Deterministic design assumes that there is no uncertainty in the modeling parameters, and as a consequence, there is no variability in the simulation outputs. Therefore, deterministic optimal designs that are obtained without taking into account uncertainty are usually unreliable. This is the case with transonic shape optimization, where the randomness in the cruise Mach number might have significant impact on the optimal geometric design. In this context, a stochastic search turns out to be more appropriate.

Approaches to stochastic optimization have followed a variety of modeling philosophies, but little has been done to systematically compare different models. The goal of this thesis is to present a comparison between two stochastic optimization algorithms, with the emphasis on applications, especially on the airfoil shape optimization. Single-objective and multi-objective optimization programs are analyzed as well.

The relationship between the expected minimum value (EMV) criterion and the minimum expected value (MEV) criterion is explored, and it is shown that, under favorable conditions, a better optimal point could be obtained via the EMV approach. Unfortunately, the advantages of using the EMV approach are far outweighed by the prohibitive exorbitant computational cost.
THE FLORIDA STATE UNIVERSITY
COLLEGE OF ARTS AND SCIENCES

SINGLE- AND MULTIPLE-OBJECTIVE STOCHASTIC PROGRAMMING MODELS WITH APPLICATIONS TO AERODYNAMICS

By

ANA-MARIA CROICU

A dissertation submitted to the Department of Mathematics in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Degree Awarded:
Fall Semester, 2005
The members of the Committee approve the dissertation of Ana-Maria Croicu defended on October 28, 2005.

---

M. Yousuff Hussaini  
Professor Directing dissertation

---

Anuj Srivastava  
Outside Committee Member

---

David Kopriva  
Committee Member

---

Qi Wang  
Committee Member

Approved:

Philip L. Bowers, Chair  
Department of Mathematics

Joseph Travis, Dean, College of Arts and Sciences

The Office of Graduate Studies has verified and approved the above named committee members.
The members of the Committee approve the dissertation of Ana-Maria Croicu defended on October 28, 2005.

M. Yousuff Hussaini  
Professor Directing dissertation

Anuj Srivastava  
Outside Committee Member

David Kopriva  
Committee Member

Qi Wang  
Committee Member

The Office of Graduate Studies has verified and approved the above named committee members.
To my dear parents, lovely daughter and wonderful husband
ACKNOWLEDGEMENTS

I would like to express my utmost gratitude to my adviser, Professor M. Y. Hussaini, both for giving me the opportunity to work in this interesting field of Applied Mathematics and for his invaluable advice, his continual guidance, encouragement and support throughout this research.

I would also wish to acknowledge my committee members, Professor D. Kopriva, Professor Q. Wang and Professor A. Srivastava, who helped me grow more mature during my years as a graduate student, for their great interest in my work and valuable suggestions.

I wish to thank Professor A. Jameson and his Research Postdoctoral Associate S. Kim for helping me with their code.

Most importantly, I cannot thank enough to my family, my parents, my daughter and my husband, for their endless support and love over the years.
# Table of Contents

List of Tables ................................................................. vii  
List of Figures ................................................................. viii  
Abstract ................................................................................. xii  

1. **Introduction** ............................................................... 1  
   1.1 Optimization under Uncertainty, or, Stochastic Programming ..... 1  
      1.1.1 About Stochastic Programming ........................................... 1  
      1.1.2 Models for Stochastic Programming Problems ....................... 2  
   1.2 Multi-objective Optimization ............................................... 6  
      1.2.1 Problem Formulation. Optimum ........................................... 6  
      1.2.2 Multi-objective Optimization Approaches ............................. 9  
      1.2.3 Multi-objective Stochastic Programming ............................. 12  
   1.3 Goals of this Research .................................................... 14  
      1.3.1 MEV versus EMV Approach .............................................. 15  
      1.3.2 Single-objective versus Multiple-objective. Nash Equilibrium Point ........................................ 15  
      1.3.3 Airfoil Shape Optimization .............................................. 16  

2. **Single-objective (SO) EMV/MEV Approach** ........................... 19  
   2.1 Problem Formulation ...................................................... 19  
   2.2 Theoretical Results ..................................................... 20  
   2.3 Examples ......................................................................... 27  
      2.3.1 Free Undamped Vibration ............................................... 27  
      2.3.2 Population Growth ....................................................... 31  
      2.3.3 The Burgers Equation .................................................... 35  
   2.4 Application: Airfoil Shape Optimization ............................... 39  
      2.4.1 EMV/MEV Approach ...................................................... 39  
      2.4.2 Results ........................................................................... 42  

3. **Multi-objective (MO) EMV/MEV Approach** ............................... 65  
   3.1 Formulation ................................................................. 65  
   3.2 Stochastic Nash Equilibrium for the Periodic Burgers Equation .... 66  
      3.2.1 Problem Formulation ...................................................... 66  


# LIST OF TABLES

2.1 Comparison of EMV and MEV approach for (SO) airfoil optimization . . . 64
3.1 The initial and final values of the objectives for the Burgers equation . . . 81
3.2 Additional results for the deterministic Burgers equation . . . . . . . . . . 81
3.3 Comparison of EMV and MEV approach for stochastic Burgers equation . . 130
3.4 Additional results for the Nash (MO) airfoil optimization . . . . . . . . . . 136
3.5 Drag and Lift for the Nash (MO) geometry . . . . . . . . . . . . . . . . . . 136
3.6 Angle of attack (MO) airfoil optimization . . . . . . . . . . . . . . . . . . 164
3.7 Comparison of EMV and MEV approach for (MO) airfoil optimization . . 164
LIST OF FIGURES

2.1 Free Undamped Vibration .................................................. 30
2.2 Population Growth ............................................................ 34
2.3 Burgers Equation – uniform $\nu$ ......................................... 37
2.4 Burgers Equation – normal $\nu$ ........................................... 38
2.5 (SO) Drag Coefficient at EMV–$M \sim U(0.7, 0.8)$ & initial shape .......... 44
2.6 (SO) Pressure (for EMV–$M \sim U(0.7, 0.8)$ & initial shape) at $M = 0.7907923$ 45
2.7 (SO) Geometry (EMV–$M \sim U(0.7, 0.8)$ & initial) ......................... 46
2.8 (SO) Drag Coefficient at MEV–$M \sim U(0.7, 0.8)$ & initial shape .......... 47
2.9 (SO) Pressure (for MEV–$M \sim U(0.7, 0.8)$ & initial shape) at $M = 0.7907923$ 48
2.10 (SO) Geometry (MEV–$M \sim U(0.7, 0.8)$ & initial) ......................... 49
2.11 (SO) Drag Coefficient at EMV–& MEV–$M \sim U(0.7, 0.8)$ shape .......... 50
2.12 (SO) Pressure (for EMV– & MEV–$M \sim U(0.7, 0.8)$ shape) at $M = 0.7907923$ 51
2.13 (SO) Geometry (EMV– & MEV–$M \sim U(0.7, 0.8)$) ......................... 52
2.14 (SO) Drag Coefficient at EMV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape .......... 53
2.15 (SO) Pressure (for EMV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape) at $M = 0.7540006$ 54
2.16 (SO) Geometry (EMV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial) ......................... 55
2.17 (SO) Drag Coefficient at MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape .......... 56
2.18 (SO) Pressure (for MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape) at $M = 0.7540006$ 57
2.19 (SO) Geometry (MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial) ......................... 58
2.20 (SO) Drag Coefficient at EMV–& MEV–M \sim N(0.75, 0.025^2) shape . . . . . . 59
2.21 (SO) Drag Coefficient at EMV–& MEV–M \sim N(0.75, 0.025^2) shape (zoom) 60
2.22 (SO) Drag Coefficient at EMV–& MEV–M \sim N(0.75, 0.025^2) shape (zoom) 61
2.23 (SO) Pressure (for EMV– & MEV– M \sim N(0.75, 0.025^2) shape) at \( M = 0.7540006 \) . . . . . . . . . . . . . . . . . . . . . . . . . . 62
2.24 (SO) Geometry (EMV– & MEV– M \sim N(0.75, 0.025^2)) . . . . . . . . . . . . 63
3.1 Uncontrolled, controlled, target solution at \( t=0.5 \) . . . . . . . . . . . . . . 77
3.2 Uncontrolled, controlled, target solution at \( t=1.-dt \) . . . . . . . . . . . . . 78
3.3 Controls at \( t=0.5 \) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 79
3.4 Controls at \( t=1.-dt \) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 80
3.5 \( J_1 \) at initial and final EMV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 88
3.6 \( J_2 \) at initial and final EMV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 89
3.7 Uncontrolled, EMV controlled, target solution \( t=0.5-\nu \sim U[0.001, 0.0015] \) . 90
3.8 Uncontrolled, EMV controlled, target solution \( t=1.-dt-\nu \sim U[0.001, 0.0015] \) . 91
3.9 EMV optimal control at \( t=0.5-\nu \sim U[0.001, 0.0015] \) . . . . . . . . . . . . . 92
3.10 EMV optimal control \( t=1.-dt-\nu \sim U[0.001, 0.0015] \) . . . . . . . . . . . . 93
3.11 \( J_1 \) at initial and final MEV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 94
3.12 \( J_2 \) at initial and final MEV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 95
3.13 Uncontrolled, MEV controlled, target solution \( t=0.5-\nu \sim U[0.001, 0.0015] \) . 96
3.14 Uncontrolled, MEV controlled, target solution \( t=1.-dt-\nu \sim U[0.001, 0.0015] \) . 97
3.15 MEV optimal control \( t=0.5-\nu \sim U[0.001, 0.0015] \) . . . . . . . . . . . . . 98
3.16 MEV optimal control \( t=1.-dt-\nu \sim U[0.001, 0.0015] \) . . . . . . . . . . . . 99
3.17 \( J_1 \) at final EMV and MEV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 100
3.18 \( J_2 \) at final EMV and MEV control \( -\nu \sim U[0.001, 0.0015] \) . . . . . . . 101
3.19 \( J_2 \) at final EMV and MEV control \( -\nu \sim U[0.001, 0.0015] \) (zoom) . . . . . . . 102
3.20 EMV and MEV controlled solution \( t=0.5, \nu = 0.00125-\nu \sim U[0.001, 0.0015] \) 103
3.21 EMV and MEV controlled solution \( t=1.-dt, \nu = 0.00125-\nu \sim U[0.001, 0.0015] \) 104
3.22 EMV and MEV optimal control $v_1 \ t=0.5-\nu \sim U[0.001, 0.0015]$ ....... 105
3.23 EMV and MEV optimal control $v_2 \ t=0.5-\nu \sim U[0.001, 0.0015]$ ....... 106
3.24 EMV and MEV optimal control $v_1 \ t=1.-dt-\nu \sim U[0.001, 0.0015]$ ....... 107
3.25 EMV and MEV optimal control $v_2 \ t=1.-dt-\nu \sim U[0.001, 0.0015]$ ....... 108
3.26 $J_1$ at initial and final EMV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 109
3.27 $J_2$ at initial and final EMV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 110
3.28 Uncontrolled, EMV controlled, target solution $t=0.5-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 111
3.29 Uncontrolled, EMV controlled, target solution $t=1.-dt-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 112
3.30 EMV optimal control $t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 113
3.31 EMV optimal control $t=1.-dt - \nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 114
3.32 $J_1$ at initial and final MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 115
3.33 $J_2$ at initial and final MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 116
3.34 Uncontrolled, MEV controlled, target solution $t=0.5-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 117
3.35 Uncontrolled, MEV controlled, target solution $t=1.-dt-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 118
3.36 MEV optimal control $t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 119
3.37 MEV optimal control $t=1.-dt - \nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 120
3.38 $J_1$ at final EMV and MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 121
3.39 $J_1$ at final EMV and MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$(zoom) ....... 122
3.40 $J_2$ at final EMV and MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 123
3.41 EMV and MEV controlled solution $t=0.5, \nu = 0.00125-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 124
3.42 EMV and MEV controlled solution $t=1.-dt,\nu = 0.00125-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 125
3.43 EMV and MEV optimal control $v_1 \ t=0.5-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 126
3.44 EMV and MEV optimal control $v_2 \ t=0.5-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 127
3.45 EMV and MEV optimal control $v_1 \ t=1.-dt-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 128
3.46 EMV and MEV optimal control $v_2 \ t=1.-dt-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ ....... 129
3.47 (MO) Geometry Nash & initial ....... 134
3.48 (MO) Pressure at Nash & initial point, \( M = 0.7907923 \) . . . . . . . . . . . . 135
3.49 (MO) Drag Coefficient at EMV– \( M \sim U(0.7, 0.8) \) & initial point . . . . . . 140
3.50 (MO) Lift Coefficient at EMV– \( M \sim U(0.7, 0.8) \) & initial point . . . . . . 141
3.51 (MO) Pressure (for EMV– \( M \sim U(0.7, 0.8) \) & initial point) at \( M = 0.7907923 \) 142
3.52 (MO) Geometry (EMV– \( M \sim U(0.7, 0.8) \) & initial) . . . . . . . . . . . . . . . 143
3.53 (MO) Drag Coefficient at MEV– \( M \sim U(0.7, 0.8) \) & initial point . . . . . . 144
3.54 (MO) Lift Coefficient at MEV– \( M \sim U(0.7, 0.8) \) & initial point . . . . . . 145
3.55 (MO) Pressure (for MEV– \( M \sim U(0.7, 0.8) \) & initial point) at \( M = 0.7907923 \) 146
3.56 (MO) Geometry (MEV– \( M \sim U(0.7, 0.8) \) & initial) . . . . . . . . . . . . . . . 147
3.57 (MO) Drag Coefficient at EMV & MEV point– \( M \sim U(0.7, 0.8) \) . . . . . . . 148
3.58 (MO) Lift Coefficient at EMV & MEV point– \( M \sim U(0.7, 0.8) \) . . . . . . . 149
3.59 (MO) Pressure (for EMV & MEV point– \( M \sim U(0.7, 0.8) \)) at \( M = 0.7907923 \) 150
3.60 (MO) Geometry (EMV & MEV– \( M \sim U(0.7, 0.8) \)) . . . . . . . . . . . . . . . 151
3.61 (MO) Drag Coefficient at EMV– \( M \sim N(0.75, 0.025^2) \) & initial point . . . 152
3.62 (MO) Lift Coefficient at EMV– \( M \sim N(0.75, 0.025^2) \) & initial point . . . 153
3.63 (MO) Pressure (for EMV– \( M \sim N(0.75, 0.025^2) \) & initial) at \( M = 0.7540006 \) 154
3.64 (MO) Geometry (EMV– \( M \sim N(0.75, 0.025^2) \) & initial) . . . . . . . . . . . 155
3.65 (MO) Drag Coefficient at MEV– \( M \sim N(0.75, 0.025^2) \) & initial point . . . 156
3.66 (MO) Lift Coefficient at MEV– \( M \sim N(0.75, 0.025^2) \) & initial point . . . 157
3.67 (MO) Pressure (for MEV– \( M \sim N(0.75, 0.025^2) \) & initial) at \( M = 0.7540006 \) 158
3.68 (MO) Geometry (MEV– \( M \sim N(0.75, 0.025^2) \) & initial) . . . . . . . . . . . 159
3.69 (MO) Drag Coefficient at EMV & MEV point– \( M \sim N(0.75, 0.025^2) \) . . . 160
3.70 (MO) Lift Coefficient at EMV & MEV point– \( M \sim N(0.75, 0.025^2) \) . . . 161
3.71 (MO) Pressure (for EMV & MEV– \( M \sim N(0.75, 0.025^2) \)) at \( M = 0.7540006 \) 162
3.72 (MO) Geometry (EMV & MEV– \( M \sim N(0.75, 0.025^2) \)) . . . . . . . . . . . . . 163
ABSTRACT

Deterministic design assumes that there is no uncertainty in the modeling parameters, and as a consequence, there is no variability in the simulation outputs. Therefore, deterministic optimal designs that are obtained without taking into account uncertainty are usually unreliable. This is the case with transonic shape optimization, where the randomness in the cruise Mach number might have significant impact on the optimal geometric design. In this context, a stochastic search turns out to be more appropriate.

Approaches to stochastic optimization have followed a variety of modeling philosophies, but little has been done to systematically compare different models. The goal of this thesis is to present a comparison between two stochastic optimization algorithms, with the emphasis on applications, especially on the airfoil shape optimization. Single-objective and multi-objective optimization programs are analyzed as well.

The relationship between the expected minimum value (EMV) criterion and the minimum expected value (MEV) criterion is explored, and it is shown that, under favorable conditions, a better optimal point could be obtained via the EMV approach. Unfortunately, the advantages of using the EMV approach are far outweighed by the prohibitive exorbitant computational cost.
CHAPTER 1

INTRODUCTION

1.1 Optimization under Uncertainty, or, Stochastic Programming

1.1.1 About Stochastic Programming

Problems of optimality under uncertainty occur frequently in a wide variety of real world problems in science, engineering and technology, which have probabilistic parameters, non-deterministic initial conditions, uncertain input situations and models based on incomplete knowledge. Specifically, a large number of problems such as engineering design ([52]), supply-allocation ([99]), production planning ([25]) and scheduling ([68]), transportation ([69]), inventory network ([37], [65]), finance ([80]) require that decisions be made in the presence of uncertainty. Uncertainty, for instance, governs the prices of fuels, the availability of electricity, and the demand for chemicals. In other words, much of life involves making optimal choices under uncertainty, i.e., choosing the optimal from some set of optional courses of action in uncertain situations. For these kind of problems, the effect of decisions made today will have consequences that can’t fully be determined until tomorrow. If these uncertainties are not taken into account in mathematical models, costly mistakes can be made when they are applied.

To be more specific, let us define the problem that motivated the present research. Let \((\Omega, \mathcal{F}, P)\) be a probability space and let \(\omega\) be an \(\mathcal{F}\)–measurable random variable. We denote by \(d\) the design variable, whose domain is confined to a Banach space \((X, \|\cdot\|)\). Let us consider a functional \(J : X \times \Omega \to IR\), which is assumed to have a minimum for any given value of the random variable \(\omega\). The minimization problem under uncertainty consists in finding an
optimal point $d^* \in D \subseteq X$, such that
\[
J(d^*, \omega) = \min_{d \in D} J(d, \omega), \text{ for all } \omega \in \Omega. \tag{1.1}
\]

To simplify our notation, let us denote the optimal point $d^*$ by
\[
d^* = \arg \min_{d \in D} J(d, \omega).
\]

The problem of optimization under uncertainty (1.1), posed this way, seems to be quite challenging because it requires finding an optimum point that minimizes the functional $J(\cdot, \omega)$ for all possible values of the random variable $\omega$. Indeed, solutions obtained from an optimization program may be optimal for certain values of the random variable $\omega$, but for other values of uncertainty the decision taken may be far from optimal. Basically, the solution of a stochastic optimization model is meant to produce a solution that is robust with respect to all possible values of the randomness.

Optimization under uncertainty, or stochastic programming, represents an extension of optimization to allow randomness in the modeling stage. In other words, in a stochastic optimization model, certain parameters are stochastic variables, having some continuous or discrete probability distributions, as defined before in (1.1).

Once the randomness is included in the formulation of the mathematical program, it is not immediately clear how to formulate a well-posed optimization problem. A number of convenient reformulations of the problem (1.1) are possible depending on when decisions must be taken relative to the realization of the random variable, and these will be summarized next.

### 1.1.2 Models for Stochastic Programming Problems

Beginning with the seminal works of Beale (1955, [7]), Dantzig (1955, [36]), Tintner (1955, [107]), Bellman (1957, [9]), Charnes and Cooper (1959, [22]), optimization under uncertainty has experienced and still experiences rapid development in both theory and applications.

Approaches to optimization under uncertainty have followed a variety of modeling philosophies, we provide here only a very brief review. Excellent reviews can be found in Rockafellar ([96]), Liu ([70]), Biegler and Grossmann ([14]), and Sahinidis ([97]). Before we present the most important models, we should stress out again that the exact solution of
stochastic optimization problems is not generally known, and the choice of approach depends upon data availability and the goals of the decision-makers.

The models found in the literature can be divided in two main categories: models where the decision has to be taken in advance of any information about the randomness $\omega$, other than the probability distribution function, and models where a two-stage optimization problem permits improvements after the random events have presented themselves. We will start by presenting models from the first category.

1. **One-stage programming**

Most frequently by far in random circumstances, the uncertainty is pushed out of sight by replacing the random variable $\omega$ by a particular estimate $\hat{\omega}$. The problem to be solved is in this case

$$\min_{d \in D} J(d, \hat{\omega}). \quad (1.2)$$

The justification usually given for such a formulation is that practical problems are hard enough to solve without getting involved in the difficult uncertainty. On the other hand, a decision based on concentrating the randomness just to a single value $\hat{\omega}$ can’t do a good job of reflecting all possible situations, no matter how well selected $\hat{\omega}$ might be.

Another approach, the so called the worst-case approach, looks at the worst outcome that might result, and makes no distinction between outcomes according to the probability distribution function. In this case, the optimization problem can be written as

$$\min_{d \in D} \max_{\omega \in \Omega} J(d, \omega). \quad (1.3)$$

The first method dealing with stochastic parameters in stochastic manner are the so-called “expected value models”.

In the approach used and justified in [52] and [51], the best design or decision is the one that minimizes the overall risk, and it is based on the Von Neumann-Morgenstern statistical decision theory [89]. In this case, the optimization problem is written as

$$\min_{d \in D} E(J(d, \omega)) \quad (1.4)$$
and the Bayes’ decision is given by

$$d_{MEV} = \arg \min_{d \in D} E (J (d, \omega)).$$

(1.5)

For our purpose, we assume that $d_{MEV}$ is well-defined, i.e. the function $E (J (d, \omega))$ has a single minimum. The index “MEV” stands for “minimum (maximum) expected value”, and it is suggested by [51].

Various “expected value models” of the type (1.4) may differ by the way the constraints are incorporated, i.e. by the definition of the set $D$. Conventional constraint formulations may be inappropriate to deal with randomness. If the constraints have nothing to do with $\omega$, then there is not much to say. But when they do depend on randomness, they need special care. There are two main approaches to formulate an optimization problem when there is randomness in the constraints ([90]). There are expectation-based constraints, which imply the following formulation of the stochastic program

$$\left\{ \begin{array}{l}
\min_{d \in X} E (J (d, \omega)) \\
E J_i (d, \omega) \leq 0, \text{ for } i = 1, 2, ..., m
\end{array} \right.$$  

(1.6)

The constraints are expected to be satisfied “on average”, but the amount of constraint violation is not restricted. Another special formulation of stochastic constraints was pioneered by Charnes and Cooper ([22]) as a means of handling the uncertain constraints by specifying confidence levels ($p_i, i = 1, 2, ..., m$) at which it is desired that the constraints hold. The problem can therefore be formulated as

$$\left\{ \begin{array}{l}
\min_{d \in X} E (J (d, \omega)) \\
P (J_i (d, \omega) \leq 0) \geq p_i, \text{ for } i = 1, 2, ..., m
\end{array} \right.$$  

(1.7)

and carry the name of “chance-constrained programming”. Despite of the fact that this last formulation is very appealing, in many applications it is questionable because of the technical ramifications. A crucial issue that should be considered is to ensure that the probabilities $p_i$ are not set so high that no feasible solution exists for the problem. Still, the chance-constrained model is widely applied.
2. Programming with recourse

Regarding the category of stochastic models “programming with recourse”, the workhorse in stochastic programming is the so-called the “two-stage stochastic optimization”, where the distribution of the future outcomes are taken into account when the first stage decisions are made. The first-stage variables are those that must be decided before the actual realization of the uncertainty. Once the random variables have presented themselves, further improvements or corrections can be made by selecting, at a certain cost, the values of the second-stage, or recourse, variables. For instance, to plan a new facility, one has to decide on its location, scale, etc., but only later one has to decide how to operate it.

The well-known two-stage stochastic linear program with recourse was introduced independently by Beale ([7]) and Dantzig ([36]). The mathematical formulation of two-stage stochastic programming is as follows

\[
\min_{d_1 \in D_1} \left( j(d_1) + E \left( \min_{d_2 \in D_2} J(d_1, d_2, \omega) \right) \right) \tag{1.8}
\]

with \( D = D_1 \times D_2 \), and \( j : D_1 \to IR \). A more widely used formulation of (1.8) can be written as

\[
\min_{d_1 \in D_1} (j(d_1) + E(Q(d_1, \omega))) \tag{1.9}
\]

\[
Q(d_1, \omega) = \min_{d_2 \in D_2} J(d_1, d_2, \omega) \tag{1.10}
\]

The problem (1.9) with variable \( d_1 \) constitutes the first stage, which needs to be decided prior to the realization of the uncertain parameter \( \omega \). The problem (1.10) with variable \( d_2 \) means the second stage. The objective is to select the first-stage variables so that the sum of the first-stage costs and the expected value of the second-stage costs is minimized.

The difference between chance-constrained programming and stochastic programming with recourse is that they use different measures for risk.

Another possible stochastic formulation, called “robust optimization”, recognizes that it may be impossible to determine a solution that is feasible for all scenarios. It forms an optimization problem that minimizes a weighted sum of cost and infeasibility. For instance, to capture the notion of risk in stochastic programming,
Mulvey, Vanderbei, and Zenios ([81]) proposed the following modification of the objective function in (1.8)

$$\min_{d_1 \in D_1} \left( j(d_1) + E \left( \min_{d_2 \in D_2} J(d_1, d_2, \omega) + \lambda f(d_2, \omega) \right) \right) \quad (1.11)$$

where $f : D_2 \times \Omega \to \mathbb{R}$ is a variability measure of the second-stage costs, and $\lambda > 0$ represents the risk tolerance of the modeler.

Finally, dating back to the 1950’s is a theory of multistage decision problems under uncertainty known as “dynamic programming” ([9]). This theory focuses on states and controls and grows out of a stage-by-stage process of control under uncertainty. We refer to ([96]) for more information about the mathematical formulation.

In addition to all the models presented above, fuzzy mathematical programming offers a powerful means of handling optimization problems under uncertainty. Many of the developments in this area are based on the seminal paper by Bellman and Zadeh ([8]). Because this approach is beyond the scope of the present work, we refer the reader to a detailed survey on fuzzy optimization has been made by Lahandjula ([64]).

## 1.2 Multi-objective Optimization

### 1.2.1 Problem Formulation. Optimum

In most real-world problems, several goals must be satisfied simultaneously to obtain an optimal solution ([29]). These problems with several objectives are called “multi-objective” or “vector” optimization problems, and were originally studied in the context of economics. However, scientists and engineers soon realized that such problems naturally arise in all areas of knowledge ([101]). The multiple objectives are typically conflicting and non-commensurable, and must be satisfied simultaneously.

John von Neumann and Oskar Morgenstern ([111]) were the first to recognize the existence of optimization problems in economics that were “a peculiar and disconcerting mixture of several conflicting problems”. However, no real contribution to the solution of such problems was made until the 1950s.

The application of multi-objective optimization to domains outside economics began with the work of Tjalling Koopmans (1951, [62]) in production theory, and with the work
of Marglin (1967, [76]) in water resources planning. The first application of multi-objective optimization in engineering was in early 1960s (Zadeh, 1963, [121]), but its use didn’t become generalized until the 1970s ([101]).

We are interested to solve multi-objective optimization problems of the form:

$$\text{minimize } [J_1(d), J_2(d), \ldots, J_k(d)]$$

(1.12)

subject to the $m$ inequality constraints, and the $p$ equality constraints:

$$g_i(d) \geq 0, i = 1, 2, \ldots, m$$

(1.13)

$$h_i(d) = 0, i = 1, 2, \ldots, p$$

(1.14)

Denote by $D$ the set of all vectors $d$ that satisfy (1.13) and (1.14), and call it the set of all feasible vectors.

Our task is to define optimality for this kind of optimization problem.

In a simple optimization problem, the notion of optimality is straightforward. The best element is the one that realizes the minimum (or the maximum) of the objective function. In a multi-objective optimization problem, the notion of optimality is not so obvious ([75]). In a multi-objective optimization, ideally the effort must be made in finding the set of trade-off optimal solutions by considering all objectives to be important. After a set of such trade-off solutions are found, a user can then use higher-level qualitative considerations to make a choice ([40]). In other words, there is no solution that is the best for all the criteria, but there exists a set of solutions that are better than other solutions in all the search space, when considering all the objectives. This set of solutions is known as the optimal solutions of the Pareto set or non-dominated solutions. This is the most commonly adopted notion of optimality, originally proposed by Francis Ysidro Edgeworth (1881, [42]), and later generalized by Vilfredo Pareto (1896, [85]).

**Definition 1.** We say that a vector of decision variables $d^* \in D$ is Pareto optimal if there does not exist another $d \in D$ such that

$$J_i(d) \leq J_i(d^*), \text{ for all } i=1,2,\ldots,k$$

and

$$J_j(d) < J_j(d^*), \text{ for at least one } j.$$
In words, this definition says that \( d^* \) is Pareto optimal if there exists no feasible vector of decision variables \( d \in D \) that would decrease some criterion without causing a simultaneous increase in at least one other criterion. Unfortunately, this concept almost always gives not a single solution, but rather a set of solutions called the **Pareto optimal set**. The vectors \( d^* \) that correspond to the solutions included in the Pareto optimal set are called **non-dominated**. The plot of the objective functions whose non-dominated vectors are in the Pareto optimal set is called the **Pareto front**.

Where applications are concerned, Pareto optimality has established itself as nearly the only viable optimality concept when a comparison between several competing desiderata is to be reached. The concept fares well from the traditional single-criterion optimization point of view, since it disallows any decision for which all the criteria could still be improved; indeed, it provides clear information concerning the compromises that must be made ([106]).

This preference for Pareto optimality has advantages and disadvantages. Wide acceptance of a concept is needed to make it palatable in practice; as a consequence, however, many interesting and useful optimality concepts have remained relatively unknown to the possible user.

In particular, optimal decisions in antagonistic game theory may be equally well used by a single decision maker who would like decisions that display the attributes of the particular game theoretic optimum. By way of example, the Nash equilibrium provides a fault tolerant optimality concept ([106]).

Nash equilibrium defines a non-cooperative multi-objective optimization strategy first proposed by John Nash (1951, [82]). Since it originated in game theory and economics, the notion of player is often used.

**Definition 2.** A decision \( d^* = (d^*_1, d^*_2, \ldots, d^*_k) \in D \subseteq \mathbb{R}^k \) is a Nash equilibrium for the collection of criteria \( J_i, i = 1, 2, \ldots, k \), iff

\[
J_i (d^*) \leq J_i \left( d^*_1, d^*_2, \ldots, d^*_{i-1}, d_i, d^*_{i+1}, \ldots, d^*_k \right),
\]

for all \( i = 1, 2, \ldots, k \), and for every \( d \in D \).

We will use the following notation

\[
d^* = \text{Nash}_{d \in D} \left( J_1 (d), J_2 (d), \ldots, J_k (d) \right)
\]
One of the main aspects of Nash optima is that the players cannot unilaterally improve themselves. That is, each player has to optimize his criterion given that all the other criteria are fixed by the rest of the players. When no player can further improve his criterion, it means that the system has reached a Nash equilibrium state.

1.2.2 Multi-objective Optimization Approaches

As we mentioned before, most of the previous work on multi-objective optimization are based on the following idea: to find the set of non-dominated solutions (Pareto front) or to approximate it with a representative subset. Afterward the decision-maker’s preference may be applied to choose the best compromise solution from the generated set ([92]).

There are two approaches to multi-objective optimization: classical methods and evolutionary methods.

1. Classical methods

Classical methods convert the multiple objective functions into a single objective function.

The common approach in this sort of problem is to choose one objective and incorporate the other objectives as constraints. This approach was first suggested by Haines et. al. in 1971 ([49]). It has the disadvantage of limiting the choices available to the designer, making the optimization process a rather difficult task.

Another common approach is the combination of all the objectives into a single objective function (weighted sum method, weighted metric methods, value function method, goal programming methods). This technique has the drawback of modeling the original problem in an inadequate manner, generating solutions that will require a further sensitivity analysis to become reasonably useful to the designer ([29]).

Some recent applications based on classical methods can be found in [113], [46].

2. Evolutionary algorithms

The field of search and optimization has changed over the last few years by the introduction of a number of non-classical, unorthodox and stochastic search optimization algorithms ([40]). Of these, the evolutionary algorithm mimics nature’s
evolutionary principles to drive its search toward an optimal solution. One of the most striking differences to classical search and optimization algorithms is that evolutionary algorithms use a population of solutions in each iteration, instead of a single solution. Since a population of solutions is processed in each iteration, the outcome of an evolutionary algorithm is also a population of solutions. If an optimization problem has a single optimum, all evolutionary algorithm population members can be expected to converge to that optimum. However, if an optimization problem has multiple optimal solutions, an evolutionary algorithm can be used to capture multiple optimal solutions in its population.

This ability of an evolutionary algorithm to find multiple optimal solutions in one single simulation run makes evolutionary algorithms unique in solving multi-objective optimization problems ([40]).

A considerable number of evolutionary algorithms have been proposed in the last few years. We will concentrate our discussion on those techniques that have been more popular among researchers.

David Schaffer (1985, [103]) implemented the first multi-objective genetic algorithm to find a set of non-dominated solutions, that he called the “Vector Evaluated Genetic Algorithm” (VEGA). He modified the selection operator of a simple genetic algorithm so that at each generation a number of sub-populations was generated by performing proportional selection according to each objective function in turn. Thus, for a problem with $k$ objectives, $k$ sub-populations of size $N/k$ each would be generated, assuming a total population of size $N$. These sub-populations would be shuffled together to obtain a new population of size $N$, on which the genetic algorithm would apply the crossover and mutation operators in the usual way ([29]). The main advantages of these technique are its simplicity and its efficiency. However, the “middling” problem prevents the technique from finding the compromise solutions that we normally aim to produce.

Fonseca and Fleming (1993, [45]) have proposed a scheme, “Multi-Objective Genetic Algorithm” (MOGA), in which the rank of a certain individual corresponds to the number of chromosomes in the current population by which it is dominated.
All non-dominated individuals are assigned rank 1, while dominated ones are penalized according to the population density corresponding to the trade-off surface. Fitness assignment is performed. This method is combined with mating restrictions and sharing on the objective function values to preserve diversity. MOGA has been a popular technique because it is relatively simple to implement, but also because of its good performance. Its main weakness is its dependence on the sharing factor ([101]).

The “Non-dominated Sorting Genetic Algorithm” (NSGA) was proposed by Srinivas and Deb (1994, [105]) and is based on several layers of classifications of the individuals. Before the selection is performed, the population is ranked on the basis of non-domination: all non-dominated individuals are classified into one category (with a dummy fitness value, which is proportional to the population size, to provide an equal reproductive potential for these individuals). To maintain the diversity of the population, these classified individuals are shared with their dummy fitness value. Then this group of classified individuals is ignored and another layer of non-dominated individuals is considered. The process continues until all individuals in the population are classified. A stochastic remainder proportionate selection was used for this approach ([29]). Some researches have reported that NSGA has a lower overall performance than MOGA, and it seems to be also more sensitive to the value of the sharing factor than MOGA ([101]).

Horn and Nafpliotis (1994, [50]) proposed the “Niched Pareto Genetic Algorithm” (NPGA), which uses a tournament selection based on Pareto dominance. Two individuals are compared against a set of members of the population (typically, 10% of the population size). When both competitors are either dominated or non-dominated (i.e., when there is a tie), the result of the tournament is decided through fitness sharing in the objective domain. Since this approach does not apply Pareto ranking to the entire population, but only to a segment of it at each run, its main strength are that it is faster than MOGA and NSGA. Furthermore, it also produces good non-dominated fronts that can be kept for a large number of generations. However, its main weakness is that besides requiring a sharing factor, this approach also require an additional parameter: the size of the tournament
Recently, some other new genetic based multi-objective optimization techniques are proposed and applied in real life situations ([29], [58], [109], [28], [92], [102], [10], [24], [13], [74]).

3. Nash algorithms

When it comes to multiple objective optimization, Pareto optima have now became a sort of standard. However, another multiple objective scheme, this time a non-cooperative one, was presented by John Nash in the early 50s. This approach introduced the notion of player and aimed at solving multiple objective optimization problems originating from game theory and economics.

Sefrioui and Periaux (2000, [104]) suggested a Nash genetic algorithm for multi-objective optimization. Motivated by a game-theoretic approach, they allowed one player to get associated with each objective function. A player tries to optimize its objective function while keeping other functions unchanged. In a periodic sequence of operations, the Nash genetic algorithm is terminated when no more improvement is recorded. At this steady-state scenario, the resulting solution is a Nash equilibrium solution and is a candidate Pareto-optimal solution ([40]).

The articles [79], [2], [93], [94], [86] suggest the fact that game theory approach (non-genetic or genetic Nash equilibrium) may be successfully applied to practical problems.

1.2.3 Multi-objective Stochastic Programming

Because stochastic phenomena exists everywhere, it is very meaningful to focus attention on stochastic one multi-objective decision-making problem. Due to the tremendous complexity, the multi-objective stochastic optimization problems have drawn an increased amount of attention from the research community. Usually, we may tackle stochastic programming by the methods presented in 1.1.2, like the “expectation model”, “chance-constrained programming”, “two-stage stochastic programming”. In other words, the stochastic problem is converted into a deterministic one. To solve the new multi-objective program, we may
employ the approaches presented earlier, like the classical methods, the weighted sum for instance, or evolutionary algorithms. Not many articles attack applications where optimal solutions (Pareto optimal solutions) under uncertainties are sought.

In [77] (2003), Marseguerra, Zio, and Podofillini proposed an approach, based on the effective coupling of genetic algorithms and Monte Carlo simulations, for the multi-objective optimization of the technical specifications of nuclear safety systems. The method transparently and explicitly accounts for the uncertainties in the model parameters by attempting to minimize both the expected value of the system unavailability and its associated variance.

Another example is the model formulation and solution procedure described in [69] (2003), which provide a means for assessing the impacts of uncertainty (in future demands and operating conditions) on fleet sizing decisions, and for supporting explicit decisions on trading off expected costs against risk.

Tonnon, Mammino, and Bernardini, in [108] (2002), addressed two common questions in tunnel design: (i) how to choose an optimum solution when more than one conflicting objective must be achieved: (ii) how to deal with data affected both by imprecision and randomness. They used Fuzzy Set Theory and Random Set Theory to develop a general interactive procedure, which is then applied to the design of tunnel support / reinforcement.

The article [47] (2003) presents a bi-criteria genetic algorithm approach to solve fuzzy job shop scheduling problems in which the objective of integral value and the uncertainty of the fuzzy makespan are minimized. In this approach, imprecise processing times are modeled as triangular fuzzy numbers, which results in a makespan that is a triangular fuzzy number.

The paper [100] (2003) is devoted to the solution of voltage control in distribution systems in the presence of uncertainties. The proposed solution is based on the multi-objective algorithm that takes into account active power losses, 'soft' voltage constraints, supply and voltage quality and variations of control resource variables. Uncertain input data and optimization criteria are specified in linguistic forms, translated to corresponding fuzzy sets.

Sakawa, Kato and Nishizaki, in [98] (2003), focus on multi-objective linear programming problems with random variable coefficients in objective functions and/or constraints. After incorporating fuzzy goals of the decision maker for the objective functions, they propose an interactive fuzzy satisfying method for the expectation model to derive a satisfying solution for the decision maker.

It is worth remarking that the multi-objective stochastic programming literature lacks in applications where the Nash equilibrium point is employed. Only a few articles tackle the idea of stochastic Nash game, where the players seek to improve their own expected payoff (see [12, 57]).

1.3 Goals of this Research

There are several challenges and opportunities in the area of optimization under uncertainty. For instance, there is a notable need for systematic comparisons between the different modeling philosophies, as Sahinidis noted in ([97]). A door in this sense has been opened by Liu and Sahinidis ([72]), who compared stochastic and fuzzy programming as applied to chemical process planning, but no comparison between different stochastic programming models has been found in the literature yet, at least as far as we are aware.

The work of Huyse ([52]), which actually motivated our investigation, deals with the problem of airfoil shape optimization. An important concern in the shape optimization of airfoils is the sensitivity of the optimal design to small fluctuations in the operating conditions (e.g. flight speed), that cannot be avoided. Therefore, we would like to achieve the best performance for all combinations of operating conditions, in other words, we are interested in designing an airfoil that minimizes the drag coefficient or maximizes the lift coefficient under the uncertainty of the Mach number. Because the optimal design (like airfoil shape) cannot be improved after the realization of the random variable, we should focus on models where the decision has to be taken in advance of any information about the randomness. In ([52]) they employ the MEV concept, which optimizes the expectation of the objective/risk function subject to some constraint. There is no reason why one should not consider the expectation of the optimized objective/risk function, namely the EMV concept.

Therefore, the purpose of this current research is twofold: to compare the MEV and EMV approaches (for single- and multi-objective stochastic programming problems), and to apply them to theoretical and practical problems, with the emphasis on the two-dimensional aerodynamic optimization.

A short description of these goals is given next.
1.3.1 MEV versus EMV Approach

One possible approach, as we mentioned earlier, is the Minimum (Maximum) Expected Value criterion (MEV), which can be formulated as

$$\min_{d \in D} E(J(d, \omega)).$$

The MEV decision, corresponding to the MEV criterion, is defined by

$$d_{MEV} = \arg \min_{d \in D} E(J(d, \omega)).$$

On the other hand, a practical solution to the minimization problem (1.1) may well be to “average” over the entire range of optima, i.e., to consider the optimization problem and the optimum solution respectively to be

$$E\left(\min_{d \in D} J(d, \omega)\right)$$

$$d_{EMV} = E\left(\arg \min_{d \in D} J(d, \omega)\right) \quad (1.15)$$

This new formulation (1.15), which we call the Expected Minimum Value criterion (EMV), is, in our opinion, a more natural representation of the initial problem under consideration (1.1). It is therefore interesting to analyze how the EMV solution $d_{EMV}$ compares with the MEV solution $d_{MEV}$. It is quite clear that the EMV criterion is rather closer to the initial optimization problem, but does it give us a “better” optimum?

The theory developed in the next chapter shows that the expectation and optimization operators are not commutative and that $EMV \leq MEV$ for continuous objective functions. Moreover, the probability of having a lower risk (lower value of the objective function) at the EMV design point compared to the MEV design strategy may be greater than 50% under certain assumptions. Applications to stochastic problems of free undamped vibrations, population growth, nonlinear Burgers equation, airfoil shape optimization are considered in support of the theoretical results.

1.3.2 Single-objective versus Multiple-objective.

Nash Equilibrium Point

The same EMV / MEV approaches could be applied in the case of a multi-objective stochastic optimization problem. Our purpose is to make use of both EMV / MEV ideas
vis-a-vis the Nash equilibrium point. Therefore, the stochastic MEV Nash equilibrium would be defined as

\[ d^{Nash}_{MEV} = \text{Nash}_{d \in D} (EJ_1 (d, \omega), EJ_2 (d, \omega), \ldots, EJ_k (d, \omega)) \]

while the EMV Nash equilibrium would be given by

\[ d^{Nash}_{EMV} = E (\text{Nash}_{d \in D} (J_1 (d, \omega), J_2 (d, \omega), \ldots, J_k (d, \omega))) \].

Performance tests are conducted on the nonlinear periodic Burgers equation and the airfoil shape optimization. These provide support for the same argument that EMV strategy may lead to lower risk.

1.3.3 Airfoil Shape Optimization

With the advances in computational fluid dynamics and modern computers, aerodynamic design optimization becomes more important than ever, and has been extensively investigated in recent years. Optimization techniques based on adjoint methods [61, 60, 56, 59], evolutionary or stochastic algorithms [115, 38, 117, 71] have been developed and implemented.

The adjoint method is extremely efficient in a gradient-based optimization technique, since the necessary gradients are obtained via the solution of the adjoint equations of the governing equations of interest. The computational cost incurred in the calculation of the complete gradient is independent of the number of design variables, and it is similar to that of the flow solution. The adjoint method was applied in this way to shape design for elliptic equations by Pironneau [88]. It was first used in transonic flow by Jameson [55]. Since then, this method has become a popular tool for aerodynamic optimization [61, 60, 56, 59]. Unfortunately, there is a drawback associated with the gradient-based methods, and implicitly with the adjoint method: they may get trapped in a local minimum.

To avoid finding a local minimum instead of a global one, stochastic methods such as genetic algorithms, simulated annealing algorithms, and so on have been applied to aerodynamic shape design. Although these algorithms were inspired by different natural processes, their implementation to optimization problems shares a common feature: the search for the global optimum through a stochastic process. Genetic algorithms belong to
a class of methods called evolutionary algorithms, and they are inspired by the process of natural selection. These evolutionary algorithms have been applied to airfoil shape design by Quagliarella and Cioppa, [91] Yamamoto and Inoue, [120], and recently by de Sousa and Ramos, [38] and Liu [71]. On the other hand, simulated annealing algorithms are inspired by the behavior of a collection of atoms immersed in a heat bath subject to a cooling schedule. Aly et al., [3] and Wang et al. [115] have applied it for the design of an optimal aerodynamic shape. The main disadvantage of these stochastic algorithms is that they usually require a great number of evaluations of the objective function. Today, there are many derivatives of these methods created to be more efficient than the original.

Unfortunately, despite of the name, the afore mentioned methods are deterministic in nature. The conditions/parameters of the problem are all known or deterministic; only the search for the optimum may be stochastic. In reality, the conditions or parameters of the problem under investigation are not all known or are random. In such situations, optimization is said to be under “uncertainty”, and the notion of “optimum” is not well-defined yet. When randomness is involved, it is difficult to ensure that an optimum will turn out to be the right one.

To our knowledge, optimization under uncertainty is quite new in airfoil optimization. Several different approaches can be found in the literature that are concerned with an optimal design that is robust to small manufacturing errors or fluctuations in the operating conditions, such that the cruise Mach number. This research deals with robust design of airfoils that optimizes drag or drag and lift over a range of free stream Mach numbers.

Let us concentrate on a single-objective optimization formulation, where the drag coefficient is minimized subject to some constraints.

As in a deterministic context, a Single-Point Optimization method can be employed as a tool for optimization under uncertainty [52, 51, 53], but the drag reduction is attained only over a narrow range of Mach numbers. An improved drag coefficient can be realized over a wider range of Mach numbers if the Multi-Point Optimization technique is applied. With the multi-point formulation, practical problems arise with the selection of the flight conditions and specification of the weights, which are left up to the designer’s discretion [52, 51, 53]. As we mentioned before, another possible approach is based on Von Neumann-Morgenstern statistical decision theory, which states that the best design or decision is the one that optimizes the Bayes’ risk, an overall risk. This is commonly known as the
Maximum (Minimum) Expected Value criterion (MEV), and its application to airfoil design is presented by Huyse, Padula, Lewis, and Li [52, 51, 53]. One last approach, called the Profile Optimization Method, is developed to achieve consistent drag reduction over a given Mach range with only a few design points compared to the Multi-Point Optimization technique [67, 66]. One drawback of the Profile Optimization Method is the greater likelihood of getting trapped in a local solution than with the Multi-Point Method.

A comparison between different approaches is provided by Li, Huyse, and Padula [67]. According to their results, the Minimum Expected Value approach gives the most significant reduction of the drag coefficient.

Bearing this in mind, we propose to develop an airfoil optimization scheme to achieve a more substantial drag reduction over a given Mach number range. Therefore, the aforementioned EMV approach is applied and analyzed vis-a-vis to the classical MEV criterion.

The problem of minimizing the drag coefficient and maximizing the lift coefficient in the same time hasn’t been tackled in the literature yet. So far, the drag coefficient has been optimized subject to a fixed lift coefficient, or the lift coefficient has been optimized subject to a fixed drag coefficient. This approach will be extended into a cooperative game (or Nash equilibrium search) with two opposing objectives: to minimize the drag coefficient and to maximize the lift coefficient. The stochastic EMV and MEV Nash equilibrium will be employed when the randomness is present.
CHAPTER 2
SINGLE-OBJECTIVE (SO) EMV/MEV APPROACH

2.1 Problem Formulation

Let us formulate the stochastic optimization problem again. Let \((\Omega, \mathcal{F}, P)\) be a probability space and let \(\omega\) be an \(\mathcal{F}\)-measurable random variable. We denote by \(d\) the design variable, whose domain is confined to a Banach space \((X, \|\cdot\|)\). Let us consider a functional \(J : X \times \Omega \rightarrow \mathbb{R}\), which is assumed to have a minimum for any given value of the random variable \(\omega\). The minimization problem under uncertainty is to find an optimal point \(d^* \in D \subseteq X\), such that

\[
J (d^*, \omega) = \min_{d \in D} J (d, \omega), \quad \text{for all } \omega \in \Omega. \tag{2.1}
\]

We would like to compare the two optimum design points proposed in Chapter 1:

\[
d_{MEV} = \arg \min_{d \in D} E (J (d, \omega)) \tag{2.2}
\]

and

\[
d_{EMV} = E \left( \arg \min_{d \in D} J (d, \omega) \right). \tag{2.3}
\]

Before we compare the two strategies, let us remark first that the following inequality holds for any continuous objective function:

**Lemma 3.** Let \((\Omega, \mathcal{F}, P)\) be a probability space, \(\omega\) be an \(\mathcal{F}\)-measurable random variable, \((X, \|\cdot\|)\) be a Banach space, and \(D \subseteq X\). If the objective function \(J : X \times \Omega \rightarrow \mathbb{R}\) is continuous, has a minimum for any given value of the random variable \(\omega \in \Omega\), and \(d_{MEV}\) is well-defined, then

\[
E \left( \min_{d \in D} J (d, \omega) \right) \leq \min_{d \in D} (E (J (d, \omega))). \tag{2.4}
\]
Proof. According to the assumptions, the MEV optimum exists, and it is defined by

\[ d_{MEV} = \arg \min_{d \in D} E (J (d, \omega)) . \]

As \( J \) is continuous, it implies that

\[ \min_{d \in D} (E (J (d, \omega))) = E (J (d_{MEV}, \omega)). \tag{2.5} \]

On the other hand, because

\[ J (d_{MEV}, \omega) \geq \min_{d \in D} J (d, \omega), \]

and the expected value operator is monotonic, we can derive that

\[ E (J (d_{MEV}, \omega)) \geq E \left( \min_{d \in D} J (d, \omega) \right) \]. \tag{2.6} \]

Finally, from (2.5) and (2.6), it immediately follows that

\[ \min_{d \in D} \left( E (J (d, \omega)) \right) = E (J (d_{MEV}, \omega)) \geq E \left( \min_{d \in D} J (d, \omega) \right), \tag{2.7} \]

which proves the inequality (2.4).

The inequality (2.4) suggests that the use of the EMV-design point might ensure a lower objective function value compared to the classical MEV optimum. To this end, we focus on the conditions required to guarantee that

\[ P_\omega = P (J (d_{EMV}, \omega) \leq J (d_{MEV}, \omega)) \geq \frac{1}{2}. \]

\section{2.2 Theoretical Results}

In order to analyze the previous probability one would need to know the probability density function of the random variable, as well as the expression of the functional to be optimized. We shall consider here simple illustrative problems that arise in many different situations, like portfolio optimization, statistical prediction, error minimization, etc. A general formulation of the optimization problem ([43]) could be

\[ \min_{d \in D} u (a (\omega) d - b (\omega)), \]

where \( u : X \to \mathbb{R} \) is a convex loss function. The most popular convex functions are of the linear or quadratic type, therefore it seems reasonable to consider these cases here. For such objective functions, the following theorems hold.
Theorem 4. Under the assumptions of the previous lemma, if $D \subseteq \mathbb{R}$ is bounded, and $J : D \times \Omega \to \mathbb{R}$ is a continuous linear function of the form

$$J(d, \omega) = a(\omega)d + b(\omega),$$

where $a, b : \Omega \to \mathbb{R}$, $\omega$ is a continuous random variable with probability density function $f : \Omega \to \mathbb{R}$, and

$$E(a(\omega)) \text{ Median}(a(\omega)) \leq 0$$

then

$$P_\prec = P(J(d_{EMV}, \omega) \leq J(d_{MEV}, \omega)) \geq \frac{1}{2}. \quad (2.10)$$

If the inequality (2.9) is strict, then

$$P_\prec = P(J(d_{EMV}, \omega) \leq J(d_{MEV}, \omega)) > \frac{1}{2}.$$  

Proof. To simplify the notation, let us introduce the identity function $I : D \to D$, defined by $I(d) = d$.

First, one can easily see that the two optimum points are given by

$$d_{MEV} = \arg \min_{d \in D} E(a(\omega)d + b(\omega))$$

$$= \arg \min_{d \in D} (E(a(\omega))d + E(b(\omega))) = \begin{cases} \arg \min_{d \in D} I(d), & \text{if } E(a(\omega)) \geq 0 \\ \arg \max_{d \in D} I(d), & \text{if } E(a(\omega)) < 0 \end{cases}$$

and

$$d_{EMV} = E\left(\arg \min_{d \in D} (a(\omega)d + b(\omega))\right)$$

$$= \left(\arg \min_{d \in D} I(d)\right) \int_{a(\omega) \geq 0} f(\omega) \, d\omega + \left(\arg \max_{d \in D} I(d)\right) \int_{a(\omega) < 0} f(\omega) \, d\omega.$$

Using these expressions, we obtain

$$J(d_{EMV}, \omega) - J(d_{MEV}, \omega) = \begin{cases} a(\omega) \left(\arg \max_{d \in D} I(d) - \arg \min_{d \in D} I(d)\right) \int_{a(\omega) < 0} f(\omega) \, d\omega, & \text{if } E(a(\omega)) \geq 0 \\ \text{or} \\ a(\omega) \left(\arg \min_{d \in D} I(d) - \arg \max_{d \in D} I(d)\right) \int_{a(\omega) \geq 0} f(\omega) \, d\omega, & \text{if } E(a(\omega)) < 0 \end{cases}.$$
Finally, since $E(a(\omega)) \leq 0$, the desired probability (2.10) is guaranteed, i.e.

$$P_0 = P(J(d_{EMV}, \omega) - J(d_{MEV}, \omega) \leq 0) = \begin{cases} P(a(\omega) \leq 0) & \geq P(a(\omega) \leq \text{Median}(a(\omega))) = \frac{1}{2}, \\ & \text{if } E(a(\omega)) \geq 0 \\ or & \geq P(a(\omega) \geq \text{Median}(a(\omega))) = \frac{1}{2}, \\ P(a(\omega) \geq 0) & \geq P(a(\omega) \geq \text{Median}(a(\omega))) = \frac{1}{2}, \\ & \text{if } E(a(\omega)) < 0 \end{cases}$$

Following the above proof, we can now observe that a similar result holds for the objective function of the type

$$J(d, \omega) = a(\omega) g(d) + b(\omega),$$

where $g : D \to \mathbb{R}$ is a function with extreme values.

We should also remark here that if $a(\omega)$ is a Gaussian random variable, then the EMV approach cannot do better than the MEV approach.

Another result is presented next.

**Theorem 5.** Under the assumptions of the previous lemma, if

$J : D \times \Omega \subset \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a continuous quadratic function of the form

$$J(d, \omega) = a(\omega) d^2 + b(\omega) d + c(\omega),$$

where $a, b, c : \Omega \to \mathbb{R}$ are continuous functions, $a(\omega) > 0$, for any value of $\omega \in \Omega$, $\frac{b}{a}$ is either increasing or decreasing, $\frac{b}{a}^{-1} \left( \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right) \right)$ is well-defined, $\omega$ is a continuous random variable with probability density function $f : \Omega \to \mathbb{R}$, and

$$\begin{cases} \text{sgn}\left( E\left(\frac{b(\omega)}{a(\omega)}\right) - E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \left( \frac{b}{a} \right)^{-1} \left( \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right) \right) \\ \leq \text{sgn}\left( E\left(\frac{b(\omega)}{a(\omega)}\right) - E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \text{Median}(\omega), \text{ if } \frac{b}{a} \text{ is increasing} \end{cases}$$

or

$$\begin{cases} \text{sgn}\left( E\left(\frac{b(\omega)}{a(\omega)}\right) - E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \left( \frac{b}{a} \right)^{-1} \left( \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right) \right) \\ \geq \text{sgn}\left( E\left(\frac{b(\omega)}{a(\omega)}\right) - E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \text{Median}(\omega), \text{ if } \frac{b}{a} \text{ is decreasing} \end{cases}$$

(2.12)
then
\[ P_\prec = P \left( J(d_{EMV}, \omega) \leq J(d_{MEV}, \omega) \right) \geq \frac{1}{2}. \] (2.13)

If the inequality (2.12) is strict, then
\[ P_\prec = P \left( J(d_{EMV}, \omega) \leq J(d_{MEV}, \omega) \right) > \frac{1}{2}. \]

**Proof.** This clearly follows from the formula for the minimum point of a quadratic form and the distribution function method ([112]). First, the MEV optimum point is given by
\[ d_{MEV} = \arg \min_{d \in D} \left( E(a(\omega)) d^2 + E(b(\omega)) d + E(c(\omega)) \right) = -\frac{1}{2} E \left( \frac{b(\omega)}{a(\omega)} \right). \]

On the other hand, the EMV strategy can be expressed by
\[ d_{EMV} = E \left( \arg \min_{d \in D} \left( a(\omega) d^2 + b(\omega) d + c(\omega) \right) \right) = -\frac{1}{2} E \left( \frac{b(\omega)}{a(\omega)} \right). \]

Therefore,
\[ J(d_{EMV}, \omega) - J(d_{MEV}, \omega) = (d_{EMV} - d_{MEV}) (a(\omega) (d_{EMV} + d_{MEV}) + b(\omega)) \]
\[ = \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} - E \left( \frac{b(\omega)}{a(\omega)} \right) \right) \left( a(\omega) \left( -\frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E \left( \frac{b(\omega)}{a(\omega)} \right) \right) \right) + b(\omega) \right). \]

Bearing these in mind, there are four possible cases.

Case (i). If \( \frac{E(b(\omega))}{E(a(\omega))} - E \left( \frac{b(\omega)}{a(\omega)} \right) \geq 0 \), and the function \( \frac{b}{a} \) is increasing, then, according to the distribution function method and the assumption (2.12)
\[ P_\prec = P \left( J(d_{EMV}, \omega) - J(d_{MEV}, \omega) \leq 0 \right) \]
\[ = P \left( \frac{b(\omega)}{a(\omega)} \leq \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E \left( \frac{b(\omega)}{a(\omega)} \right) \right) \right) \]
\[ \left( \frac{b}{a} \right)^{-1} \left( \frac{E(b(\omega))}{E(a(\omega))} + E \left( \frac{b(\omega)}{a(\omega)} \right) \right) \]
\[ = \int_{-\infty}^{\text{Median}(\omega)} f(\omega) d\omega \geq \int_{-\infty}^{\text{Median}(\omega)} f(\omega) d\omega = \frac{1}{2}. \]
Case (ii). Similarly, if \( \frac{E(b(\omega))}{E(a(\omega))} - E\left(\frac{b(\omega)}{a(\omega)}\right) \geq 0 \), and the function \( \frac{b}{a} \) is decreasing, the desired probability

\[
P_\prec = P\left( J\left(d_{EMV}, \omega\right) - J\left(d_{MEV}, \omega\right) \leq 0 \right)
\]

\[
= P\left( \frac{b(\omega)}{a(\omega)} \leq \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \right)
\]

\[
= \int_{\left(\frac{b}{a}\right)^{-1}\left(\frac{1}{2}\left(\frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right)\right)\}}^{\infty} f(\omega) \, d\omega \geq \int_{\text{Median}(\omega)}^{\infty} f(\omega) \, d\omega = \frac{1}{2}.
\]

Case (iii). In the same light, if \( \frac{E(b(\omega))}{E(a(\omega))} - E\left(\frac{b(\omega)}{a(\omega)}\right) \leq 0 \), and the function \( \frac{b}{a} \) is increasing,

\[
P_\prec = P\left( J\left(d_{EMV}, \omega\right) - J\left(d_{MEV}, \omega\right) \leq 0 \right)
\]

\[
= P\left( \frac{b(\omega)}{a(\omega)} \geq \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \right)
\]

\[
= \int_{\left(\frac{b}{a}\right)^{-1}\left(\frac{1}{2}\left(\frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right)\right)\}}^{\infty} f(\omega) \, d\omega \geq \int_{\text{Median}(\omega)}^{\infty} f(\omega) \, d\omega = \frac{1}{2}.
\]

Case (iv). Finally, based on the same argument, if \( \frac{E(b(\omega))}{E(a(\omega))} - E\left(\frac{b(\omega)}{a(\omega)}\right) \leq 0 \), and the function \( \frac{b}{a} \) is decreasing, the same conclusion holds, i.e.

\[
P_\prec = P\left( J\left(d_{EMV}, \omega\right) - J\left(d_{MEV}, \omega\right) \leq 0 \right)
\]

\[
= P\left( \frac{b(\omega)}{a(\omega)} \geq \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right) \right) \right)
\]

\[
= \int_{\left(\frac{b}{a}\right)^{-1}\left(\frac{1}{2}\left(\frac{E(b(\omega))}{E(a(\omega))} + E\left(\frac{b(\omega)}{a(\omega)}\right)\right)\right)\}}^{-\infty} f(\omega) \, d\omega \geq \int_{-\infty}^{\text{Median}(\omega)} f(\omega) \, d\omega = \frac{1}{2}.
\]

This completes the proof. \( \Box \)
We conclude from the last two theorems that the EMV optimum point is more desirable compared to the MEV optimum point, provided there is asymmetry and/or non-linearity in the randomness. In real-world situations, it is reasonable to assume that the objective functions lack symmetry or linearity and the random variables are non-Gaussian. Therefore the EMV approach may be more appropriate and preferable in real world situations.

Having made this observation, let us focus on a simple Gaussian stochastic function and analyze the implications of the EMV and MEV strategy.

First, let us define the covariance function $\text{cov} [J] : \bar{D} \times \bar{D} \to \mathbb{R}$ associated with a stochastic function $J : D \times \Omega \to \mathbb{R}$ by

$$
\text{cov} [J] (d_1, d_2) = \text{cov} (J (d_1, \cdot), J (d_2, \cdot))
$$

or

$$
\text{cov} [J] (d_1, d_2) = E ((J (d_1, \omega) - E J (d_1, \omega)) (J (d_2, \omega) - E J (d_2, \omega)))
$$

We shall prove the following result.

**Theorem 6.** If the stochastic objective function $J (d, \cdot)$ is a normal random variable, for all $d \in D$, and if the covariance function $\text{cov} [J]$ is continuous on the closed interval $\bar{D} \times \bar{D}$, then there is no other optimal strategy better than $d_{MEV}$, i.e.

$$
P(J (d, \omega) - J (d_{MEV}, \omega) \leq 0) \leq \frac{1}{2}, \text{ all } d \in D.
$$

**Proof.** We will make use of the Karhunen-Loève expansion, which is a suitable tool for the approximation of stochastic functions. For the stochastic function $J$, we define the corresponding operator $T_J = L^2 (D) \to L^2 (D)$ by ([5])

$$
T_J v (\cdot) = \int_D \text{cov} [J] (x, \cdot) v (x) dx, \forall v \in L^2 (D).
$$

This operator $T_J$ has a decreasing sequence of non-negative eigenvalues $(\lambda_i)_{i=1}^{\infty}$, i.e. $\lambda_i \geq 0$, $\lambda_i \geq \lambda_{i+1}$, for $i = 1, 2, 3, \ldots$. Let $(b_i)_{i=1}^{\infty}$ be the sequence of corresponding orthonormal eigenfunctions, i.e. $T_J b_i = \lambda_i b_i$, and $(b_i, b_j)_{L^2 (D)} = \delta_{ij}$, for $i, j = 1, 2, 3, \ldots$

Therefore, the truncated Karhunen-Loève expansion $J_N$ of the stochastic function $J$ is defined to be

$$
J_N (d, \omega) = E J (d, \omega) + \sum_{i=1}^{N} \sqrt{\lambda_i} b_i (d) Y_i (\omega), \forall N \in \mathbb{N}_+,
$$
where the real random variables \((Y_i)_{i=1}^\infty\) are uncorrelated, have mean zero and unit variance, i.e. \(EY_i = 0, E(Y_iY_j) = \delta_{ij}\), for \(i, j = 1, 2, 3, \ldots\). These random variables are uniquely determined when \(\lambda_i > 0\) by

\[
Y_i(\omega) = \frac{1}{\sqrt{\lambda_i}} \int_{D} (J(x, \omega) - EJ(x, \omega)) b_i(x) \, dx, \quad \forall i \in \mathbb{N}_+.
\]

Then, by Mercer’s Theorem ([95]), it follows that

\[
\lim_{N \to \infty} \left( \sup_{D} E \left( (J - J_N)^2 \right) \right) = 0.
\]

In addition, according to [73], if the covariance \(cov[J]\) is continuous on a closed interval \(\bar{D} \times \bar{D}\), then the random function \(J(d, \cdot)\) is normal if and only if the random variables \(Y_i\) are normal. If, moreover, \(J(d, \cdot)\) is real-valued, then \(Y_i\) are independent and the proper decomposition of \(J(d, \cdot)\) converges almost sure.

Having these in mind, we would like to analyze the difference

\[
J(d, \omega) - J(d_{MEV}, \omega)
\]

which, based on the Karhunen-Loève expansion equals,

\[
EJ(d, \omega) - EJ(d_{MEV}, \omega) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} (b_i(d) - b_i(d_{MEV})) Y_i(\omega)
= \alpha + \sum_{i=1}^{\infty} \sqrt{\lambda_i} (b_i(d) - b_i(d_{MEV})) Y_i(\omega),
\]

where \(\alpha = EJ(d, \omega) - EJ(d_{MEV}, \omega) \geq 0\), according to the definition of the MEV strategy.

We will employ the moment-generating function method for finding the probability distribution of the function \(U = J(d, \omega) - J(d_{MEV}, \omega)\). The method finds the moment-generating function for \(U\), and compares it with other well-known generating-functions. If two moment-generating functions exist and are equal to each other, then the corresponding random variables have the same probability distribution. Recall that the moment-generating function is defined by

\[
m_U(t) = E \left( e^{tU} \right),
\]

and it satisfies the following properties

\[
m_{aZ+b}(t) = e^{bt} m_Z(at), \quad \forall a, b \in \mathbb{R},
\]

26
\[ m_{Z_1 + Z_2 + \ldots + Z_n}(t) = (m_{Z_1}(t)) (m_{Z_2}(t)) \ldots (m_{Z_n}(t)), \]
\[ \forall Z_1, Z_2, \ldots, Z_n \text{ independent random variables} \]
\[ m_Z(t) = e^{\mu t + \frac{1}{2} t^2}, \text{ if } Z \text{ has a normal distribution } N(\mu, \sigma^2). \]

Because the random variables \( Y_i \) are normal and independent, the corresponding moment-generating function is given by
\[ m_J(d, \omega) = E \left( e^{t (J(d, \omega) - J(d_{MEV}, \omega))} \right) \]
\[ = e^{\alpha t} \prod_{i=1}^{\infty} m_{Y_i} \left( \sqrt{\lambda_i} (b_i(d) - b_i(d_{MEV})) t \right) \]
\[ = e^{\alpha t} \prod_{i=1}^{\infty} e^{\lambda_i (b_i(d) - b_i(d_{MEV}))^2 t^2 / 2} \]
\[ = e^{\alpha t + \sum_{i=1}^{\infty} \lambda_i (b_i(d) - b_i(d_{MEV}))^2 t^2 / 2}. \]

In other words, the function \( J(d, \omega) - J(d_{MEV}, \omega) \) is a normal random variable with mean \( \alpha \geq 0 \), and variance \( \sum_{i=1}^{\infty} \lambda_i (b_i(d) - b_i(d_{MEV}))^2 \).

Therefore,
\[ P(J(d, \omega) - J(d_{MEV}, \omega) \leq 0) \leq P(J(d, \omega) - J(d_{MEV}, \omega) \leq \alpha) = \frac{1}{2}, \]
which completes the proof.

This result shows that the EMV criterion might not work better if the randomness involved is normal or symmetric.

### 2.3 Examples

#### 2.3.1 Free Undamped Vibration

Consider the problem of an undamped free vibration ([16]). Suppose that a unit mass that hangs at the end of a vertical spring (with spring constant \( \delta \in (0, 4\pi^2] \) for instance) is initially displaced by a distance \( \omega \) according to Fisher’s \( F_{25,25} \) distribution and then released. We are interested to find the spring constant that provides the minimal elongation after a second under the assumed uncertainty of the initial elongation. The corresponding mathematical formulation is as follows:

\[ \min_{\delta \in (0, 4\pi^2]} J(\delta, \omega) = \min_{\delta \in (0, 4\pi^2]} X(1, \delta, \omega) \quad (2.14) \]
where \( X(t, \delta, \omega) \) is the solution of the initial value problem

\[
\begin{align*}
\frac{d^2X}{dt^2} + \delta X &= 0, \quad t > 0, \\
X(0) &= \omega, \\
X'(0) &= 0,
\end{align*}
\]  

(2.15)

and \( \omega \sim F_{25,25}(\cdot + 1.05) \).  

(2.16)

Obviously, \( X(t, \delta, \omega) = \omega \cos \left( \sqrt{\delta} t \right) \), and we are looking for a stochastic minimum value of the functional \( J(\delta, \omega) = \omega \cos \left( \sqrt{\delta} t \right) \), where \( \omega \) is a \( F \) distribution random variable of the form (2.16). The choice of this kind of randomness is justified by the fact that among the higher transcendental functions, a frequently used function within the area of statistical inference is the inverted beta distribution, also called the \( F \) distribution. The shift applied to it brings even more asymmetry to the problem under investigation.

By ([44]) and (2.16), the expected value and the median of the random variable \( \omega \) are given by

\[
\begin{align*}
E(\omega) &= 0.036957 > 0 \\
\text{Median}(\omega) &= -0.05 < 0
\end{align*}
\]

According to the theorem 4 and (2.11), the EMV should give us a more accurate solution. We shall prove that by computing both the MEV and the EMV optimum points. Thus,

\[
\delta_{MEV} = \arg \min_{\delta \in (0,4\pi^2)} (E(J(\delta, \omega))) = \arg \min_{\delta \in (0,4\pi^2)} \left( (0.036957) \cos \sqrt{\delta} \right) = \pi^2.
\]

On the other hand,

\[
\delta_{EMV} = E \left( \arg \min_{\delta \in (0,4\pi^2)} J(\delta, \omega) \right) = 4\pi^2 \int_{-1.05}^{0} f(\omega) \, d\omega + \pi^2 \int_{0}^{\infty} f(\omega) \, d\omega
\]

\[
= 4\pi^2 \int_{-1.05}^{0} \frac{\Gamma(25)}{\Gamma^2 \left( \frac{25}{2} \right)} \frac{(\omega + 1.05)^{23/2}}{(\omega + 2.05)^{25}} \, d\omega + \pi^2 \int_{0}^{\infty} \frac{\Gamma(25)}{\Gamma^2 \left( \frac{25}{2} \right)} \frac{(\omega + 1.05)^{23/2}}{(\omega + 2.05)^{25}} \, d\omega
\]

\[
= 4\pi^2 (0.640286) + \pi^2 (0.359713) = (2.920857) \pi^2.
\]

The absolute minimum value of the objective function, as well as the value of the objective function at these two “optimum points” are presented in Figure 2.1.
The probability of our interest $P_<$ is given by

$$P_< = P(J(\delta_{EMV}, \omega) \leq J(\delta_{MEV}, \omega)) = P(\omega \leq 0)$$

$$= \int_{-1.05}^{0} \frac{\Gamma(25) (\omega + 1.05)^{23/2}}{\Gamma^{2}(\frac{25}{2}) (\omega + 2.05)^{25}} d\omega = 64.03\%.$$ 

In conclusion, since the desired probability $P_<$ is well above 50%, EMV is definitely a better alternative to MEV. The next two examples lead to similar conclusion.
Figure 2.1: Free Undamped Vibration
2.3.2 Population Growth

Consider a population growth problem that requires the determination of the initial population $\delta > 0$, which will ensure that the population over the first time unit is as close as possible to a prescribed value (say 1, for simplicity), if the rate of growth $\omega$ depends on the environmental conditions. Let us assume that the rate of growth is a uniform $(0, 1)$ random variable.

Let $X(t)$ be the population of the given species at time $t$. The simplest model associated with the variation of $X(t)$ is the exponential growth ([16])

$$\left\{ \begin{array}{l} \frac{dX}{dt} = \omega X, \quad t > 0 \\ X(0) = \delta. \end{array} \right. \tag{2.17}$$

The functional $J$ to be minimized may be written as a norm of the difference between the computed and the desired solution, namely

$$J(\delta, \omega) = \frac{1}{2} \int_0^1 (X(t, \delta, \omega) - 1)^2 dt. \tag{2.18}$$

Taking into account the fact that the solution of (2.17) is given by $X(t, \delta, \omega) = \delta e^{\omega t}$, the functional $J$ can be rewritten as

$$J(\delta, \omega) = \frac{1}{2} \int_0^1 (\delta e^{\omega t} - 1)^2 dt = \frac{1}{2} \int_0^1 \left( \delta^2 e^{2\omega t} - 2\delta e^{\omega t} + 1 \right) dt$$

$$= \frac{1}{2} \left( \delta^2 \frac{e^{2\omega t}}{2\omega} - 2\delta \frac{e^{\omega t}}{\omega} + t \right) \bigg|_0^1 = \delta^2 \frac{e^{2\omega}}{4\omega} - \delta \frac{e^{\omega}}{\omega} + \frac{1}{2}$$

$$= a(\omega) \delta^2 + b(\omega) \delta + c(\omega).$$

First, let us remark that the functions $a, b, c : [0, 1] \to \mathbb{R}, a(\omega) = \frac{e^{2\omega} - 1}{4\omega}, b(\omega) = -\frac{e^{\omega} - 1}{\omega}, c(\omega) = \frac{1}{2}$ are continuous, $a(\omega) > 0$, and $\frac{b(\omega)}{a(\omega)} = -\frac{4}{e^\omega + 1}$ is increasing over $[0, 1]$.

Besides,

$$\left\{ \begin{array}{l} E(a(\omega)) = \int_0^1 \frac{e^{2\omega} - 1}{4\omega} d\omega = 0.920967, \\
E(b(\omega)) = \int_0^1 -\frac{e^{\omega} - 1}{\omega} d\omega = -1.317902, \\
E\left(\frac{b(\omega)}{a(\omega)}\right) = \int_0^1 -\frac{4}{e^\omega + 1} d\omega = -1.519542 \end{array} \right.$$
Therefore
\[
\begin{cases}
\frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left( \frac{b(\omega)}{a(\omega)} \right) \right) = -1.475269, \\
\text{sgn} \left( E\left( \frac{b(\omega)}{a(\omega)} \right) - \frac{E(b(\omega))}{E(a(\omega))} \right) = \text{sgn}(-0.088545) = -1.
\end{cases}
\]

On the other hand,
\[
\left( \frac{b}{a} \right)^{-1} \left( \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left( \frac{b(\omega)}{a(\omega)} \right) \right) \right) = \ln \left( -1 - \frac{4}{2 \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left( \frac{b(\omega)}{a(\omega)} \right) \right)} \right)
\]
\[
= \ln \left( -1 + \frac{4}{1.475269} \right) = 0.537294,
\]
which implies
\[
\text{sgn} \left( E\left( \frac{b(\omega)}{a(\omega)} \right) - \frac{E(b(\omega))}{E(a(\omega))} \right) \left( \frac{b}{a} \right)^{-1} \left( \frac{1}{2} \left( \frac{E(b(\omega))}{E(a(\omega))} + E\left( \frac{b(\omega)}{a(\omega)} \right) \right) \right) \leq \text{sgn} \left( E\left( \frac{b(\omega)}{a(\omega)} \right) - \frac{E(b(\omega))}{E(a(\omega))} \right) \text{Median}(\omega)
\]

Thus, by virtue of the Theorem 5, the EMV will give us a better answer to our optimization problem. Let us prove that by computing the corresponding \( \delta_{MEV} \) and \( \delta_{EMV} \). First, we shall analyze the MEV optimum, i.e.,
\[
\delta_{MEV} = \arg\min_{\delta > 0} (E(J(\delta, \omega))) = \frac{\int_{0}^{1} \frac{e^{\omega-1}}{\omega} d\omega}{2 \int_{0}^{1} \frac{e^{\omega-1}}{4\omega} d\omega} = 0.715498,
\]
and then the EMV optimum
\[
\delta_{EMV} = E \left( \arg\min_{\delta > 0} J(\delta, \omega) \right) = E \left( \frac{2}{e^\omega + 1} \right) = 2 \int_{0}^{1} \frac{d\omega}{e^\omega + 1} = 0.759771.
\]

For a graphic comparison of the two methods, we plot the minimum of the objective function
\[
\min_{\delta > 0} J(\delta, \omega) = J(\delta(\omega), \omega)
\]
given by the design variable which provides the lowest possible value of \( J \), i.e.,
\[
\delta(\omega) = \frac{1}{2} - \frac{1}{\omega} \frac{e^\omega - 1}{e^\omega + 1},
\]
along with the objective function at the EMV optimum \( \delta_{EMV} \), and at the MEV optimum \( \delta_{MEV} \), respectively (Figure 2.2).
In conclusion, we observe that the desired probability $P_\prec$

$$
(\frac{1}{2})^{-1}\left(\frac{1}{2}E\left(\frac{b(\omega)}{E(a(\omega))}\right) + E\left(\frac{b(\omega)}{a(\omega)}\right)\right)
$$

$$
P_\prec = P(J(\delta_{EMV}, \omega) \leq J(\delta_{MEV}, \omega)) = \int_{0}^{0.537294} f(\omega) \, d\omega
$$

$$
= \int_{0}^{0.537294} d\omega = 53.73\%
$$

which justifies once again our advocacy of EMV criterion.
Figure 2.2: Population Growth
2.3.3 The Burgers Equation

We are concerned with the Burgers equation

\[
\begin{align*}
  y_t - \nu y_{xx} + yy_x &= 0 \\
  y(x, 0) &= \sin \pi x \\
  y(x + 1, t) &= y(x, t)
\end{align*}
\]  \tag{2.19}

with periodic boundary conditions, and a given initial value function where the viscosity \( \nu \) is a random variable. We are interested to find the instant of time when the solution of (2.19) at a specified location, \( x = 0.5 \), for instance, takes a certain value, say \( y = 0.3 \). The problem is quite challenging because of the uncertainty in viscosity and the nonlinearity of the problem.

In other words, we would like to find the minimum of the objective function

\[
\min_{t > 0} J(t, \nu) = \min_{t > 0} |y(0.5, t, \nu) - 0.3|.
\]  \tag{2.20}

In order to evaluate the objective function, we employ the analytical solution \( y(x, t) \) of the Burgers equation available through use of the Hopf-Cole transformation ([30]):

\[
y(x, t) = \frac{2\pi \nu}{\sum_{n=0}^{\infty} \exp \{-n^2\pi^2\nu t\} A_n \sin (n\pi x)} \sum_{n=0}^{\infty} \exp \{-n^2\pi^2\nu t\} A_n \cos (n\pi x),
\]

where

\[
\begin{align*}
  A_0 &= \int_0^1 \exp \{- (2\pi \nu)^{-1} [1 - \cos (\pi x)]\} \, dx \\
  A_n &= 2 \int_0^1 \exp \{- (2\pi \nu)^{-1} [1 - \cos (\pi x)]\} \cos (n\pi x) \, dx, \quad n > 0.
\end{align*}
\]

To find the solution of the optimization problem (2.20), we use the MEV and the EMV approaches. For simplicity, we restrict the design space \( t \in T = (0.0, 2.0) \) (examining the results at different times ([84]). As for the randomness of \( \nu \), we consider uniform and normal distributions.

If the viscosity \( \nu \) is a uniform random variable in the interval \([0.2, 0.7]\) (Figure 2.3), the MEV and the EMV approach provide the following “minima”:

\[
\begin{align*}
  t_{MEV} &= 0.408000 \\
  t_{EMV} &= 0.412034.
\end{align*}
\]
It follows that
\[ P_\prec = P (J(t_{EMV}, \nu) \leq J(t_{MEV}, \nu)) = 51.90\%. \]

Finally, if the viscosity \( \nu \) is a normal random variable \( N(1., (0.2)^2) \) (Figure 2.4), the “optimum” time values are given by
\[
\begin{aligned}
    t_{MEV} &= 0.276660 \\
    t_{EMV} &= 0.268387
\end{aligned}
\]
and the probability \( P_\prec \) is
\[ P_\prec = P (J(t_{EMV}, \nu) \leq J(t_{MEV}, \nu)) = 56.03\%. \]

Once again, irrespective of the choice of randomness, we arrive at the same conclusion as we established earlier, namely that the EMV criterion may furnish a “better” optimum, especially if we are dealing with non-linearities and non-symmetries.
Figure 2.3: Burgers Equation – uniform $\nu$
Figure 2.4: Burgers Equation – normal $\nu$
2.4 Application: Airfoil Shape Optimization

2.4.1 EMV/MEV Approach

We now present a comparison between the EMV and MEV strategy applied to a transonic airfoil optimization problem. The main objective here is to find the optimal shape $\theta$ of a two-dimensional airfoil that minimizes the drag coefficient $C_D$ subject to a prescribed lift coefficient, and some other geometrical constraints, namely $h_i$, under random cruise Mach numbers $M$, i.e.

$$\left\{ \begin{array}{l}
\min_{\theta \in \Theta} C_D (\theta, M) , \ M \in \Omega \\
h_i (\theta, M) \geq 0, i = 1, 2, ..., n
\end{array} \right. \quad (2.21)$$

The desired shape, $\theta$, is an element of a design space, $\Theta$. The random Mach number, $M$, is defined on a given probability space $\Omega$ and it has the probability density function $f(M)$.

The deterministic drag coefficient minimization problem can be solved by making use of the compressible Euler equations and the adjoint method. The Appendix A provides a brief description of this procedure. Our goal is, once again, to develop a methodology that would incorporate the uncertainty in the cruise Mach number.

As we mentioned earlier, a number of convenient reformulations of problem (2.21) are possible. According to the Von Neumann-Morgenstern statistical decision theory [89], the best course of action in the presence of uncertainty is to select the design that leads to the lowest expected drag, also called the MEV design point.

There is no reason why one should not consider the expectation of the minimum, the EMV optimum. Croicu and Hussaini (2.2 and [34]) have proved that in some illustrative cases, the EMV criterion provides a higher probability of lower risk compared to the MEV approach. It is interesting to analyze how the EMV compares to the MEV optimum solution in the aerodynamic shape design, especially because we are concerned with obtaining the most favorable geometry of an airfoil over a range of Mach numbers. The impact of Mach number uncertainty may prove significant in a quantitative design.

Let us describe the two optimum points next, starting with the one proposed in the present research work.
The EMV optimum shape is therefore defined to be

$$\theta_{EMV} = E \left( \arg \min_{\theta \in \Theta} C_D (\theta, M) \right) = \int_{\Omega} \arg \min_{\theta \in \Theta} C_D (\theta, M) f(M) \, dM \quad (2.22)$$

In order to approximate the right hand side of (2.22), we let $M_1, M_2, ..., M_n$ be a random sample of size $n$ of the random variable $M$ with probability density function $f$. For each sample $M_k, k = 1, 2, ..., n$, the optimization problem (2.21) yields a solution $\theta (M_k), k = 1, 2, ..., n$. Therefore, the averaged optimal shape is considered to be

$$\theta^n_{EMV} = \frac{1}{n} \sum_{k=1}^{n} \theta (M_k) = \frac{1}{n} \sum_{k=1}^{n} \arg \min_{\theta \in \Theta} C_D (\theta, M_k)$$

If the right hand side of (2.22) is finite, then by the Strong Law of large Numbers [15], the estimator $\theta^n_{EMV}$ converges to the desired optimum $\theta_{EMV}$, i.e.,

$$\theta^n_{EMV} \to \theta_{EMV} \text{ as } n \to \infty.$$ 

The overall design procedure (based on the methodology described in Appendix A) can be summarized as follows:

1. Initialize the deterministic parameters, and parameterize the configuration of interest using a set of design variables. Define an initial shape;
2. Sample a value of the random variable $M$, according to its probability distribution function $f(M)$, and initialize the shape;
3. Solve the flow equations for the flow variables;
4. Solve the adjoint equations for the co-state variables subject to appropriate boundary conditions;
5. Evaluate the gradients and update the aerodynamic shape based on the direction of steepest descent (for instance);
6. Return to step #3 until an optimum configuration is attained;
7. Return to step #2 until a desired number of sample points are analyzed;
8. Average all the optimum configurations obtained at step #6, which will give us the EMV optimum design.

MEV

The MEV optimum shape defined by Huyse, Padula, Lewis, and Li [52, 51, 53] to be

$$\theta_{MEV} = \arg \min_{\theta \in \Theta} E(C_D(\theta, M)) = \arg \min_{\theta \in \Theta} \int_{\Omega} C_D(\theta, M) f(M) dM$$

is analyzed here too. Because the function $C_D(\theta, \cdot)$ is continuous, $f(\cdot)$ is of constant sign and integrable ($\int_{\Omega} f(M) dM = 1$), according to the First Mean Value Theorem [4], we can derive that there exists a value $M^{MEV}(\theta)$ such that

$$\int_{\Omega} C_D(\theta, M) f(M) dM = C_D(\theta, M^{MEV}(\theta)).$$

(2.24)

The equation (2.24) is very useful if we would like to make use of a numerical deterministic code that minimizes the drag coefficient for known parameters. In this light, the Monte Carlo type estimator is defined by

$$\theta^*_{MEV} = \arg \min_{\theta \in \Theta} \left[ \frac{1}{n} \sum_{k=1}^{n} C_D(\theta, M_k) \right] = \arg \min_{\theta \in \Theta} C_D(\theta, M^{MEV}(\theta)),$$

and it converges to the optimum $\theta_{MEV}$, as $n \to \infty$.

Therefore, the procedure for implementing the MEV strategy could be described as follows:

1. Initialize the deterministic parameters, and parameterize the configuration of interest using a set of design variables, and define the initial shape;

2. Sample all the values of the random variable $M$, according to its probability distribution function $f(M)$;

3. For every value of the random variable $M$, solve the flow equations for the flow variables and evaluate the objective function;

4. Average all the objective function values obtained at the previous step, and identify its corresponding $M^{MEV}$;

5. For $M = M^{MEV}$ and the current airfoil shape, solve the flow equations for the flow variables;
6. For $M = M^{MEV}$ and the current airfoil shape, solve the adjoint equations for the co-state variables subject to appropriate boundary conditions;

7. Evaluate the gradients and update the aerodynamic shape based on the direction of steepest descent (for instance);

8. Return to step #3 until an MEV optimum configuration is attained.

2.4.2 Results

The design of a transonic airfoil that performs well over a range of different Mach numbers is investigated ([35]) using the EMV and MEV approaches and Jameson’s code [55]. Jameson’s code is an implementation of a gradient-based optimization technique in which the control variable, namely the airfoil shape, is parametrized using a set of 65 design variables. The cost function to be minimized is the drag coefficient. The gradient information is obtained via the adjoint equation. This adjoint equation is used to calculate the sensitivity derivatives of the cost function with respect to the design variables in order to get a direction of improvement. The flow is calculated using the steady state inviscid Euler equations. The initial shape is the well-known NACA-0012 profile with the angle of attack $\alpha = 0.5^\circ$. Because this shape is not suitable for the transonic regime, substantial improvements are to be expected. The number of iterations for the optimization algorithm (which is also the stopping criterion) is 20.

We should mention that the specification of the Mach number is sufficient to carry out the optimal shape. Therefore the implementation of the EMV method is quite straightforward. On the other hand, one call of Jameson’s code for a given Mach number, and for the number of iterations set to zero allows us to compute the cost function, i.e. the drag coefficient for this Mach number. Having this, the implementation of the MEV approach follows easily, of course, with some modifications in Jameson’s source code.

We assume two distributions of the Mach number: uniform distribution $[0.7, 0.8]$, and Gaussian distribution $M \sim \mathcal{N}(\bar{M}, \sigma^2)$ with the mean $\bar{M} = 0.75$ and the standard deviation $\sigma = 0.025$.

Figures 2.5, 2.8, 2.14, 2.17 show the drag coefficient before and after the optimization procedure. For both optimization approaches and for each distribution, we choose 1000 samples of the Mach number. (We should mention here that the Monte-Carlo procedure
might be accelerated by a variance reduction based on the sensitivity derivatives, see [19, 20] for further details.) Then an optimal shape is found using the EMV and the MEV methods. In order to better compare the results, we calculated the corresponding drag coefficient at 100 equally spaced points in [0.7, 0.8]. A drag reduction is experienced over the entire set of Mach numbers, no matter what the nature of the randomness or the approach is. Most of the drag reduction comes from the reduction of the shock wave in the upper surface, see the Figures 2.6, 2.9, 2.15, 2.18.

In addition, a comparison between the EMV and MEV approaches can be carried out. According to Figures 2.11, 2.20, a lower risk is observed over a larger set of Mach numbers in the EMV case. However, the probability $P_{<}$ of finding a better solution by employing the EMV strategy is 78% for the uniform random variable, and 46.36% for the Gaussian randomness. Because of the small standard deviation of the normal random variable, it’s not surprisingly to find out that EMV and MEV are very similar. These results are in correlation with the pressure coefficient distributions found in the Figures 2.12, 2.23, where a drag coefficient reduction matches a shock wave reduction.

The EMV & MEV optimal shapes are presented in the Figures 2.7, 2.10, 2.13, 2.16, 2.19, 2.24).

On the other hand, the CPU time spent to find the EMV design point is about 16,777.70 seconds for the $U[0.7, 0.8]$, and 16,881.45 seconds for the $\mathcal{N}(0.75, 0.025^2)$. For the MEV strategy, it is about 486.21 seconds for the uniform random variable, and 492.45 seconds for the normal random variable (see Table 2.1). All computations in this thesis were performed in double precision (64-bit arithmetic) on a Dell PC (P4, 1.8 GHZ, 512 MB). In other words, there is a price paid for a lower cost function: a higher CPU time.
Figure 2.5: (SO) Drag Coefficient at EMV–$M \sim U(0.7, 0.8)$ & initial shape
Figure 2.6: (SO) Pressure (for EMV–$M \sim U(0.7, 0.8)$ & initial shape) at $M = 0.7907923$
Figure 2.7: (SO) Geometry (EMV–M $\sim U(0.7, 0.8)$ & initial)
Figure 2.8: (SO) Drag Coefficient at MEV–$M \sim U(0.7, 0.8)$ & initial shape
Figure 2.9: (SO) Pressure (for MEV–$M \sim U(0.7, 0.8)$ & initial shape) at $M = 0.7907923$
Figure 2.10: (SO) Geometry (MEV–M ~ U(0.7, 0.8) & initial)
Figure 2.11: (SO) Drag Coefficient at EMV–& MEV–M $\sim U(0.7, 0.8)$ shape
Figure 2.12: (SO) Pressure (for EMV- & MEV-M $\sim U(0.7, 0.8)$ shape) at $M = 0.7907923$
Figure 2.13: (SO) Geometry (EMV- & MEV- $M \sim U(0.7, 0.8)$)
Figure 2.14: (SO) Drag Coefficient at EMV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape
Figure 2.15: (SO) Pressure (for EMV–M \sim \mathcal{N}(0.75, 0.025^2) & initial shape) at M = 0.7540006
Figure 2.16: (SO) Geometry (EMV–M $\sim N(0.75, 0.025^2)$ & initial)
Figure 2.17: (SO) Drag Coefficient at MEV–M \( \sim \mathcal{N}(0.75, 0.025^2) \) & initial shape
Figure 2.18: (SO) Pressure (for MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial shape) at $M = 0.7540006$
Figure 2.19: (SO) Geometry (MEV-M \sim \mathcal{N}(0.75, 0.025^2) \& initial)
Figure 2.20: (SO) Drag Coefficient at EMV–& MEV–M $\sim N(0.75, 0.025^2)$ shape
Figure 2.21: (SO) Drag Coefficient at EMV–& MEV–M ~ \( N(0.75, 0.025^2) \) shape (zoom)
Figure 2.22: (SO) Drag Coefficient at EMV–& MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ shape (zoom)
Figure 2.23: (SO) Pressure (for EMV− & MEV− $M \sim \mathcal{N}(0.75, 0.025^2)$ shape) at $M = 0.7540006$
Figure 2.24: (SO) Geometry \((EMV \& MEV \sim N(0.75, 0.025^2))\)
<table>
<thead>
<tr>
<th></th>
<th>CPU time (seconds)</th>
<th>$P_{&lt;}$</th>
</tr>
</thead>
</table>
| $M \sim U (0.7, 0.8)$ | (EMV) 16777.70  
               (MEV) 486.21 | $(C_D)$ 78% |
| $M \sim N (0.75, 0.025^2)$ | (EMV) 16881.45  
                  (MEV) 492.45 | $(C_D)$ 46.36% |

Table 2.1: Comparison of EMV and MEV approach for (SO) airfoil optimization
CHAPTER 3

MULTI-OBJECTIVE (MO) EMV/MEV APPROACH

3.1 Formulation

Our main purpose here is to develop a strategy for stochastic multi-objective optimization with direct impact on the airfoil shape optimization, where a design that minimizes the drag coefficient and maximizes the lift coefficient is sought.

It is widely recognized that the numerical solution of optimal problems governed by the unsteady compressible Euler or Navier-Stokes equations is a challenging problem that requires careful mathematical formulation, accurate state solution, efficient gradient computation, and convergent optimization algorithms ([23]). As a simplified model of the Navier-Stokes equation, the one-dimensional Burgers equation represents many of the properties of Navier-Stokes equations. Given this, the viscous Burgers equation has received significant attention ([21], [63]), and recent research has focused on the control of Burgers flow as a model for control of Navier-Stokes flows ([39], [27], [48], [11], [94], [23]). On the other hand, uncertainties associated with Burgers equation were studied recently ([114], [54]), as well as with CFD simulations ([78], [118]).

The main goal is to develop a framework for the viscous Burgers equation and apply it to airfoil shape optimization. The EMV and MEV methods are employed to deal with the randomness, and the Nash equilibrium point is chosen as a possible optimum for the multi-objective program.

Therefore, the next section is concerned with a multi-objective control under uncertainty problem associated with the Burgers equation, and the following one with an application to aerodynamics.
3.2 Stochastic Nash Equilibrium for the Periodic Burgers Equation

3.2.1 Problem Formulation

In [94], Ramos, Glowinski, Periaux considered a Nash multi-objective control problem for the Burgers equation with Dirichlet-Neumann boundary conditions. Our goal is to extend this problem when uncertainty in viscosity is taken into consideration, but for the Burgers equation with periodic boundary conditions.

We are concerned with instantaneous control, where the control target is to match the given desired state in the $L^2$-sense by adjusting two body forces $v_1(x,t), v_2(x,t)$ with minimal energy and work. The state equation is

$$
\begin{cases}
  y_t - \nu y_{xx} + yy_x = f + v_1 x_1 + v_2 x_2, \forall (x,t) \in (0, 2\pi) \times (0, T) \\
  y(0,t) = y(2\pi, t), \forall t \in (0, T) \\
  y(x,0) = y_0(x), \forall x \in (0, 2\pi)
\end{cases}
$$

(3.1)

Let us consider $\omega_i, \omega_{d_i}, \omega_{T_i} \subseteq (0, 2\pi)$, $i=1,2$, and the target functions $y_{d_i} \in L^2(\omega_{d_i} \times (0, T))$ and $y_{T_i} \in L^2(\omega_{T_i})$, $i = 1, 2$. Because of the periodicity, if the control volumes $\omega_{d_i}, \omega_{T_i}$ comprise the endpoints, we assume to have the following relations satisfied:

$$
\begin{cases}
  y_{d_1}(0,t) = y_{d_2}(2\pi,t), \forall t \in (0, T) \\
  y_{T_1}(0) = y_{T_2}(2\pi)
\end{cases}
$$

We take as the control space $U_1 = U_2 = L^2((0, 2\pi) \times (0, T)) = U$. The goal of each control $v_i, i = 1, 2$ is to drive the solution $y(x,t)$ close to $y_{d_i}(x,t)$ in $\omega_{d_i} \times (0, T), i = 1, 2$ and $y(x,T)$ close to $y_{T_i}(x)$ in $\omega_{T_i}, i = 1, 2$, at a minimal cost for the control $v_i, i = 1, 2$ ([32]).

First of all, we will develop a procedure to deal with the control problem without any uncertainty, and will employ the EMV/MEV approach to take care of the randomness in the viscosity parameter only at the end. To better understand the implications, we will assume viscosity to be a uniform or a Gaussian random variable. Because variations in the numerical value of the viscosity may have a negative impact on the solution of the Burgers equation, we will select only small variations in the uncertain coefficient. As we are already aware, this selection will have an influence on the EMV/MEV results. Based on what we’ve seen in the previous chapter, we should not expect to notice much difference in the results of the two approaches.
3.2.2 Nash Equilibrium

Formulation and Definition

We model the control problem by the two cost functions:

\[
J_i(v_1, v_2) = \frac{\alpha_i}{2} \|v_i\|_U^2 + \frac{k_i}{2} \|y(v_1, v_2) - y_{d_i}\|_{L^2(\omega_i)}^2 + \frac{l_i}{2} \|y(v_1, v_2, T) - y_{r_i}\|_{L^2(\omega_i)}^2, \quad i = 1, 2
\]

or

\[
J_i(v_1, v_2) = \frac{\alpha_i}{2} \int_0^T \int_0^{2\pi} v_i^2 \, dx \, dt + \frac{k_i}{2} \int_0^T \int_{\omega_i} (y - y_{d_i})^2 \, dx \, dt + \frac{l_i}{2} \int_{\omega_i} (y - y_{r_i})^2 \, dx, \quad i = 1, 2
\]

where

\[
\alpha_i > 0, \quad k_i \geq 0, \quad l_i \geq 0, \quad k_i + l_i > 0.
\]

The first term in the cost functions measures the size of the controls. The second and third measure the physical objective.

Therefore, our goal is to determine the control functions \(v_1, v_2\) that minimize the functionals \(J_1\) and \(J_2\) simultaneously. The effect of uncertainty, as we mentioned earlier, is neglected at this point, but it will be discussed in the next subsection. To achieve our goal, the Nash equilibrium point is employed as a solution for the proposed multi-objective optimization problem. For more information about the Nash equilibrium points, one can consult [6].

A Nash equilibrium is a pair \((u_1, u_2) \in U_1 \times U_2\) such that

\[
\begin{align*}
J_1(u_1, u_2) &\leq J_1(v_1, u_2), \quad \forall v_1 \in U_1 \\
J_2(u_1, u_2) &\leq J_2(u_1, v_2), \quad \forall v_2 \in U_2
\end{align*}
\]

Equivalently, according to [94], a Nash equilibrium \((u_1, u_2) \in U_1 \times U_2\) is a solution of the coupled optimality system

\[
\begin{align*}
\frac{\partial J_1}{\partial v_1}(u_1, u_2) &= 0 \\
\frac{\partial J_2}{\partial v_2}(u_1, u_2) &= 0
\end{align*}
\]

An equivalent formulation of the optimality system (3.5) can be obtained if we follow an adjoint system procedure as in [93]. In other words, our goal is to get an expression for the partial derivatives.
The Adjoint System

Let us consider, for the moment, \( w_2 \in U_2, v_1 \in U_1 \), a small perturbation \( \delta_1 v_1 \) of \( v_1 \), and a small perturbation \( \delta_1 y \) of \( y \). Denote

\[
\begin{align*}
Y &= y + \delta_1 y \\
V_1 &= v_1 + \delta_1 v_1
\end{align*}
\]

and substitute them into (3.1). It follows that

\[
\begin{align*}
y_t + \frac{\partial (\delta_1 y)}{\partial t} - \nu y_{xx} - \nu \frac{\partial^2 (\delta_1 y)}{\partial x^2} + (y + \delta_1 y) \left( y_x + \frac{\partial (\delta_1 y)}{\partial x} \right) \\
&= f + (v_1 + \delta_1 v_1) \chi_{\omega_1} + w_2 \chi_{\omega_2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
y(0, t) + \delta_1 y(0, t) &= y(2\pi, t) + \delta_1 y(2\pi, t), \forall t \in (0, T) \\
y(x, 0) + \delta_1 y(x, 0) &= y_0(x), \forall x \in (0, 2\pi)
\end{align*}
\]

Then, \( \delta_1 y \) is a solution of the linearized system

\[
\begin{align*}
\frac{\partial (\delta_1 y)}{\partial t} - \nu \frac{\partial^2 (\delta_1 y)}{\partial x^2} + y \frac{\partial (\delta_1 y)}{\partial x} + (\delta_1 y) y_x = (\delta_1 v_1) \chi_{\omega_1}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
(\delta_1 y)(0, t) &= (\delta_1 y)(2\pi, t), \forall t \in (0, T) \\
(\delta_1 y)(x, 0) &= 0, \forall x \in (0, 2\pi)
\end{align*}
\]

(3.6)

We have

\[
\delta_1 J_1 (v_1, w_2) = \int_0^T \int_0^{2\pi} \frac{\partial J_1}{\partial v_1} (v_1, w_2) \delta_1 v_1 dx dt \\
= \alpha_1 \int_0^T \int_0^{2\pi} v_1 \delta_1 v_1 dx dt \\
+ k_1 \int_0^T \int_{\omega_1} [y(v_1, w_2) - y_{\Omega_1}] \delta_1 y dx dt \\
+ l_1 \int_{\omega T_1} [y(T; v_1, w_2) - y_{T_1}] \delta_1 y(T) dx
\]

(3.7)

Now let us introduce a reasonably smooth and periodic function \( p_1 \) defined over \([0, 2\pi] \times [0, T] \). We multiply the first equation of (3.6) by \( p_1 \):

\[
p_1 \frac{\partial}{\partial t} (\delta_1 y) - \nu p_1 \frac{\partial^2}{\partial x^2} (\delta_1 y) + p_1 y \frac{\partial}{\partial x} (\delta_1 y) + p_1 (\delta_1 y) y_x = p_1 (\delta_1 v_1) \chi_{\omega_1}
\]

68
and integrate over \([0, T] \times [0, 2\pi]\):

\[
\int_0^T \int_0^{2\pi} p_1 \frac{\partial}{\partial t} (\delta_1 y) \, dx \, dt - \int_0^T \int_0^{2\pi} \nu p_1 \frac{\partial^2}{\partial x^2} (\delta_1 y) \, dx \, dt \\
+ \int_0^T \int_0^{2\pi} p_1 y \frac{\partial}{\partial x} (\delta_1 y) \, dx \, dt \\
+ \int_0^T \int_0^{2\pi} p_1 (\delta_1 y) y \, dx \, dt = \int_0^T \int_0^{2\pi} p_1 (\delta_1 v_1) \chi_{\omega_1} \, dx \, dt
\]

Integration by parts yields:

\[
\int_0^{2\pi} p_1 (\delta_1 y) \, dx - \int_0^T \int_0^{2\pi} \frac{\partial p_1}{\partial t} (\delta_1 y) \, dx \, dt - \int_0^T \nu \frac{\partial}{\partial x} (\delta_1 y) \bigg|_0^{2\pi} \, dt \\
+ \int_0^T \int_0^{2\pi} \nu \frac{\partial}{\partial x} (\delta_1 y) \, dx \, dt + \int_0^T p_1 y (\delta_1 y) \bigg|_0^{2\pi} \, dt \\
- \int_0^T \int_0^{2\pi} \frac{\partial}{\partial x} (p_1 y) (\delta_1 y) \, dx \, dt \\
+ \int_0^T \int_0^{2\pi} p_1 (\delta_1 y) y \, dx \, dt = \int_0^T \int_0^{2\pi} p_1 (\delta_1 v_1) \chi_{\omega_1} \, dx \, dt
\]

It follows from (3.6) and integration by parts, once again, that:

\[
\int_0^{2\pi} p_1 (T) (\delta_1 y) (T) \, dx + \int_0^T p_1 y (\delta_1 y) \bigg|_0^{2\pi} \, dt \\
- \int_0^T \nu \frac{\partial}{\partial x} (\delta_1 y) \bigg|_0^{2\pi} \, dt + \int_0^T \nu \frac{\partial}{\partial x} (\delta_1 y) \bigg|_0^{2\pi} \, dt \\
+ \int_0^T \int_0^{2\pi} \left[ - \frac{\partial p_1}{\partial t} - \nu \frac{\partial^2 p_1}{\partial x^2} - \frac{\partial}{\partial x} (p_1 y) + p_1 y \right] (\delta_1 y) \, dx \, dt \\
= \int_0^T \int_0^{2\pi} p_1 (\delta_1 v_1) \chi_{\omega_1} \, dx \, dt
\]

(3.8)

In order to simplify the expression of \( \frac{\partial J_1}{\partial v_1} (v_1, w_2) \), we choose \( p_1 \) as the solution of the following periodic backward adjoint system:

\[
\begin{cases}
- \frac{\partial p_1}{\partial t} - \nu \frac{\partial^2 p_1}{\partial x^2} - \frac{\partial}{\partial x} y = k_1 [y - y_{d1}] \chi_{\omega_1}, & \text{in } [0, 2\pi] \times [0, T] \\
p_1 (x, T) = l_1 [y (x, T) - y_{T_1} (x)] \chi_{\omega_{T_1}}, & \text{in } [0, 2\pi] \\
p_1 (0, t) = p_1 (2\pi, t), & \text{in } [0, T]
\end{cases}
\]

(3.9)
Therefore, from (3.7), (3.8), (3.9), we can derive that
\[ \int_{0}^{T} \int_{0}^{2\pi} \frac{\partial J_1}{\partial v_1}(v_1, w_2) \, \delta_1 v_1 \, dx \, dt = \int_{0}^{T} \int_{0}^{2\pi} \left[ \alpha_1 v_1 + p_1 \chi_{\omega_1} \right] \delta_1 v_1 \, dx \, dt. \]

Since \( \delta_1 v_1 \) is arbitrary, we have proved that
\[ \frac{\partial J_1}{\partial v_1}(v_1, w_2) = \alpha_1 v_1 + p_1 \chi_{\omega_1} \tag{3.10} \]

In the similar way,
\[ \frac{\partial J_2}{\partial v_2}(w_1, v_2) = \alpha_2 v_2 + p_2 \chi_{\omega_2} \]

where, \( p_2 \) satisfies a similar periodic system:
\[ \begin{cases} -\frac{\partial p_2}{\partial t} - \nu \frac{\partial^2 p_2}{\partial x^2} - \frac{\partial p_2}{\partial x} y = k_2 \left[ y - y_{d_2} \right] \chi_{\omega_{d_2}}, \text{ in } [0, 2\pi] \times [0, T] \\ p_2(x, T) = l_2 \left[ y(x, T) - y_{r_2}(x) \right] \chi_{\omega_{r_2}}, \text{ in } [0, 2\pi] \\ p_2(0, t) = p_2(2\pi, t), \text{ in } [0, T] \end{cases} \tag{3.11} \]

Algorithm

To get the solution of the optimality system (3.5), we can employ a gradient-type procedure proposed by Croicu in [31].

ALGORITHM-Nash Equilibrium:

1. Take \((u_1^0, u_2^0) \in U_1 \times U_2\), and set \( k = 0 \).

2. Solve the coupled systems of partial differential equations:
\[ \begin{cases} y_t - \nu y_{xx} + y y_x = f + u_1^k \chi_{\omega_1} + u_2^k \chi_{\omega_2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\ y(0, t) = y(2\pi, t), \forall t \in (0, T) \\ y(x, 0) = y_0(x), \forall x \in (0, 2\pi) \end{cases} \tag{3.12} \]

\[ \begin{cases} -(p_1)_t - \nu (p_1)_{xx} - y(p_1)_x = k_1 \left[ y - y_{d_1} \right] \chi_{\omega_{d_1}}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\ p_1(0, t) = p_1(2\pi, t), \forall t \in (0, T) \\ p_1(x, T) = l_1 \left[ y(x, T) - y_{r_1}(x) \right] \chi_{\omega_{r_1}}, \forall x \in (0, 2\pi) \end{cases} \tag{3.13} \]
\[
\begin{aligned}
-(p_2)_t - \nu (p_2)_{xx} - y (p_2)_x = k_2 [y - y_{d2}] \chi_{\omega d2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
p_2 (0, t) = p_2 (2\pi, t), \forall t \in (0, T) \\
p_2 (x, T) = l_2 [y (x, T) - y_T (x)] \chi_{\omega r2}, \forall x \in (0, 2\pi)
\end{aligned}
\]  

(3.14)

3. Compute the partial derivatives:

\[
\begin{aligned}
\frac{\partial J_1}{\partial v_1} (u_1^k, u_2^k) &= \alpha_1 u_1^k + \chi_{\omega 1} \\
\frac{\partial J_2}{\partial v_2} (u_1^k, u_2^k) &= \alpha_2 u_2^k + \chi_{\omega 2}
\end{aligned}
\]

4. Get the appropriate step-sizes \( s_1^k, s_2^k > 0 \), such that

\[
\begin{aligned}
J_1 \left( u_1^k - s_1^k \frac{\partial J_1}{\partial v_1} (u_1^k, u_2^k), u_2^k \right) &= \min_{s_1 > 0} J_1 \left( u_1^k - s_1 \frac{\partial J_1}{\partial v_1} (u_1^k, u_2^k), u_2^k \right) \\
J_2 \left( u_1^k, u_2^k - s_2^k \frac{\partial J_2}{\partial v_2} (u_1^k, u_2^k) \right) &= \min_{s_2 > 0} J_2 \left( u_1^k, u_2^k - s_2 \frac{\partial J_2}{\partial v_2} (u_1^k, u_2^k) \right)
\end{aligned}
\]

5. Take

\[
\begin{aligned}
u_1^{k+1} &= u_1^k - s_1^k \frac{\partial J_1}{\partial v_1} (u_1^k, u_2^k) \\
u_2^{k+1} &= u_2^k - s_2^k \frac{\partial J_2}{\partial v_2} (u_1^k, u_2^k)
\end{aligned}
\]

6. Test for convergence, or if a number of specified iterations NITER have been performed;

If YES: we have the solution \((u_1, u_2) = (u_1^{k+1}, u_2^{k+1})\);

If NO: put \( k = k + 1 \) and go to the second step.

For a short description of the minimization procedure employed at step#4, we refer the reader to Appendix B.

**Numerical Discretization**

The proposed algorithm for determining the Nash equilibrium requires, at each step, solving three coupled systems of partial differential equations (3.12), (3.13), (3.14). All
three systems have something in common: they have periodic boundary conditions. The main difference between them is that (3.12) requires a forward integration in time, but (3.13), and (3.14) require a backward integration in time. Before we address the time integration, let’s discuss the space discretization and the discrete representation of the solution. Let’s analyze (3.12) first.

There are basically two steps required to obtain a numerical approximation to a solution of a differential equation. First, an appropriate finite or discrete representation of the solution must be chosen. The second step is to obtain equations for the coefficients of that representation from the original equation. If \( y \) is a periodic square integrable function, then \( y \) can be expanded in a Fourier series of the form ([110]):

\[
y(x, t) = \sum_{k=-\infty}^{\infty} \hat{y}_k(t) e^{ikx}, \quad y(., t) \in L^2(0, 2\pi), \forall t \in (0, T)
\]

Denote by

\[ S_N = \text{span} \{ e^{ikx} \}_{k=-\frac{N}{2}}^{\frac{N}{2}-1}, \text{ where } N \text{ is an even positive integer.} \]

then, the spectral (Galerkin) approximation to \( y(x, t) \) is given by:

\[
Y(x, t) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{y}_k(t) e^{ikx}, \quad Y(., t) \in S_N, \forall t \in (0, T).
\]  

(3.15)

The scalar product valid here is

\[
(u, v) = \int_0^{2\pi} u\bar{v} \, dx, \quad u, v \in L^2(0, 2\pi),
\]

so that the orthogonality property of the complex exponential functions is

\[
\int_0^{2\pi} e^{ikx} e^{-ilx} \, dx = \begin{cases} 
2\pi & \text{if } k = l \\
0 & \text{if } k \neq l
\end{cases}
\]

(3.16)

We can calculate the coefficients \( \hat{y}_k(t), k = \frac{N}{2}, \ldots, \frac{N}{2} - 1 \), by using the Galerkin technique. First, the approximation (3.15) is substituted into the partial differential equation of the system (3.12), or, for the sake of simplicity, into the equation (3.1). Because \( Y(x, t) \) is just an approximation of \( y(x, t) \), by using the approximation \( Y(x, t) \) we introduce a residual, \( R \):

\[
Y_t - \nu Y_{xx} + YY_x - f - v_1 \chi \omega_1 - v_2 \chi \omega_2 = R
\]

72
We’re using the Galerkin method, so we set the residual to be orthogonal to $S_N$:

$$(R, V) = 0, \forall V \in S_N$$

or

$$(Y_t - \nu Y_{xx} + YY_x - f - v_1 \chi_{\omega_1} - v_2 \chi_{\omega_2}, V) = 0, \forall V \in S_N.$$  

Because of the definition of $S_N$, we can write further

$$\begin{align*}
\frac{1}{2\pi} \int_0^{2\pi} (Y_t - \nu Y_{xx} + YY_x - f - v_1 \chi_{\omega_1} - v_2 \chi_{\omega_2}) e^{-ikx} dx = 0, k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1 \quad (3.17)
\end{align*}$$

We evaluate each term from (3.17), (for $k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1$), and we use orthogonality (3.16):

$$\begin{align*}
\frac{1}{2\pi} \int_0^{2\pi} Y_t e^{-ikx} dx &= \frac{1}{2\pi} \int_0^{2\pi} \left( \sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} \frac{d\hat{y}_m}{dt} e^{imx} \right) e^{-ikx} dx = \frac{1}{2\pi} \int_0^{2\pi} \frac{d\hat{y}_k}{dt} dx = \frac{d\hat{y}_k}{dt} \quad (3.18)
\end{align*}$$

$$\begin{align*}
\frac{-\nu}{2\pi} \int_0^{2\pi} Y_{xx} e^{-ikx} dx &= \frac{-\nu}{2\pi} \int_0^{2\pi} \left( \sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{y}_m (im)^2 e^{imx} \right) e^{-ikx} dx = \frac{-\nu}{2\pi} \int_0^{2\pi} \hat{y}_k k^2 dx = \nu k^2 \hat{y}_k \quad (3.19)
\end{align*}$$

Recall that

$$Y(x, t) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{y}_k(t) e^{ikx},$$

$$Y_x(x, t) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{y}_k(t) (ik) e^{ikx},$$

so

$$(YY_x)_k = \sum_{p,q=-\frac{N}{2}}^{\frac{N}{2}-1} iq\hat{y}_p(t) \hat{y}_q(t)$$

Then,

$$\begin{align*}
\frac{1}{2\pi} \int_0^{2\pi} YY_x e^{-ikx} dx = (YY_x)_k \quad (3.20)
\end{align*}$$

There is a delicate question that concerns the evaluation of the convolution sum $(YY_x)_k$ which must be as efficient as possible. This can be obtained through the so-called “pseudospectral technique” ([87]). This technique performs the differentiations in the
spectral space (the space of coefficients $\hat{y}_k$) and the products in the physical space (the space of the values $Y(x_j)$ at the collocation points $x_j = \frac{2\pi j}{N}, j = 0, 1, ..., N-1$). The link between the two spaces is made by the FFT (Fast Fourier Transform). This technique introduces an aliasing error, which can be removed. The aliasing removal technique employed here was considered by Orszag, in 1971 ([83]). The method extends the spectrum (and therefore the number of collocation points) of the involved quantities, so that the alias terms, introduced in the pseudospectral calculation of the resulting quantities, are not actually present. It can be proved that if we choose $M \geq \frac{3N}{2}$ collocation points $x^*_j = \frac{2\pi j}{M}, j = 0, 1, ..., M - 1$ to calculate the products, then the aliasing error is removed.

We continue evaluating the terms from (3.17). If we consider the approximation

$$f \approx \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_k(t) e^{ikx},$$

then

$$-\frac{1}{2\pi} \int_0^{2\pi} f e^{-ikx} dx = -\hat{f}_k$$

(3.21)

Similarly, for

$$v_1\chi_{\omega_1} \approx \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \widehat{(v_1\chi_{\omega_1})}_k(t) e^{ikx},$$

we get that

$$-\frac{1}{2\pi} \int_0^{2\pi} v_1\chi_{\omega_1} e^{-ikx} dx = -\widehat{(v_1\chi_{\omega_1})}_k$$

(3.22)

If

$$v_2\chi_{\omega_2} \approx \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \widehat{(v_2\chi_{\omega_2})}_k(t) e^{ikx},$$

then

$$-\frac{1}{2\pi} \int_0^{2\pi} v_2\chi_{\omega_2} e^{-ikx} dx = -\widehat{(v_2\chi_{\omega_2})}_k$$

(3.23)

Now using the equations (3.18), (3.19), ..., (3.23), the system (3.12) becomes:

$$\begin{cases} \\
\frac{d\hat{y}_k}{dt} + \nu k^2 \hat{y}_k + (YY_x)_k = \hat{f}_k + \widehat{(v_1\chi_{\omega_1})}_k + \widehat{(v_2\chi_{\omega_2})}_k, & k = \frac{N}{2}, ..., \frac{N}{2} - 1 \\
\hat{y}_k(0) = \frac{1}{2\pi} \int_0^{2\pi} y_0(x) e^{-ikx} dx 
\end{cases}$$

(3.24)
In the same way, making the notations

\[ Y (x, t) = \sum_{k=-N}^{N} \hat{y}_k (t) e^{ikx}, \]
\[ P_1 (x, t) = \sum_{k=-N}^{N} \hat{p}_{1k} (t) e^{ikx}, \]
\[ P_2 (x, t) = \sum_{k=-N}^{N} \hat{p}_{2k} (t) e^{ikx}, \]

\[ F_1 (x, t) = \sum_{k=-N}^{N} \hat{F}_{1k} (t) e^{ikx}, \]
\[ F_2 (x, t) = \sum_{k=-N}^{N} \hat{F}_{2k} (t) e^{ikx}, \]
\[ G_1 (x, t) = \sum_{k=-N}^{N} \hat{G}_{1k} (t) e^{ikx}, \]
\[ G_2 (x, t) = \sum_{k=-N}^{N} \hat{G}_{2k} (t) e^{ikx}, \]

the systems (3.13), and (3.14) become

\[ \begin{cases} 
-\frac{d\hat{p}_{1k}}{dt} + \nu k^2 \hat{p}_{1k} + (Y(P_1)_x) = \hat{F}_{1k}, k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1 \\
\hat{p}_{1k} (T) = \hat{G}_{1k} = \frac{1}{2\pi} \int_{0}^{2\pi} l_1 (y - y_{T_1}) \chi_{\omega_T} e^{-ikx} dx 
\end{cases} \]  
(3.25)

\[ \begin{cases} 
-\frac{d\hat{p}_{2k}}{dt} + \nu k^2 \hat{p}_{2k} + (Y(P_2)_x) = \hat{F}_{2k}, k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1 \\
\hat{p}_{2k} (T) = \hat{G}_{2k} = \frac{1}{2\pi} \int_{0}^{2\pi} l_1 (y - y_{T_2}) \chi_{\omega_T} e^{-ikx} dx 
\end{cases} \]  
(3.26)

To integrate in time, we need to consider numerical schemes that are absolute stable for our problems.
To integrate in time the initial value problem (3.24), we use Williamson’s third order low storage Runge-Kutta scheme ([116]), where we choose the time-step

\[ \Delta t = dt = \frac{C}{\nu N^2} \leq \frac{C}{\nu \left(\frac{N}{2}\right)^2}, \, C \approx 1.7 \]

To integrate backward in time the systems (3.25), and (3.26), we use the backward Euler scheme, where the time-step is given by

\[ \Delta t = dt = \frac{1}{\nu N^2} \leq \frac{1}{\nu \left(\frac{N}{2}\right)^2}. \]

Results

Assuming that the reader is familiar with the concept of Nash equilibrium (see Appendix C for an illustration of Nash equilibrium), we consider the following data for the Burgers equation:

\[ T = 1.0, \nu = 0.00125, N = 32, NITER = 4 \]

\[ \omega_1 = \omega_{d_1} = \omega_{T_1} = (0, \pi), \omega_2 = \omega_{d_2} = \omega_{T_2} = (\pi, 2\pi) \]

\[ \alpha_1 = \alpha_2 = k_1 = k_2 = l_1 = l_2 = 1.0 \]

\[ f(x, t) = \sin x, y_0(x) = 0. \]

\[ y_{d_1}(x, t) = \sin t \sin x, y_{d_2}(x, t) = t \sin x \]

\[ y_{T_1}(x) = \sin(T) \sin(x), y_{T_2}(x) = \sin x \]

with the initial control \( v_1, v_2 \) being

\[ v_1^{\text{initial}}(x, t) = v_2^{\text{initial}}(x, t) = \sin(x) \]

Figures 3.1 and 3.2 show the uncontrolled, controlled with Nash strategy and target state solution at time 0.5 and 1.-dt, respectively. Figures 3.3 and 3.4 display the computed controls \( v_1, v_2 \) at the same moment of times. In Table 3.1 and 3.2, we give some further information about our solution.

In this case the target functions considered are not compatible (i.e., when being really close to one of the target functions implies to be far away from the others). The numerical results obtained for the Burgers equation are consistent with what we can expect from a non-cooperative Nash strategy, namely in the region \( \omega_{d_1}, \omega_{d_2} \) the solution is close to the target, but not the target.
Figure 3.1: Uncontrolled, controlled, target solution at $t=0.5$
Figure 3.2: Uncontrolled, controlled, target solution at $t=1.-dt$
Figure 3.3: Controls at $t=0.5$
Figure 3.4: Controls at $t=1.-dt$
Table 3.1: The initial and final values of the objectives for the Burgers equation

<table>
<thead>
<tr>
<th></th>
<th>$J_1$</th>
<th>$J_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(v_{1}^{\text{initial}}, v_{2}^{\text{initial}})$</td>
<td>1.40346549</td>
<td>1.29437838</td>
</tr>
<tr>
<td>$(v_{1}^{\text{final}}, v_{2}^{\text{final}})$</td>
<td>0.0105658785</td>
<td>0.00666189933</td>
</tr>
</tbody>
</table>

Table 3.2: Additional results for the deterministic Burgers equation

<table>
<thead>
<tr>
<th></th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$|y^{\text{uncontrolled}} - y^{\text{desired}}|$</td>
<td>0.196469217</td>
</tr>
<tr>
<td>$|y^{\text{controlled}} - y^{\text{desired}}|$</td>
<td>0.000675401747</td>
</tr>
<tr>
<td>$|\frac{\partial J_1}{\partial v_1}(v_{1}^{\text{final}}, v_{2}^{\text{final}})|$</td>
<td>6.45691419E-05</td>
</tr>
<tr>
<td>$|\frac{\partial J_2}{\partial v_2}(v_{1}^{\text{final}}, v_{2}^{\text{final}})|$</td>
<td>8.20361592E-05</td>
</tr>
<tr>
<td>CPU time (seconds)</td>
<td>287 seconds</td>
</tr>
</tbody>
</table>
3.2.3 Stochastic Nash Equilibrium Algorithm

For the uncertainty analysis, we assume the input viscosity $\nu$ to be stochastic, and it is considered to follow either a uniform or a Gaussian distribution. Again, we look at our multi-objective problem as a game, with 2 players, where the Nash equilibrium point is sought. Because of the uncertainty in the viscosity, we will employ the EMV or MEV approach, namely

$$d_{\text{Nash}}^{\text{EMV}} = E \left( \text{Nash}(v_1, v_2) \right)$$

$$d_{\text{Nash}}^{\text{MEV}} = \text{Nash}(v_1, v_2)$$

The EMV and MEV stochastic Nash algorithms are a combination of the algorithms presented in 3.2.2 and 2.4. A Monte-Carlo type of approach is employed, with NSAMP number of samples. The implementation of the EMV criterion is straightforward. The MEV criterion needs some additional manipulations, especially because it makes use of the First Mean Value Theorem. The algorithms are presented next.

**ALGORITHM-Stochastic EMV Nash Equilibrium:**

1. Put $r = 1$.

2. Independently sample a value of $\nu_r$ according to the prescribed probability density function.

3. Put $k = 0$ and take $(u_1^k, u_2^k) \in U_1 \times U_2$.

4. Solve the coupled systems of partial differential equations:

$$
\begin{cases}
    y_t - \nu_r y_{xx} + yy_x = f + u_1^k \chi_1 + u_2^k \chi_2, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
    y(0, t) = y(2\pi, t), \forall t \in (0, T) \\
    y(x, 0) = y_0(x), \forall x \in (0, 2\pi)
\end{cases}
$$

(3.29)
\[
\begin{aligned}
-p_1(t) - \nu_r (p_1)_{xx} - y (p_1)_{x} &= k_1 [y - y_{d_1}] \chi_{\omega_1}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
p_1(0, t) &= p_1(2\pi, t), \forall t \in (0, T) \\
p_1(x, T) &= l_1 [y(x, T) - y_{T_1}(x)] \chi_{\omega_{T_1}}, \forall x \in (0, 2\pi)
\end{aligned}
\tag{3.30}
\]

\[
\begin{aligned}
-p_2(t) - \nu_r (p_2)_{xx} - y (p_2)_{x} &= k_2 [y - y_{d_2}] \chi_{\omega_2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
p_2(0, t) &= p_2(2\pi, t), \forall t \in (0, T) \\
p_2(x, T) &= l_2 [y(x, T) - y_{T_2}(x)] \chi_{\omega_{T_2}}, \forall x \in (0, 2\pi)
\end{aligned}
\tag{3.31}
\]

5. Compute the partial derivatives:
\[
\begin{aligned}
\frac{\partial J_1}{\partial v_1} (u^k_1, u^k_2) &= \alpha_1 u^k_1 + p_1 \chi_{\omega_1} \\
\frac{\partial J_2}{\partial v_2} (u^k_1, u^k_2) &= \alpha_2 u^k_2 + p_2 \chi_{\omega_2}
\end{aligned}
\]

6. Get the appropriate step-sizes \(s_1^k, s_2^k > 0\), such that
\[
\begin{aligned}
J_1 \left( u^k_1 - s_1^k \frac{\partial J_1}{\partial v_1} (u^k_1, u^k_2), u^k_2 \right) &= \min_{s_1 > 0} J_1 \left( u^k_1 - s_1 \frac{\partial J_1}{\partial v_1} (u^k_1, u^k_2), u^k_2 \right) \\
J_2 \left( u^k_1, u^k_2 - s_2^k \frac{\partial J_2}{\partial v_2} (u^k_1, u^k_2) \right) &= \min_{s_2 > 0} J_2 \left( u^k_1, u^k_2 - s_2 \frac{\partial J_2}{\partial v_2} (u^k_1, u^k_2) \right)
\end{aligned}
\]

7. Take
\[
\begin{aligned}
u_1^{k+1} &= u_1^k - s_1^k \frac{\partial J_1}{\partial v_1} (u_1^k, u_2^k) \\
u_2^{k+1} &= u_2^k - s_2^k \frac{\partial J_2}{\partial v_2} (u_1^k, u_2^k)
\end{aligned}
\]

8. Test for convergence or if a number of specified iterations NITER have been performed:

If YES: we have the solution \((u_1^r, u_2^r) = (u_1^{k+1}, u_2^{k+1})\);

If NO: put \(k = k + 1\) and go to the fourth step.

9. If \(r < \text{NSAMP}\) then put \(r = r + 1\) go to the second step.
10. Average all the computed controls

\[
\begin{align*}
u_{EMV}^1 &= \frac{1}{NSAMP} \sum_{r=1}^{NSAMP} u_r^1 \\
u_{EMV}^2 &= \frac{1}{NSAMP} \sum_{r=1}^{NSAMP} u_r^2
\end{align*}
\]

On the other hand,

**ALGORITHM-Stochastic MEV Nash Equilibrium:**

1. Put \( k = 0 \) and take \((u_1^k, u_2^k) \in U_1 \times U_2\).

2. Independently sample all the values of \( \nu_r \), \( r=1,\ldots,NSAMP \), according to the prescribed probability density function.

3. For every \( r=1,\ldots,NSAMP \), solve the following system of partial differential equations:

\[
\begin{align*}
y_t - \nu_r y_{xx} + yy_x &= f + u_1^k \chi_{\omega_1} + u_2^k \chi_{\omega_2}, \forall (x,t) \in (0,2\pi) \times (0,T) \\
y(0,t) &= y(2\pi,t), \forall t \in (0,T) \\
y(x,0) &= y_0(x), \forall x \in (0,2\pi)
\end{align*}
\] (3.32)

and evaluate \( J_1(u_1^k, u_2^k, \nu_r) \), \( J_2(u_1^k, u_2^k, \nu_r) \).

4. Identify \( \nu_k^1, \nu_k^2 \) such that

\[
\begin{align*}
J_1(u_1^k, u_2^k, \nu_k^1) &= \frac{1}{NSAMP} \sum_{r=1}^{NSAMP} J_1(u_1^k, u_2^k, \nu_r) \\
J_2(u_1^k, u_2^k, \nu_k^2) &= \frac{1}{NSAMP} \sum_{r=1}^{NSAMP} J_2(u_1^k, u_2^k, \nu_r)
\end{align*}
\]

5. Solve the coupled systems of partial differential equations:
\[
\begin{aligned}
&\begin{cases}
y_t - \nu_k^1 y_{xx} + y_{yx} = f + u_1^k \chi_{\omega_1} + u_2^k \chi_{\omega_2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
y(0, t) = y(2\pi, t), \forall t \in (0, T) \\
y(x, 0) = y_0(x), \forall x \in (0, 2\pi)
\end{cases} \\
&\begin{cases}
-(p_1)_t - \nu_k^1 (p_1)_{xx} - y(p_1)_x = k_1 [y - y_{d_1}] \chi_{\omega_{d_1}}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
p_1(0, t) = p_1(2\pi, t), \forall t \in (0, T) \\
p_1(x, T) = l_1 [y(x, T) - y_{T_1}(x)] \chi_{\omega_{T_1}}, \forall x \in (0, 2\pi)
\end{cases} \\
&\begin{cases}
y_t - \nu_k^2 y_{xx} + y_{yx} = f + u_1^k \chi_{\omega_1} + u_2^k \chi_{\omega_2}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
y(0, t) = y(2\pi, t), \forall t \in (0, T) \\
y(x, 0) = y_0(x), \forall x \in (0, 2\pi)
\end{cases} \\
&\begin{cases}
-(p_2)_t - \nu_k^2 (p_2)_{xx} - y(p_2)_x = k_2 [y - y_{d_2}] \chi_{\omega_{d_2}}, \forall (x, t) \in (0, 2\pi) \times (0, T) \\
p_2(0, t) = p_2(2\pi, t), \forall t \in (0, T) \\
p_2(x, T) = l_2 [y(x, T) - y_{T_2}(x)] \chi_{\omega_{T_2}}, \forall x \in (0, 2\pi)
\end{cases}
\end{aligned}
\]

6. Compute the partial derivatives:
\[
\begin{cases}
\frac{\partial J_1}{\partial v_1}(u_1^k, u_2^k, \nu_k^1) = \alpha_1 u_1^k + p_1 \chi_{\omega_1} \\
\frac{\partial J_2}{\partial v_2}(u_1^k, u_2^k, \nu_k^1) = \alpha_2 u_2^k + p_2 \chi_{\omega_2}
\end{cases}
\]

7. Get the appropriate step-sizes \(s_1^k, s_2^k > 0\), such that
\[
\begin{cases}
J_1 \left( u_1^k - s_1^k \frac{\partial J_1}{\partial v_1}(u_1^k, u_2^k), u_2^k, \nu_k^1 \right) = \min_{s_1 > 0} J_1 \left( u_1^k - s_1 \frac{\partial J_1}{\partial v_1}(u_1^k, u_2^k), u_2^k, \nu_k^1 \right) \\
J_2 \left( u_1^k, u_2^k - s_2^k \frac{\partial J_2}{\partial v_2}(u_1^k, u_2^k), \nu_k^2 \right) = \min_{s_2 > 0} J_2 \left( u_1^k, u_2^k - s_2 \frac{\partial J_2}{\partial v_2}(u_1^k, u_2^k), \nu_k^2 \right)
\end{cases}
\]
8. Take

\[
\begin{align*}
&u_1^{k+1} = u_1^k - s_1^k \frac{\partial J_1}{\partial u_1^k} (u_1^k, u_2^k, \nu_k^1) \\
&u_2^{k+1} = u_2^k - s_2^k \frac{\partial J_2}{\partial u_2^k} (u_1^k, u_2^k, \nu_k^2)
\end{align*}
\]

9. Test for convergence or if a number of specified iterations NITER have been performed:

If YES: we have the solution \((u_1^{MEV}, u_2^{MEV}) = (u_1^{k+1}, u_2^{k+1})\);

If NO: put \(k = k + 1\) and go to the second step.

**Results**

In order to illustrate the EMV and MEV stochastic Nash procedures discussed previously, we assume the same data as in Section 3.2.2, two distributions for the kinematic viscosity parameter: uniform distribution \(U[0.001, 0.0015]\), and Gaussian distribution \(\mathcal{N}(0.00125, 0.001^2)\), and NITER=4.

Small variations of the viscosity \(\nu\) are necessary to ensure the absolute stability of the numerical scheme, and unfortunately, they will have a negative impact on the numerical results.

It is quite clear from the Figures 3.5, 3.6, 3.11, 3.12, 3.26, 3.27, 3.32, 3.33 that the final controls minimize both objective functions on the whole range of viscosity, no matter what the nature of the randomness. The plots of the controlled solution shown in Figures 3.7, 3.8, 3.13, 3.14, 3.28, 3.29, 3.34, 3.35 justify it even more. The controls are presented in the Figures 3.9, 3.10, 3.15, 3.16, 3.30, 3.31, 3.36, 3.37.

In order to verify the suitability of the EMV approach, we constructed a few “EMV versus MEV” plots. According to Figures 3.17, 3.18, 3.19, 3.39, 3.40, we obtain a lower or at least the same risk at the EMV optimum compared to the MEV strategy. Unfortunately, the differences between the controlled EMV and MEV solution or between the EMV and MEV controls are so small that they could not be seen in Figures 3.20, 3.21, 3.41, 3.42, 3.22, 3.23, 3.24, 3.25, 3.43, 3.44, 3.45, 3.46.

For the purpose of completeness, we have also included some additional information on the EMV/MEV procedure in Table 3.3.

Overlooking the CPU time, the results obtained in the present subsection give rise to the impression that the EMV strategy is still superior to the MEV criterion. It is therefore
encouraging to apply the EMV and MEV stochastic Nash equilibrium to a wide range of problems.
Figure 3.5: $J_1$ at initial and final EMV control $\nu \sim U[0.001, 0.0015]$
Figure 3.6: $J_2$ at initial and final EMV control $-\nu \sim U[0.001, 0.0015]$
Figure 3.7: Uncontrolled, EMV controlled, target solution $t=0.5 - \nu \sim U[0.001, 0.0015]$
Figure 3.8: Uncontrolled, EMV controlled, target solution $t=1.-\nu \sim U[0.001, 0.0015]$
Figure 3.9: EMV optimal control at $t=0.5 - \nu \sim U[0.001, 0.0015]$
Figure 3.10: EMV optimal control $t=1.-dt \sim U[0.001, 0.0015]$
Figure 3.11: $J_1$ at initial and final MEV control $\nu \sim U[0.001, 0.0015]$
Figure 3.12: $J_2$ at initial and final MEV control $-\nu \sim U[0.001, 0.0015]$
Figure 3.13: Uncontrolled, MEV controlled, target solution $t=0.5 - \nu \sim U[0.001, 0.0015]$
Figure 3.14: Uncontrolled, MEV controlled, target solution $t=1.0-\nu \sim U[0.001, 0.0015]$
Figure 3.15: MEV optimal control $t=0.5 \rightarrow \nu \sim U[0.001, 0.0015]$
Figure 3.16: MEV optimal control $t=1.-dt \sim U[0.001, 0.0015]$
Figure 3.17: $J_1$ at final EMV and MEV control $-\nu \sim U[0.001, 0.0015]$
Figure 3.18: $J_2$ at final EMV and MEV control $-v \sim U[0.001, 0.0015]$
Figure 3.19: $J_2$ at final EMV and MEV control $\nu \sim U[0.001, 0.0015]$(zoom)
Figure 3.20: EMV and MEV controlled solution $t=0.5$, $\nu = 0.00125 - \nu \sim U[0.001, 0.0015]$
Figure 3.21: EMV and MEV controlled solution $t=1.-dt$, $\nu = 0.00125-\nu \sim U[0.001, 0.0015]$
Figure 3.22: EMV and MEV optimal control $v_1 t=0.5 - \nu \sim U[0.001, 0.0015]$
Figure 3.23: EMV and MEV optimal control $v_2 \ t=0.5 - \nu \sim U[0.001, 0.0015]$
Figure 3.24: EMV and MEV optimal control $v_1$ at time $1.-dt$ ~ $U[0.001, 0.0015]$
Figure 3.25: EMV and MEV optimal control $v_2 t=1.-\nu \sim U[0.001, 0.0015]$
Figure 3.26: $J_1$ at initial and final EMV control $\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.27: $J_2$ at initial and final EMV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.28: Uncontrolled, EMV controlled, target solution $t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.29: Uncontrolled, EMV controlled, target solution $t=1.-dt - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.30: EMV optimal control $t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.31: EMV optimal control $t=1.-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.32: $J_1$ at initial and final MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.33: $J_2$ at initial and final MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.34: Uncontrolled, MEV controlled, target solution $t=0.5-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.35: Uncontrolled, MEV controlled, target solution $t=1.-dt \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.36: MEV optimal control $t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.37: MEV optimal control $t=1.-dt - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.38: $J_1$ at final EMV and MEV control $\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.39: $J_1$ at final EMV and MEV control $-\nu \sim \mathcal{N}(0.00125, 0.001^2)$ (zoom)
Figure 3.40: $J_2$ at final EMV and MEV control $- \nu \sim N(0.00125, 0.001^2)$
Figure 3.41: EMV and MEV controlled solution $t=0.5, \nu = 0.00125-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.42: EMV and MEV controlled solution $t=1.-dt, \nu = 0.00125-\nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.43: EMV and MEV optimal control $v_1 t=0.5 - \nu \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.44: EMV and MEV optimal control $v_2$ at $t=0.5 \sim N(0.00125, 0.001^2)$
Figure 3.45: EMV and MEV optimal control $v_1 \sim \mathcal{N}(0.00125, 0.001^2)$
Figure 3.46: EMV and MEV optimal control $v_2 t=1.-dt \sim \mathcal{N}(0.00125, 0.001^2)$
<table>
<thead>
<tr>
<th></th>
<th>CPU time (seconds)</th>
<th>$P_{\leq}$</th>
<th>$|y^{\text{uncontrolled}} - y^{\text{desired}}|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M \sim U (0.001, 0.0015)$</td>
<td>(EMV) 247621 (MEV) 463</td>
<td>$(J_1) 100%$ $(J_2) 100%$</td>
<td>(EMV) 0.000675400242 (MEV) 0.000675402671</td>
</tr>
<tr>
<td>$M \sim N (0.75, 0.025^2)$</td>
<td>(EMV) 263180 (MEV) 460</td>
<td>$(J_1) 100%$ $(J_2) 100%$</td>
<td>(EMV) 0.000675402208 (MEV) 0.000675402217</td>
</tr>
</tbody>
</table>

Table 3.3: Comparison of EMV and MEV approach for stochastic Burgers equation
3.3 Stochastic Nash Equilibrium for the Airfoil Shape Optimization

3.3.1 Problem Formulation

We now wish to generalize the airfoil shape optimization problem presented in §2.4 when multiple conflicting objectives are present. It is meaningful to formulate the following program ([33]):

\[
\begin{align*}
\min_{\theta \in \Theta, \alpha \in \Lambda} & \quad C_D (\theta, \alpha, M), M \in \Omega \\
\max_{\theta \in \Theta, \alpha \in \Lambda} & \quad C_L (\theta, \alpha, M), M \in \Omega \\
H_i (\theta, \alpha, M) & \geq 0, i = 1, 2, \ldots, m
\end{align*}
\]

Here the goal is to find the optimal shape \( \theta \) of a two-dimensional airfoil and the optimal angle of attack \( \alpha \) that minimizes the drag coefficient \( C_D \) and maximizes the lift coefficient \( C_L \) subject to some geometrical constraints, \( H_i \), under random Mach numbers, \( M \). As in Section 2.4, the desired shape \( \theta \) is an element of a design space \( \Theta \). The desired angle of attack \( \alpha \) is a real number restricted to \( \Lambda \subset \mathbb{R} \), and the random Mach number \( M \) is defined on a given probability space \( \Omega \) with probability density function \( f(M) \).

3.3.2 Nash Equilibrium

Definition and Algorithm

Now let us turn our attention to the following multi-objective optimization, where the effect of the uncertainty is neglected for the moment (\( M \) has a deterministic value):

\[
\begin{align*}
\min_{\theta \in \Theta, \alpha \in \Lambda} & \quad C_D (\theta, \alpha) \\
\max_{\theta \in \Theta, \alpha \in \Lambda} & \quad C_L (\theta, \alpha) \\
H_i (\theta, \alpha) & \geq 0, i = 1, 2, \ldots, m
\end{align*}
\]  

(3.37)

The Nash equilibrium point will be used as a solution for (3.37). In other words, we are looking for a point \((\bar{\theta}, \bar{\alpha}) \in \Theta \times \Lambda\) that is a Nash equilibrium for the non-cooperative game, i.e.

\[
\begin{align*}
C_D (\bar{\theta}, \bar{\alpha}) & \leq C_D (\theta, \bar{\alpha}), \forall \theta \in \Theta \\
C_L (\bar{\theta}, \bar{\alpha}) & \geq C_L (\bar{\theta}, \alpha), \forall \alpha \in \Lambda
\end{align*}
\]
Strictly speaking, the essence of this non-cooperative game lies in the two players, each one with its own objective. The first player’s utility function is the drag coefficient, while the second player’s utility is the lift coefficient. The first player can make decisions only regarding the airfoil shape, in contrast with the second one who can decide only on the angle of attack. The Nash equilibrium point is attained when both players are satisfied with their choice, i.e. they cannot improve their own objective any further provided that the strategy of the other does not change.

We will use the notation

\[
(\bar{\theta}, \bar{\alpha}) = \text{Nash}_{(\theta, \alpha) \in \Theta \times A} \left( C_D(\theta, \alpha), C_L(\theta, \alpha) \right)
\]

Thus the algorithm can be summed up as follows:

**ALGORITHM-Nash Equilibrium:**

1. Initialize the deterministic parameters, define the initial shape and angle of attack \((\theta^0, \alpha^0) \in \Theta \times A\), and let \(k = 0\).

2. Find an optimal shape \(\theta^{k+1} = \arg \min_{\theta \in \Theta} C_D(\theta, \alpha^k)\), according to the procedure presented in Appendix A.

3. Find an optimal angle of attack \(\alpha^{k+1} = \arg \min_{\alpha \in A} (-C_L(\theta^k, \alpha))\), according to the procedure presented in Appendix B.

4. Test for convergence, or if a number of specified iterations NITER have been performed:

   If YES: we have the solution \((\bar{\theta}, \bar{\alpha}) = (\theta^{k+1}, \alpha^{k+1})\);

   If NO: put \(k = k + 1\) and go to the second step.

**Results**

The initial shape is again the well-known NACA-0012 profile with the initial angle of attack \(\alpha = 0.5^\circ\), and the deterministic Mach number \(M = 0.7907923\). The number of iterations is set up \(NITER = 4\).

The Nash calculations make use of Jameson’s code once again, and show a substantial decrease in the drag coefficient from 0.00991043 to 0.00468347, and increase in the lift.
coefficient, from 0.14377801 to 0.21432692. The initial angle of attack is 0.50000000, while the final value is 0.79996145.

The Figure 3.47 shows a comparison of the initial and final geometry. As we expected, improvements are seen. On the other hand, the initial pressure and the final pressure at the Nash optimum point are plotted in Figure 3.48. We see that the drag minimization comes from the reduction of the shock wave on the upper surface, while the lift maximization comes from a bigger difference between the pressure over the upper surface and over the bottom surface.

For the sake of completeness, we include Tables 3.4 and 3.5. The Table 3.4 presents the results obtained at the intermediate steps #2, and #3 of the Nash algorithm. Furthermore, after observing the Nash outcome, the drag and lift coefficient are computed at the final Nash geometry and initial angle of attack (see Table 3.5). A considerable drag reduction is experienced at a small decrease in the lift coefficient. Therefore, even if the angle of attack is maintained constant, the Nash geometry is a good candidate for the drag minimization. If the values of the angle of attack have any practical implementation is another issue.
Figure 3.47: (MO) Geometry Nash & initial
Figure 3.48: (MO) Pressure at Nash & initial point, $M = 0.7907923$
<table>
<thead>
<tr>
<th>Shape</th>
<th>Angle of Attack</th>
<th>$C_D$</th>
<th>$C_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NACA-0012</td>
<td>0.50000000</td>
<td>0.00991043</td>
<td>0.14377801</td>
</tr>
<tr>
<td>First Geometry</td>
<td>0.50000000</td>
<td>0.00174608</td>
<td>0.07548249</td>
</tr>
<tr>
<td></td>
<td>0.79988801</td>
<td>0.00741795</td>
<td>0.22396138</td>
</tr>
<tr>
<td>Second Geometry</td>
<td>0.79988801</td>
<td>0.00177581</td>
<td>0.15634242</td>
</tr>
<tr>
<td></td>
<td>0.79995722</td>
<td>0.00573536</td>
<td>0.21554132</td>
</tr>
<tr>
<td>Third Geometry</td>
<td>0.79995722</td>
<td>0.00168484</td>
<td>0.16544698</td>
</tr>
<tr>
<td></td>
<td>0.79999381</td>
<td>0.00505927</td>
<td>0.21514075</td>
</tr>
<tr>
<td>Final Geometry</td>
<td>0.79999381</td>
<td>0.00164774</td>
<td>0.16935484</td>
</tr>
<tr>
<td></td>
<td>0.79996145</td>
<td>0.00468347</td>
<td>0.21432692</td>
</tr>
</tbody>
</table>

Table 3.4: Additional results for the Nash (MO) airfoil optimization

<table>
<thead>
<tr>
<th>Shape</th>
<th>Angle of Attack</th>
<th>$C_D$</th>
<th>$C_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final Geometry</td>
<td>0.50000000</td>
<td>0.00324682</td>
<td>0.13413532</td>
</tr>
</tbody>
</table>

Table 3.5: Drag and Lift for the Nash (MO) geometry


3.3.3 Stochastic Nash Equilibrium

Algorithm

We shall assume that \( M \) is a random variable and apply the EMV an MEV strategy for the stochastic Nash equilibrium, in the same way it was applied in the Subsection 3.2.3

**ALGORITHM-Stochastic EMV Nash Equilibrium:**

1. Initialize the deterministic parameters, put \( r = 1 \).

2. Sample a value of the random variable \( M_r \), according to its probability distribution function \( f(M) \), and define the initial shape and angle of attack \( (\theta^0, \alpha^0) \in \Theta \times A \), and let \( k = 0 \).

3. Find an optimal shape \( \theta^{k+1} = \arg \min_{\theta \in \Theta} C_D(\theta, \alpha^k, M_r) \), according to the procedure presented in Appendix A.

4. Find an optimal angle of attack \( \alpha^{k+1} = \arg \min_{\alpha \in A} (-C_L(\theta^k, \alpha, M_r)) \), according to the procedure presented in Appendix B.

5. Test for convergence or if a number of specified iterations \( \text{NITER} \) have been performed:

   If YES: we have the solution \( (\bar{\theta}^r, \bar{\alpha}^r) = (\theta^{k+1}, \alpha^{k+1}) \);

   If NO: put \( k = k + 1 \) and go to the third step.

6. If \( r < \text{NSAMP} \) then put \( r = r + 1 \) and go to the second step.

7. Average all the computed Nash optima

\[
\begin{align*}
\theta^{EMV} &= \frac{1}{\text{NSAMP}} \sum_{r=1}^{\text{NSAMP}} \bar{\theta}^r \\
\alpha^{EMV} &= \frac{1}{\text{NSAMP}} \sum_{r=1}^{\text{NSAMP}} \bar{\alpha}^r
\end{align*}
\]

For the MEV algorithm, the same First Mean Theorem is employed in order to deal with the expectation.
ALGORITHM-Stochastic MEV Nash Equilibrium:

1. Initialize the deterministic parameters, and define the initial shape and angle of attack \((\theta^0, \alpha^0) \in \Theta \times A\), and let \(k = 0\).

2. Independently sample all the values of the random variable \(M_r, r = 1, ..., \text{NSAMP}\), according to its probability distribution function \(f(M)\).

3. Identify the corresponding \(M^D_k, M^L_k\) such that

\[
\begin{align*}
C_D (\theta^k, \alpha^k, M^D_k) &= \frac{1}{\text{NSAMP}} \sum_{r=1}^{\text{NSAMP}} C_D (\theta^k, \alpha^k, M_r) \\
C_L (\theta^k, \alpha^k, M^L_k) &= \frac{1}{\text{NSAMP}} \sum_{r=1}^{\text{NSAMP}} C_L (\theta^k, \alpha^k, M_r)
\end{align*}
\]

4. Find an optimal shape \(\theta^{k+1} = \arg\min_{\theta \in \Theta} C_D (\theta, \alpha^k, M^D_k)\), according to the procedure presented in Appendix A.

5. Find an optimal angle of attack \(\alpha^{k+1} = \arg\min_{\alpha \in A} (-C_L (\theta^k, \alpha, M^L_k))\), according to the procedure presented in Appendix B.

6. Test for convergence or if a number of specified iterations \(\text{NITER}\) have been performed:

   If YES: we have the solution \((\theta^{MEV}, \alpha^{MEV}) = (\theta^{k+1}, \alpha^{k+1})\);

   If NO: put \(k = k + 1\) and go to the third step.

Results

The same random variables and data as in Subsection 2.4 have been used for the evaluation of the EMV/MEV stochastic Nash equilibrium (\(\text{NITER} = 4, \text{NSAMP} = 1000\)).

The EMV and MEV Nash equilibrium strategies are given in Table 3.6 and Figures 3.52, 3.56, 3.60, 3.64, 3.68, 3.72. Substantial changes have been made to the airfoil shape and angle of attack.

The drag coefficient has been minimized and the lift coefficient has been maximized, no matter what the type of the randomness or the algorithm (Figures 3.49, 3.50, 3.53, 3.54, 3.61, 3.62, 3.65, 3.66). The pressure plots in Figures 3.51, 3.55, 3.63, 3.67 justify these
results, namely a reduction in the drag coefficient usually corresponds to a reduction of the shock wave on the upper surface, and an increase in the lift coefficient to an increase of the difference between the pressure distribution over the upper surface and the bottom surface.

From the viewpoint of effectiveness, Figures 3.57, 3.58, 3.69, 3.70 (and 3.59, 3.71), and Table 3.7 suggest that the EMV strategy is more efficient from the cumulative $P_<$ point of view, while the MEV is more efficient from the CPU time point of view.
Figure 3.49: (MO) Drag Coefficient at EMV– $M \sim U(0.7, 0.8)$ & initial point
Figure 3.50: (MO) Lift Coefficient at EMV\( \sim M \sim U(0.7, 0.8) \) & initial point
Figure 3.51: (MO) Pressure (for EMV–M ∼ U(0.7, 0.8) & initial point) at M = 0.7907923
Figure 3.52: (MO) Geometry (EMV–M $\sim U(0.7, 0.8)$ & initial)
Figure 3.53: (MO) Drag Coefficient at MEV– $M \sim U(0.7, 0.8)$ & initial point
Figure 3.54: (MO) Lift Coefficient at MEV~ $M \sim U(0.7, 0.8)$ & initial point
Figure 3.55: (MO) Pressure (for MEV–M ∼ U(0.7, 0.8) & initial point) at M = 0.7907923
Figure 3.56: (MO) Geometry (MEV–M ~ U(0.7, 0.8) & initial)
Figure 3.57: (MO) Drag Coefficient at EMV & MEV point—$M \sim U(0.7, 0.8)$
Figure 3.58: (MO) Lift Coefficient at EMV & MEV point—$M \sim U(0.7, 0.8)$
Figure 3.59: (MO) Pressure (for EMV & MEV point $-M \sim U(0.7, 0.8)$) at $M = 0.7907923$
Figure 3.60: (MO) Geometry (EMV & MEV – $M \sim U(0.7, 0.8)$)
Figure 3.61: (MO) Drag Coefficient at EMV—$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial point
Figure 3.62: (MO) Lift Coefficient at EMV—$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial point
Figure 3.63: (MO) Pressure (for EMV–M $\sim \mathcal{N}(0.75, 0.025^2)$ & initial) at $M = 0.7540006$
Figure 3.64: (MO) Geometry (EMV–M \sim N(0.75, 0.025^2) & initial)
Figure 3.65: (MO) Drag Coefficient at MEV – $M \sim N(0.75, 0.025^2)$ & initial point
Figure 3.66: (MO) Lift Coefficient at MEV—$M \sim N(0.75, 0.025^2)$ & initial point
Figure 3.67: (MO) Pressure (for MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial) at $M = 0.7540006$
Figure 3.68: (MO) Geometry (MEV–$M \sim \mathcal{N}(0.75, 0.025^2)$ & initial)
Figure 3.69: (MO) Drag Coefficient at EMV & MEV point $\sim M \sim \mathcal{N}(0.75, 0.025^2)$
Figure 3.70: (MO) Lift Coefficient at EMV & MEV point—$M \sim \mathcal{N}(0.75, 0.025^2)$
Figure 3.71: (MO) Pressure (for EMV & MEV \( M \sim \mathcal{N}(0.75, 0.025^2) \)) at \( M = 0.7540006 \)
Figure 3.72: (MO) Geometry (EMV & MEV $-M \sim \mathcal{N}(0.75, 0.025^2)$)
Table 3.6: Angle of attack (MO) airfoil optimization

<table>
<thead>
<tr>
<th></th>
<th>$\alpha_{\text{initial}}$</th>
<th>$\alpha_{\text{final}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M \sim U (0.7, 0.8)$</td>
<td>(EMV) 0.5000000000 (MEV) 0.5000000000</td>
<td>(EMV) 0.7999532 (MEV) 0.7999837</td>
</tr>
<tr>
<td>$M \sim N (0.75, 0.025^2)$</td>
<td>(EMV) 0.5000000000 (MEV) 0.5000000000</td>
<td>(EMV) 0.7999550 (MEV) 0.7999899</td>
</tr>
</tbody>
</table>

Table 3.7: Comparison of EMV and MEV approach for (MO) airfoil optimization

<table>
<thead>
<tr>
<th></th>
<th>$CPU ; \text{time (seconds)}$</th>
<th>$P_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M \sim U (0.7, 0.8)$</td>
<td>(EMV) 107783.59 (MEV) 620.58</td>
<td>$(C_D)$ 39% $(C_L)$ 100%</td>
</tr>
<tr>
<td>$M \sim N (0.75, 0.025^2)$</td>
<td>(EMV) 106298.13 (MEV) 622.15</td>
<td>$(C_D)$ 33.83% $(C_L)$ 100%</td>
</tr>
</tbody>
</table>
CHAPTER 4

CONCLUSIONS AND FUTURE WORK

Most applied problems with inherent uncertainty require finding an optimum point that is, in general, highly dependent on the randomness, and makes the optimization process quite difficult. Therefore, it is natural to place it under the “expected value” perspective. But when the operators of expectation and optimization do not commute, it raises the choice between the EMV and MEV criterion.

Some theoretical results have been proved (more generalized theorems being subject of further research), and a variety of stochastic optimization problems have been solved using the EMV and MEV approach, ranging from a free un-damped vibration to an airfoil shape optimization; from a single-objective optimization to a multi-objective optimization. The EMV, as well as the MEV, has been shown to be viable, either for small problems or large scale problems, for one or more cost functions.

To cope with large problems, a Monte-Carlo type of approach was developed to implement the EMV/MEV strategy. An important issue, which was not addressed in the present thesis, is the possibility of computing a confidence interval on the approximation, as well as the accuracy of the approximation. These would be excellent tools with which to assess the performance of EMV and MEV and topics for future work.

On the other hand, it has been found that the EMV approach is a useful way to formulate the stochastic problem. It is easy to implement, and has a better outcome in some cases. In contrast, the MEV procedure is not that straightforward to implement, but is much more time efficient. Moreover, for small variability in the randomness, the two approaches do not differ much.

From another perspective, a stochastic EMV and MEV Nash equilibrium has been defined
and included in the study as well. In particular, the effect of EMV and MEV approach in the airfoil shape optimization is discussed. Several problems remain to be solved. Among these are the specification of the confidence intervals and the closely related problems of defining more realistic non-cooperative games. In any event, our preliminary Monte-Carlo results indicate that for various distributions a better design can be obtained using the EMV procedure. Unfortunately, the advantages of using the EMV approach are far outweighed by the prohibitive computational cost.

Finally, there are advantages and shortcomings associated with each approach, we just wanted to point out a few of them. One must compare the exorbitant computational cost to find an EMV optimal design with the benefit of having a lower risk. Most of the real-life problems are non-linear and non-symmetric in nature, therefore under favorable circumstances the EMV criteria can outperform the classical MEV approach, and the use of either one is a decision-maker choice.
APPENDIX A

A DETERMINISTIC APPROACH TO AIRFOIL SHAPE OPTIMIZATION

In general, the progress of an optimization process is measured in terms of a cost or objective functional $I$, which in aerodynamic shape design could be, for instance, the drag coefficient, the lift coefficient, lift to drag ratio, the difference from a specified pressure distribution, etc. The airfoil shape is parametrized with a set of design variables.

According to Jameson [56], for flow about an airfoil, the aerodynamic properties that define the cost function are not only functions of the physical location of the boundary (the shape of the airfoil), which can be represented by a function $F$, but also of the flow field variables $w$. The objective function is

$$I = I(w, F)$$

and a change in $F$ results in a change in the objective function given by

$$\delta I = \frac{\partial I^T}{\partial w} \delta w + \frac{\partial I^T}{\partial F} \delta F. \quad (A.1)$$

Using a control theory approach, which has dramatic computational advantages over a finite difference method, the governing equations of the flow field are introduced as a constraint in such a way that the final expression for the gradient does not require reevaluation of the flow field. To achieve this, $\delta w$ must be eliminated from (A.1). Introduce the governing equation,

$$R(w, F) = 0. \quad (A.2)$$

which expresses the dependence of $w$ and $F$ within the flow field domain $D$.

We will use the compressible Euler equations as the mathematical model of the flow. Before we state them, let us denote the Cartesian coordinates and velocity components by
$x_1, x_2, x_3$ and $u_1, u_2, u_3$, and use the summation convention. Then, the three-dimensional Euler equations can be written as

$$\frac{\partial w}{\partial t} + \frac{\partial f_i}{\partial x_i} = 0, \text{ in } D \tag{A.3}$$

where

$$w = \begin{cases} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{cases}, \quad f_i = \begin{cases} \rho u_i \\ \rho u_1 u_i + p \delta_{i1} \\ \rho u_2 u_i + p \delta_{i2} \\ \rho u_3 u_i + p \delta_{i3} \\ \rho u_i H \end{cases}. \tag{A.4}$$

Also,

$$p = (\gamma - 1) \rho \left\{ E - \frac{1}{2} \left( u_i^2 \right) \right\} \tag{A.5}$$

and

$$\rho H = \rho E + p. \tag{A.6}$$

Aerodynamic design calculations using the Navier-Stokes equations as model of the fluid flow can also be carried out, see the work of Kim, Alonso, and Jameson [61, 60, 56].

Because we would like to eliminate $\delta w$ from the equation (A.1), we will determine it from the equation

$$\delta R = \left[ \frac{\partial R}{\partial w} \right] \delta w + \left[ \frac{\partial R}{\partial \mathcal{F}} \right] \delta \mathcal{F} = 0. \tag{A.7}$$

Next, by introducing a Lagrange multiplier $\psi$ and combining the equations (A.1) and (A.7), it follows that

$$\delta I = \frac{\partial I^T}{\partial w} \delta w + \frac{\partial I^T}{\partial \mathcal{F}} \delta \mathcal{F} - \psi^T \left( \left[ \frac{\partial R}{\partial w} \right] \delta w + \left[ \frac{\partial R}{\partial \mathcal{F}} \right] \delta \mathcal{F} \right)$$

$$= \left\{ \frac{\partial I^T}{\partial w} - \psi^T \left[ \frac{\partial R}{\partial w} \right] \right\} \delta w + \left\{ \frac{\partial I^T}{\partial \mathcal{F}} - \psi^T \left[ \frac{\partial R}{\partial \mathcal{F}} \right] \right\} \delta \mathcal{F}. \tag{A.8}$$

The expression of the $\delta I$ can be simplified if we choose $\psi$ to satisfy the so called adjoint equation

$$\left[ \frac{\partial R}{\partial w} \right]^T \psi = \frac{\partial I}{\partial w} \tag{A.8}$$

and we find that

$$\delta I = \left\{ \frac{\partial I^T}{\partial \mathcal{F}} - \psi^T \left[ \frac{\partial R}{\partial \mathcal{F}} \right] \right\} \delta \mathcal{F} = G \delta \mathcal{F}. \tag{A.9}$$

The main advantage is that (A.9) is independent of $\delta w$, with the result that the gradient of the cost function $I$ with respect to an arbitrary number of design variables is determined.

168
without the need for additional flow field evaluations. Because in our case equation (A.2) is a
partial differential equation, the adjoint equation (A.8) is also a partial differential equation,
and appropriate boundary conditions must be determined.

Therefore, the design procedure can be described as follows.

1. Initialize the deterministic parameters involved in the optimization procedure, param-
eterize the configuration of interest using a set of design variables, and define the initial
shape;

2. Solve the flow equations for the flow variables $\rho, u_1, u_2, u_3, p$;

3. Solve the adjoint equations for the co-state variables $\psi$ subject to appropriate boundary
conditions, which will usually depend on the form of the cost function $I$;

4. Evaluate the gradients $G$ and update the aerodynamic shape based on the direction of
steepest descent (for instance);

5. Return to step #2 until an optimum configuration is attained.

In our study, the aerodynamic cost function is chosen to be the drag coefficient.
APPENDIX B

AN APPROXIMATION TO THE MINIMUM OF A FUNCTION

The function program DFMN or FMIN from NETLIB is a slightly modified version of the ALGOL60 procedure "localmin" given in [17]. The algorithm finds an approximation to the minimum of a function $f$ defined on an interval $[a, b]$. Unless $a$ is very close to $b$, $f$ is never evaluated at the endpoints $a$ and $b$, so $f$ needs only to be defined on $(a, b)$. If the minimum is actually at $a$ or $b$ then an interior point distant no more than $2tol$ from $a$ or $b$ will be returned, where $tol$ is a tolerance. The minimum found may be local, but not-global.

At a typical step there are six significant points $a, b, u, v, w, x$, not all distinct. The positions of these points change during the algorithm. Initially $(a, b)$ is the interval on which $f$ is defined, and

$$v = w = x = a + \left( \frac{3 - \sqrt{5}}{2} \right)(b - a)$$

The magic number $\left( \frac{3 - \sqrt{5}}{2} \right) = 0.381966$ is chosen arbitrarily so that the first step is the same as for a golden section search.

At the start of a cycle the points $a, b, u, v, w, x$ always serve as follows: a local minimum lies in $[a, b]$; of all the points at which $f$ has been evaluated, $x$ is the one with the least value of $f$, or the point of the most recent evaluation if there is a tie; $w$ is the point with the next lowest value of $f$; $v$ is the previous value of $w$; and $u$ is the last point at which $f$ has been evaluated (undefined the first time).

Let $m = \frac{1}{2}(a + b)$ be the midpoint of the interval known to contain the minimum. If

$$|x - m| \leq 2tol - \frac{1}{2}(b - a),$$

170
then the procedure terminates with $x$ as the approximate position of the minimum. Otherwise, numbers $p$ and $q$ ($q \geq 0$) are computed so that $x + \frac{p}{q}$ is the turning point of the parabola passing through $(v, f(v))$, $(w, f(w))$, and $(x, f(x))$. If two or more of these points coincide, or if the parabola degenerates to a straight line, then $q = 0$. The numbers $p$ and $q$ are given by

$$p = \pm [(x - v)^2 (f(x) - f(w)) - (x - w)^2 (f(x) - f(v))] \quad (B.1)$$

$$q = \mp 2 [(x - v) (f(x) - f(w)) - (x - w) (f(x) - f(v))] \quad (B.2)$$

From (B.1) and (B.2), the correction $\frac{p}{q}$ should be small if $x$ is close to a minimum where the second derivative is positive, so the effect of rounding errors in computing $p$ and $q$ is minimized.

Let $e$ be the value of $\frac{p}{q}$ at the second-last cycle. If $|e| \leq tol, q = 0, x + \frac{p}{q} \notin (a, b)$, or $\left| \frac{p}{q} \right| \geq \frac{1}{2} |e|$, then a “golden section” step is performed, i.e. the next value of $u$ is

$$u = \begin{cases} 
\left( \frac{\sqrt{5} - 1}{2} \right) x + \left( \frac{3 - \sqrt{5}}{2} \right) a, & \text{if } x \geq m \\
\left( \frac{\sqrt{5} - 1}{2} \right) x + \left( \frac{3 - \sqrt{5}}{2} \right) b, & \text{if } x < m
\end{cases} \quad (B.3)$$

Otherwise $u$ is taken as $u = x + \frac{p}{q}$ (a ”parabolic interpolation” step), except that the distances $|u - x|, u - a,$ and $b - u$ must be at least $tol$. Then $f$ is evaluated at the new point $u$, the points $a, b, v, w, x$ are updated as necessary, and the cycle is repeated. We see that $f$ is never evaluated at two points closer together than $tol$.

The algorithm terminates in the following way: $x = b - tol$ (or, symmetrically, $a + tol$) after a parabolic interpolation step has been performed with the condition $|u - x| \geq tol$ enforced. The parabolic interpolation point lies very close to $x$ and $b$, so $u$ is forced to be $x - tol$. If $f(u) > f(x)$ then $a$ moves to $u, b - a$ becomes $2tol$, and the termination criterion is satisfied. Note that two consecutive steps of $tol$ are done just before termination. If a golden search were done whenever the last, rather than second-last, value of $\left| \frac{p}{q} \right|$ was $tol$ or less, then termination with two consecutive steps of $tol$ would be prevented, and unnecessary golden section steps would be performed.
Let us illustrate the concept of Nash equilibrium for a non-cooperative game by an example. Our goal is to determine the Nash equilibrium point for a game specified by the following objectives:

\[
\begin{align*}
    f : K_1 \times K_2 \to \mathbb{R}, & \quad f(x, y) = 2x^2 - 2xy + 5y^2 - 6x - 6y \\
g : K_1 \times K_2 \to \mathbb{R}, & \quad g(x, y) = x^2 + xy + y^2 - 3x - 6y
\end{align*}
\]

We recall that the Nash equilibrium point is defined to be a point \((a, b) \in K_1 \times K_2\) that satisfies

\[
\begin{align*}
f(a, b) & \leq f(x, b), \forall x \in K_1 \\
g(a, b) & \leq g(a, y), \forall y \in K_2
\end{align*}
\]  
(C.1)

Let us find this point \((a, b)\) on different sets.

1. In the case when \(K_1 = K_2 = \mathbb{R}\), the Nash equilibrium point is given by the solution of the following system

\[
\begin{align*}
    \frac{\partial f}{\partial x} & = 4x - 2y - 6 = 0 \\
    \frac{\partial g}{\partial y} & = x + 2y - 6 = 0
\end{align*}
\]

which imply

\[
a = \frac{12}{5}, \quad b = \frac{9}{5},
\]

so the Nash equilibrium point is \((\frac{12}{5}, \frac{9}{5})\).
2. The case when \( K_1 = [0, 2], K_2 = [1, 3] \) requires more detailed evaluation ([31]). From the definition (C.1), it can be deduced that

\[
\begin{align*}
2a^2 - 2ab + 5b^2 - 6a - 6b &\leq 2x^2 - 2xb + 5b^2 - 6x - 6b, \forall x \in [0, 2] \\
a^2 + ab + b^2 - 3a - 6b &\leq a^2 + ay + y^2 - 3a - 6y, \forall y \in [1, 3]
\end{align*}
\]

Therefore

\[
\begin{align*}
2(a - x)(a + x) - 2b(a - x) - 6(a - x) &\leq 0, \forall x \in [0, 2] \\
a(b - y) + (b - y)(b + y) - 6(b - y) &\leq 0, \forall y \in [1, 3]
\end{align*}
\]

Or

\[
\begin{align*}
(a - x)[2(a + x) - 2b - 6] &\leq 0, \forall x \in [0, 2] \\
(b - y) [(b + y) + a - 6] &\leq 0, \forall y \in [1, 3]
\end{align*}
\]

In other words

\[
\begin{align*}
(a - x)[x + (a - b - 3)] &\leq 0, \forall x \in [0, 2] \\
(b - y) [y + (a + b - 6)] &\leq 0, \forall y \in [1, 3]
\end{align*}
\]

(C.2)

Notice that the following relation is satisfied

\[
x + (a - b - 3) \leq 0, \forall x \in [0, 2], \forall a \in [0, 2], \forall b \in [1, 3].
\]

Hence, we only have to impose the condition

\[
a - x \geq 0, \forall x \in [0, 2] \text{ which gives } a = 2.
\]

Under this circumstance, the second inequality of the system (C.2) becomes

\[
(b - y) [y + (b - 4)] \leq 0, \forall y \in [1, 3].
\]

It is obvious that the condition we have to impose here is \( b = 4 - b \). This means \( b = 2 \).

As a conclusion, the Nash equilibrium point is \((2, 2)\).
REFERENCES


Ana-Maria Croicu

Ana-Maria Croicu was born on April 21st, 1971, in Cluj-Napoca, Romania. She completed her Bachelor degree in Electrical Engineering at the Technical University of Cluj-Napoca in 1994, and in Mathematics at Babes-Bolyai University of Cluj-Napoca in 1995. In 1995 she was awarded a Master of Science degree in Electrical Engineering from Technical University of Cluj-Napoca and was appointed as an Assistant Professor of Mathematics at the same university.

She enrolled in the doctoral program at Babes-Bolyai University in the fall of 1996, and, under the advisement of Professor I. Kolumban, obtained her PhD degree in Pure Mathematics in summer of 2001. In between, she also enrolled in the doctoral program in Applied Mathematics at FSU in the summer of 2000. Here at FSU she was strongly influenced by the fine research projects of Professor M.Y. Hussaini. She is expecting to receive her Ph.D. in Applied Mathematics from FSU with a dissertation on stochastic optimization with applications. Overall, Ana-Maria’s research interests include variational inequalities, Nash equilibrium, computational electromagnetism and fluid dynamics, stochastic optimization, aerodynamics. He has published 7 articles, co-authored one textbook and has a few other articles in preparation for publication.

While she was working as an Assistant Professor at Technical University of Cluj-Napoca, Ana-Maria was awarded two Tempus fellowships from the European Union at the Technical University of Graz, Austria (1997), and Katholieke Universiteit Leuven, Belgium (1999). After she joined Professor M.Y. Hussaini research group in 2000, she was awarded a Research fellowship (2000-2002) and a Teaching fellowship (2002-2005).