341 53	0.002 95	0.871 61	−0.009 14
0.002 20	−0.010 87	0.355 98	0.003 59	0.880 50	−0.009 23
0.003 32	−0.012 60	0.370 57	0.004 18	0.889 10	−0.009 26
0.004 79	−0.013 86	0.385 30	0.004 71	0.897 39	−0.009 25
0.006 80	−0.014 73	0.400 15	0.005 18	0.905 36	−0.009 19
0.009 38	−0.015 42	0.415 12	0.005 59	0.913 03	−0.009 09
0.012 45	−0.015 96	0.430 18	0.005 92	0.920 38	−0.008 94
0.016 00	−0.016 37	0.445 33	0.006 17	0.927 42	−0.008 74
0.020 00	−0.016 65	0.460 55	0.006 32	0.934 13	−0.008 49
0.024 45	−0.016 82	0.475 84	0.006 33	0.940 51	−0.008 20
0.029 34	−0.016 89	0.491 12	0.006 18	0.946 57	−0.007 87
0.034 66	−0.016 85	0.506 32	0.005 90	0.952 29	−0.007 49
0.040 39	−0.016 73	0.521 44	0.005 53	0.957 68	−0.007 07
0.046 55	−0.016 52	0.536 45	0.005 08	0.962 73	−0.006 60
0.053 11	−0.016 23	0.551 35	0.004 57	0.967 44	−0.006 06
0.060 08	−0.015 87	0.566 13	0.004 01	0.971 83	−0.005 46
0.067 44	−0.015 44	0.580 78	0.003 41	0.975 93	−0.004 80
0.075 18	−0.014 95	0.595 29	0.002 77	0.979 74	−0.004 11
0.083 31	−0.014 40	0.609 65	0.002 11	0.983 28	−0.003 41
0.091 82	−0.013 80	0.623 84	0.001 43	0.986 52	−0.002 73
0.100 69	−0.013 15	0.637 87	0.000 73	0.989 47	−0.002 09
0.109 92	−0.012 46	0.651 72	0.000 03	0.992 11	−0.001 52
0.119 50	−0.011 73	0.665 38	−0.000 68	0.994 42	−0.001 02
0.129 42	−0.010 97	0.678 84	−0.001 38	0.996 37	−0.000 63
0.139 68	−0.010 18	0.692 11	−0.002 07	0.997 93	−0.000 33
0.150 26	−0.009 36	0.705 17	−0.002 74	0.999 07	−0.000 14
0.161 16	−0.008 53	0.718 01	−0.003 40	0.999 76	−0.000 03

Table C.3: Trial 35 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.679 96	0.078 90	0.250 08	0.170 25
0.999 76	−0.000 01	0.667 80	0.083 34	0.241 31	0.168 79
0.999 05	−0.000 04	0.655 64	0.087 81	0.232 62	0.167 13
0.997 85	−0.000 09	0.643 51	0.092 27	0.224 02	0.165 30
0.996 16	−0.000 13	0.631 40	0.096 74	0.215 50	0.163 31
0.993 98	−0.000 16	0.619 33	0.101 18	0.207 08	0.161 17
0.991 30	−0.000 18	0.607 30	0.105 60	0.198 76	0.158 88
0.988 10	−0.000 16	0.595 33	0.109 98	0.190 54	0.156 44
0.984 40	−0.000 09	0.583 43	0.114 29	0.182 44	0.153 88
0.980 19	0.000 04	0.571 59	0.118 55	0.174 46	0.151 18
0.975 47	0.000 27	0.559 82	0.122 72	0.166 60	0.148 37
0.970 25	0.000 60	0.548 14	0.126 81	0.158 88	0.145 44
0.964 54	0.001 06	0.536 55	0.130 80	0.151 29	0.142 40
0.958 36	0.001 66	0.525 05	0.134 67	0.143 84	0.139 26
0.951 73	0.002 43	0.513 64	0.138 42	0.136 53	0.136 01
0.944 67	0.003 36	0.502 35	0.142 03	0.129 38	0.132 67
0.937 20	0.004 46	0.491 15	0.145 50	0.122 39	0.129 25
0.929 35	0.005 73	0.480 07	0.148 81	0.115 55	0.125 74
0.921 11	0.007 18	0.469 11	0.151 95	0.108 88	0.122 15
0.912 51	0.008 79	0.458 26	0.154 91	0.102 38	0.118 48
0.903 57	0.010 59	0.447 53	0.157 67	0.096 06	0.114 75
0.894 31	0.012 57	0.436 93	0.160 21	0.089 92	0.110 96
0.884 73	0.014 72	0.426 45	0.162 49	0.083 96	0.107 10
0.874 87	0.017 06	0.416 03	0.164 49	0.078 19	0.103 18
0.864 73	0.019 57	0.405 62	0.166 25	0.072 61	0.099 21
0.854 35	0.022 25	0.395 23	0.167 83	0.067 21	0.095 18
0.843 72	0.025 10	0.384 88	0.169 21	0.062 01	0.091 12
0.832 89	0.028 12	0.374 58	0.170 43	0.056 99	0.087 02
0.821 85	0.031 30	0.364 35	0.171 47	0.052 18	0.082 90
0.810 63	0.034 63	0.354 19	0.172 35	0.047 56	0.078 75
0.799 25	0.038 11	0.344 12	0.173 06	0.043 14	0.074 59
0.787 72	0.041 73	0.334 14	0.173 61	0.038 93	0.070 41
0.776 06	0.045 47	0.324 26	0.173 99	0.034 93	0.066 24
0.764 29	0.049 34	0.314 50	0.174 20	0.031 13	0.062 07
0.752 42	0.053 31	0.304 87	0.174 25	0.027 54	0.057 91
0.740 47	0.057 39	0.295 37	0.174 11	0.024 17	0.053 77
0.728 44	0.061 56	0.286 02	0.173 80	0.021 01	0.049 66
0.716 37	0.065 81	0.276 82	0.173 29	0.018 07	0.045 57
0.704 26	0.070 12	0.267 81	0.172 54	0.015 34	0.041 53
0.692 11	0.074 49	0.258 91	0.171 52	0.012 84	0.037 52

Continued on next page.

x	y	x	y	x	y
0.010 55	0.033 58	0.171 31	−0.012 23	0.719 98	−0.004 32
0.008 49	0.029 69	0.182 59	−0.011 61	0.732 58	−0.004 59
0.006 65	0.025 88	0.194 15	−0.010 97	0.744 96	−0.004 84
0.005 03	0.022 14	0.205 97	−0.010 32	0.757 13	−0.005 08
0.003 64	0.018 49	0.218 05	−0.009 67	0.769 07	−0.005 31
0.002 47	0.014 94	0.230 38	−0.009 01	0.780 78	−0.005 52
0.001 52	0.011 49	0.242 94	−0.008 35	0.792 24	−0.005 71
0.000 81	0.008 16	0.255 73	−0.007 69	0.803 46	−0.005 89
0.000 32	0.004 96	0.268 73	−0.007 04	0.814 42	−0.006 04
0.000 05	0.001 91	0.281 94	−0.006 39	0.825 12	−0.006 18
0.000 02	−0.000 98	0.295 34	−0.005 76	0.835 55	−0.006 29
0.000 21	−0.003 70	0.308 93	−0.005 14	0.845 71	−0.006 38
0.000 63	−0.006 22	0.322 69	−0.004 55	0.855 58	−0.006 45
0.001 28	−0.008 50	0.336 61	−0.003 97	0.865 16	−0.006 49
0.002 17	−0.010 53	0.350 68	−0.003 41	0.874 45	−0.006 51
0.003 30	−0.012 20	0.364 89	−0.002 89	0.883 44	−0.006 51
0.004 87	−0.013 44	0.379 22	−0.002 40	0.892 12	−0.006 48
0.007 02	−0.014 36	0.393 67	−0.001 94	0.900 49	−0.006 42
0.009 69	−0.015 15	0.408 23	−0.001 53	0.908 54	−0.006 34
0.012 85	−0.015 81	0.422 88	−0.001 16	0.916 27	−0.006 23
0.016 46	−0.016 35	0.437 61	−0.000 85	0.923 68	−0.006 09
0.020 52	−0.016 79	0.452 40	−0.000 60	0.930 75	−0.005 92
0.025 01	−0.017 13	0.467 26	−0.000 44	0.937 48	−0.005 73
0.029 93	−0.017 37	0.482 12	−0.000 38	0.943 87	−0.005 50
0.035 26	−0.017 54	0.496 94	−0.000 41	0.949 92	−0.005 25
0.041 00	−0.017 62	0.511 70	−0.000 50	0.955 62	−0.004 96
0.047 14	−0.017 62	0.526 40	−0.000 63	0.960 96	−0.004 63
0.053 68	−0.017 56	0.541 02	−0.000 80	0.965 95	−0.004 26
0.060 61	−0.017 43	0.555 56	−0.001 00	0.970 60	−0.003 82
0.067 91	−0.017 24	0.570 00	−0.001 23	0.974 94	−0.003 34
0.075 59	−0.016 99	0.584 34	−0.001 47	0.978 97	−0.002 84
0.083 64	−0.016 69	0.598 56	−0.001 74	0.982 69	−0.002 33
0.092 05	−0.016 34	0.612 67	−0.002 01	0.986 10	−0.001 84
0.100 81	−0.015 95	0.626 64	−0.002 30	0.989 18	−0.001 38
0.109 91	−0.015 51	0.640 47	−0.002 59	0.991 93	−0.000 98
0.119 35	−0.015 04	0.654 15	−0.002 88	0.994 31	−0.000 64
0.129 12	−0.014 53	0.667 67	−0.003 18	0.996 31	−0.000 38
0.139 21	−0.013 99	0.681 02	−0.003 47	0.997 90	−0.000 19
0.149 61	−0.013 43	0.694 20	−0.003 76	0.999 06	−0.000 07
0.160 31	−0.012 84	0.707 19	−0.004 04	0.999 76	−0.000 02

Table C.4: Trial 36 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.686 32	0.080 56	0.256 56	0.172 10
0.999 77	−0.000 00	0.674 28	0.084 99	0.247 70	0.170 66
0.999 07	−0.000 00	0.662 23	0.089 44	0.238 91	0.169 04
0.997 89	0.000 00	0.650 20	0.093 89	0.230 21	0.167 25
0.996 24	0.000 02	0.638 20	0.098 34	0.221 59	0.165 30
0.994 11	0.000 07	0.626 22	0.102 78	0.213 06	0.163 19
0.991 49	0.000 15	0.614 28	0.107 18	0.204 64	0.160 94
0.988 38	0.000 29	0.602 39	0.111 54	0.196 32	0.158 54
0.984 77	0.000 48	0.590 56	0.115 85	0.188 11	0.156 01
0.980 69	0.000 75	0.578 79	0.120 09	0.180 02	0.153 36
0.976 11	0.001 10	0.567 09	0.124 25	0.172 05	0.150 58
0.971 06	0.001 57	0.555 46	0.128 33	0.164 21	0.147 69
0.965 54	0.002 15	0.543 92	0.132 30	0.156 50	0.144 68
0.959 56	0.002 87	0.532 46	0.136 16	0.148 93	0.141 57
0.953 15	0.003 74	0.521 09	0.139 90	0.141 51	0.138 36
0.946 31	0.004 76	0.509 82	0.143 51	0.134 24	0.135 06
0.939 07	0.005 94	0.498 66	0.146 97	0.127 12	0.131 66
0.931 44	0.007 27	0.487 59	0.150 28	0.120 16	0.128 18
0.923 44	0.008 77	0.476 64	0.153 41	0.113 36	0.124 62
0.915 07	0.010 44	0.465 79	0.156 36	0.106 74	0.120 98
0.906 35	0.012 28	0.455 06	0.159 12	0.100 29	0.117 27
0.897 31	0.014 29	0.444 45	0.161 66	0.094 02	0.113 50
0.887 96	0.016 47	0.433 96	0.163 95	0.087 93	0.109 66
0.878 32	0.018 82	0.423 51	0.165 96	0.082 03	0.105 76
0.868 40	0.021 35	0.413 07	0.167 74	0.076 32	0.101 80
0.858 22	0.024 05	0.402 65	0.169 32	0.070 79	0.097 79
0.847 81	0.026 91	0.392 27	0.170 73	0.065 45	0.093 74
0.837 17	0.029 93	0.381 94	0.171 96	0.060 30	0.089 65
0.826 33	0.033 10	0.371 66	0.173 02	0.055 35	0.085 53
0.815 30	0.036 43	0.361 46	0.173 92	0.050 59	0.081 38
0.804 10	0.039 90	0.351 34	0.174 65	0.046 04	0.077 22
0.792 75	0.043 51	0.341 31	0.175 22	0.041 69	0.073 05
0.781 26	0.047 25	0.331 37	0.175 62	0.037 54	0.068 87
0.769 66	0.051 10	0.321 55	0.175 86	0.033 60	0.064 69
0.757 95	0.055 07	0.311 85	0.175 92	0.029 88	0.060 52
0.746 15	0.059 13	0.302 28	0.175 82	0.026 36	0.056 36
0.734 28	0.063 28	0.292 85	0.175 52	0.023 06	0.052 23
0.722 35	0.067 52	0.283 58	0.175 04	0.019 97	0.048 12
0.710 37	0.071 81	0.274 47	0.174 32	0.017 10	0.044 05
0.698 35	0.076 17	0.265 49	0.173 33	0.014 45	0.040 02

Continued on next page.

x	y	x	y	x	y
0.012 02	0.036 03	0.163 89	−0.011 71	0.711 58	0.000 30
0.009 81	0.032 11	0.174 99	−0.011 05	0.724 45	0.000 12
0.007 82	0.028 25	0.186 36	−0.010 38	0.737 12	−0.000 06
0.006 06	0.024 47	0.198 01	−0.009 69	0.749 58	−0.000 24
0.004 52	0.020 76	0.209 92	−0.009 00	0.761 83	−0.000 43
0.003 20	0.017 15	0.222 08	−0.008 30	0.773 85	−0.000 62
0.002 11	0.013 64	0.234 49	−0.007 59	0.785 64	−0.000 80
0.001 25	0.010 24	0.247 12	−0.006 89	0.797 18	−0.000 98
0.000 61	0.006 96	0.259 98	−0.006 19	0.808 48	−0.001 16
0.000 20	0.003 82	0.273 05	−0.005 49	0.819 51	−0.001 34
0.000 01	0.000 83	0.286 31	−0.004 80	0.830 27	−0.001 51
0.000 05	−0.001 99	0.299 77	−0.004 13	0.840 76	−0.001 68
0.000 32	−0.004 63	0.313 40	−0.003 47	0.850 97	−0.001 83
0.000 82	−0.007 06	0.327 20	−0.002 83	0.860 88	−0.001 98
0.001 55	−0.009 25	0.341 16	−0.002 21	0.870 49	−0.002 12
0.002 52	−0.011 16	0.355 26	−0.001 62	0.879 80	−0.002 25
0.003 74	−0.012 66	0.369 49	−0.001 05	0.888 79	−0.002 37
0.005 45	−0.013 75	0.383 84	−0.000 52	0.897 46	−0.002 47
0.007 78	−0.014 60	0.398 31	−0.000 02	0.905 80	−0.002 56
0.010 62	−0.015 31	0.412 87	0.000 44	0.913 82	−0.002 63
0.013 92	−0.015 90	0.427 51	0.000 85	0.921 49	−0.002 68
0.017 68	−0.016 37	0.442 23	0.001 20	0.928 82	−0.002 71
0.021 88	−0.016 74	0.457 01	0.001 48	0.935 79	−0.002 72
0.026 51	−0.017 02	0.471 81	0.001 67	0.942 42	−0.002 71
0.031 57	−0.017 20	0.486 59	0.001 78	0.948 68	−0.002 66
0.037 03	−0.017 29	0.501 32	0.001 85	0.954 58	−0.002 59
0.042 91	−0.017 31	0.516 01	0.001 88	0.960 11	−0.002 48
0.049 18	−0.017 25	0.530 65	0.001 88	0.965 28	−0.002 32
0.055 85	−0.017 12	0.545 22	0.001 85	0.970 09	−0.002 11
0.062 91	−0.016 93	0.559 71	0.001 80	0.974 57	−0.001 85
0.070 34	−0.016 68	0.574 13	0.001 72	0.978 72	−0.001 57
0.078 15	−0.016 37	0.588 45	0.001 63	0.982 54	−0.001 27
0.086 32	−0.016 01	0.602 67	0.001 52	0.986 03	−0.000 98
0.094 85	−0.015 60	0.616 77	0.001 40	0.989 17	−0.000 72
0.103 72	−0.015 15	0.630 76	0.001 27	0.991 94	−0.000 49
0.112 95	−0.014 65	0.644 61	0.001 13	0.994 34	−0.000 30
0.122 50	−0.014 12	0.658 33	0.000 98	0.996 35	−0.000 16
0.132 38	−0.013 56	0.671 89	0.000 82	0.997 93	−0.000 07
0.142 58	−0.012 97	0.685 30	0.000 65	0.999 08	−0.000 02
0.153 08	−0.012 35	0.698 53	0.000 48	0.999 77	−0.000 00

Table C.5: Trial 37 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.723 57	0.053 78	0.241 03	0.110 72
0.999 77	0.000 04	0.711 57	0.055 99	0.231 08	0.110 18
0.999 08	0.000 17	0.699 45	0.058 21	0.221 27	0.109 51
0.997 94	0.000 41	0.687 22	0.060 43	0.211 60	0.108 72
0.996 38	0.000 76	0.674 89	0.062 65	0.202 09	0.107 82
0.994 40	0.001 24	0.662 48	0.064 88	0.192 73	0.106 80
0.992 04	0.001 84	0.649 99	0.067 09	0.183 55	0.105 68
0.989 31	0.002 55	0.637 42	0.069 30	0.174 54	0.104 45
0.986 23	0.003 36	0.624 81	0.071 49	0.165 72	0.103 13
0.982 81	0.004 26	0.612 14	0.073 66	0.157 08	0.101 71
0.979 07	0.005 23	0.599 43	0.075 80	0.148 64	0.100 19
0.975 01	0.006 25	0.586 69	0.077 92	0.140 41	0.098 59
0.970 62	0.007 30	0.573 92	0.080 00	0.132 38	0.096 90
0.965 90	0.008 38	0.561 15	0.082 04	0.124 57	0.095 13
0.960 83	0.009 47	0.548 36	0.084 04	0.116 99	0.093 27
0.955 40	0.010 59	0.535 58	0.085 99	0.109 63	0.091 33
0.949 61	0.011 74	0.522 81	0.087 90	0.102 52	0.089 31
0.943 46	0.012 94	0.510 06	0.089 75	0.095 65	0.087 22
0.936 96	0.014 18	0.497 34	0.091 54	0.089 04	0.085 04
0.930 12	0.015 47	0.484 65	0.093 27	0.082 68	0.082 78
0.922 95	0.016 81	0.472 00	0.094 94	0.076 60	0.080 44
0.915 46	0.018 21	0.459 40	0.096 54	0.070 81	0.078 02
0.907 65	0.019 65	0.446 86	0.098 07	0.065 32	0.075 47
0.899 53	0.021 15	0.434 37	0.099 54	0.060 12	0.072 77
0.891 11	0.022 71	0.421 96	0.100 93	0.055 12	0.069 90
0.882 41	0.024 31	0.409 64	0.102 25	0.050 32	0.066 92
0.873 42	0.025 97	0.397 40	0.103 50	0.045 73	0.063 84
0.864 15	0.027 68	0.385 27	0.104 68	0.041 35	0.060 68
0.854 63	0.029 44	0.373 25	0.105 77	0.037 17	0.057 45
0.844 85	0.031 25	0.361 35	0.106 79	0.033 20	0.054 16
0.834 83	0.033 11	0.349 58	0.107 71	0.029 44	0.050 82
0.824 58	0.035 02	0.337 95	0.108 55	0.025 90	0.047 45
0.814 11	0.036 96	0.326 47	0.109 29	0.022 57	0.044 04
0.803 42	0.038 95	0.315 16	0.109 93	0.019 46	0.040 62
0.792 53	0.040 98	0.304 01	0.110 47	0.016 57	0.037 18
0.781 46	0.043 04	0.293 04	0.110 89	0.013 90	0.033 75
0.770 20	0.045 14	0.282 26	0.111 18	0.011 45	0.030 32
0.758 77	0.047 26	0.271 68	0.111 34	0.009 24	0.026 90
0.747 18	0.049 42	0.261 32	0.111 32	0.007 25	0.023 52
0.735 45	0.051 59	0.251 11	0.111 11	0.005 50	0.020 17

Continued on next page.

x	y	x	y	x	y
0.003 98	0.016 87	0.198 73	−0.061 99	0.674 88	−0.028 92
0.002 70	0.013 63	0.208 93	−0.063 06	0.687 73	−0.026 56
0.001 66	0.010 46	0.219 29	−0.064 08	0.700 53	−0.024 24
0.000 87	0.007 38	0.229 81	−0.065 03	0.713 26	−0.021 98
0.000 33	0.004 39	0.240 47	−0.065 91	0.725 92	−0.019 77
0.000 05	0.001 53	0.251 28	−0.066 72	0.738 49	−0.017 63
0.000 03	−0.001 20	0.262 22	−0.067 46	0.750 96	−0.015 57
0.000 30	−0.003 76	0.273 28	−0.068 12	0.763 30	−0.013 58
0.000 87	−0.006 10	0.284 46	−0.068 71	0.775 51	−0.011 69
0.001 91	−0.008 20	0.295 74	−0.069 22	0.787 56	−0.009 89
0.003 50	−0.010 18	0.307 13	−0.069 64	0.799 45	−0.008 19
0.005 54	−0.012 18	0.318 61	−0.069 98	0.811 15	−0.006 60
0.007 99	−0.014 18	0.330 16	−0.070 23	0.822 64	−0.005 12
0.010 82	−0.016 19	0.341 79	−0.070 39	0.833 92	−0.003 75
0.014 01	−0.018 20	0.353 49	−0.070 45	0.844 96	−0.002 49
0.017 55	−0.020 21	0.365 24	−0.070 42	0.855 75	−0.001 35
0.021 42	−0.022 21	0.377 04	−0.070 28	0.866 26	−0.000 34
0.025 63	−0.024 20	0.388 87	−0.070 02	0.876 49	0.000 56
0.030 16	−0.026 18	0.400 74	−0.069 66	0.886 41	0.001 34
0.035 01	−0.028 15	0.412 62	−0.069 16	0.896 01	0.002 01
0.040 17	−0.030 10	0.424 50	−0.068 53	0.905 28	0.002 55
0.045 63	−0.032 02	0.436 39	−0.067 75	0.914 19	0.002 99
0.051 39	−0.033 93	0.448 25	−0.066 76	0.922 73	0.003 32
0.057 43	−0.035 81	0.460 15	−0.065 53	0.930 90	0.003 54
0.063 77	−0.037 67	0.472 18	−0.064 09	0.938 66	0.003 66
0.070 38	−0.039 50	0.484 32	−0.062 48	0.946 01	0.003 69
0.077 26	−0.041 29	0.496 56	−0.060 73	0.952 94	0.003 63
0.084 41	−0.043 05	0.508 91	−0.058 86	0.959 43	0.003 48
0.091 83	−0.044 78	0.521 35	−0.056 88	0.965 47	0.003 26
0.099 49	−0.046 47	0.533 87	−0.054 80	0.971 03	0.002 98
0.107 41	−0.048 12	0.546 47	−0.052 64	0.976 11	0.002 65
0.115 56	−0.049 73	0.559 15	−0.050 41	0.980 71	0.002 29
0.123 95	−0.051 29	0.571 89	−0.048 12	0.984 80	0.001 91
0.132 57	−0.052 81	0.584 68	−0.045 78	0.988 41	0.001 54
0.141 42	−0.054 28	0.597 52	−0.043 40	0.991 51	0.001 18
0.150 47	−0.055 70	0.610 39	−0.041 00	0.994 13	0.000 85
0.159 74	−0.057 07	0.623 28	−0.038 58	0.996 26	0.000 57
0.169 20	−0.058 38	0.636 19	−0.036 15	0.997 90	0.000 33
0.178 86	−0.059 64	0.649 10	−0.033 72	0.999 07	0.000 15
0.188 71	−0.060 84	0.662 00	−0.031 31	0.999 77	0.000 04

Table C.6: Trial 38 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.730 26	0.056 43	0.247 48	0.112 47
0.999 77	0.000 05	0.718 42	0.058 62	0.237 39	0.111 98
0.999 09	0.000 21	0.706 45	0.060 81	0.227 44	0.111 36
0.997 97	0.000 50	0.694 37	0.063 00	0.217 63	0.110 62
0.996 43	0.000 93	0.682 19	0.065 19	0.207 97	0.109 77
0.994 50	0.001 50	0.669 90	0.067 37	0.198 48	0.108 81
0.992 19	0.002 20	0.657 53	0.069 55	0.189 16	0.107 75
0.989 54	0.003 03	0.645 08	0.071 71	0.180 01	0.106 58
0.986 55	0.003 97	0.632 57	0.073 85	0.171 05	0.105 32
0.983 25	0.005 00	0.619 99	0.075 98	0.162 28	0.103 96
0.979 64	0.006 11	0.607 37	0.078 07	0.153 70	0.102 51
0.975 73	0.007 26	0.594 70	0.080 14	0.145 33	0.100 97
0.971 51	0.008 45	0.582 00	0.082 17	0.137 17	0.099 34
0.966 96	0.009 65	0.569 28	0.084 16	0.129 23	0.097 63
0.962 07	0.010 86	0.556 54	0.086 11	0.121 51	0.095 83
0.956 83	0.012 09	0.543 79	0.088 02	0.114 03	0.093 95
0.951 23	0.013 35	0.531 05	0.089 87	0.106 78	0.091 99
0.945 28	0.014 65	0.518 31	0.091 67	0.099 79	0.089 94
0.938 99	0.016 00	0.505 59	0.093 42	0.093 05	0.087 82
0.932 37	0.017 38	0.492 89	0.095 11	0.086 57	0.085 61
0.925 42	0.018 82	0.480 23	0.096 73	0.080 37	0.083 31
0.918 14	0.020 30	0.467 60	0.098 30	0.074 45	0.080 92
0.910 55	0.021 83	0.455 02	0.099 80	0.068 85	0.078 41
0.902 66	0.023 41	0.442 49	0.101 24	0.063 53	0.075 73
0.894 46	0.025 03	0.430 03	0.102 61	0.058 41	0.072 88
0.885 98	0.026 70	0.417 65	0.103 91	0.053 49	0.069 91
0.877 22	0.028 42	0.405 35	0.105 15	0.048 77	0.066 84
0.868 18	0.030 18	0.393 15	0.106 31	0.044 26	0.063 68
0.858 88	0.031 99	0.381 06	0.107 40	0.039 96	0.060 44
0.849 33	0.033 84	0.369 08	0.108 40	0.035 85	0.057 15
0.839 53	0.035 73	0.357 23	0.109 32	0.031 96	0.053 79
0.829 50	0.037 67	0.345 51	0.110 16	0.028 28	0.050 40
0.819 24	0.039 64	0.333 94	0.110 90	0.024 81	0.046 97
0.808 77	0.041 64	0.322 52	0.111 54	0.021 56	0.043 52
0.798 09	0.043 68	0.311 26	0.112 08	0.018 53	0.040 05
0.787 22	0.045 75	0.300 17	0.112 51	0.015 71	0.036 57
0.776 16	0.047 84	0.289 27	0.112 82	0.013 12	0.033 10
0.764 92	0.049 96	0.278 56	0.112 99	0.010 75	0.029 63
0.753 52	0.052 10	0.268 05	0.113 00	0.008 61	0.026 19
0.741 96	0.054 26	0.257 71	0.112 82	0.006 70	0.022 78

Continued on next page.

x	y	x	y	x	y
0.005 02	0.019 42	0.191 14	−0.059 97	0.667 01	−0.024 95
0.003 58	0.016 10	0.201 21	−0.061 03	0.680 06	−0.022 52
0.002 37	0.012 84	0.211 46	−0.062 03	0.693 07	−0.020 15
0.001 41	0.009 67	0.221 86	−0.062 96	0.706 03	−0.017 83
0.000 69	0.006 58	0.232 42	−0.063 82	0.718 93	−0.015 58
0.000 22	0.003 59	0.243 12	−0.064 62	0.731 76	−0.013 41
0.000 01	0.000 73	0.253 96	−0.065 34	0.744 48	−0.011 32
0.000 07	−0.001 99	0.264 92	−0.065 98	0.757 09	−0.009 32
0.000 40	−0.004 53	0.276 01	−0.066 55	0.769 58	−0.007 43
0.001 05	−0.006 85	0.287 21	−0.067 04	0.781 92	−0.005 64
0.002 18	−0.008 92	0.298 52	−0.067 45	0.794 09	−0.003 96
0.003 85	−0.010 89	0.309 92	−0.067 77	0.806 09	−0.002 40
0.005 98	−0.012 87	0.321 40	−0.068 00	0.817 89	−0.000 96
0.008 51	−0.014 85	0.332 97	−0.068 14	0.829 47	0.000 35
0.011 41	−0.016 83	0.344 60	−0.068 18	0.840 81	0.001 53
0.014 68	−0.018 81	0.356 29	−0.068 12	0.851 90	0.002 58
0.018 30	−0.020 78	0.368 03	−0.067 96	0.862 72	0.003 50
0.022 26	−0.022 75	0.379 81	−0.067 68	0.873 25	0.004 28
0.026 54	−0.024 70	0.391 63	−0.067 29	0.883 47	0.004 94
0.031 15	−0.026 64	0.403 46	−0.066 76	0.893 37	0.005 46
0.036 08	−0.028 56	0.415 31	−0.066 10	0.902 92	0.005 86
0.041 32	−0.030 46	0.427 16	−0.065 28	0.912 11	0.006 13
0.046 85	−0.032 34	0.438 98	−0.064 25	0.920 92	0.006 28
0.052 69	−0.034 19	0.450 86	−0.062 96	0.929 34	0.006 31
0.058 81	−0.036 02	0.462 87	−0.061 45	0.937 35	0.006 24
0.065 22	−0.037 82	0.475 00	−0.059 78	0.944 94	0.006 06
0.071 91	−0.039 59	0.487 25	−0.057 95	0.952 09	0.005 79
0.078 87	−0.041 33	0.499 61	−0.056 00	0.958 78	0.005 42
0.086 09	−0.043 03	0.512 08	−0.053 94	0.965 00	0.004 96
0.093 57	−0.044 70	0.524 64	−0.051 78	0.970 71	0.004 44
0.101 31	−0.046 32	0.537 30	−0.049 53	0.975 92	0.003 88
0.109 30	−0.047 91	0.550 05	−0.047 21	0.980 60	0.003 29
0.117 52	−0.049 45	0.562 87	−0.044 83	0.984 76	0.002 70
0.125 98	−0.050 94	0.575 75	−0.042 40	0.988 41	0.002 14
0.134 66	−0.052 39	0.588 70	−0.039 93	0.991 53	0.001 62
0.143 57	−0.053 79	0.601 69	−0.037 43	0.994 16	0.001 15
0.152 68	−0.055 14	0.614 72	−0.034 92	0.996 28	0.000 75
0.162 01	−0.056 43	0.627 78	−0.032 41	0.997 92	0.000 43
0.171 53	−0.057 67	0.640 86	−0.029 90	0.999 08	0.000 20
0.181 24	−0.058 85	0.653 94	−0.027 41	0.999 77	0.000 05

Table C.7: Trial 39 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.736 79	0.059 42	0.254 89	0.115 35
0.999 77	0.000 06	0.725 13	0.061 60	0.244 71	0.114 88
0.999 10	0.000 25	0.713 33	0.063 78	0.234 67	0.114 28
0.998 00	0.000 60	0.701 40	0.065 97	0.224 77	0.113 56
0.996 49	0.001 09	0.689 36	0.068 14	0.215 02	0.112 73
0.994 59	0.001 75	0.677 22	0.070 31	0.205 42	0.111 79
0.992 35	0.002 55	0.664 98	0.072 47	0.196 00	0.110 74
0.989 77	0.003 49	0.652 66	0.074 61	0.186 74	0.109 59
0.986 88	0.004 56	0.640 26	0.076 73	0.177 67	0.108 34
0.983 69	0.005 71	0.627 79	0.078 83	0.168 78	0.106 99
0.980 20	0.006 95	0.615 26	0.080 90	0.160 08	0.105 55
0.976 43	0.008 23	0.602 68	0.082 94	0.151 59	0.104 02
0.972 36	0.009 54	0.590 06	0.084 94	0.143 31	0.102 40
0.967 98	0.010 87	0.577 41	0.086 91	0.135 23	0.100 69
0.963 27	0.012 20	0.564 73	0.088 83	0.127 38	0.098 90
0.958 20	0.013 54	0.552 03	0.090 71	0.119 76	0.097 02
0.952 79	0.014 91	0.539 33	0.092 54	0.112 38	0.095 06
0.947 03	0.016 32	0.526 62	0.094 32	0.105 24	0.093 02
0.940 94	0.017 78	0.513 92	0.096 04	0.098 35	0.090 89
0.934 51	0.019 27	0.501 23	0.097 71	0.091 72	0.088 68
0.927 76	0.020 80	0.488 57	0.099 32	0.085 36	0.086 38
0.920 70	0.022 38	0.475 94	0.100 88	0.079 28	0.083 98
0.913 32	0.024 00	0.463 34	0.102 37	0.073 52	0.081 46
0.905 64	0.025 67	0.450 79	0.103 81	0.068 02	0.078 77
0.897 67	0.027 38	0.438 30	0.105 19	0.062 73	0.075 92
0.889 40	0.029 13	0.425 89	0.106 51	0.057 64	0.072 94
0.880 86	0.030 92	0.413 57	0.107 76	0.052 74	0.069 86
0.872 05	0.032 75	0.401 33	0.108 94	0.048 05	0.066 70
0.862 97	0.034 62	0.389 20	0.110 04	0.043 57	0.063 46
0.853 64	0.036 53	0.377 18	0.111 06	0.039 28	0.060 15
0.844 06	0.038 48	0.365 29	0.112 00	0.035 21	0.056 79
0.834 25	0.040 46	0.353 52	0.112 85	0.031 34	0.053 38
0.824 20	0.042 47	0.341 90	0.113 62	0.027 69	0.049 93
0.813 94	0.044 51	0.330 42	0.114 28	0.024 25	0.046 46
0.803 47	0.046 58	0.319 10	0.114 84	0.021 03	0.042 97
0.792 80	0.048 68	0.307 95	0.115 29	0.018 03	0.039 47
0.781 94	0.050 79	0.296 98	0.115 62	0.015 25	0.035 97
0.770 90	0.052 93	0.286 20	0.115 81	0.012 69	0.032 48
0.759 69	0.055 08	0.275 62	0.115 84	0.010 36	0.029 01
0.748 32	0.057 25	0.265 20	0.115 68	0.008 26	0.025 56

Continued on next page.

x	y	x	y	x	y
0.006 39	0.022 15	0.183 55	−0.057 00	0.659 67	−0.020 46
0.004 75	0.018 78	0.193 51	−0.058 03	0.672 92	−0.018 00
0.003 35	0.015 48	0.203 64	−0.059 00	0.686 15	−0.015 60
0.002 19	0.012 23	0.213 94	−0.059 90	0.699 33	−0.013 26
0.001 26	0.009 08	0.224 40	−0.060 73	0.712 47	−0.011 00
0.000 59	0.006 01	0.235 01	−0.061 50	0.725 53	−0.008 83
0.000 17	0.003 06	0.245 76	−0.062 19	0.738 50	−0.006 75
0.000 00	0.000 23	0.256 64	−0.062 81	0.751 37	−0.004 77
0.000 11	−0.002 44	0.267 65	−0.063 35	0.764 12	−0.002 90
0.000 49	−0.004 93	0.278 78	−0.063 82	0.776 73	−0.001 14
0.001 20	−0.007 17	0.290 02	−0.064 20	0.789 18	0.000 48
0.002 41	−0.009 18	0.301 35	−0.064 50	0.801 46	0.001 98
0.004 17	−0.011 11	0.312 78	−0.064 71	0.813 54	0.003 35
0.006 37	−0.013 05	0.324 29	−0.064 82	0.825 41	0.004 58
0.008 96	−0.014 98	0.335 88	−0.064 84	0.837 04	0.005 66
0.011 94	−0.016 92	0.347 53	−0.064 76	0.848 42	0.006 60
0.015 28	−0.018 84	0.359 23	−0.064 58	0.859 52	0.007 40
0.018 97	−0.020 76	0.370 99	−0.064 28	0.870 33	0.008 06
0.023 00	−0.022 66	0.382 77	−0.063 86	0.880 83	0.008 57
0.027 36	−0.024 55	0.394 59	−0.063 32	0.891 00	0.008 94
0.032 04	−0.026 42	0.406 41	−0.062 63	0.900 81	0.009 17
0.037 04	−0.028 28	0.418 24	−0.061 77	0.910 26	0.009 26
0.042 35	−0.030 11	0.430 06	−0.060 71	0.919 32	0.009 23
0.047 96	−0.031 92	0.441 93	−0.059 39	0.927 98	0.009 07
0.053 87	−0.033 70	0.453 94	−0.057 83	0.936 22	0.008 79
0.060 07	−0.035 45	0.466 08	−0.056 11	0.944 02	0.008 40
0.066 55	−0.037 18	0.478 36	−0.054 23	0.951 37	0.007 91
0.073 31	−0.038 87	0.490 75	−0.052 23	0.958 24	0.007 31
0.080 34	−0.040 53	0.503 27	−0.050 11	0.964 62	0.006 62
0.087 64	−0.042 15	0.515 89	−0.047 89	0.970 47	0.005 86
0.095 20	−0.043 73	0.528 62	−0.045 58	0.975 77	0.005 06
0.103 01	−0.045 27	0.541 44	−0.043 20	0.980 53	0.004 25
0.111 07	−0.046 77	0.554 35	−0.040 75	0.984 75	0.003 46
0.119 36	−0.048 23	0.567 34	−0.038 26	0.988 42	0.002 71
0.127 89	−0.049 63	0.580 41	−0.035 73	0.991 56	0.002 03
0.136 64	−0.050 99	0.593 53	−0.033 18	0.994 19	0.001 43
0.145 62	−0.052 30	0.606 70	−0.030 61	0.996 31	0.000 93
0.154 80	−0.053 56	0.619 92	−0.028 05	0.997 94	0.000 53
0.164 19	−0.054 77	0.633 16	−0.025 49	0.999 09	0.000 24
0.173 78	−0.055 91	0.646 41	−0.022 96	0.999 77	0.000 06

Table C.8: Trial 40 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.749 76	0.064 91	0.269 22	0.120 56
0.999 78	0.000 08	0.738 43	0.067 09	0.258 85	0.120 15
0.999 13	0.000 34	0.726 95	0.069 25	0.248 61	0.119 61
0.998 07	0.000 78	0.715 33	0.071 41	0.238 50	0.118 95
0.996 62	0.001 41	0.703 59	0.073 56	0.228 55	0.118 18
0.994 82	0.002 24	0.691 72	0.075 70	0.218 74	0.117 30
0.992 70	0.003 24	0.679 74	0.077 82	0.209 10	0.116 31
0.990 29	0.004 40	0.667 66	0.079 92	0.199 62	0.115 22
0.987 60	0.005 69	0.655 48	0.081 99	0.190 32	0.114 03
0.984 64	0.007 08	0.643 22	0.084 04	0.181 20	0.112 74
0.981 42	0.008 55	0.630 88	0.086 05	0.172 27	0.111 36
0.977 94	0.010 07	0.618 48	0.088 04	0.163 53	0.109 88
0.974 18	0.011 61	0.606 01	0.089 98	0.155 00	0.108 31
0.970 13	0.013 16	0.593 49	0.091 89	0.146 68	0.106 65
0.965 76	0.014 70	0.580 92	0.093 76	0.138 57	0.104 91
0.961 05	0.016 25	0.568 32	0.095 58	0.130 69	0.103 07
0.956 00	0.017 83	0.555 69	0.097 36	0.123 04	0.101 15
0.950 62	0.019 44	0.543 05	0.099 08	0.115 63	0.099 14
0.944 92	0.021 09	0.530 38	0.100 76	0.108 47	0.097 04
0.938 89	0.022 77	0.517 72	0.102 39	0.101 56	0.094 86
0.932 54	0.024 49	0.505 05	0.103 96	0.094 92	0.092 58
0.925 89	0.026 24	0.492 40	0.105 49	0.088 55	0.090 20
0.918 93	0.028 03	0.479 76	0.106 96	0.082 49	0.087 69
0.911 67	0.029 86	0.467 16	0.108 40	0.076 69	0.085 01
0.904 12	0.031 72	0.454 62	0.109 79	0.071 09	0.082 16
0.896 29	0.033 62	0.442 15	0.111 12	0.065 69	0.079 19
0.888 18	0.035 55	0.429 75	0.112 39	0.060 49	0.076 11
0.879 80	0.037 50	0.417 44	0.113 59	0.055 48	0.072 94
0.871 15	0.039 49	0.405 23	0.114 72	0.050 67	0.069 68
0.862 25	0.041 51	0.393 11	0.115 78	0.046 06	0.066 36
0.853 10	0.043 55	0.381 12	0.116 75	0.041 66	0.062 98
0.843 71	0.045 62	0.369 24	0.117 64	0.037 47	0.059 54
0.834 09	0.047 70	0.357 50	0.118 44	0.033 48	0.056 07
0.824 25	0.049 81	0.345 89	0.119 14	0.029 71	0.052 56
0.814 19	0.051 94	0.334 44	0.119 75	0.026 15	0.049 03
0.803 92	0.054 08	0.323 15	0.120 24	0.022 80	0.045 48
0.793 45	0.056 23	0.312 02	0.120 61	0.019 68	0.041 92
0.782 79	0.058 39	0.301 07	0.120 85	0.016 77	0.038 36
0.771 95	0.060 56	0.290 32	0.120 93	0.014 09	0.034 81
0.760 94	0.062 74	0.279 71	0.120 83	0.011 64	0.031 28

Continued on next page.

x	y	x	y	x	y
0.009 41	0.027 77	0.168 57	−0.052 01	0.644 04	−0.011 51
0.007 41	0.024 30	0.178 26	−0.052 98	0.657 70	−0.008 93
0.005 64	0.020 88	0.188 14	−0.053 89	0.671 37	−0.006 43
0.004 11	0.017 51	0.198 20	−0.054 74	0.685 02	−0.004 00
0.002 82	0.014 21	0.208 43	−0.055 52	0.698 64	−0.001 67
0.001 76	0.010 98	0.218 82	−0.056 24	0.712 21	0.000 56
0.000 95	0.007 85	0.229 36	−0.056 88	0.725 71	0.002 68
0.000 38	0.004 82	0.240 04	−0.057 45	0.739 14	0.004 67
0.000 07	0.001 91	0.250 87	−0.057 94	0.752 45	0.006 53
0.000 01	−0.000 86	0.261 82	−0.058 36	0.765 65	0.008 24
0.000 23	−0.003 46	0.272 88	−0.058 69	0.778 70	0.009 80
0.000 74	−0.005 86	0.284 06	−0.058 94	0.791 58	0.011 21
0.001 60	−0.007 96	0.295 34	−0.059 10	0.804 28	0.012 46
0.002 98	−0.009 86	0.306 72	−0.059 17	0.816 77	0.013 54
0.004 90	−0.011 73	0.318 17	−0.059 14	0.829 02	0.014 45
0.007 26	−0.013 60	0.329 70	−0.059 01	0.841 03	0.015 19
0.010 01	−0.015 45	0.341 29	−0.058 77	0.852 75	0.015 75
0.013 13	−0.017 30	0.352 94	−0.058 42	0.864 18	0.016 14
0.016 62	−0.019 13	0.364 64	−0.057 94	0.875 29	0.016 36
0.020 46	−0.020 95	0.376 36	−0.057 33	0.886 05	0.016 40
0.024 63	−0.022 76	0.388 11	−0.056 58	0.896 45	0.016 29
0.029 14	−0.024 54	0.399 87	−0.055 65	0.906 46	0.016 01
0.033 97	−0.026 30	0.411 62	−0.054 50	0.916 06	0.015 58
0.039 11	−0.028 04	0.423 44	−0.053 07	0.925 24	0.015 00
0.044 57	−0.029 75	0.435 41	−0.051 40	0.933 97	0.014 28
0.050 33	−0.031 43	0.447 55	−0.049 54	0.942 23	0.013 43
0.056 38	−0.033 08	0.459 83	−0.047 53	0.950 00	0.012 45
0.062 73	−0.034 70	0.472 26	−0.045 37	0.957 26	0.011 35
0.069 35	−0.036 29	0.484 83	−0.043 10	0.963 97	0.010 14
0.076 26	−0.037 84	0.497 54	−0.040 71	0.970 09	0.008 86
0.083 43	−0.039 36	0.510 38	−0.038 24	0.975 60	0.007 55
0.090 87	−0.040 83	0.523 34	−0.035 69	0.980 50	0.006 27
0.098 57	−0.042 26	0.536 41	−0.033 07	0.984 81	0.005 04
0.106 51	−0.043 65	0.549 60	−0.030 41	0.988 52	0.003 90
0.114 70	−0.045 00	0.562 88	−0.027 71	0.991 68	0.002 88
0.123 13	−0.046 30	0.576 25	−0.024 99	0.994 29	0.002 01
0.131 79	−0.047 54	0.589 69	−0.022 26	0.996 38	0.001 29
0.140 67	−0.048 74	0.603 21	−0.019 53	0.997 98	0.000 72
0.149 76	−0.049 89	0.616 78	−0.016 82	0.999 11	0.000 32
0.159 06	−0.050 98	0.630 40	−0.014 14	0.999 78	0.000 08

Table C.9: Trial 41 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.749 76	0.064 91	0.269 22	0.120 56
0.999 78	0.000 08	0.738 43	0.067 09	0.258 85	0.120 15
0.999 13	0.000 34	0.726 95	0.069 25	0.248 61	0.119 61
0.998 07	0.000 78	0.715 33	0.071 41	0.238 50	0.118 95
0.996 62	0.001 41	0.703 59	0.073 56	0.228 55	0.118 18
0.994 82	0.002 24	0.691 72	0.075 70	0.218 74	0.117 30
0.992 70	0.003 24	0.679 74	0.077 82	0.209 10	0.116 31
0.990 29	0.004 40	0.667 66	0.079 92	0.199 62	0.115 22
0.987 60	0.005 69	0.655 48	0.081 99	0.190 32	0.114 03
0.984 64	0.007 08	0.643 22	0.084 04	0.181 20	0.112 74
0.981 42	0.008 55	0.630 88	0.086 05	0.172 27	0.111 36
0.977 94	0.010 07	0.618 48	0.088 04	0.163 53	0.109 88
0.974 18	0.011 61	0.606 01	0.089 98	0.155 00	0.108 31
0.970 13	0.013 16	0.593 49	0.091 89	0.146 68	0.106 65
0.965 76	0.014 70	0.580 92	0.093 76	0.138 57	0.104 91
0.961 05	0.016 25	0.568 32	0.095 58	0.130 69	0.103 07
0.956 00	0.017 83	0.555 69	0.097 36	0.123 04	0.101 15
0.950 62	0.019 44	0.543 05	0.099 08	0.115 63	0.099 14
0.944 92	0.021 09	0.530 38	0.100 76	0.108 47	0.097 04
0.938 89	0.022 77	0.517 72	0.102 39	0.101 56	0.094 86
0.932 54	0.024 49	0.505 05	0.103 96	0.094 92	0.092 58
0.925 89	0.026 24	0.492 40	0.105 49	0.088 55	0.090 20
0.918 93	0.028 03	0.479 76	0.106 96	0.082 49	0.087 69
0.911 67	0.029 86	0.467 16	0.108 40	0.076 69	0.085 01
0.904 12	0.031 72	0.454 62	0.109 79	0.071 09	0.082 16
0.896 29	0.033 62	0.442 15	0.111 12	0.065 69	0.079 19
0.888 18	0.035 55	0.429 75	0.112 39	0.060 49	0.076 11
0.879 80	0.037 50	0.417 44	0.113 59	0.055 48	0.072 94
0.871 15	0.039 49	0.405 23	0.114 72	0.050 67	0.069 68
0.862 25	0.041 51	0.393 11	0.115 78	0.046 06	0.066 36
0.853 10	0.043 55	0.381 12	0.116 75	0.041 66	0.062 98
0.843 71	0.045 62	0.369 24	0.117 64	0.037 47	0.059 54
0.834 09	0.047 70	0.357 50	0.118 44	0.033 48	0.056 07
0.824 25	0.049 81	0.345 89	0.119 14	0.029 71	0.052 56
0.814 19	0.051 94	0.334 44	0.119 75	0.026 15	0.049 03
0.803 92	0.054 08	0.323 15	0.120 24	0.022 80	0.045 48
0.793 45	0.056 23	0.312 02	0.120 61	0.019 68	0.041 92
0.782 79	0.058 39	0.301 07	0.120 85	0.016 77	0.038 36
0.771 95	0.060 56	0.290 32	0.120 93	0.014 09	0.034 81
0.760 94	0.062 74	0.279 71	0.120 83	0.011 64	0.031 28

Continued on next page.

x	y	x	y	x	y
0.009 41	0.027 77	0.168 57	−0.052 01	0.644 04	−0.011 51
0.007 41	0.024 30	0.178 26	−0.052 98	0.657 70	−0.008 93
0.005 64	0.020 88	0.188 14	−0.053 89	0.671 37	−0.006 43
0.004 11	0.017 51	0.198 20	−0.054 74	0.685 02	−0.004 00
0.002 82	0.014 21	0.208 43	−0.055 52	0.698 64	−0.001 67
0.001 76	0.010 98	0.218 82	−0.056 24	0.712 21	0.000 56
0.000 95	0.007 85	0.229 36	−0.056 88	0.725 71	0.002 68
0.000 38	0.004 82	0.240 04	−0.057 45	0.739 14	0.004 67
0.000 07	0.001 91	0.250 87	−0.057 94	0.752 45	0.006 53
0.000 01	−0.000 86	0.261 82	−0.058 36	0.765 65	0.008 24
0.000 23	−0.003 46	0.272 88	−0.058 69	0.778 70	0.009 80
0.000 74	−0.005 86	0.284 06	−0.058 94	0.791 58	0.011 21
0.001 60	−0.007 96	0.295 34	−0.059 10	0.804 28	0.012 46
0.002 98	−0.009 86	0.306 72	−0.059 17	0.816 77	0.013 54
0.004 90	−0.011 73	0.318 17	−0.059 14	0.829 02	0.014 45
0.007 26	−0.013 60	0.329 70	−0.059 01	0.841 03	0.015 19
0.010 01	−0.015 45	0.341 29	−0.058 77	0.852 75	0.015 75
0.013 13	−0.017 30	0.352 94	−0.058 42	0.864 18	0.016 14
0.016 62	−0.019 13	0.364 64	−0.057 94	0.875 29	0.016 36
0.020 46	−0.020 95	0.376 36	−0.057 33	0.886 05	0.016 40
0.024 63	−0.022 76	0.388 11	−0.056 58	0.896 45	0.016 29
0.029 14	−0.024 54	0.399 87	−0.055 65	0.906 46	0.016 01
0.033 97	−0.026 30	0.411 62	−0.054 50	0.916 06	0.015 58
0.039 11	−0.028 04	0.423 44	−0.053 07	0.925 24	0.015 00
0.044 57	−0.029 75	0.435 41	−0.051 40	0.933 97	0.014 28
0.050 33	−0.031 43	0.447 55	−0.049 54	0.942 23	0.013 43
0.056 38	−0.033 08	0.459 83	−0.047 53	0.950 00	0.012 45
0.062 73	−0.034 70	0.472 26	−0.045 37	0.957 26	0.011 35
0.069 35	−0.036 29	0.484 83	−0.043 10	0.963 97	0.010 14
0.076 26	−0.037 84	0.497 54	−0.040 71	0.970 09	0.008 86
0.083 43	−0.039 36	0.510 38	−0.038 24	0.975 60	0.007 55
0.090 87	−0.040 83	0.523 34	−0.035 69	0.980 50	0.006 27
0.098 57	−0.042 26	0.536 41	−0.033 07	0.984 81	0.005 04
0.106 51	−0.043 65	0.549 60	−0.030 41	0.988 52	0.003 90
0.114 70	−0.045 00	0.562 88	−0.027 71	0.991 68	0.002 88
0.123 13	−0.046 30	0.576 25	−0.024 99	0.994 29	0.002 01
0.131 79	−0.047 54	0.589 69	−0.022 26	0.996 38	0.001 29
0.140 67	−0.048 74	0.603 21	−0.019 53	0.997 98	0.000 72
0.149 76	−0.049 89	0.616 78	−0.016 82	0.999 11	0.000 32
0.159 06	−0.050 98	0.630 40	−0.014 14	0.999 78	0.000 08

Table C.10: Trial 42 Final Solution Coordinates.

x	y	x	y	x	y
1	0	0.754 42	0.069 96	0.279 43	0.126 36
0.999 78	0.000 09	0.743 28	0.072 29	0.268 98	0.125 87
0.999 14	0.000 37	0.732 00	0.074 62	0.258 66	0.125 26
0.998 09	0.000 85	0.720 57	0.076 93	0.248 46	0.124 54
0.996 67	0.001 54	0.709 03	0.079 22	0.238 41	0.123 69
0.994 90	0.002 43	0.697 36	0.081 50	0.228 50	0.122 74
0.992 82	0.003 51	0.685 58	0.083 76	0.218 74	0.121 69
0.990 46	0.004 76	0.673 70	0.086 00	0.209 15	0.120 53
0.987 83	0.006 14	0.661 73	0.088 20	0.199 72	0.119 27
0.984 95	0.007 64	0.649 68	0.090 38	0.190 47	0.117 91
0.981 81	0.009 21	0.637 54	0.092 51	0.181 39	0.116 46
0.978 42	0.010 84	0.625 34	0.094 61	0.172 51	0.114 91
0.974 75	0.012 49	0.613 08	0.096 67	0.163 83	0.113 27
0.970 80	0.014 14	0.600 77	0.098 68	0.155 34	0.111 54
0.966 54	0.015 80	0.588 42	0.100 64	0.147 07	0.109 72
0.961 93	0.017 47	0.576 03	0.102 55	0.139 02	0.107 82
0.956 99	0.019 17	0.563 61	0.104 41	0.131 18	0.105 83
0.951 72	0.020 91	0.551 17	0.106 20	0.123 58	0.103 75
0.946 13	0.022 68	0.538 72	0.107 94	0.116 22	0.101 58
0.940 22	0.024 50	0.526 26	0.109 62	0.109 11	0.099 33
0.933 99	0.026 36	0.513 80	0.111 23	0.102 25	0.096 98
0.927 47	0.028 26	0.501 36	0.112 77	0.095 66	0.094 53
0.920 64	0.030 20	0.488 93	0.114 26	0.089 37	0.091 96
0.913 52	0.032 17	0.476 53	0.115 68	0.083 33	0.089 23
0.906 11	0.034 19	0.464 17	0.117 03	0.077 49	0.086 33
0.898 42	0.036 24	0.451 86	0.118 31	0.071 84	0.083 32
0.890 45	0.038 32	0.439 62	0.119 52	0.066 39	0.080 20
0.882 22	0.040 44	0.427 44	0.120 65	0.061 14	0.076 99
0.873 73	0.042 59	0.415 34	0.121 70	0.056 08	0.073 70
0.864 98	0.044 77	0.403 33	0.122 67	0.051 23	0.070 35
0.855 99	0.046 97	0.391 42	0.123 55	0.046 58	0.066 93
0.846 77	0.049 20	0.379 61	0.124 34	0.042 13	0.063 47
0.837 31	0.051 45	0.367 91	0.125 04	0.037 90	0.059 96
0.827 64	0.053 73	0.356 34	0.125 64	0.033 87	0.056 42
0.817 75	0.056 02	0.344 91	0.126 13	0.030 06	0.052 85
0.807 66	0.058 32	0.333 61	0.126 52	0.026 47	0.049 27
0.797 37	0.060 64	0.322 47	0.126 79	0.023 09	0.045 68
0.786 89	0.062 96	0.311 48	0.126 92	0.019 94	0.042 08
0.776 23	0.065 30	0.300 67	0.126 91	0.017 01	0.038 50
0.765 41	0.067 63	0.290 00	0.126 71	0.014 30	0.034 92

Continued on next page.

x	y	x	y	x	y
0.011 82	0.031 37	0.159 98	−0.046 96	0.638 23	−0.005 56
0.009 57	0.027 85	0.169 58	−0.047 84	0.652 11	−0.003 06
0.007 55	0.024 38	0.179 38	−0.048 65	0.665 99	−0.000 63
0.005 76	0.020 95	0.189 36	−0.049 41	0.679 87	0.001 71
0.004 21	0.017 58	0.199 52	−0.050 10	0.693 71	0.003 96
0.002 89	0.014 29	0.209 85	−0.050 73	0.707 52	0.006 09
0.001 82	0.011 07	0.220 35	−0.051 29	0.721 26	0.008 09
0.000 99	0.007 95	0.230 99	−0.051 78	0.734 91	0.009 97
0.000 41	0.004 94	0.241 79	−0.052 20	0.748 47	0.011 70
0.000 08	0.002 06	0.252 72	−0.052 55	0.761 90	0.013 29
0.000 01	−0.000 68	0.263 78	−0.052 81	0.775 19	0.014 71
0.000 21	−0.003 25	0.274 96	−0.053 00	0.788 31	0.015 98
0.000 69	−0.005 62	0.286 24	−0.053 10	0.801 25	0.017 07
0.001 50	−0.007 70	0.297 63	−0.053 12	0.813 97	0.017 99
0.002 83	−0.009 54	0.309 11	−0.053 04	0.826 46	0.018 73
0.004 72	−0.011 31	0.320 68	−0.052 86	0.838 69	0.019 29
0.007 05	−0.013 08	0.332 31	−0.052 58	0.850 64	0.019 67
0.009 78	−0.014 82	0.344 01	−0.052 19	0.862 28	0.019 87
0.012 90	−0.016 56	0.355 76	−0.051 68	0.873 60	0.019 89
0.016 38	−0.018 27	0.367 55	−0.051 04	0.884 57	0.019 74
0.020 22	−0.019 96	0.379 37	−0.050 25	0.895 16	0.019 41
0.024 41	−0.021 64	0.391 22	−0.049 31	0.905 36	0.018 92
0.028 93	−0.023 28	0.403 05	−0.048 14	0.915 14	0.018 28
0.033 78	−0.024 91	0.414 96	−0.046 70	0.924 48	0.017 48
0.038 96	−0.026 50	0.427 04	−0.045 02	0.933 36	0.016 54
0.044 44	−0.028 07	0.439 28	−0.043 16	0.941 76	0.015 47
0.050 24	−0.029 61	0.451 68	−0.041 14	0.949 66	0.014 28
0.056 34	−0.031 11	0.464 23	−0.038 99	0.957 04	0.012 95
0.062 73	−0.032 58	0.476 93	−0.036 72	0.963 84	0.011 52
0.069 41	−0.034 01	0.489 78	−0.034 35	0.970 03	0.010 03
0.076 38	−0.035 41	0.502 76	−0.031 89	0.975 60	0.008 52
0.083 61	−0.036 77	0.515 87	−0.029 36	0.980 53	0.007 04
0.091 12	−0.038 09	0.529 10	−0.026 76	0.984 85	0.005 64
0.098 89	−0.039 36	0.542 45	−0.024 13	0.988 58	0.004 35
0.106 91	−0.040 59	0.555 90	−0.021 46	0.991 73	0.003 21
0.115 18	−0.041 78	0.569 45	−0.018 77	0.994 33	0.002 23
0.123 69	−0.042 92	0.583 08	−0.016 08	0.996 41	0.001 42
0.132 43	−0.044 01	0.596 79	−0.013 40	0.998 00	0.000 80
0.141 40	−0.045 04	0.610 56	−0.010 75	0.999 12	0.000 35
0.150 58	−0.046 03	0.624 38	−0.008 13	0.999 78	0.000 09