Florida International University FIU Digital Commons

FIU Electronic Theses and Dissertations

University Graduate School

6-28-2019

Many-Objective Hybrid Optimization Under Uncertainty With Applications

Sohail Reddy Florida International University, sredd001@fiu.edu

Follow this and additional works at: https://digitalcommons.fiu.edu/etd

Part of the Aerodynamics and Fluid Mechanics Commons, Other Mechanical Engineering Commons, and the Systems Engineering and Multidisciplinary Design Optimization Commons

Recommended Citation

Reddy, Sohail, "Many-Objective Hybrid Optimization Under Uncertainty With Applications" (2019). *FIU Electronic Theses and Dissertations*. 4369. https://digitalcommons.fiu.edu/etd/4369

This work is brought to you for free and open access by the University Graduate School at FIU Digital Commons. It has been accepted for inclusion in FIU Electronic Theses and Dissertations by an authorized administrator of FIU Digital Commons. For more information, please contact dcc@fiu.edu.

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

MANY-OBJECTIVE HYBRID OPTIMIZATION UNDER UNCERTAINTY WITH APPLICATIONS

A dissertation submitted in partial fulfillment of the

requirements for the degree of

DOCTOR OF PHILOSOPHY

 in

MECHANICAL ENGINEERING

by

Sohail R. Reddy

2019

To: Dean John Volakis College of Engineering and Computing

This dissertation, written by Sohail R. Reddy, and entitled Many-Objective Hybrid Optimization Under Uncertainty With Applications, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this dissertation and recommend that it be approved.

M. Ali Ebadian Dwayne McDaniel Sakhrat Khizroev Marcelo J. Colaco Kalaynmoy Deb Carlo Poloni George S. Dulikravich, Major Professor Date of Defense: June 28, 2019

The dissertation of Sohail R. Reddy is approved.

Dean John Volakis College of Engineering and Computing

Andres Gil Vice President for Research and Economic Development and Dean of the University Graduate School

Florida International University, 2019

© Copyright 2019 by Sohail R. Reddy All rights reserved.

DEDICATION

Dedicated to my family for their constant support and sacrifice, without them this work would not have been possible.

ACKNOWLEDGMENTS

First and foremost, I would like to thank Professor George Dulikravich for his guidance and continuous support during the coarse of my studies. Thank you for providing me with the freedom, opportunity and resources to work on several interesting projects. I appreciate all his contributions of time and ideas to make my Ph.D. experience productive and stimulating. His excitement and enthusiasm for research was contagious and motivational for me.

I would also like to thank my committee members: Prof. Ali Ebadian, Prof. Dwayne McDaniel, Prof. Sakhrat Khizroev, Prof. Marcelo J. Colaco, Prof. Kalaynmoy Deb and Prof. Carlo Poloni. Thank you for the fruitful discussions, encouragement and insightful comments, that helped widen my research from various perspective.

I am also indebted to Dr. Garret Vanderplaats and Prof. Daniel Watzenig for allowing me to perform research with them. Thank you for hosting me at your company; it not only gave me invaluable experience but also great friends and memories to cherish.

I would also like to thank my friends: Anthony Abrahao, Janhavi Chitale, Mackenson Telusma and Matthias Scharrer for all the fun we have had and memories we made in the last five years.

I am also extremely grateful to FIU University Graduate School for the FIU Presidential Fellowship and the FIU Dissertation Year Fellowship.

ABSTRACT OF THE DISSERTATION MANY-OBJECTIVE HYBRID OPTIMIZATION UNDER UNCERTAINTY WITH APPLICATIONS

by

Sohail R. Reddy

Florida International University, 2019

Miami, Florida

Professor George S. Dulikravich, Major Professor

A novel method for solving many-objective optimization problems under uncertainty was developed. Many-objective optimization typically refers to problems with more than three objectives. It is well known that no single optimization algorithm performs best for all problems. Therefore, the developed method, a many-objective hybrid optimizer (MOHO), uses five constitutive algorithms and actively switches between them throughout the optimization process allowing for robust optimization. MOHO monitors the progress made by each of the five algorithms and allows the best performing algorithm more attempts at finding the optimum. This removes the need for user input for selecting algorithm as the best performing algorithm is automatically selected thereby increasing the probability of converging to the optimum. An uncertainty quantification framework, based on sparse polynomial chaos expansion, to propagate the uncertainties in the input parameter to the objective functions was also developed and verified. Where the samples and analysis runs needed for standard polynomial chaos expansion increases exponentially with the dimensionality, the presented sparse polynomial chaos approach efficiently propagates the uncertainty with only a few samples, thereby greatly reducing the computational cost. The performance of MOHO was investigated on a total of 65 analytical test problems from the DTLZ and WFG test suite, for which the analytical solution is known. It was also shown that MOHO is capable of solving single-objective problems. The Single-Objective Hybrid Optimizer (SOHO) was used to solve the parameter identification problem in the electrochemical model of a Lithium-Ion (Li⁺) battery. The complex non-linear electrochemical model adds additional difficulty due to its multidisciplinary nature and is, therefore, a good test case. The SOHO algorithm decreased the time required to estimate the parameters from three weeks to less than one day. MOHO is also applied to two additional real-life cases of aerodynamic shape design of subsonic and hypersonic bodies. Aerodynamic shape optimization is often computationally expensive and is, therefore, a good test case to investigate MOHO's ability to reduce the computational time through robust optimization and accelerated convergence. The subsonic design optimization had three objectives: maximize lift and minimize drag and moment. The hypersonic design optimization had two objectives: maximize volume and minimize drag. Two accelerated solvers based on fast multipole method and Newton impact theory are developed for simulating The results show that MOHO performed, on subsonic and hypersonic flows. average, better than all five remaining algorithms in 52% of the DTLZ+WFG problems. The results of robust optimization of a subsonic body and hypersonic bodies were in good agreement with theory. The MOHO developed is capable of solving many-objective, multi-objective and single objective, constrained and unconstrained optimization problems with and without uncertainty with little user input.

TABLE OF CONTENTS

| CHAPTER | PAGE |
|--|------|
| 1. INTRODUCTION | . 1 |
| $1.1 \text{Motivation} \dots \dots$ | . 2 |
| 1.2 Objective of the Research | . 8 |
| 1.3 Organization of the Dissertation | . 10 |
| 2. OPTIMIZATION | . 11 |
| 2.1 Multi-Objective Optimization | . 12 |
| 2.2 Many Objective Optimization (MOO) | . 14 |
| 2.3 Probabilistic Optimization | . 16 |
| 2.4 Many Objective Hybrid Optimization (MOHO) | . 18 |
| 2.4.1 Constitutive Algorithms | . 19 |
| 2.4.2 Switching Rules | . 22 |
| 2.5 Benchmarking MOHO | . 24 |
| 2.5.1 Test Problems for Many-Objective Optimization | . 24 |
| 2.5.2 Indicator of Performance | . 26 |
| 2.5.3 Algorithm Parameter Settings | . 27 |
| 2.5.4 Performance of MOHO on Unconstrained Test Problems | . 29 |
| 2.5.5 Performance of MOHO on Constrained Test Problems | . 39 |
| 2.0 Optimization. Summary | . 44 |
| 3. UNCERTAINTY QUANTIFICATION | . 45 |
| 3.1 Monte Carlo Sampling | . 46 |
| 3.2 Polynomial Chaos Expansion | . 47 |
| 3.3 Uncertainty Quantification: Summary | . 49 |
| 4. NUMERICAL MODELING | . 51 |
| 4.1 Geometry Parameterization and Deformation | . 51 |
| 4.2 Electrochemical Model of Doyle-Fuller-Newman | . 53 |
| 4.3 Fast Multipole Method for Potential Flow Problems | . 56 |
| 4.3.1 The Fast Multipole Method | . 59 |
| 4.3.2 Verification of Potential Flow Solver | . 64 |
| 4.4 Modified Newton Impact Theory for Hypersonic Flow | . 67 |
| 4.4.1 Validation of Modified Newton Impact Theory Solver | . 70 |
| 4.5 Numerical Modeling: Summary | . 73 |
| 5. PARAMETER ESTIMATION IN BATTERY MODEL | . 74 |
| 5.1 Problem Definition of Inverse Parameter Estimation | . 75 |
| 5.2 Parameter Estimation in DFN Model | . 78 |
| 5.3 Parameter Estimation in Battery Model: Summary | . 81 |

| 6. | OPTIMIZATION OF SUBSONIC BODIES |
|-----|---|
| 6.1 | Problem Definition of Subsonic Aerodynamic Shape Design |
| 6.2 | Deterministic Optimization of Subsonic Bodies |
| 6.3 | Optimization of Subsonic Bodies Under Uncertainty |
| 6.4 | Optimization of Subsonic Bodies: Summary |
| 7. | OPTIMIZATION OF HYPERSONIC BODIES |
| 7.1 | Problem Definition of Hypersonic Shape Design |
| 7.2 | Deterministic Optimization of Hypersonic Bodies |
| 7.3 | Optimization of Hypersonic Bodies Under Uncertainty |
| 7.4 | Optimization of Hypersonic Bodies: Summary |
| 8. | CONCLUSION |
| 8.1 | Summary of Thesis |
| 8.2 | Future Work |
| RE | FERENCES |
| AP | PENDICES |
| VI | ΓΑ |

LIST OF TABLES

| TABLE PAGE | | |
|---|---|--|
| 2.1 Properties of DTLZ and WFG test problems used to validate the optimization algorithms | 5 | |
| 2.2 The reference point used in the computation of Hypervolume for each problem | 7 | |
| 2.3 Population size and generations for each test problem | 3 | |
| 3.1 Polynomials of the Wiener-Askey scheme for some random variables 47 | 7 | |
| 5.1 Error statistics of the three selected cases in parameter identification 80 |) | |
| 6.1 Objective functions values of the baseline aircraft and the two optimized designs obtained within the deterministic framework | 3 | |
| 6.2 Mean and standard deviation of the aerodynamic coefficients for the subsonic designs obtained within the deterministic framework 92 | 2 | |
| 6.3 Mean and standard deviation of the aerodynamic coefficients for the subsonic designs obtained within the probabilistic framework 96 | 3 | |
| 7.1 Objective functions for the starting shape and the two optimized hypersonic designs obtained within the deterministic framework 104 | ł | |
| 7.2 Mean and standard deviation of the objective function, under control point uncertainty of $\sigma = 0.05m$, for the hypersonic designs obtained within the deterministic framework |) | |
| 7.3 Mean and standard deviation of the objective function, under control point uncertainty of $\sigma = 0.1m$, for the hypersonic designs obtained within the deterministic framework |) | |
| 7.4 Mean and standard deviation of the objective function, under varying degrees of control point uncertainty, for the hypersonic designs obtained within the probabilistic framework | 2 | |
| A.1 Mean and standard deviation of the IGD obtained by each algorithm for the DTLZ problems. Best performance is coloured in red 131 | | |
| A.2 Mean and standard deviation of the hypervolume obtained by each algorithm for the WFG problems. Best performance is coloured in red132 | 2 | |

LIST OF FIGURES

| FIGURE PAGE |
|---|
| 2.1 Representation of the non-domination criteria used in multi-objective optimization |
| 2.2 Percent of population that is non-dominated at each generation for varying number of objectives |
| 2.3 Representation of reference point based niching used to preserve diversity in many-objective optimization |
| 2.4Convergence histories of the newly developed algorithms on: a) DTLZ1, b) DTLZ2 and c) DTLZ4DTLZ421 |
| 2.5 The Pareto front for a three objective DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.6 Parallel coordinates plot showing the Pareto front for a 10 objective DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.7 Parallel coordinates plot showing the Pareto front for a 10 objective DTLZ3 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.8 Parallel coordinates plot showing the Pareto front for a 15 objective DTLZ4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.9 The Pareto front for a three objective WFG4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.10 Parallel coordinates plot showing the Pareto front for a 15 objective WFG5 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.11 Parallel coordinates plot showing the Pareto front for a 15 objective WFG6 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.12 The Pareto front for a three objective C1-DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |
| 2.13 The Pareto front for a three objective C1-DTLZ3 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO |

| 2.14 | The Pareto front for a three objective C3-DTLZ4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO | 43 |
|------|---|----|
| 4.1 | Surface boundary representation of: a) undeformed lattice and object and b) deformed lattice and the corresponding deformed object | 52 |
| 4.2 | Surface boundary representation of undeformed NASA 25D aircraft (blue) with: a) deformed wing (green) and b) deformed underbody (green) | 53 |
| 4.3 | Schematic view of the battery cell sub-domains: Ω_l is the electrolyte, Ω_a and Ω_c are the anode (negative) and cathode (positive) electrode areas, $\Omega_{s,a}$ and $\Omega_{s,c}$ are the electrodes' particles $\ldots \ldots \ldots \ldots$. | 54 |
| 4.4 | An example of: a) quadtree decomposition and b) sample hierarchical tree structure | 64 |
| 4.5 | Verification of potential flow solver on a) an ellipsoid, subject to b) normal gradient boudary condition, showing c) FMM solution and d) the analytical solution | 66 |
| 4.6 | Illustration of a) octree decomposition of a sphere and b) computing time required to solve the potential flow by different methods | 67 |
| 4.7 | Haack-Sears body showing the frontal and "shadow" region | 70 |
| 4.8 | Validation of modified Newton impact theory solver on a) triangulated sphere, showing b) experimentally and numerically computed coefficient of drag | 71 |
| 4.9 | Illustration of a) Hypersonic half-sphere-cone waverider and its experimentally and numerically obtained: b) coefficient of lift and c) coefficient of drag | 72 |
| 5.1 | The cycling of the Li ⁺ battery showing: a) the applied current and b) the measured voltage | 76 |
| 5.2 | Parameter estimation of the DFN model showing: a) the convergence history of the SOHO algorithm, b) error probability distribution of Case 1, c) error probability distribution of Case 2, d) error probability distribution of Case 3 | 79 |
| 5.3 | Measured and estimated voltage response obtained using the converged DFN model | 80 |
| 6.1 | 3-D view of the baseline aircraft geometry with: a) the FFD lattice used, b) sample deformations on the upper surface of the wing and c) sample deformations on the lower surface of the wing | 84 |

| 6.2 | Pareto front, obtained by each algorithm, within the deterministic framework, for the subsonic shape optimization problem showing the a) objective function space and b) normalized objective function space |
|-----|--|
| 6.3 | Surface distribution of the deformation for design obtained using: a) the potential flow solver and b) the surrogate model; the surface pressure coefficient for: c) design obtained using the potential flow solver and d) design obtained using the surrogate model; the surface pressure coefficient difference between the baseline and: e) design obtained using the potential flow solver and f) design obtained using the surrogate model |
| 6.4 | The probability density functions, obtained using different uncertainty quantification methods, of baseline configuration for : a) C_l , b) C_d and c) C_m ; configuration obtained using potential flow solver for: d) C_l , e) C_d and f) C_m ; configuration obtained using the surrogate for: d) C_l , e) C_d and f) C_m ; configuration obtained using the surrogate for: d) C_l , e) C_d and f) C_m |
| 6.5 | Pareto front obtained by each algorithm, within the probabilistic framework, for the subsonic shape optimization problem using: a) 100 population members and b) 20 population members 93 |
| 6.6 | Surface distribution of the deformation for: a) low drag design, b) high lift design and c) an optimal trade-off design; the surface pressure coefficient for: d) low drag design, e) high lift design and f) an optimal trade-off design; the surface pressure coefficient difference between the baseline and: g) low drag design, h) high lift design and i) an optimal trade-off design |
| 6.7 | Probability density functions of aerodynamic coefficients for each design, obtained within the probabilistic framework: a) C_l , b) C_d and c) C_m 97 |
| 7.1 | The surface mesh and lattice points for the cylinder used as the starting point for hypersonic shape optimization |
| 7.2 | Pareto front, obtained by each algorithm, within the deterministic framework, for the hypersonic shape optimization problem showing the: a) objective function space and b) normalized objective function space |
| 7.3 | Pareto front obtained by MOHO at different number of generations 103 |
| 7.4 | Pareto designs, obtained within the deterministic framework by MOHO, selected from the: a) minimum drag region and b) optimal region 104 |
| 7.5 | The probability density functions, obtained using different uncertainty quantification methods, of the coefficient of drag for: a) starting cylinder, b) low drag design and c) optimal trade-off design; volume for: d) starting cylinder, e) low drag design and f) optimal trade-off design |

- 7.6 The probability density functions, due to varying degree of uncertainty in the control point, of the coefficient of drag for: a) starting cylinder,
 b) low drag design and c) optimal trade-off design; volume for: d) starting cylinder, e) low drag design and f) optimal trade-off design . 108
- 7.8 Pareto front, obtained by each algorithm, within the probabilistic framework, for the hypersonic shape optimization problem under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$. . 111
- 7.10 Probability density functions for each design, obtained within the probabilistic framework, of coefficient of drag under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$; volume under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$. 113

CHAPTER 1

INTRODUCTION

Before the advent of computers, engineering analysis design had been restricted to analytical methods, experimentation, intuition, trial-and-error and experience. Iterative design stages of past decades required constant design updates and experimentation, often becoming very expensive. Modern development in computers and mathematical models has made designs safer, more efficient and less expensive. With increasing computing power becoming less expensive and more accessible, the time and resources to go from concept to production has drastically decreased.

Although sophisticated analysis methods, such as finite elements and finite volume, have made it possible to predict the performance of the model, the optimization methodologies have made it possible to deterministically update the design to reach a particular goal. Numerical optimization typically refers to the identification of some model parameters or design variables that maximize or minimize a particular response.

The original field of optimization started in the times of Fermat, who used calculus-based methods to find the optimum, and Newton, who invented iterative methods to search for the optimum. Since then, the field has matured and methods have been developed to perform optimization of multiple objectives, under enforced constraints while accounting for uncertainty. However, there exist several methods developed to solve specific problems, and very few that can be generalized to solve a large assortment of problems. This work aims to address this scarcity.

In this work a novel algorithm is developed that is able to cope with several objectives, constraints, and model uncertainty with limited input from the user.

The algorithm incorporates several existing and new optimization techniques and learns about the characteristics of the problem being solved. It automatically switches back-and-forth between algorithms, allowing the most effective algorithm more attempts at finding a better optimum design. The performance of the developed algorithm is investigated on analytical test cases as well as three real-life test problems of inverse parameter identification, subsonic aerodynamic shape optimization and hypersonic aerodynamic shape optimization.

1.1 Motivation

The field of numerical optimization is mature yet still a productive area of research. Several algorithms, capable of solving difficult problems more efficiently, continue to be published. Academic journals such as the IEEE Transactions on Evolutionary Computation have seen an increased influx of works in algorithm development.

Optimization algorithms can typically be separated into two categories: gradient based and non-gradient based. Gradient based algorithms such as Sequential Quadratic Programming (SQP) and Broyden-Fletcher-Goldfarb-Shanno (BFGS) use the gradient of the objective functions to update the search direction. Non-gradient based algorithms, sometimes referred to as evolutionary and heuristics algorithms, use random search processes to update the search.

Several evolutionary and heuristic algorithms exist that are capable of solving complex optimization problems. Algorithms such as the genetic algorithm (GA), particle swarm (PS), dragonfly algorithm (DA) and differential evolution (DE) use a population based approach to navigate the design and objective function space. There is, however, a scarcity in high level hybridization of multiple algorithms into one. It is well known that no single optimization algorithm performs best for all conceivable problems. Since the objective function topology is not known in advance, selecting an appropriate optimization algorithm can be difficult, if not impossible. One approach to overcome this lack of knowledge is to employ several different optimization algorithms and actively switch between them based on their performance. This hybridization of algorithms can allow the better performing algorithm, for the current test problem, more attempts at locating the optimum. Talbi [1] defined two forms of hybridization techniques: relay and teamwork. The teamwork hybridization approach uses several algorithms simultaneously to solve a problem, whereas the relay approach actively switches back-and-forth between algorithms based on some switching criteria. This study develops a high level relay-type hybrid optimization algorithm capable of solving different types of optimization problems.

One of the early examples of high-level relay type hybrid algorithm is the work of Lin et al. [2], which combine the GA and simulated annealing (SA) algorithms. The optimization starts with the SA algorithm and uses the GA algorithm to enrich the solutions found. In the work of Talbi et. al [3], tabu search algorithm was used to improve the population obtained by a GA. Like Lin et al. [2], Mahfoud and Goldberg [4] also used the GA to enrich the solution obtained using SA. These works, however, do not actively switch between the two algorithms. For such an approach, one must look at the work of Foster and Dulikravich [5, 6]. They used several gradient and non-gradient based algorithms with active switching between them to accelerate the convergence to the global minimum. This active switching also helped avoid premature convergence to the local minimum. Their single objective hybrid optimizer uses eight switching criteria to determine when an algorithm prematurely converges and automatically switches to

another algorithm. Poloni et al. [7] hybridized the genetic algorithm with a conjugate gradient method. Their work made use of Neural Network surrogate to analytically compute the derivatives needed by the gradient based method. The work of Emmerich et al. [8] employed a hybrid gradient descent coupled with a GA to perform multi-objective optimization. Their work first used GA to perform a global search followed by the gradient descent to perform the local search. This makes their hybridization sequential. The work of Satoru et al. 9 investigated the performance of a hybrid GA-SQP approach also in a sequential manner. The GA was first used to conduct a broad design space search followed by the SQP algorithm to refine the local search. A hybrid global and local search approach was used by Balsa-Canto et al. [10] to perform parameter estimation in dynamical Their work determined whether a local minimum was approached to systems. switch between the DE algorithm and the multiple shooting algorithm. Drawback of the sequential method is that each algorithm only gets one attempt at locating the optimum. Because an algorithm might perform better in one region of the design space than another, it might not get enough attempts in its dominant region to exhaustively perform the search.

The global-local search approach was also investigated by Kelner et al. [11] but within the multi-objective optimization framework. Their work utilized both the evolutionary and gradient based algorithms at each iteration to refine the Pareto front. The work of Moral [12] developed a multi-objective hybrid optimization algorithm by combining the SPEA [13], MOPSO [14] and NSDE [15] algorithms. It continuously switched between constitutive algorithms, thereby avoiding local Pareto fronts. Their work outlines five criteria for switching, based on the convergence and diversity of the population. These five criteria will be discussed further. Colaco et al. [16] also uses a similar strategy to switch between constitutive algorithms. Their work approximates the single objective function using radial basis functions and then performs the optimization using this This constant switching between algorithm alleviates the approximation. drawback experienced in sequential switching. The continuous switching allows each algorithm an attempt at finding an optimum in the region of current search. Once the search region changes, an algorithm that best performs on the current objective function topology can be selected. This approach is used by Dulikravich et al. [17]. The single-objective hybrid algorithm suite of Dulikravich et al. [17] uses several non-gradient based algorithms. Their work uses a search-vector concept to control the switching. The search-vector and an algorithm's population centroid are used to determine the algorithm that should be used. This approach works well for single-objective optimization problem. For the many-objective problems, the search vector is often difficult, if not impossible, to obtain.

The adoption of relay-type hybridization for solving multi-objective optimization problems is scarce. This is mainly due to the difficulty in determining appropriate switching criteria. Nonetheless, efforts have been made to improve multi-objective optimization using a relay-type hybridization. A hybrid framework was developed by Sindhya et al. [18] that combined an evolutionary algorithm with a local search algorithm. Their work utilized the NSGA-II [19] and the MOEA/D [20] algorithms to perform the global search and an SQP algorithm to perform the local search. Automatic switching is used based on a metric of diversity in the population. This approach only takes into account the diversity of the Pareto optimal solutions but not the convergence. For optimal results, both diversity and convergence should be taken into account.

Wang et al. [21] also used a global-local search approach without switching. Bosman [22] used an efficient gradient-based and evolutionary algorithm combination to develop a real-valued multi-objective hybrid optimizer. The solution of a multi-objective optimization problem using gradient based methods typically requires a weighted sum of all objectives to form a single objective. Bosman's work performs this gradient-based optimization without constructing a single objective function.

An alternate approach to hybridization is presented by Sekhar and Devi [23] where a GA-PSO hybrid and a DE-PSO hybrid is used to solve a single objective transmission system design problem. Their work used a divide-and-conquer approach where each algorithm in the hybridization was used to solve a separate sub-problem. This is known as teamwork hybridization. A thorough review of a PSO-DE hybridization is performed by Xin et al. [24]. The work of Liu and Yang [25] developed a hybrid particle swarm - Nelder-Mead (NM) algorithm. Their approach uses both the PSO and NM algorithms at each iteration. Although this approach guarantees that each algorithm attempts the optimization at each iteration, it also requires excessive number of objective function calculations. This is because the objective function has to be calculated even for the algorithm that is under performing.

Reddy et al. [26, 27] accelerated the parameter estimation in Lithium-Ion batteries using a relay-type single objective hybrid optimizer. Their switching strategy randomly selected between NSGA-III, MOEA/DD and NSDE-R if the current algorithm failed to improve in its objective function. Their work reduced the time required to solve the problem from three weeks, when a non-hybrid algorithm was used, to 14 hours using the hybrid approach. Lepagnot [28] developed a multi-objective high-level relay hybrid algorithm consisting of a local search, differential evolution and particle swarm algorithms. Their approach first used the local search algorithm, followed by the differential evolution and particle swarm algorithm in a sequential manner.

All hybrid multi-objective optimization algorithms thus far do not perform well for problems with more than four to five objectives. These types of problems, named many-objective optimization problems, pose great difficulty in preserving diversity in the Pareto solution. This is even more true when accounting for uncertainty. It will be later shown that for each objective in the deterministic optimization framework, there are two objectives in the probabilistic framework. Uncertainty is a very significant vet inevitable issue in design optimization. Often uncertainties present in the model input, i.e. through geometry defects, improper calibration of measurement equipment, can lead to uncertainties in and deterioration of design performance. Few researchers have investigated solving the probabilistic optimization problem using algorithms for multi-objective optimization. Cheng et al. [29] developed a hybrid DE-SQP algorithm for robust optimization. Zhang et al. [30] used a finite difference combined with a stochastic gradient method to solve Bayesian inverse problems under uncertainty.

It can be seen that the amount of work in hybrid optimization under uncertainty is far-and-few. This work develops a many-objective hybrid optimizer (MOHO) capable of solving constrained and unconstrained, many-objective optimization problems under uncertainty. The MOHO algorithm will feature five constitutive algorithms and will use a single deterministic criteria to control the switching. It will use several methods for modeling the uncertainties in objective functions. The developed MOHO algorithm will be analyzed using 65 analytical test problems for which the optimum solution is known. The single objective version of MOHO, named SOHO, will be used to perform parameter estimation on a highly non-linear electrochemical model of a battery. MOHO will also be used to perform design optimization of subsonic and hypersonic bodies, both, within the deterministic framework and probabilistic framework. These real-life test cases are used due to the popularity of aerodynamic shape design. The increased computational cost of such aerodynamic design and analysis, pose a challenge for optimization algorithms to efficiently solve the design problem with few objective function evaluations. The design of subsonic aerodynamic bodies will utilize an in-house developed fast multipole method solver to accelerate the computation of the flow-field around the geometry. The hypersonic design analysis will be performed using an in-house developed modified Newton impact theory solver.

1.2 Objective of the Research

The ultimate objective of this research is to develop a robust framework for probabilistic many-objective optimization. The developed optimization and analysis frameworks will be analyzed on an analytical test suite as well as three real-life problems. The objectives of this research are the following:

- 1. Develop and validate a many-objective hybrid optimizer (MOHO) that:
 - (a) Can optimize for several objectives
 - (b) Can incorporate constraints
 - (c) Can account for uncertainty
 - (d) Requires little user input
 - (e) Is robust
- 2. Develop and validate an uncertainty quantification framework
- 3. Validate Single-Objective Hybrid Optimizer (SOHO) for inverse parameter identification problems:

- (a) Non-intrusively estimate 44 parameters in a highly non-linear electrochemical model of Li⁺ battery
- (b) Drastically reduce time required to estimate the parameters.
- 4. Validate MOHO on subsonic aerodynamic shape optimization:
 - (a) Develop Fast Multipole solver
 - (b) Validate Fast Multipole solver
 - (c) Validate MOHO for subsonic case within the deterministic framework
 - (d) Validate MOHO for subsonic case within the probabilistic framework
- 5. Validate MOHO on hypersonic aerodynamic shape optimization
 - (a) Develop modified Newton impact theory Solver
 - (b) Validate modified Newton impact theory Solver
 - (c) Validate MOHO for hypersonic case within the deterministic framework
 - (d) Validate MOHO for hypersonic case within the probabilistic framework

The main contributions of this work are as follows:

- 1. Two new optimization algorithms featuring increased rate of convergence
- 2. A new robust hybrid optimizer with automatic switching
- 3. A new deterministic criteria to control the automated switching between constitutive algorithms
- 4. Development of an efficient uncertainty quantification framework
- 5. Development of accelerated external flow analysis frameworks

1.3 Organization of the Dissertation

The remaining chapters of the dissertation are organized as follows.

In Chapter 2, the multi-objective, many-objective and probabilistic optimization frameworks are presented. The constitutive algorithms and switching rules in MOHO are also presented in this chapter. Finally, Chapter 2 also presents the results of MOHO on the analytical test cases. Chapter 3 presents the framework for uncertainty quantification. In Chapter 4, the numerical framework for geometry parameterization, electrochemical model of a Li⁺ battery, potential flow (Fast Multipole) solution and modified Newton impact theory solution is presented and validated. In Chapter 5, the parameter estimation in electrochemical model of a battery is performed. In Chapter 6, the results of subsonic aerodynamic shape The results of both, optimization within the optimization are presented. deterministic framework and probabilistic framework, are presented here. In Chapter 7, the results of hypersonic aerodynamic shape optimization are presented. The results of both, optimization within the deterministic framework and probabilistic framework, are presented here. Chapter 8 presents the summary of the dissertation and some area of future work.

CHAPTER 2

OPTIMIZATION

Numerical optimization typically involves identifying a set of parameters or design variables to maximize or minimize a particular objective or set of objectives. Most optimization problems typically fall under the following traditional categories:

- 1. Single objective vs. Multi-Objectives vs. Many-Objectives
- 2. Constrained vs. Unconstrained
- 3. Deterministic vs. Probabilistic
- 4. Continuous vs. Discrete

All optimization methods use some type of procedure for updating the design variables in search for a better optimum. These updating procedures can be deterministic, like in the case where the gradient of the objective function is used to obtain the search direction. These deterministic updated procedures are typically used by gradient based optimization algorithms. There also exist stochastic methods such as those used by evolutionary or heuristic optimization algorithms. The proposed set of variables, obtained using the updating procedure, are then evaluated and compared against those from the previous iteration. The proposed set of variables are either kept or discarded depending on their superiority over the designs from the previous iterations. This step is typically referred to as the selection step. The updating and selection process is continued until a user-specified number of iterations has been reached, or no further improvement in optimum is obtained. This section outlines the different types of optimization problems that MOHO is able to solve. This section also presents the frameworks that are used to solve various optimization problems.

2.1 Multi-Objective Optimization

m

Multi-objective optimization problems typically involve minimizing or maximizing two or more conflicting objectives. The standard multi-variate, multi-objective, constrained optimization problem [31] is defined as

in
$$\vec{f}(\vec{x})$$

 $\vec{f} = \{f_1(\vec{x}), \dots, f_l(\vec{x})\}$
 $\vec{x} = \{x_1, \dots, x_m\}$
subject to : $x \in [a_i, b_i], \quad i = 1, \dots, m$
 $h_j(\vec{x}) = 0, \quad j = 1, \dots, n$
 $g_k(\vec{x}) \le 0, \quad k = 1, \dots, o$
(2.1)

where \vec{x} is the vector of m design variables, \vec{f} is the vector of l objectives, h_j are the n equality constraints, and g_k are the o inequality constraints. If l = 1the problem reduces to a single objective problem and if n = o = 0 the problem reduces to an unconstrained optimization problem. It should be noted that the single objective optimization problem can also be solved within the multi-objective optimization framework if l > 1 by setting the value of all objectives to the value of the objective that is being minimized/maximized. This approach is used to solve all single objective optimization problems in this work.

In single objective optimization, it is easy to identify which design is better simply by comparing the values of the single objective. In multi-objective optimization, an alternate method to judge the superiority of one design over another is needed. This is done using the non-domination criteria presented by Deb [31]. Using the non-domination criteria, design A is said to dominate design B if A is better than B in at least one objective and no worse than B in the remaining objectives. Otherwise, the two designs are non-dominated with each other. The set of these non-dominated designs form the Pareto frontier. The solution of the multi-objective optimization problem is this Pareto front of non-dominated designs. Figure 2.1 shows the Pareto frontier where the black points represent the non-dominated designs and the red points represent the dominated designs. The proposed designs at the next iteration are obtained through an update procedure using these non-dominated designs. The update procedure is specific to each algorithm and will be presented in greater detail.



Figure 2.1: Representation of the non-domination criteria used in multi-objective optimization

2.2 Many Objective Optimization (MOO)

Optimization problems with two to three objectives are commonly referred to as multi-objective optimization problems. Optimization problems with four or more objectives are referred to as many-objective optimization problems. These types of optimization problems pose additional difficulties and are an active field of research.

It is well known that with an increase in the number of objectives, an increasingly larger percent of the total population becomes non-dominated [31, 13]. Since most algorithms use the non-domination criteria to compare superiority of designs, no new information is obtained if the entire population is non-dominated. Figure 2.2 shows the percent of the population that is non-dominated, for different numbers of objectives, as the optimization procedure continues for longer generations. It can be seen that for optimization problems with five objectives or more, almost 100% of the population becomes non-dominated in the early phase of the search.



Figure 2.2: Percent of population that is non-dominated at each generation for varying number of objectives

A successful optimization yields Pareto designs that are both well converged and uniformly distributed on the analytical Pareto front. This uniform distribution is referred to as diversity preservation. Many-objective optimization problems also pose a difficulty with diversity preservation. Consider a simple genetic algorithm initialized with 100 members where all 100 members are non-dominated. The recombination operator will yield an additional 100 members (one child for each original member) all of which are non-dominated. From this set of 200 non-dominated members, a robust method is needed to carry into the next generation the 100 members that best preserve convergence and diversity.

This work uses a space partitioning technique to preserve diversity in the Pareto set [32]. The objective function space is partitioned using reference points created using uniform random number generators. The Pareto designs that are closest to the sparsely populated region of the objective function space are carried forward into the next generation. This type of reference point based selection is shown in Fig. 2.3, where the red points are the members of the population and the black points are the reference points. Each member is associated with the single reference point with the shorted perpendicular distance. Preference is given to the designs closest to the sparsely populated reference points. Uniformly distributed reference points will result in uniformly distributed Pareto designs, while a biased distribution of reference points will yield the Pareto designs nearest to those reference points. This can be very beneficial when designs in a specific region of the Pareto front are sought after.



Figure 2.3: Representation of reference point based niching used to preserve diversity in many-objective optimization

2.3 Probabilistic Optimization

The multi-objective optimization framework presented in Section 2.1 considers that the design variables, objective function and constraints are all deterministic values. In most applications, although neglected, all three sets contain some level of random error. The error in design variables can arise from defects in manufacturing or machine imprecision, while errors in objective functions and constraints can arise from simplified mathematical models, machine imprecision, errors in input parameters, etc. Objective function values obtained through experimentation also contain some form of error. Therefore, it can often become imperative to also incorporate such uncertainty into the optimization framework. The deterministic framework can be converted to a probabilistic framework by incorporating uncertainty ξ in objective functions and constraints as

$$\min \quad \vec{f}\left(\vec{x}, \vec{\xi}\right)$$

$$\vec{f} = \left\{ f_1\left(\vec{x}, \vec{\xi}\right), \dots, f_l\left(\vec{x}, \vec{\xi}\right) \right\}$$

$$\vec{x} = \left\{ x_1, \dots, x_m \right\}$$

$$subject \ to : x \in [a_i, b_i], \quad i = 1, \dots, m$$

$$h_j\left(\vec{x}, \vec{\xi}\right) = 0, \quad j = 1, \dots, n$$

$$g_k\left(\vec{x}, \vec{\xi}\right) \le 0, \quad k = 1, \dots, o$$

$$(2.2)$$

The probabilistic optimization problem, however, is not solvable within the current framework and must be converted to a deterministic optimization problem while incorporating the uncertainty. This is done by using statistical moments of the objectives and constraints. The probabilistic optimization problem can be posed as a deterministic optimization problem as

$$min \quad \vec{\mu} = \{\mu_{f_1}, \mu_{f_2}, \dots, \mu_{f_l}\}$$

$$min \quad \vec{\sigma} = \{\sigma_{f_1}, \sigma_{f_2}, \dots, \sigma_{f_l}\}$$

$$subject \ to: \qquad (2.3)$$

$$Prob\left[h_{j}\left(\vec{x},\vec{\xi}\right)=0\right] \geq R_{j}, \quad j=1,\ldots,n$$
$$Prob\left[g_{k}\left(\vec{x},\vec{\xi}\right)\leq0\right] \geq R_{k}, \quad k=1,\ldots,o$$

where μ_{f_i} is the mean value of the i^{th} objective function f, σ_{f_i} is the standard deviation of the i^{th} objective function f, $Prob[\cdot]$ is the probability and R_i is the threshold probability of satisfying the i^{th} constraint. Immediately, it can be seen that each probabilistic objective yields two deterministic objectives. Therefore, a probabilistic optimization problem that is multi-objective in nature, can easily become a many-objective optimization problem when solved in the deterministic framework. The uncertainty in design variables must also be propagated through

the analysis code to the objective function to obtain the probabilistic response. This is the main difficulty in solving probabilistic optimization problems as this uncertainty propagation is computationally expensive. Methods for propagating uncertainty are discussed in later sections.

2.4 Many Objective Hybrid Optimization (MOHO)

There exist several optimization algorithms, each using a different update and selection procedure. It is also known that the objective function topology is dependent on the input variables and the analysis code. It is therefore safe to assume that there is no single algorithm that can outperform all other algorithms for every conceivable optimization problem. This is apparent in the "No Free Lunch Theorem for Search" proposed by Wolpert and Macready [33]. The theorem states

"... all algorithms that search for an extremum of a cost function perform exactly the same, when averaged over all possible cost functions. In particular, if algorithm A outperforms algorithm B on some cost functions, then loosely speaking there must exist exactly as many cost functions where B outperforms A."

Therefore the probability of converging to the optimum designs can be increased by combining several algorithms into a single suite. This hybridization of several algorithms and controlled switching between the algorithms can increase the convergence speed to optimum design on the Pareto front. This work develops a Many-Objective Hybrid Optimizer (MOHO) to address these key issues. Several attempts have been made to hybridize particular algorithms by combining the update and selection procedure but this is a lower level hybridization. The hybridization in this work is a high-level relay since MOHO does not alter the individual algorithms. This combination of several algorithms into a single suite leads to MOHO's performance to be heavily dependent on not only the individual constitutive algorithms but also the switching mechanism used. Both dependencies are investigated in the the upcoming sections.

2.4.1 Constitutive Algorithms

Since the topology of the objective function is not known a priori, it is impossible to select the correct algorithm for that particular problem. Therefore, the MOHO suite should include a selection of algorithms that perform differently in their update and selection procedures. The MOHO suite currently contains five constitutive algorithms. A general overview of the algorithms is given. Due to the large number of established algorithms in the MOHO suite, the inner workings of the individual algorithms are not presented. The reader is referred to the algorithm's respective reference. The MOHO suite contains the following algorithms:

- 1. NSGA-III [32]: The Non-Dominated Sorting Genetic Algorithm-III (NSGA-III) is a third-generation genetic algorithm developed for many-objective optimization problems. It uses a combination of Simulated Binary Crossover (SBX) [34] and polynomial mutation [31] to produce a candidate design from randomly selected parent designs. The candidate designs are then evaluated and a non-dominated sort is performed. A reference point based niching is performed to select the designs to be carried into the next generation.
- 2. **MOEA-DD** [35]: The Many-Objective Evolutionary Algorithm Based on Dominance and Decomposition (MOEA-DD) is an evolutionary algorithm that uses similar recombination operators as NSGA-III. Whereas the

NSGA-III selects its parent designs randomly, the MOEA-DD algorithm selects the parent design from its immediate neighborhood. The MOEA-DD uses a penalty-boundary-intersection metric (PBI) to select the designs to be carried into the next generation.

- 3. **SPEA-R** [36]: The Strength Pareto Evolutionary Algorithm Based on Reference Point (SPEA-R) is also an evolutionary algorithm that uses the same recombination operators as MOEA-DD. It, however, prioritizes diversity preservation in the Pareto front before the convergence to the Pareto front.
- 4. **NSDE-R1B** [37]: The Non-Dominated Sorting Differential Evolution (NSDE) algorithm uses the same selection operator as NSGA-III but a different recombination operator. Whereas the previous three algorithms combine two parent designs to create a candidate designs, the NSDE-R1B algorithm uses the "rand/1/bin" (R1B) mutation operator to create a candidate design from three parent designs.
- 5. **NSDE-D3** [37]: The Non-Dominated Sorting Differential Evolution algorithm uses the same selection operator as NSGA-III but a different recombination operator. The NSDE-D3 uses the "donor3" (D3) mutation operator that performs a weighted combination of three parent designs. The random weights can be generated using any distribution but are uniformly distributed in this work.

The NSDE-R1B and NSDE-D3 algorithms were recently developed for use in MOHO. The NSDE-R1B and NSDE-D3 algorithms showed an increased rate of convergence. Figure 2.4 shows the convergence of four algorithms on the DTLZ

problems. It can be seen that the NSDE-R1B and NSDE-D3 algorithms have better convergence and a higher rate of convergence.



Figure 2.4: Convergence histories of the newly developed algorithms on: a) DTLZ1, b) DTLZ2 and c) DTLZ4

It should be mentioned that MOHO is also able to interchange recombination and selection operators from different algorithms to also hybridize the constituent algorithms, but this is out of the scope of this work.
2.4.2 Switching Rules

Although the MOHO suite contains a diverse set of algorithms, its performance is also dependent on the switching criteria used. The switching criteria for a single objective optimization problem is simple since it is easy to monitor the progress of the algorithm. In single objective optimization, an algorithm has made progress if it found a point at the current iteration that is better than the best known point at the previous iteration. This approach cannot be used for many-objective optimization problems since the solution of the optimization problem is now a set of designs.

The hybrid optimizer developed by Moral [12] assigned a score out of five at each generation. If the total scored by an algorithm was less than two, another algorithm was selected at random for the next generation. An algorithm is assigned a score of one for each of the following criteria it fulfills:

- 1. Change in non-dominated set size
- 2. A point from the new generation dominates
- 3. Change in dominated hypervolume
- 4. Average distance change between the designs in objective function space
- 5. Spread of non-dominated set

Most metrics used for multi-objective problems fail for many-objective problems. As shown in Fig. 2.2, as the number of objectives increases, a greater percent of the population becomes non-dominated. This very quickly leads to all the population members becoming non-dominated and the change in non-dominated set size from generation to generation becomes zero. For this reason, neither the first nor the second criteria used by previous versions of MOHO can be applied to many-objective problems. Computing the average distance of the designs in objective function space with respect to the origin, provides incomplete information about the Pareto solution. It is possible that all designs converge to a single point that is farthest away from the origin. In this case, the average distance metric will indicate that the current Pareto front is better when actually the Pareto designs fail to preserve diversity. Similarly, this is true for the spread metric since it only considers the extreme values of the objectives on the Pareto front.

Hypervolume is a metric that can quantify both the convergence and diversity of the Pareto front [38]. Hypervolume measures the size of the objective function space dominated by solutions S and bounded by z^r where z^r is the reference point dominated by all Pareto optimal solutions. Therefore, the hypervolume metric does not require the analytical Pareto front to be known. The hypervolume is computed as

$$HV(S) = Vol(_{x \in S} [f_1(x), z_1^r] \times \dots [f_M(x), z_M^r])$$
(2.4)

where $Vol(\cdot)$ is the Lebesgue measure. A larger value of HV indicates a better converged and more diversified solution set. The reference point is taken as the Nadir point [39] from the initialization population. The Nadir point is the worstcase objective vector where the worst value of each objective function from the entire population set is collected into a single vector. Unlike in previous attempts at multi-objective hybridization where the alternative algorithm is selected at random, the current MOHO uses a more deterministic criteria, Probability of Success (POS), for selecting the next algorithm. During the optimization process, the number of attempts made by each algorithm and the number of successful attempts of each algorithm is monitored and updated at each generation. The POS value is then computed as

$$POS(\cdot) = \frac{N_{success}(\cdot)}{N_{total}(\cdot)}$$
(2.5)

where \cdot is a particular algorithm, $N_{success}$ are the number of successful attempts, and N_{total} are the total attempts made by the algorithm. Here, a successful attempt is one where the hypervolume increases from the current maximum value. At initialization, each algorithm in the MOHO suite is given a POS value of one. This value is updated at each generation for each algorithm. The algorithm with the largest POS is selected for the next generation. If multiple algorithms have the same largest POS value, one is selected at random. This switching criteria allows the algorithm that is performing best for the current problem to be selected and the algorithm performing worst to not be selected.

2.5 Benchmarking MOHO

Each optimization algorithm in the MOHO suite is analyzed on a family of test problems for which the analytical Pareto front is known. The test problems vary in degree of difficulty and number of variables, objectives and constraints.

2.5.1 Test Problems for Many-Objective Optimization

The test problems used to evaluate the performance of MOHO are taken from the DTLZ [40] and WFG [41] test suites. These test suites were chosen because they can be scaled up to any number of design variables N and any number of objectives M. It should be mentioned that the test suites only contain unconstrained problems. The test suites contain problems of varying difficulty, each with different properties (linear, concave, etc.). They provide a diverse set of objective function topologies that the optimization algorithm can be validated against. Table 2.1 shows these properties. Their analytical Pareto fronts are also known, which allow for the convergence analysis of the algorithms. The Pareto front for the DTLZ1 problem is a half-unit-hyperplane, while for DTLZ2 to DTLZ4, the Pareto front is a unit hypersphere. The WFG4 to WFG9 test problems have a hyperelliptic Pareto front with radii $r_i = 2i$ where $i \in \{1 \dots M\}$.

The total number of design variables for the DTLZ cases is N = M + k - 1. For DTLZ1, k = 5 and for DTLZ2, DTLZ3 and DTLZ4, k = 10. The total number of variables for the WFG suite is N = k + l, where l = 20 and $k = 2 \times (M - 1)$. The algorithms were investigated on problems having three, five, eight, 10 and 15 objectives.

| Test problem | Properties |
|--------------|--|
| DTLZ1 | Linear, multi-modal |
| DTLZ2 | Concave |
| DTLZ3 | Concave, multi-modal |
| DTLZ4 | Concave, biased |
| WFG1 | Mixed, biased |
| WFG2 | Convex, disconnected, multi-modal, non-separable |
| WFG3 | Linear, degenerate, non-separable |
| WFG4 | Concave, multi-modal |
| WFG5 | Concave, deceptive |
| WFG6 | Concave, non-separable |
| WFG7 | Concave, biased |
| WFG8 | Concave, biased, non-separable |
| WFG9 | Concave, biased, multi-modal, deceptive, non-separable |

Table 2.1: Properties of DTLZ and WFG test problems used to validate the optimization algorithms

The constrained test problems were obtained by incorporating constraints to the original DTLZ problems. In particular, the C1-DTLZ1, C1-DTLZ3 and C3-DTLZ4 [42] test problems are considered to evaluate the algorithms on constrained test problems. The C1 constrained test problems maintain the same optimal Pareto front as their unconstrained version, but the feasible search space is greatly reduced. The C3 constrained test problems alter the unconstrained Pareto front by placing it in an infeasible region. The Pareto front for the C3 type constrained problems is the constraint boundary itself.

2.5.2 Indicator of Performance

This work utilizes two metrics to quantify the performance of the optimization algorithms: inverted generalized distance (IGD) and hypervolume (HV).

Because all algorithms in the MOHO suite are reference point-guided algorithms, only the analytical Pareto points closest to these reference points, called targeted Pareto points, should be used to construct the accuracy measure. Since the analytical Pareto front is known, the targeted Pareto points, Z_{tar} , can be obtained by finding the intersection between the analytical Pareto front and the ray connecting the reference point and the origin. The accuracy measure should then use these targeted Pareto points and the converged Pareto point obtained from the optimization algorithms. This work makes use of the inverted generalized distance (IGD) metric [43] given by

$$\operatorname{IGD}\left(P, \boldsymbol{Z_{tar}}\right) = \frac{1}{|\boldsymbol{Z_{tar}}|} \sum_{i=1}^{|\boldsymbol{Z_{tar}}|} \min_{j=1}^{|\boldsymbol{P}|} d\left(\boldsymbol{z_i}, \boldsymbol{p_i}\right)$$
(2.6)

where $d(\mathbf{z}_i, \mathbf{p}_i) = ||\mathbf{z}_i, \mathbf{p}_i||_2$. The IGD measures both convergence to the analytical Pareto front and the diversity of the Pareto solution, if the targeted Pareto points are well distributed. The smaller the value of IGD, the better the approximated Pareto solution.

The IGD metric is only used for the DTLZ test problems because of their well defined Pareto front. It is difficult to generate the targeted Pareto points for the WFG problems, due to their complicated Pareto fronts. Therefore, the hypervolume is used instead of the IGD metric for the WFG problems. The reference points required for the hypervolume calculations in Eq. (2.4) are given in Table 2.2. The hypervolume is normalized to [0, 1] by dividing by $z = \prod_{i=1}^{M} z_i^r$. For three to 10-objective test problems, the exact hypervolume is calculated using the WFG algorithm [44] . For problems with greater than 10 objectives, the hypervolume is approximated using Monte Carlo sampling [45]. The sampling size was held constant, as recommended, at 10,000 samples [45].

Table 2.2: The reference point used in the computation of Hypervolume for each problem

| Test problem | Reference point |
|-----------------|---------------------------------------|
| DTLZ1 | $\left(1.0,\ldots,1.0 ight)^T$ |
| DTLZ2 to DTLZ4 | $(2.0,\ldots,2.0)^T$ |
| WFG1 to WFG9 $$ | $(3.0, \ldots, 2.0 \times M + 1.0)^T$ |

2.5.3 Algorithm Parameter Settings

The algorithm-specific and test-problem-specific parameters are presented in this section.

Due to the stochastic nature of the optimization, each algorithm was ran 20 times on each test problem. Due to the varying degree of difficulty of problems in the test suites, the number of generation and population size were made test problem specific. Table 2.3 shows the population size and the number of generations for each test problem.

| | | Number of generations | | | | |
|----|-----------------|-----------------------|-------|-------|-------|------|
| M | Population size | DTLZ1 | DTLZ2 | DTLZ3 | DTLZ4 | WFG |
| 3 | 92 | 400 | 250 | 1000 | 600 | 400 |
| 5 | 212 | 600 | 350 | 1000 | 1000 | 600 |
| 8 | 156 | 750 | 500 | 1000 | 1000 | 750 |
| 10 | 276 | 1000 | 750 | 1500 | 2000 | 1000 |
| 15 | 136 | 1500 | 1000 | 2000 | 3000 | 1500 |

Table 2.3: Population size and generations for each test problem

The parameters that control the recombination operator of the algorithm are kept constant for all test problems. The properties for each algorithm are as follows:

- NSGA-III: The crossover probability, p_c , and the mutation probability, p_m are 1.0 and 1/N respectively. The crossover distribution index, η_c , and mutation distribution index, η_m , are set to 30 and 20 respectively.
- **MOEA-DD**: The crossover and mutation parameters are the same as those for the NSGA-III algorithm. The penalty parameter θ in PBI is set to 5.0. The neighborhood size, T, is set to 20 and the probability, δ , used to select in the neighborhood is chosen to be 0.9.
- **SPEA-R**: The crossover and mutation parameters are the same as those for the NSGA-III algorithm.
- NSDE-R1B: The scale factor F is held constant at 0.5 and the crossover probability Cr is also held constant at 0.7.
- **NSDE-D3**: The scale factor and crossover probability are the same as those for the NSDE-R1B algorithm.

2.5.4 Performance of MOHO on Unconstrained Test Problems

The performance of MOHO is investigated on analytical unconstrained test problems with three, five, eight, 10 and 15 objectives. Table A.1 and Table A.2 show the IGD values for the DTLZ test problems and hypervolume values for the WFG test problems respectively. For both test problem suites, it can be seen that MOHO performs better on average then the other algorithms.

One main issue when solving optimization problems with more than three objectives, is the visualization of the Pareto front. In this work, the higher dimensional Pareto fronts are shown using a parallel coordinate graph, e.g. Fig. 2.6, where each line represents a Pareto design. The minimum and maximum values of the parallel coordinate graph should coincide with the ideal point and the Nadir point of the analytical Pareto front. For example, in the case of the DTLZ1 problem, Fig. 2.6, the ideal point has a value of zero for all the objectives and the Nadir point has a value 0.5 for all the objective. Therefore, the value of each objective for each design in parallel coordinate graph should be between 0.0 and 0.5. Similarly, the value of each objective for each design in parallel coordinate graph should be between 0.0 and 1.0 for the DTLZ2 to DTLZ4 test problems. The distribution of the parallel coordinates indicates the distribution of the Pareto design through the objective function space. A uniform, structure parallel coordinate graph indicates a uniformly distributed Pareto front.

Figure 2.5 shows the Pareto fronts obtained for the three objective DTLZ1 problem where the grey surface represents the analytical Pareto front, the half-unit hyperplane. It can be seen that the SPEA-R algorithm is unable to properly converge to the analytical Pareto front or preserve diversity. The MOEA-DD

algorithm also isn't able to properly preserve the diversity, whereas the other algorithms are able to both converge and preserve diversity. It can be seen that even with an under performing algorithm, MOHO is able to produce good results.







Figure 2.5: The Pareto front for a three objective DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.6 shows the Pareto fronts for a 10 objective DTLZ1 problem. The structured arrangement of the designs indicate a good convergence to the analytical Pareto front and to the reference points. The MOEA-DD and SPEA-R again fail to produce good results for the 10 objective DTLZ1 test problem, whereas the remaining algorithms all produce good convergence. This suggests that the MOEA-DD and SPEA-R algorithms are non-ideal for the linear multi-modal problems.





Figure 2.6: Parallel coordinates plot showing the Pareto front for a 10 objective DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.7 shows the Pareto fronts for a 10 objective DTLZ3 problem. The analytical Pareto front for this problem is a unit hypersphere. Unlike the DTLZ1 problem, the MOEA-DD algorithm performs well for the DTLZ3 problem while the SPEA-R algorithm continues to under perform. The remaining algorithms also perform well for the DTLZ3 problem. Again, MOHO is able to produce good convergence and preserve diversity despite the presence of SPEA-R in its switching suite. Like the DTLZ1 problem, the DTLZ3 test problem is also multi-modal, again indicating that the SPEA-R algorithm is unable to converge well for multi-modal DTLZ problems.





Figure 2.7: Parallel coordinates plot showing the Pareto front for a 10 objective DTLZ3 problem obtained, using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.8 shows the Pareto fronts for a 15 objective DTLZ4 problem. The analytical Pareto front for this problem is also a unit hypersphere. It can be seen that the SPEA-R algorithm performs better for the DTLZ4 problem than the DTLZ1 and DTLZ3 problems but still under performs when compared to the other five algorithms. The MOHO algorithm continues to produce stable results.





Figure 2.8: Parallel coordinates plot showing the Pareto front for a 15 objective DTLZ4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.9 shows the Pareto fronts obtained for the three objective WFG4 problem where the grey surface represents the analytical Pareto front, a hyperellipse. It can be seen that the final Pareto front obtained by MOEA-DD converges to a particular area of the objective function space, rather than uniformly over the entire hyperellipse. The WFG4 problem like the DTLZ3 problem is concave and multi-modal, but unlike in previous cases, the SPEA-R algorithm is able to both converge to the analytical Pareto front and preserve diversity. For the WFG test suite, the MOEA-DD algorithm under performs

compared to the remaining five. It can be seen that even with an under performing algorithm, MOHO is able to produce good results.







Figure 2.9: The Pareto front for a three objective WFG4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.10 shows the Pareto fronts for a 15 objective WFG5 problem. The analytical Pareto front for this problem is also a unit hypersphere. The WFG5 problem features a concave, deceptive topology. It can again be seen that the MOEA-DD algorithm under performs when compared to the remaining five. The NSDE-R1B and NSDE-D3 algorithms also are not able to produce Pareto solutions in the lower bound of the first few objectives. Despite having three algorithms not suited for the WFG5 test problem, MOHO is able to produce a Pareto front that both converges to the analytical Pareto front and satisfies diversity.





Figure 2.10: Parallel coordinates plot showing the Pareto front for a 15 objective WFG5 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.11 shows the Pareto fronts for a 15 objective WFG6 problem. The WFG6 is a concave, non-separable problem with a hyperelliptic Pareto front. It can again be seen that the MOEA-DD algorithm under performs significantly. The NSDE-R1B and NSDE-D3 algorithms for the WFG6 problem produce Pareto solutions comparable to those obtained by NSGA-III, SPEA-R and MOHO.





Figure 2.11: Parallel coordinates plot showing the Pareto front for a 15 objective WFG6 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

2.5.5 Performance of MOHO on Constrained Test Problems

The performance of MOHO is investigated on analytical constrained test problems.

Figure 2.12 shows the Pareto fronts obtained for the three objective C1-DTLZ1 problem where the grey surface represents the analytical Pareto front, the half-unit hyperplane. The C1-DTLZ1 problem features only a small feasible region close to the analytical Pareto front. It can be seen that the SPEA-R algorithm again fails to

properly converge to the analytical Pareto front or preserve diversity, drawing the search to the upper region of the Pareto front. The remaining five algorithms, all produce accurate results for this test problem.







Figure 2.12: The Pareto front for a three objective C1-DTLZ1 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.13 shows the Pareto fronts obtained for the three objective C1-DTLZ3 problem where the grey surface represents the analytical Pareto front, the unit hypersphere. The C1-DTLZ3 problem features a banded infeasible region adjacent to the Pareto front. For this case, the MOEA-DD, SPEA-R and NSDE-D3 algorithms, all fail to converge to the Pareto front. Those that converge, NSGA-III, NSDE-R1B and MOHO, all show good convergence to the Pareto front and to the reference points. Again, it can be seen that despite having under performing algorithms available in its switching pool, MOHO is still able to produce converged results.





Figure 2.13: The Pareto front for a three objective C1-DTLZ3 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

Figure 2.14 shows the Pareto fronts obtained for the three objective C3-DTLZ4 problem where the grey surface represents the analytical Pareto front. The C3-DTLZ4 test problem completely redefines the original Pareto front. The original hyperspherical Pareto front now lies in the infeasible region therefore making the new Pareto front, the surface of the constraint boundary. It can be seen that all six algorithms converge to the Pareto front and preserve diversity satisfactorily. The algorithms that features the best convergence and diverse populations are the MOEA-DD and MOHO algorithms.









Figure 2.14: The Pareto front for a three objective C3-DTLZ4 problem, obtained using: a) NSGA-III, b) MOEA-DD, c) SPEA-R, d) NSDE-R1B, e) NSDE-D3 and f) MOHO

2.6 Optimization: Summary

A robust algorithm for many-objective optimization was presented. The MOHO algorithm performs controlled switching between five constitutive algorithms to accelerate convergence and avoid local Pareto fronts. A deterministic switching criteria was developed to control the switching between the constitutive algorithms. The developed MOHO algorithm was validated on a set of analytical test problems from the DTLZ and WFG test suites with varying degrees of difficulty. It was shown that the MOHO algorithm can robustly solve problems with up to 15 objectives. The developed MOHO was investigated for both constrained and unconstrained problems. MOHO performed better in more than 50% of the test problems. It was shown that MOHO is able to reliably converge to the Pareto front despite having under performing algorithms in its algorithm suite.

CHAPTER 3

UNCERTAINTY QUANTIFICATION

Uncertainty quantification is the science of quantifying and characterizing uncertainty in a numerical model or real world system. It primarily deals with quantifying the uncertainty in output given the uncertainty in the input and vice versa. Uncertainty is inherently present in the system and can appear in the model through measurement errors, uncertain material properties and manufacturing defects. Uncertainty can typically be put in two categories: aleatoric and epistemic. Aleatoric uncertainty is uncertainty that is beyond our current ability to measure. Examples of aleatoric uncertainty would involve rolling a die or shooting an arrow. Epistemic uncertainty is uncertainty arising from lack of knowledge about a particular system. Examples of epistemic uncertainty include material properties and manufacturing dimensions. When performing uncertainty quantification for design optimization, only epistemic uncertainties are considered.

For probabilistic optimization, the uncertainty in output due to uncertainties present in the the inputs must be quantified. This is commonly referred to as uncertainty propagation. That is, uncertainties in the input are propagated through the model to the outputs. The goal of uncertainty propagation is to obtain low-order statistical moments of the outputs (mean and variance). There exist several methods for propagation of uncertainty, each with its own advantages. Uncertainties can be propagated through the model using an intrusive approach, requiring modification to the existing analysis code, or a black-box non-intrusive approach. Only nonintrusive propagation is considered in this work to generalize the application of the uncertainty quantification. The approaches available in MOHO in propagating uncertainties are presented in this section.

3.1 Monte Carlo Sampling

One of the simplest methods for propagating uncertainties through a model is through the Monte Carlo method (MCM) [46]. The Monte Carlo method is a sampling method that generates a set of N deterministic samples $x_{i_{i\in[1,N]}}$, from their corresponding probability distribution function $\pi(x)$. Each deterministic sample is then evaluated using the model to obtain a deterministic response, $f(x_i)$. The deterministic responses can then be used to compute the mean, μ_f , and standard deviation, σ_f , of the response using Eq. (3.1). By the central limit theorem it can be shown that the Monte Carlo method displays $1/\sqrt{N}$ convergence. This shows that the convergence is independent of the dimensionality of the model and only depends on the sample size.

$$\mu_f = \frac{1}{N} \sum_{i=1}^{N} f_i \qquad \sigma_f^2 = \frac{1}{N-1} \left(\sum_{i=1}^{N} \left(f_i - \mu_f \right)^2 \right)$$
(3.1)

Monte Carlo sampling can become infeasible if the computational time required to compute the model response is too large. For this reason, Monte Carlo sampling should only be used when the time required to compute a single response is under one second as typically several thousand samples are required for an accurate calculation of the statistical moments. Therefore, Monte Carlo sampling is usually performed using a surrogate model.

3.2 Polynomial Chaos Expansion

An alternative approach for propagating uncertainty is through Polynomial Chaos Expansion (PCE) [47, 48]. This approach decomposes the model response into a deterministic component and a stochastic component as shown in Eq. (3.2)

$$f(\xi, D) = \sum_{i=0}^{\infty} \alpha_i(D) \Psi_i(\xi)$$
(3.2)

where the coefficients α are dependent on the deterministic component D, and the basis functions Ψ are dependent on the stochastic component ξ . The basis functions are chosen such that they are orthogonal with respect to the probability distribution function of the input parameters, Eq. (3.3). Some common probabilities and their corresponding orthogonal Wiener-Askey basis functions [49] are given in Table 3.1.

$$\int \Psi_i(\xi) \Psi_j(\xi) \pi(\xi) d\xi = \delta_{i,j}$$
(3.3)

| $\pi(x)$ | Orthogonal Polynomials | Support Range |
|-----------------|------------------------|--------------------|
| Uniform | Legendre | [-1, 1] |
| Gaussian/Normal | Hermite | $[-\infty,\infty]$ |
| Gamma | Laguerre | $[0,\infty]$ |
| Beta | Jacobi | [-1,1] |

Table 3.1: Polynomials of the Wiener-Askey scheme for some random variables

Polynomial Chaos Expansions can be used intrusively or non-intrusively. An intrusive approach requires changes to the analysis code and, therefore, is not considered in this work. Several methods have been developed for non-intrusive polynomial chaos expansion. This work makes extensive use of the point-collocation approach [48]. In practice the infinite series in Eq. (3.2) must be truncated at some value P using a suitable truncation scheme. Since the joint distribution function of the design variables is the product of the probability distribution function of each variable, it can be shown that the total number of basis functions P + 1 is given by, Eq. (3.4), where n is the number of random variables and s is the order of the PCE.

$$N_t = P + 1 = \frac{(n+s)!}{n!s!} \tag{3.4}$$

This results in a linear system of size P shown in Eq. (3.5),

$$\begin{pmatrix} \Psi_{0}\left(\xi_{0}\right) & \Psi_{1}\left(\xi_{0}\right) & \cdots & \Psi_{p}\left(\xi_{0}\right) \\ \Psi_{0}\left(\xi_{1}\right) & \Psi_{1}\left(\xi_{1}\right) & \cdots & \Psi_{p}\left(\xi_{1}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_{0}\left(\xi_{P}\right) & \Psi_{1}\left(\xi_{P}\right) & \cdots & \Psi_{p}\left(\xi_{P}\right) \end{pmatrix} \begin{pmatrix} \alpha_{0} \\ \alpha_{1} \\ \vdots \\ \alpha_{P} \end{pmatrix} = \begin{pmatrix} f\left(\xi_{0}, D\right) \\ f\left(\xi_{1}, D\right) \\ \vdots \\ f\left(\xi_{P}, D\right) \end{pmatrix}$$
(3.5)

where $\Psi_i(\xi_j)$ is the *i*th basis function evaluated at the *j*th sample. Once the coefficients α are computed, the mean and variance can be obtained using Eq. (3.6) and Eq. (3.7) respectively.

$$\mu_f = \langle f \rangle \approx \sum_{i=0}^{P} \alpha_i \left(D \right) \left\langle \Psi_i \left(\xi \right) \right\rangle = \alpha_0 \tag{3.6}$$

$$\sigma_f^2 = \left\langle \left(f - \mu_f\right)^2 \right\rangle \approx \left\langle \left(\sum_{i=0}^P \alpha_i\left(D\right) \Psi_i\left(\xi\right)\right)^2 \right\rangle = \sum_{i=1}^P \alpha_i^2\left(D\right) \left\langle \Psi_i^2\left(\xi\right) \right\rangle \tag{3.7}$$

It is evident from Eq. (3.4) that the standard PCE approach suffers from the "curse of dimensionality" [50]. To construct a PCE of third order (s = 3) with five random variables (n = 5), a total of 55 (P = 54) samples are needed. This can sometimes become prohibitively expensive if the computational cost of a single analysis is too large. An ideal approach would be to construct the PCE using as

few deterministic samples as possible, even if the total required number of samples is not attainable. This can be done using a sparse representation of the PCE [51].

It is known that an underdetermined system has an infinite number of solutions. In PCEs, only a small fraction of the coefficients may carry significant weight, allowing for the assumption that many coefficients are zero. This leads to a sparse vector of unknowns. This assumption allows for the system to be regularized, leading to a well-posed problem. Doostan and Owhadi [52] discuss the theory, formulation and stability of this approach in great detail. The goal is to seek a solution to the linear system with the fewest number of non-zero coefficients. This problem can be solved using an L_1 -minimization:

$$\min ||a||_1 \text{ subject to } ||\Psi\alpha - f||_2 \le \delta \tag{3.8}$$

where δ is the truncation error. The L_1 -minimization problem can be solved using quadratic programming. This work uses the least-angle regression (LARS) algorithm [53] to solve the L_1 -minimization problem since it is not significantly affected by the dimensionality of the problem. The PCE obtained by solving the linear system in Eq. (3.5) will be referred to as the standard PCE whereas the expansion obtained by solving Eq. (3.8) will be referred to as the sparse PCE.

3.3 Uncertainty Quantification: Summary

This section presented the framework for uncertainty quantification. Three different approached were presented to propagate the uncertainty in input variables, through the system, to the output variables. The three approaches available in the suite are: Monte Carlo sampling, standard polynomial chaos expansion and sparse polynomial chaos expansion. Whereas, the Monte Carlo and standard polynomial chaos requires several samples to quantify uncertainty, the sparse polynomial chaos can perform the quantification using fewer samples. All three approaches will be be investigated in real life test problems.

CHAPTER 4

NUMERICAL MODELING

This chapter discusses the numerical framework, and governing equations used for each analysis. It presents the framework used for geometry parameterization, analysis of subsonic flows and analysis of hypersonic flows.

4.1 Geometry Parameterization and Deformation

The process of design, analysis and optimization often requires constant changes to an existing geometry or domain being analyzed. The method used to parameterize the geometry is sometimes not known and is often defined using simplified point clouds with no information on connectivity of the point clouds. One such example is point cloud data obtained from MRI and CT scans, or surface data used for 3D printing. A method is needed to efficiently manipulate such data without prior knowledge of parameterization and data formats. Since the solution of an optimization problem is heavily dependent on the number of variables used to define the problem, an efficient approach that can reduce the number of variables while providing detailed control of the geometry is needed. This work uses an efficient freeform deformation (FFD) technique to parameterize the geometry with only a few parameters.

The FFD technique used here originated in the field of computer graphics and animation. This method encloses the geometry or region of interest in a lattice of control points. These control points are then deformed and the resulting lattice deformation is propagated to the enclosed geometry. It was originally developed by Sederberg and Parry [54] and expanded on by others. This FFD approach is used herein. The authors described this approach as follows: "A good physical analogy for FFD is to consider a parallelpiped of clear, flexible plastic in which is embedded an object, or several objects, which we wish to deform. The object is imagined to also be flexible, so that it deforms along with the plastic that surrounds it."



Figure 4.1: Surface boundary representation of: a) undeformed lattice and object and b) deformed lattice and the corresponding deformed object.

Figure 4.1 shows a visual example of this approach. This approaches uses trivariate Bernstein polynomials to propagate the deformation in the lattice to the enclosed object.

Any point on the object of interest in 3D space can be written as, Eq. (4.1),

$$\vec{X} = \vec{X_0} + s\vec{S} + t\vec{T} + u\vec{U}$$
(4.1)

where $\vec{X} = \{x, y, z\}$, \vec{X}_0 is the lattice origin, $\vec{S}, \vec{T}, \vec{U}$ are the frame vectors for the lattice, and s, t, u are the normalized coordinate defined by Eq. (4.2)

$$s = \frac{\vec{T} \times \vec{U} \left(\vec{X} - \vec{X_0} \right)}{\vec{T} \times \vec{U} \cdot \vec{S}} \qquad t = \frac{\vec{S} \times \vec{U} \left(\vec{X} - \vec{X_0} \right)}{\vec{S} \times \vec{U} \cdot \vec{T}} \qquad u = \frac{\vec{S} \times \vec{T} \left(\vec{X} - \vec{X_0} \right)}{\vec{S} \times \vec{T} \cdot \vec{U}} \quad (4.2)$$

where $s \in [0, 1]$, $t \in [0, 1]$, $u \in [0, 1]$ if the point lies within the lattice. Let l, m, nbe the number of subdivisions of the lattice in the x, y and z direction. Then the coordinates of the lattice point \vec{P}_{ijk} can be written as Eq. (4.3)

$$\vec{P}_{ijk} = \vec{X_0} + \frac{i}{l}\vec{S} + \frac{j}{m}\vec{T} + \frac{k}{n}\vec{U}$$
(4.3)

If a lattice point \vec{P}_{ijk} is moved to a new location \tilde{P}_{ijk} , then the resulting deformed geometry can be obtained using Eq. (4.4)

$$\vec{X}_{def} = \sum_{i=0}^{l} \binom{l}{i} (1-s)^{l-i} s^{i} \left[\sum_{j=0}^{m} \binom{m}{j} (1-t)^{m-j} t^{j} \left(\sum_{k=0}^{n} \binom{n}{k} (1-u)^{n-k} u^{k} \tilde{P}_{ijk} \right) \right]$$
(4.4)

Figure 4.2 shows the applicability of the FFD framework for localized deformations, where only certain areas of interest are to be deformed.



Figure 4.2: Surface boundary representation of undeformed NASA 25D aircraft (blue) with: a) deformed wing (green) and b) deformed underbody (green)

4.2 Electrochemical Model of Doyle-Fuller-Newman

The dynamics of Lithium-Ion batteries is of a highly multi-physics nature. The physics of the processes in Li⁺ cells are governed by strongly coupled, highly nonlinear system of partial differential equations. Although simplifications can be made to the mathematical model of electrochemistry in the Li⁺ battery, the simulation of such simplified processes is still computationally expensive. For this reason, an efficient implementation of the mathematical model is needed. Due to the nature of materials inside a cell several simplifying assumptions have been made, often applied in the field of battery modelling, to enable computational simulation of the electrical and chemical processes inside a cell. One crucial assumption is made: all electrode particles are spheres of radius $R_{s,i}$, where $i \in \{a, c\}$ denotes the anode and cathode domain. This results in a simplified one-dimensional diffusion equation.

This work models the electrochemistry inside the battery using the Doyle-Fuller-Newman (DFN) model [55]. Figure 4.3 shows the model domain schematically, including the layered structure of a cell, as well as the sub-domains annotations.



Figure 4.3: Schematic view of the battery cell sub-domains: Ω_l is the electrolyte, Ω_a and Ω_c are the anode (negative) and cathode (positive) electrode areas, $\Omega_{s,a}$ and $\Omega_{s,c}$ are the electrodes' particles

Each electrode is represented by homogeneously distributed spherical particles as the limiting factor, connected via electrolyte. The model equation system results in

$$\frac{\partial c_{s,i}}{\partial t} - \frac{1}{r^2} \nabla \cdot \left(r^2 D_{s,i} \nabla c_{s,i} \right) = 0 \qquad \text{in} \quad \Omega_{s,i} \times \Omega_i$$

$$\epsilon_l \frac{\partial c_l}{\partial t} - \nabla \cdot \left(\frac{RT}{F^2} t + \frac{\kappa \left(c_l \right)}{c_l} \nabla c_l + t_+ \frac{\kappa \left(c_l \right)}{F} \nabla \phi_l \right) = A_i j_{BV} \qquad \text{in} \quad \Omega_i \qquad (4.5)$$

$$- \nabla \cdot \left(\frac{RT}{F} \left(2t_+ - 1 \right) \frac{\kappa \left(c_l \right)}{c_l} \nabla c_l + \kappa \left(c_l \right) \nabla \phi_l \right) = F A_i j_{BV} \qquad \text{in} \quad \Omega_i$$

where c_s denotes the concentration of lithium inside the solid electrode, R is the universal gas constant, F is the Faraday constant, j_{BV} is the electrode current density defined by the Butler-Volmer expression, Eq. (4.7), r is the spheres' radial dimension and D_s is the solid diffusion coefficient in electrode i, with $\Omega_{s,i} = (0, R_{s,i})$. Due to the homogenous distribution of the particles and the assumption of a small dimension orthogonal to the layered structure, a single one-dimensional cut through the electrolyte domain models the electrolyte geometry, i.e. $\Omega_l = (0, L)$, $\Omega_a = (0, L_a)$ and $\Omega_c = (L - L_c, L)$ see Figure 4.3.

The constant inner surface $A_i = 3\epsilon_s/R_{s,i}$ arises as the constant particle surface to particle volume ratio, ϵ_l and ϵ_s denote the active volume fraction in the liquid and solid phase, respectively, t_+ is the charge transfer constant and $\kappa(c_l)$ is the conductivity in the liquid phase. Taking into account all electrolyte and electrode quantities allows to set the particle boundary condition to

$$D_{s,i}\frac{\partial c_{s,i}}{\partial \mathbf{n}} = j_{BV,i}\left(\phi_l, \phi_s, U_{OCP}\left(c_s\right)\right) \quad \text{on} \quad \Gamma_{s,i} \tag{4.6}$$

$$j_{BV,i} = i_0 (c_s) \left(\frac{c_l}{c_{l,0}} \exp\left(\frac{\alpha F}{RT} \left(\phi_s - \phi_l - U_{OCP} \left(c_s\right)\right)\right) \right) - i_0 \left(\exp\left(\frac{-1 \left(1 - \alpha\right) F}{RT} \left(\phi_s - \phi_l - U_{OCP} \left(c_s\right)\right) \right) \right)$$
(4.7)

where ϕ_s denotes the electrode potential and $U_{OCP}(c_s)$ is the open circuit potential of the electrode at a given lithium concentration c_s . The model of $U_{OCP}(c_s)$ utilized in this work is based on the Redlich-Kister expansion [56] given as

$$U_{OCP}\left(\xi\right) = E_0 + \frac{RT}{F} \ln\left(\frac{1-\xi}{\xi}\right) + \frac{RT}{F} \sum_{k=0}^n A_k \left(\left(2\xi - 1\right)^{k+1} - \frac{2\xi k \left(1-\xi\right)}{\left(2\xi - 1\right)^{1-k}}\right)$$
(4.8)

 $\xi = c_s/c_{total}$ is a measure for the lithiation state of an intercalation electrode. The system of partial differential equations were dicretized using the finite element method. For breviety, the complete derivation and numerical implementation of this model is not presented in this work but can be found in the work of Pichler [57].

4.3 Fast Multipole Method for Potential Flow Problems

Aerodynamic shape optimization has been an active area of research since the 1980s. Traditional methods typically involve parameterizing the geometry and solving the flow field around a geometry to obtain certain aerodynamic coefficients be optimized. The flow field analysis used can significantly influence the results of the design optimization. While high-fidelity computational fluid dynamics (CFD) solvers, such as Reynolds Averaged Navier-Stokes (RANS) solvers [58], can yield accurate flow fields, they typically require days or weeks of computing time. Certain simplifications can be made to the RANS model, such as neglecting viscosity [59], to accelerate the solution but even this can take hours or days. Such methods also require additional computing time to construct a computational volume mesh.

In cases where the computational time is of more importance than the accuracy, lower fidelity panel methods can be used to accelerate the flow field calculation. Panel methods [60] solve the linearized potential flow equation using indirect boundary element method. This approach only requires the boundary, on which the flow variables are required, to be meshed, thereby reducing the total

computational cost of each analysis. The linear potential flow equation assumes steady-state, inviscid, irrotational flow with no body forces. The derivation of the linearized potential flow equation and the boundary integral equation are presented herein.

Let u, v and w be the velocities in the x, y and z direction respectively. For three-dimensional steady-state flows the continuity equation can be written as

$$\frac{\partial}{\partial x}\left(\rho u\right) + \frac{\partial}{\partial y}\left(\rho v\right) + \frac{\partial}{\partial z}\left(\rho w\right) = 0 \tag{4.9}$$

and the steady-state momentum balance equation, neglecting viscous effects and body forces, as

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla p}{\rho} \tag{4.10}$$

where p is the pressure and ρ is the density. Since the speed of sound, a, can be expressed as $a^2 = \partial p / \partial \rho$, using the chain rule, the pressure gradient can be written as

$$\nabla p = \frac{\partial p}{\partial \rho} \cdot \nabla \rho = a^2 \nabla \rho \tag{4.11}$$

Substituting Eq. (4.11) into Eq. (4.10) yields

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{a^2}{\rho} \nabla \rho \tag{4.12}$$

Multiplying the x, y and z components of Eq. (4.12) by u, v and w, adding them and substituting Eq. (4.9) into the resulting equation yields [61]

$$\left(\frac{u^2}{a^2} - 1\right)\frac{\partial u}{\partial x} + \left(\frac{v^2}{a^2} - 1\right)\frac{\partial v}{\partial y} + \left(\frac{w^2}{a^2} - 1\right)\frac{\partial w}{\partial z} + \frac{uv}{a^2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{uw}{a^2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) + \frac{vw}{a^2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) = 0$$

$$(4.13)$$
The velocity potential for irrotational flows can be expressed as $\mathbf{u} = \nabla \Phi$. Using this definition of the velocity potential, Eq. (4.13) can be rewritten as

$$\left(\frac{u^2}{a^2} - 1\right)\frac{\partial^2\Phi}{\partial x^2} + \left(\frac{v^2}{a^2} - 1\right)\frac{\partial^2\Phi}{\partial y^2} + \left(\frac{w^2}{a^2} - 1\right)\frac{\partial^2\Phi}{\partial z^2} + \frac{2uv}{a^2}\left(\frac{\partial^2\Phi}{\partial x\partial y}\right) + \frac{2uw}{a^2}\left(\frac{\partial^2\Phi}{\partial x\partial z}\right) + \frac{2vw}{a^2}\left(\frac{\partial^2\Phi}{\partial y\partial z}\right) = 0$$

$$(4.14)$$

Using the notation $\Phi_x = \partial \Phi / \partial x$ and $\Phi_{xx} = \partial^2 \Phi / \partial x^2$, Eq. (4.14) can be written compactly as

$$\left(\frac{u^2}{a^2} - 1\right)\Phi_{xx} + \left(\frac{v^2}{a^2} - 1\right)\Phi_{yy} + \left(\frac{w^2}{a^2} - 1\right)\Phi_{zz} + \frac{2uv}{a^2}\Phi_{xy} + \frac{2uw}{a^2}\Phi_{xz} + \frac{2vw}{a^2}\Phi_{yz} = 0$$
(4.15)

Let us consider the flow to be primarily in a single dominant direction x, with relatively small components of velocity in y and z directions. Then it can be assumed that $v/a \ll 1$ and $w/a \ll 1$. This assumption reduces Eq. (4.15) to

$$(1 - M_{\infty}^2) \Phi_{xx} + \Phi_{yy} + \Phi_{zz} = 0$$
(4.16)

where $M_{\infty} = u/a$ is the freestream Mach number. Using the small perturbation theory, the total velocity potential Φ can be represented as $\Phi = V_{\infty}x + \phi$, where ϕ is the perturbation velocity potential [61]. Then Eq. (4.16) can be written for perturbation velocity as [62]

$$(1 - M_{\infty}^2)\phi_{xx} + \phi_{yy} + \phi_{zz} = 0$$
(4.17)

The elliptic partial differential equation presented in Eq. (4.17) can be converted into Laplace's equaton using the Prandtl-Glauert transformation [63]. Let us define the Prandtl-Glauert factor, β as $\beta = \sqrt{1 - M_{\infty}^2}$. Using the geometrically scaled properties, $\bar{x} = x$, $\bar{y} = \beta y$, $\bar{z} = \beta z$ and $\bar{\phi} = \beta^2 \phi$, Eq. (4.17) can be written as

$$\nabla^2 \bar{\phi} = 0 \tag{4.18}$$

The unscaled properties can be obtained from the scaled properties using Gothert's rule [64].

The Laplace's equation can be transformed into a boundary integral equation (BIE) and can be solved using the boundary element method.

The equivalent boundary integral equation form of the Laplace's equation can be stated as [65, 66], Eq. (4.19)

$$c\left(\vec{x}\right)\phi\left(\vec{x}\right) = \int_{S} \left[\frac{\partial\phi_{f}\left(\vec{y}\right)}{\partial n_{y}}G_{L}\left(\vec{x},\vec{y}\right) - \frac{\partial G_{L}\left(\vec{x},\vec{y}\right)}{\partial n_{y}}\phi_{f}\left(\vec{y}\right)\right]dS\left(\vec{y}\right)$$
(4.19)

where Green's function, $G_L(\vec{x}, \vec{y}) = 1/4\pi |\vec{x} - \vec{y}|$, is the fundamental solution to Eq. (4.18), $c(\vec{x}) = 0.5$ for smooth boundaries, and the boundary condition is given by the surface velocity as $\partial \phi_f(\vec{y}) / \partial n_y = -\vec{U}_{\infty} \cdot \vec{n}$.

It should be noted that the fundamental solution is non-zero everywhere and tends to zero only at infinity. If using the standard boundary element method, this characteristic requires the storage and solution of a dense, non-symmetric linear system. The dense nature of the coefficient matrix requires $O(N^2)$ storage whereas the solution of the linear system using direct method requires $O(N^3)$ operations. This computational bottleneck can be avoided using an iterative solver and the Fast Multipole Method (FMM) [67] to accelerate the matrix-vector product.

4.3.1 The Fast Multipole Method

In recent years, the Fast Multipole Method (FMM) has been used to accelerate the solution of boundary integral equations. The FMM approach has been used for elastodynamics [68], heat transfer [69], fluid dynamics [70] and aeroacoustics [71], to name a few. Solution of boundary integral equations typically require the solution of a dense system. If direct linear solvers are used, it would require $O(N^3)$ operations and $O(N^2)$ storage space, where N is the number of nodes. Iterative solvers can be used to accelerate the solution of the linear system, but this would require explicit dense matrix-vector multiplication. Both, the large number of operations required to solve the linear system and the storage space required, can be overcome using the fast multipole method. The fast multipole method never explicitly computes or stores the coefficient matrix. Instead, it directly computes the matrix-vector product. The fast multipole method accelerates the matrix-vector product by partitioning the analysis domain and using direct, high-fidelity calculations for the nearfield elements and approximations for farfield elements. Therefore, the FMM approach requires O(N) operations to solve the linear system and O(N) space for storage. The derivation of the FMM approach for the Laplace's equation is presented herein.

For the fast multipole method to be applicable, the Green's function must first be expanded as [72]

$$G_L\left(\vec{x}, \vec{y}\right) = \frac{1}{4\pi} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \overline{S_n^m}\left(\vec{ox}\right) R_n^m\left(\vec{oy}\right)$$
(4.20)

where o is the center of expansion. This expansion decouples the source point x and field point y, allowing for the influences of the nearby elements to be handled differently to those at farfield. It can be seen from Eq. (4.19) that the integration is with respect to $dS(\vec{y})$. Therefore, only the R_n^m terms in the multipole expansion need to be integrated.

The solid harmonics functions R_n^m and S_n^m , where $\overline{S_n^m}$ is the complex conjugate of the function $\overline{S_n^m}$, are defined as

$$R_n^m\left(\vec{X}\right) = \frac{1}{(n+m)!} P_n^m\left(\cos\theta\right) e^{im\phi} r^n \tag{4.21}$$

$$S_n^m\left(\vec{X}\right) = (n-m)! P_n^m\left(\cos\theta\right) e^{im\phi} \frac{1}{r^{n+1}}$$
(4.22)

where P_n^m is the associated Legendre function defined in spherical coordinates r, ϕ and θ of some vector \vec{X} , which can be \vec{ox} or \vec{oy} for example. The vectors \vec{ox} and \vec{oy} point from the box center, \vec{o} , to a farfield source location, \vec{x} , and a nearfield source location, \vec{y} , respectively. The box centers \vec{o} are taken to be the centroid of the boxes obtained from space partitioning (Quadtree or Octree decomposition). The two solid harmonics functions together construct the spherical harmonics functions, which form an infinite set of orthogonal functions on a sphere and thus may be used to represent functions defined on the surface of a sphere.

The boundary integral in Eq. (4.19) can be written using the expansion given in Eq. (4.20) as

$$\int_{S} \left[\frac{\partial \phi_f(\vec{y})}{\partial n_y} G_L(\vec{x}, \vec{y}) - \frac{\partial G_L(\vec{x}, \vec{y})}{\partial n_y} \phi_f(\vec{y}) \right] dS = \frac{1}{4\pi} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \overline{S_n^m}(\vec{ox}) M_n^m(\vec{o})$$
(4.23)

Therefore, the multipole expansions, $M_{n}^{m}\left(\vec{o}\right)$ can be written as

$$M_{n}^{m}\left(\vec{o}\right) = \int_{S} \left[\frac{\partial\phi_{f}\left(\vec{y}\right)}{\partial n_{y}}R_{n}^{m}\left(\vec{oy}\right) - \frac{\partial R_{n}^{m}\left(\vec{oy}\right)}{\partial n_{y}}\phi_{f}\left(\vec{y}\right)\right]dS$$
(4.24)

The multipole to multipole (M2M), Eq. (4.25), multipole to local (M2L), Eq. (4.26), and local to local (L2L), Eq. (4.27), expansion and translation operators [72, 71] are given as

$$M_{n}^{m}(\vec{o}') = \sum_{n'=0}^{\infty} \sum_{m'=-n'}^{n'} R_{n'}^{m'} \left(\vec{o'o}\right) M_{n-n'}^{m-m'}(\vec{o})$$
(4.25)

$$L_{n}^{m}(\vec{o}') = \sum_{n'=0}^{\infty} \sum_{m'=-n'}^{n'} \overline{S_{n+n'}^{m+m'}} \left(\vec{oo'}\right) M_{n'}^{m'}(\vec{o})$$
(4.26)

$$L_{n}^{m}(\vec{o}') = \sum_{n'=0}^{\infty} \sum_{m'=-n'}^{n'} R_{n'-n}^{m'-m} \left(\vec{oo'}\right) L_{n'}^{m'}(\vec{o})$$
(4.27)

where $(\vec{oo'})$ represents a vector pointing from o to o'. Once the local expansions are obtained, the Laplace's BIE can be written as

$$\int_{S} \left[\frac{\partial \phi_f(\vec{y})}{\partial n_y} G_L(\vec{x}, \vec{y}) - \frac{\partial G_L(\vec{x}, \vec{y})}{\partial n_y} \phi_f(\vec{y}) \right] dS = \frac{1}{4\pi} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} R_n^m(\vec{ox}) L_n^m(\vec{o}) \quad (4.28)$$

The procedure for implementation for the fast multipole method is as follows:

- 1. Partition analysis domain: The computational domain is first partitioned using an iterative hierarchical domain decomposition method [73] (Octree in 3D and Quadtree in 2D). For a 2D example, at the first iteration a simple square encompassing the 2D domain is created. The square is then decomposed into 4 smaller squares of equal sizes. This decomposition is continued until the number of nodes/elements within a square is less than some specified number. It should be noted that only those squares that contain more than the allowable number of nodes/elements are decomposed. This decomposed. This decomposition creates a hierarchical tree where the childless boxes are called leaves and all other boxes are called parent boxes. Figure 4.4 shows the decomposition and sample hierarchical tree created for a 2D problem.
- 2. Compute multipole expansion: Once the computational domain is decomposed, the multipole expansions must be computed in each leaf cell, level three in Fig. 4.4b, using Eq. (4.24) using the elements within the leaf. In Eq. (4.24), the known solution parameters are taken as those specified

from the boundary conditions and the unknowns are taken as the initial guess.

- 3. Translate multipole moments upward: The multipole moments computed in the leaves are then translated up the hierarchical structure to the parent cells from its children cells using the M2M translation, Eq. (4.25). The moments are translated up to the top tree level, level 0 in Fig. 4.4b, and is known as the upward pass.
- 4. Translate multipole moments downward: The multipole moments computed at the top level are then translated down the hierarchical structure using a combinating of L2L Eq. (4.26) and M2L Eq. (4.27) translation. The L2L translation is used when translating the local moment from the parent down to its children. The M2L translation is used to translate the multipole moments from the cells in the interaction list to the local moment. The moments are translated down to the lower level and is known as the downward pass.
- 5. Direct Evalution: While the farfield influence is modeled using the multipole expansion, the influence of nearfield elements is evaluated directly using Eq. (4.19). Nearfield elements are those that are in the leaves that neighbor the current leaf. It should be noted that direct evaluation is only performed at the leaf level.
- 6. Local Expansion: Once the local moments are translated to the leaves, the boundary integral at each source point \$\vec{x}\$ can be obtained using Eq. (4.28). This is done at each collocation point present in that leaf cell.



Figure 4.4: An example of: a) quadtree decomposition and b) sample hierarchical tree structure

For a more detailed description about the multipole translation operators and the nearfield-farfield decomposition list, the reader is referred to [67, 72]. The upward and downward pass together implicitly yield the matrix-vector product of the coefficient matrix and the vector of field variable and its gradient. Therefore, an iterative solver can be used to accelerate the solution of the linear system due to the acceleration in the matrix-vector multiplication. The fast multipole method implementation in this work uses multi-level tree structure, is parallelized and makes use of constant elements. It can, however, be easily extended to include higher order elements. The direct integration of the neighbouring elements is performed analytically [74]. The GMRES iterative solver [75, 76, 77] is used to solve the linear system.

4.3.2 Verification of Potential Flow Solver

The developed potential flow solver is verified on an analytical solution of flow around an ellipsoid [78]. For an ellipsoid with principle axis a, b and c, the velocity potential that satisfies Eq. (4.18) is given by

$$\phi(x, y, z) = U_{\infty} \left[\left(\frac{\alpha}{2 - \alpha} + 1 \right) x + \left(\frac{\beta}{2 - \beta} + 1 \right) y + \left(\frac{\gamma}{2 - \gamma} + 1 \right) z \right]$$
(4.29)

where

$$\alpha = abc \int_0^\infty \frac{dx}{(a^2 + x)\sqrt{a^2 + x} + (b^2 + y) + (c^2 + z)}$$
(4.30)

$$\beta = abc \int_0^\infty \frac{dy}{(b^2 + y)\sqrt{(a^2 + x) + (b^2 + y) + (c^2 + z)}}$$
(4.31)

$$\gamma = abc \int_0^\infty \frac{dz}{(c^2 + z)\sqrt{(a^2 + x) + (b^2 + y) + (c^2 + z)}}$$
(4.32)

Figure 4.5 shows the analytical solution and the FMM solution of an ellipsoid subject to the Neumann boundary condition. It can be seen that the FMM solver produces accurate results, even when using constant boundary elements.



Figure 4.5: Verification of potential flow solver on a) an ellipsoid, subject to b) normal gradient boudary condition, showing c) FMM solution and d) the analytical solution

The computational time required to solve the BIE using the standard boundary element method and the fast multipole method is compared on a circular geometry with different mesh sizes. Figure 4.6 shows this comparison where Fig. 4.6a shows the octree decomposition of the sphere used in the FMM. Figure 4.6b shows the computing time required by the standard boundary element method, serial fast multipole method (FMM) and parallel fast multipole method using two cores (FMM_p). It can be seen that the computational time for the standard boundary element method increases cubically while the computation times for the fast multipole methods increases linearly with the number of elements. This decrease in computing time allows for more designs to be analyzed and objective functions, that are required for the optimization, to be calculated in a shorter time.



Figure 4.6: Illustration of a) octree decomposition of a sphere and b) computing time required to solve the potential flow by different methods for different mesh sizes

4.4 Modified Newton Impact Theory for Hypersonic Flow

Analysis of subsonic and supersonic flows typically require the solution of the continuity, momentum balance and energy balance equations. For hypersonic flows, chemical reactions and heat transfer also need to be accounted for. This drastically increases the computational cost for a high-fidelity hypersonic flow analysis. Because design optimization typically requires tens-of thousands-of flow field calculations, an efficient model that accurately captures the major aerodynamic phenomenon is required. One such model is based on Newtonian dynamics. The aerodynamics of hypersonic vehicles can be efficiently modeled using modified Newton impact flow theory (MNIT) [79, 80]. The MNIT approach is extremely simple, fast and accurate for hypersonic regimes. The Newton impact theory has been shown to model reality

when freestream Mach, M_{∞} , tends to infinity and the ratio of specific heats, γ , tends to one.

The MNIT models the local coefficient of pressure on the surface of an inclined panel as a function of the square of the local panel inclination angle and the maximum coefficient of pressure. This model assumes the flow impacting surface looses all momentum normal to the surface and then move tangentially to the surface. For a surface inclined at an angle θ to the freestream, the change in normal velocity is $\Delta V_n = V_{\infty} \cdot \sin \theta$. The mass flux on the inclined surface is then $\dot{m} = \rho_{\infty} V_{\infty} A \sin \theta$, where A is the surface area and ρ_{∞} is the freestream density. The time rate of change of momentum can be calculated as

$$\dot{\mathbf{p}} = \rho_{\infty} V_{\infty}^2 A \sin^2 \theta \tag{4.33}$$

Using Newton's second law, which states that the time rate of change of momentum is equal to the force exerted, the force F, on the surface is computed using Eq. (4.33). The pressure on the surface can be stated as $F/A = p - p_{\infty}$, where p_{∞} is the freestream pressure. Using the definition of dynamic pressure, $q = 0.5\rho V_{\infty}^2$, and rearranging, yields the standard Newton impact theory definition of the surface coefficient of pressure.

$$C_p = 2\sin^2\theta \tag{4.34}$$

Equation (4.34) can also be written in terms of ϕ , ($\phi = 90 - \theta$), the angle between freestream and the surface normal, as

$$C_p = 2\cos^2\phi \tag{4.35}$$

This formulation was modified by Lees [79], Eq. (4.36), which replaced the coefficient of sine squared term in Eq. (4.34), with the coefficient of pressure at the stagnation point behind a normal shock.

$$C_p = C_{p0} \sin^2 \theta \tag{4.36}$$

The coefficient of pressure at the stagnation point, C_{p0} , is given as

$$C_{p0} = \frac{2}{\gamma M_{\infty}^2} \left[\left(\frac{\gamma + 1}{2\gamma M_{\infty}^2 - \gamma + 1} \right)^{\frac{1}{\gamma - 1}} \left(\frac{\gamma + 1}{2} M_{\infty}^2 \right)^{\frac{\gamma}{\gamma - 1}} - 1 \right]$$
(4.37)

It should be noted that Eq. (4.36) is only used in the frontal region of the object shown in Fig. 4.7. The frontal region is the region that is directly exposed to the oncoming freestream.

The pressure coefficient can also be defined in a standard form as

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\gamma p_\infty M_\infty^2} \tag{4.38}$$

Since the pressure in a vacuum is zero (p = 0), the coefficient of pressure in a vacuum becomes [81]

$$C_p = \frac{-2}{\gamma M_\infty^2} \tag{4.39}$$

Equation (4.39) is used to compute the coefficient of pressure on the panels in the shadow region. The force on a panel is defined as $F = -pA\hat{n}$, where A is the panel area, \hat{n} is the panel unit normal and p is the pressure on the panel defined as

$$p = p_{\infty} + \frac{1}{2} C_p \gamma p_{\infty} M_{\infty}^2 \tag{4.40}$$



Figure 4.7: Haack-Sears body showing the frontal and "shadow" region

4.4.1 Validation of Modified Newton Impact Theory Solver

The developed MNIT solver is first validated using experimental data of a sphere in a hypersonic wind tunnel for increasing Mach numbers. Figure 4.8a shows the triangulated sphere geometry used. Figure 4.8b shows the coefficient of drag obtained using the developed MNIT solver and the experimental set up of Masson et. al [82]. It can be seen that the numerical value tends towards the experimental mean for higher Mach numbers. This is expected since the MNIT is only valid within the hypersonic regime.



Figure 4.8: Validation of modified Newton impact theory solver on a) triangulated sphere, showing b) experimentally and numerically computed coefficient of drag

The developed MNIT solver is also validated against experimental data obtained from the wind tunnel [83] for a half-sphere-cone geometry shown in Fig. 4.9a. The radius of curvature for the hemispherical nose measured 0.365in and the diameter of the semicircular rear plane measured 2.43in. The total length of the cone was 4.0in and the planform area was 6.025 in^2 . The model was analyzed at a freestream Mach number of 12.6, a Reynolds number (based on model length) of 4.9×10^5 and at various angles of attack. The lift and drag forces obtained using the MNIT solver were normalized using the planform area and dynamic pressure of 2320.78 psf. The experimentally obtained and numerically calculated coefficients of drag and lift are shown in Fig. 4.9c and Fig. 4.9b respectively. It can be seen that the MNIT solver is in good agreement with experimental measurements.



Figure 4.9: Illustration of a) Hypersonic half-sphere-cone waverider and its experimentally and numerically obtained: b) coefficient of lift and c) coefficient of drag

4.5 Numerical Modeling: Summary

This section presented the theory and numerical framework of the analysis tools used in this work. A versatile framework for geometry parameterization and morphing, that requires no knowledge for format, is presented. The electrochemical model of Doyle-Fuller-Newman was presented. An accelerated framework for the solution of potential flow around a subsonic body was developed and verified against an analytical solution of flow over an ellipsoid. It was shown that the fast multipole potential flow solver was 100 times faster than the standard boundary element method. A modified Newton impact theory solver was developed for the rapid analysis of hypersonic flow. The developed hypersonic solver was validated against experimental data. These developed numerical model will be used in the application of MOHO for real life test problems.

CHAPTER 5

PARAMETER ESTIMATION IN BATTERY MODEL

The applications of Lithium-Ion batteries have drastically increased over the last decade. With the continuous implementation of the Li-Ion (Li⁺) cells in household appliances, automotive, aerospace and defense industries, accurate modeling and simulation of them is paramount. Accurate analysis of the battery can at time require the internal state of the cell to be known. This internal state can include abstract quantities (e.g. state of charge (SoC) and state of health) and physical quantities (e.g. potentials and concentrations). Some of these quantities can be measured through experimentation. In several cases, the material properties of the cell can also be of interest. These material properties sometimes cannot be measured directly and must be estimated, often non-intrusively. This gives rise to the traditional parameter estimation problem. Parameter estimation techniques attempt to identify certain parameters in a model using only the model response. Parameter estimation techniques can be non-intrusive and non-destructive depending on whether the model response can be obtained non-intrusively and non-destructively. The parameter estimation problem in this work can be stated as follows: Given only the voltage, how can the material properties and model parameters of the Lithium-Ion cell model be estimated?

Santhanagopalan et al. [84] used the Levenberg-Marquardt algorithm to identify five parameters in the Doyle-Fuller-Newman (DFN) under constant charge and discharge conditions. Scharrer et al. [85] made use of a space-mapping parameter surrogate model to the DFN model to successfully identify three parameters. Their work made use of a Morris-One-At-A-Time sensitivity analysis to identify the three most sensitive parameters in the model.

Forman et al. 86 performed parameter identification of 88 parameters using a genetic algorithm. To date, this is the latest attempt in estimating a significant number of parameters in the DFN model. Recently, Jin et al. [87] also performed sensitivity analysis to identify the five most sensitive parameters. They then used Levenberg-Marquardt algorithm to estimate the values of these five parameters. A parallel genetic algorithm was used by Zhang et al. [88] to identify 29 parameters in the pseudo-two-dimensional DFN model. They reported a computing time of 22.3 hours to identify the 29 parameters. Uddin et al. [89] estimated a total of three parameters in the DFN model using the differential evolution algorithm. Previous works have reported solution times ranging from 22 hours up to three In the work of Forman et al. weeks. [86] it was stated that the parameter estimation took approximately three weeks. This work drastically accelerates the parameter estimation of several parameters in the DFN model by using the MOHO algorithm in the single objective mode, called the Single-Objective Hybrid Optimizer (SOHO).

5.1 Problem Definition of Inverse Parameter Estimation

The measured voltage data obtained through cycling a Panasonic NCR18650B commercial cell is used in this work. The cell is first charged at C/3 rate (C-rate = 3.35A) until the voltage reaches 4.113V followed by a constant voltage charge at 4.113V until the current tapered down to 160mA (\approx C/20 rate), then discharged at C/3 rate until 3.498V again followed by a constant voltage discharge at 3.498V for 40 minutes or until the current dropped to 160mA, respectively. This is repeated three times, afterwards two full capacity estimation cycles according to the data sheet are executed: the cell is charged at C/2 rate until 4.2V, constant voltage charged at 4.2V until the current dropped to C/50 rate and discharged at 1C rate until 2.5V.

The cell is then charged to 3.498V again and discharged to a specific SoC level for a total of seven cycles (85%, 75%, 65%, 55%, 45%, 35% and 25%). At each level a set of current pulses are applied such that the dynamic behavior of the cell is reflected as much as possible in the voltage. The pulse sequence subsequently applies C/5, 1.25C and 1.35C pulses in charge (+) and discharge (-) direction for 10s, followed by 15 minutes rest after each pulse. The pulse sequence ends with a combined 5s-pulse sequence of +C/5, +C/5, -C/5, -C/5, -1.35C, +1C with 5s rest in-between and a discrete discharge/charge stair profile of 0.2C, 0.35C, 0.5C, 0.75C, 1.25C for 10s per level. Figure 5.1 shows the voltage and current measured throughout this time. All tests were done using an Arbin BT-2000 battery testing system and Memmert incubator with peltier cooling (model IPP600) for maintaining the temperature at 25°C by forced air cooling.



Figure 5.1: The cycling of the $\rm Li^+$ battery showing: a) the applied current and b) the measured voltage

The traditional approach to solve the parameter identification problem involves minimizing the difference between the measured response and predicted response. If the cell voltage curve obtained through experimental measurement is $V_E(t)$ and the cell voltage curve obtained by solving the mathematical model for a given parameter set is $V(\pi, t)$, then the correct parameter set π can be estimated by solving the optimization problem given by

$$\pi_{est} = \underset{\pi \in \Pi}{\arg\min} \int_{0}^{T} \left(V_{E}(t) - V(\pi, t) \right)^{2} dt$$
(5.1)

The minimization algorithm subsequently updates the parameter set π to minimize the error norm. It should be mentioned that each computation of that error norm requires the solution of the mathematical model using the given parameter set π . For this reason, an algorithm that can efficiently minimize the error with a few model evaluations is very appealing. This minimization algorithm must be robust and should be able to avoid local minima. For this reason, a newly developed hybrid optimizer is used to solve the above optimization problem.

The DFN-model used in this work is defined using 44 parameters. The parameters to be estimated are: the separator resistance, along with the particle radii, diffusion coefficients, reaction rates and active mass of both the cathode and the anode, electrode area, separator porosity and the tortuosity of the cathode, anode and separator. A total of 15 terms (n = 15) in the Redlich-Kister expansion are used to define the OCP curve for each the anode and the cathode. The first RK coefficient for the anode is always set to zero. Thus, the total number of RK coefficients is 29 for both the anode and the cathode.

The parameter estimation problem is solved by minimizing the L^2 -norm of the difference between the calculated and measured voltage curves. The calculated curve is obtained by solving the mathematical model while the measured voltage

curve is obtained experimentally. It should be mentioned that the so-called inverse crime [90] is avoided in this work since the two voltage curves are obtained using different methods and because of the inherent measurement errors present in the experimentally obtained voltage curve. The SOHO algorithm will search for the parameters, within a user-specific bound, that best minimizes this L^2 -norm.

It should be mentioned that the solution of the DFN model was terminated if either the time step became less than 10^{-6} or if the maximum allowable working time was exceeded. This greatly reduces computing time as infeasible parameter combinations runs are not evaluated. These termination criteria add additional degree of non-linearity and discontinuity to the cost function space. It also adds several flat regions where the gradient is zero. For this reason, a gradient based method will find it very difficult to converge to the correct values of the model parameters. The SOHO algorithm is not affected by any of these function space modifications.

5.2 Parameter Estimation in DFN Model

The DFN model was defined using a total of 44 parameters. The results of the parameter estimation problem are presented in this section.

Figure 5.2a shows the convergence history for the DFN model estimation problem, where the residual is defined as the L^2 -norm of the error between the measured and calculated voltage. Here, the residual is seen to sharply decrease within the first 10,000 evaluations. Figures 5.2b, Fig. 5.2c and Fig. 5.2d show error distributions at three different locations along the convergence history. It can be seen that even parameter sets in the early regions of the convergence history (Case 1) have majority of the errors within 50mV, with a significant number of



them centered close to the 0mV region. In the Case 3, the majority of the errors are within 25mV.

Figure 5.2: Parameter estimation of the DFN model showing: a) the convergence history of the SOHO algorithm, b) error probability distribution of Case 1, c) error probability distribution of Case 2, d) error probability distribution of Case 3.

The error statistic and the convergence information of the three selected cases of the DFN model are shown in Table 5.1. Even though the computing time of the DFN-model is of considerable magnitude, the SOHO algorithm is able to estimate the parameters in the DFN model in less than one day. This is a significant improvement in convergence time over the previous studies, which took approximately three weeks to obtain converged results. It should be mentioned that the computing time for the Newman model used in [86], i.e. 63 seconds, is similar to the DFN model used in this work (30 seconds on average).

| | Case 1 | Case 2 | Case 3 |
|---|--------|--------|--------|
| Evaluations to convergence | 8700 | 21100 | 83500 |
| Time to convergence (s) | 5455 | 13330 | 52354 |
| Mean absolute error (mV) | 18.91 | 15.82 | 6.47 |
| Standard deviation of absolute error (mV) | 24.49 | 10.95 | 7.16 |

Table 5.1: Error statistics of the three selected cases in parameter identification

Figure 5.3 shows the estimated voltage and measured voltage using the 44 converged parameters. It can be seen that the results of the DFN model are similar to those measured. The charging and discharge peaks coincide well for the entire time range.



Figure 5.3: Measured and estimated voltage response obtained using the converged DFN model

This shows that SOHO is able find optimum solutions even in a highly nonlinear higher dimensional space. The accelerated convergence of SOHO is clearly visible in the convergence history as well as the reduction in estimation times from three-weeks to under one day.

5.3 Parameter Estimation in Battery Model: Summary

This section utilized the Single-Objective Hybrid Optimizer (SOHO) to perform parameter estimation in a highly nonlinear electrochemical model of a Li⁺ battery. A total of 44 parameters that best minimized the difference between the measured and calculated voltage were identified. It was shown that the SOHO algorithm is able to converge to relatively low residual in only 5500 iterations. The converged parameters from the SOHO algorithm resulted in a mean absolute error of only 6.47 mV. The presented framework reduces the time required to estimate the parameters from three weeks to less than one day.

CHAPTER 6

OPTIMIZATION OF SUBSONIC BODIES

Aerodynamic shape optimization has been a popular application of optimization techniques, often pushing the capabilities of existing software. Aerodynamic design and analysis requires computational fluid dynamics (CFD) methods which are often time and memory consuming. Due to the increased computational cost, the optimization technique should find an optimum solution with few objective function calculations.

Several researchers have applied techniques of different complexity and fidelity in their pursuits of an optimum aerodynamic design. Dulikravich performed a review of approaches used for aerodynamic shape optimization and inverse design [91, 92]. Their survey demonstrated the effectiveness of accelerated, lower fidelty models as analysis tools in aerodynamic shape design. Skinner and Behtash [93] performed a comprehensive review of 229 research articles on the topic of aerodynamic shape design. They show that various researchers [94, 95] used the freeform deformation approach to parameterize the aerodynamic shape. The type of aerodynamic analysis selected for aerodynamic shape design can play a significant role in the robustness of the design. Where high fidelity analysis such as finite volume and finite elements is prohibitively expensive, panel methods can be used to accelerate the flow field calculations with lower fidelity. Panel methods that solve the potential flow equation have been previously been used for aerodynamic shape design [96, 97, 98, 99]. Although fast multipole methods exist to solve the potential flow around a body [100, 101], Skinner and Behtash's [93] review does not mention any work involving aerodynamic shape optimization using the fast multipole method.

Most non-gradient aerodynamic shape optimization works use surrogate models to accelerate the computation of the objective functions. Using accelerated CFD techniques can decrease the time required to design an optimized shape. This section utilizes the fast multipole method, presented in Section 4.3, to compute the flow field variables on the aircraft's surface. The results are compared to those obtained using surrogate based optimization. The aerodynamic shape optimization in this section attempts to optimize the wing design of an aircraft for maximum lift and minimum drag and moment. The problem formulation and the results of the deterministic and robust optimization are presented herein.

6.1 Problem Definition of Subsonic Aerodynamic Shape Design

The tools developed and presented thus far are applied to the aerodynamic optimization of a subsonic aircraft. The wing design of a baseline aircraft geometry is modified to maximize the coefficient of lift and minimize the coefficients of drag and moment.

The baseline aircraft features a wing span of 28.8*m*, a combined surface area, A, of both wings of 135 m^2 and a mean aerodynamic chord (MAC) length of 6*m*. The aircraft is assumed to be traveling at a Mach number of 0.6 ($U_{\infty} = 205.8m/s$) at a 0° angle of attack. The freestream pressure, p_{∞} , and air density, ρ_{∞} , are held constant at 1 *atm* and 1.225 kg/m^3 respectively.

The freeform deformation approach presented in Section 4.1 is used to deform the geometry using a total of 12 control points; six on the upper surface and six on the lower surface. The deformations were kept symmetric with respect to the plane along the center line of the aircraft to ensure symmetric wing deformation. The 12 control points were only allowed to move either upwards or downwards in the z-direction by at most 0.25m. Figure 6.1 shows the lattice encapsulating one of the wings and the possible deformations that are possible with this approach.



Figure 6.1: 3-D view of the baseline aircraft geometry with: a) the FFD lattice used, b) sample deformations on the upper surface of the wing and c) sample deformations on the lower surface of the wing

The fast multipole method was used to solve the potential flow equation on the surface of the aircraft. Since the potential flow theory is invalid for transonic regime, the freestream Mach number was held at 0.6 to ensure no locally transonic regions develop on the aircraft surface. Once the velocity components are obtained using the FMM solver, the surface coefficient of pressure, C_p , can be computed using

$$C_p = 1 - \frac{\vec{V} \cdot \vec{V}}{U_{\infty}^2} \tag{6.1}$$

The aerodynamic coefficients, C_l , C_d and C_m , are computed using

$$C_{l} = \frac{F_{L}}{\frac{1}{2}\rho U_{\infty}^{2}A} \qquad C_{d} = \frac{F_{d}}{\frac{1}{2}\rho U_{\infty}^{2}A} \qquad C_{m} = \frac{M}{\frac{1}{2}\rho U_{\infty}^{2}AL_{MAC}}$$
(6.2)

where F_L , F_d and M are the lift force, drag force and the moment respectively. A surrogate model based on Hardy's multiquadrics radial basis functions [102] is used to model the approximate lift, drag and moment coefficients for a particular design. The shape factor in the radial basis function formulation is tuned using Rippa's approach [103]. A total of 1,000 designs were created, using Sobol's quasirandom sequence [104], and analyzed using the fast multipole method. These 1,000 designs were used to construct the three surrogate models, one each for coefficients of lift, drag and moment.

6.2 Deterministic Optimization of Subsonic Bodies

The aerodynamic optimization of the aircraft is first done within the deterministic framework. That is, no uncertainty was assumed in the control point locations. Due to the increased computational cost of the fast multipole solver over the surrogate model, the optimization using the potential flow solver was performed using an initial population of 92 members and was ran for 20 generations. The optimization using the surrogate was performed using 100 members and was ran for a total of 1,000 generations.

The three-dimensional Pareto front obtained using the potential flow solver and the surrogate model is shown in Fig. 6.2. It can be seen that the two approaches yield similar Pareto fronts. The extents of the Pareto front limits are larger for the case using the surrogate model due to the increased population size and generations of the optimization.



Figure 6.2: Pareto front, obtained by each algorithm, within the deterministic framework, for the subsonic shape optimization problem showing the a) objective function space and b) normalized objective function space

Figure 6.3 shows the deformations from the baseline geometry, the surface C_p for the baseline configuration and the two Pareto designs as well as the difference in C_p between the optimum and the baseline configuration. Here the term "optimum" means the "preferred" design. It should be mentioned that the entire Pareto front is, by definition, optimal and that an alternative design that best fits the designers' needs can also be selected. It can be seen that both the FMM and surrogate approach yield similar deformations. The resulting surface coefficient of pressure is also similar, therefore, yielding similar aerodynamic coefficients. Although the surface C_p appears similar in all three cases, the difference in C_p , ΔC_p , about the baseline shows stark differences. Firstly, it can be seen that the pressure on the upper surface of the wing is lower, contributing to increased lift. It can also be seen the pressure is lower towards the trailing edge of the wing, increasing the pitch-down moment and reducing the net moment about the aircraft's centroid. Therefore, the results of the optimization coincide well with theory.







Figure 6.3: Surface distribution of the deformation for design obtained using: a) the potential flow solver and b) the surrogate model; the surface pressure coefficient for: c) design obtained using the potential flow solver and d) design obtained using the surrogate model; the surface pressure coefficient difference between the baseline and: e) design obtained using the potential flow solver and f) design obtained using the surrogate model

Table 6.1 shows the objective functions of two designs selected from a similar region of the Pareto front. It can be seen that both the lift and moment objectives were achieved with only a slight increase in drag. The optimized design obtained using the surrogate was also analyzed using the fast multipole solver. It can be reported that the coefficients of lift, drag and moment using the potential flow solver were 0.1501, 4.22E-3 and -9.28E-3 respectively. The relative error from the potential flow solver for the values obtained from the surrogate for the coefficients of lift, drag and 3% respectively. This shows that the constructed surrogate is accurate.

Table 6.1: Objective functions values of the baseline aircraft and the two optimized designs obtained within the deterministic framework

| | C_l | C_d | C_m |
|-----------------|--------------|---------------|-----------------|
| Baseline | 0.1252 | 3.6E-03 | -0.281 |
| $Optimal_{FMM}$ | 0.1508~(20%) | 4.32E-3 (19%) | -9.05E-3 (-96%) |
| $Optimal_{RBF}$ | 0.1555~(24%) | 4.39E-3 (21%) | -9.91E-3 (-96%) |

6.3 Optimization of Subsonic Bodies Under Uncertainty

Aircraft components typically have a factor of safety of 1.4 to 1.5. Deviation from design conditions can significantly affect the performance and reliability of the design. Accounting for uncertainty can lead to a robust design whose performance does not degrade at off-design conditions. The previous section did not account for uncertainty in the geometric defects. In this section, the aerodynamic shape optimization problem is solved within the probabilistic framework.

This section introduces defects in geometry through uncertainty in control point locations. The uncertainty in geometric parameters, no matter how small, also introduce uncertainty in the flow solution and therefore the aerodynamic coefficients. The uncertainty in control points is modeled as additive and Gaussian with zero mean and some specified standard deviation. Due to the increased number of samples needed for uncertainty propagation and high computational cost of the fast multipole method, the probabilistic optimization problem is only solved using the surrogate model. This is not an issue as the surrogate was previously verified.

Although the deterministic analysis has been verified, the uncertainty propagation should also be verified for this particular problem. The stochastic moments obtained using the Polynomial Chaos Expansion framework are compared to those obtained using Monte Carlo sampling. The Monte Carlo sampling is performed using 14,000 and 1,400 points to see the effects of sampling size. The uncertainty in control points was assumed to be 10% of the maximum displacement ($\sigma = 0.025m$). Figure 6.4 shows the probability distributions of each objective for the baseline configuration and the previously selected Pareto designs from the deterministic optimization. Immediately it can be seen that the certain PCE methods are not able to correctly model the probability distribution.

The standard second order PCE yields a large standard deviation in the lift coefficient. Not only is the accuracy of the PCE function specific but it also appears to be domain specific. For example, the standard linear PCE performs well for modeling the lift uncertainty in the baseline configuration but under performs in modeling the probability distribution for the two Pareto designs. It can also be seen that the PCE performs similarly for all the objectives of a design.

Comparing the probability distributions in Fig. 6.4, it is evident that the sparse PCE performs best. This is because the over-fitting issue, that is present in most polynomial interpolation methods, is avoided. Over-fitting tends to give rise to oscillations as the interpolating polynomial tried to pass exactly through the samples. The sparse PCE only computes the most significant coefficients and therefore does not experience large local oscillations. For the case of robust aerodynamic shape optimization, the sparse PCE of second order will be used to model the uncertainty in the aerodynamic coefficients.











Figure 6.4: The probability density functions, obtained using different uncertainty quantification methods, of baseline configuration for : a) C_l , b) C_d and c) C_m ; configuration obtained using potential flow solver for: d) C_l , e) C_d and f) C_m ; configuration obtained using the surrogate for: d) C_l , e) C_d and f) C_m

Table 6.2 shows the stochastic moments for the designs obtained using the deterministic optimization framework. The designs obtained using robust optimization will be compared to these designs.

Coefficient of Lift Coefficient of Drag Coefficient of Moment Geometry σ μ σ μ σ μ Baseline 0.1531 9.42E-04 4.34E-032.50E-05-1.09E-02 5.57E-04

4.39E-03

4.11E-03

3.29E-05

1.24E-04

-6.23E-03

-1.09E-01

9.06E-04

4.01E-02

 $Optimal_{FMM}$

Optimal_{*RBF*}

0.1522

0.0394

8.85E-04

3.84E-02

Table 6.2: Mean and standard deviation of the aerodynamic coefficients for the subsonic designs obtained within the deterministic framework

Figure 6.5 shows the Pareto fronts obtained using 100 population members and 20 population members. The error bars for the standard deviation are easier to see when less members are visualized. Firstly, it can be seen that the two Pareto fronts are similar indicating that the optimizer is able to converge to the Pareto front even with a small search group. Figure 6.5b shows that the lower region of the Pareto front contains a large amount of uncertainty in the aerodynamic coefficients. The

area of low uncertainty can be seen to be populated by all algorithms. Three designs are selected from the Pareto front and analyzed. They are taken from the low drag, high lift and a middle section of the Pareto front.



Figure 6.5: Pareto front obtained by each algorithm, within the probabilistic framework, for the subsonic shape optimization problem using: a) 100 population members and b) 20 population members

Figure 6.6 shows the deformations about the baseline, surface pressure coefficient and the difference in pressure coefficient about the baseline. Table 6.3 shows the stochastic moments of the three designs. Figure 6.6 shows that the low drag and the trade-off design both feature larger deformations of similar magnitude near the wing tips. The low drag geometry yields a lower coefficient of lift, indicating higher pressure on the lower surface than the baseline geometry. The pressure on the upper surface is increased due to the optimized deformation, resulting in a lower pressure difference, especially at the wing tips. Although this reduction in pressure difference reduces the overall lift, it also reduces the induced drag due to wing tip effects. This is the reason that the low drag designs feature deformation at the wing tips. This is conversely true for the case of high lift. The majority of the deformations are inboard from the wing tip where the majority of the lift forces are produced. It can
be seen that the coefficient of pressure decreases on the upper surface of the wing, resulting in increased lift force. This in turn increases the induced drag due to wing tip vortices. Once again, the results of the optimization agree with theory.











Figure 6.6: Surface distribution of the deformation for: a) low drag design, b) high lift design and c) an optimal trade-off design; the surface pressure coefficient for: d) low drag design, e) high lift design and f) an optimal trade-off design; the surface pressure coefficient difference between the baseline and: g) low drag design, h) high lift design and i) an optimal trade-off design

Table 6.3: Mean and standard deviation of the aerodynamic coefficients for the subsonic designs obtained within the probabilistic framework

| | Coefficient of Lift | | Coefficient of Drag | | Coefficient of Moment | |
|-----------|---------------------|----------|---------------------|----------------------|-----------------------|-----------|
| Geometry | μ | σ | μ | σ | μ | σ |
| Low Drag | 0.1412 | 2.11E-03 | 4.15E-03 | 2.03E-05 | -3.37E-03 | 2.34E-03 |
| High Lift | 0.1800 | 7.29E-04 | 5.20E-03 | $3.87 \text{E}{-}05$ | -2.14E-02 | 5.67 E-04 |
| Optimal | 0.1508 | 1.10E-03 | 4.29E-03 | 2.62E-05 | -9.61E-03 | 6.59E-04 |

Figure 6.7 shows the probability density function of each objective for the three optimized designs and the baseline geometry. It is clear to see that the low drag design experiences large changes in the moment and lift coefficients. This is also shown in the Pareto front in Fig. 6.5b where the low drag region features significant uncertainty in the objectives. It should be mentioned that other Pareto designs can be selected from the Pareto front that best satisfies the design requirements. It is proof that the probabilistic design optimization framework developed can efficiently identify robust designs in all areas of the objective function space.



Figure 6.7: Probability density functions of aerodynamic coefficients for each design, obtained within the probabilistic framework: a) C_l , b) C_d and c) C_m

6.4 Optimization of Subsonic Bodies: Summary

This section presented the results of MOHO on a subsonic shape optimization problem. The freeform deformation toolbox was used to parameterize the aircraft wing geometry. The fast multipole solver was used to perform the external flow analysis and obtain the aerodynamic coefficients. The three simultaneous objectives were to maximize the coefficient of lift and minimize the coefficients of drag and pitching moment. The optimum designs obtained within the deterministic framework featured a 20% increase in lift and drag and a 96% decrease in pitching moment. The design problem was also solved within the probabilistic framework. The sparse polynomial chaos expansion approach was verified against results obtained using Monte Carlo sampling. It was shown that the sparse approach yield more stable result than the standard polynomial chaos expansion. The sparse polynomial chaos expansion was used to propagate uncertainty within the probabilistic optimization framework. It was shown that the robust configurations featured a smaller standard deviation than those design that were obtained within the deterministic framework. This showed that the MOHO algorithm is capable of solving problems with several objectives efficiently. It also shown that the uncertainty propagation.

CHAPTER 7

OPTIMIZATION OF HYPERSONIC BODIES

With current world events, several nations have invested significant resources in to the design of hypersonic bodies. The field of hypersonic aerodynamic design flourished between the 1970 and 1990 with great interest in hypersonic waveriders [105, 106]. Since then, the design considerations have shifted from waveriders to rockets and missiles.

Recent works in hypersonic nose-cone body optimization include the work of Cui and Yang [107] who used a simplex method of Nelder and Mead and a thin layer Navier-Stokes solver to design arc-wing missiles. The work of Lesieutre et. al [108] used a panel method with a Sequential Unconstrained Minimization Technique (SUMT) to perform multidisciplinary design optimization of missile configurations and fin planforms. Bowcutt [109] also performed multidisciplinary optimization of air-breathing hypersonic vehicles. Theisinger and Braun [110] performed a multiobjective optimization of a hypersonic entry aeroshell to maximize drag and stability upon reentry. Their work utilized a freeform deformation based on Non-Uniform Rational B-Splines (NURBS) to parameterize the model. The flow-field analysis was performed using the modified Newton impact theory. Foster and Dulikravich [111] performed a single objective optimization of a hypersonic nose-cone shape using a modified Newton impact theory solver and a hybrid genetic and gradient search algorithm. Sheffer and Dulikravich [112] used Pshenichny-Danilin gradient method to minimize drag while preserving length and volume of the hypersonic shape.

This section performs multi-objective aerodynamic shape optimization of a generic nose cone design to minimize the wave drag and maximize the volume. The hypersonic analysis is accelerated using the previously validated modified Newton impact theory presented in Section 4.4. The problem formulation and the results of the deterministic and robust optimization are presented herein.

7.1 Problem Definition of Hypersonic Shape Design

The MOHO suite is applied to develop optimized bodies for hypersonic flight. The optimization problem definition and formation is presented here.

This section considers two conflicting objectives for the design problem: minimize coefficient of drag and maximize volume. The maximize volume requirement is inspired by the requirement to carry more payload in the case of a suborbital rockets or more space for warheads in the case of intercontinental ballistic missiles (ICBM). The freeform deformation (FFD) presented in Section. 4.1 is used to deform the geometries where the control points are set as the design variables. Since the FFD approach deforms the original geometry, an inefficient aerodynamic body, a cylinder, is selected as the starting geometry. This means that the optimization will have to yield an efficient body despite starting from such an inefficient shape.

The cylinder and the FFD lattice used are shown in Fig. 7.1. The cylinder has a radius of 0.95m, a length of 18m and a discretized volume of $50.89m^3$.



Figure 7.1: The surface mesh and lattice points for the cylinder used as the starting point for hypersonic shape optimization

The velocity is held constant at Mach 14 (4800m/s). The surface is discretized using ≈ 13000 triangles. The coefficient of drag is computed using the MNIT solver, whereas the volume, V_{total} , is computed using only the surface mesh as

$$V_{i} = \frac{\vec{X}_{1} \cdot \left(\vec{X}_{2} \times \vec{X}_{3}\right)}{6}, V_{total} = \sum_{i=1}^{N} V_{i}$$
(7.1)

where V_i is the volume of the i^{th} triangle and X_i are the coordinates of the i^{th} vertex. The lattice points are only allowed to move in the normal direction ensuring that the shape is axi-symmetric. The lattice was created using a total of 13, 4 and 4 lattice points in the x, y, and z directions respectively. The set of control points at each xstation moved axially by the same distance to ensure axi-symmetry. This results in a total of 13 design variables, the normal displacement of the lattice points at each axial location. Each lattice point was allowed to move up to 2m either inwards or outwards.

7.2 Deterministic Optimization of Hypersonic Bodies

The performance of the MOHO suite for the design of an optimized hypersonic body was first investigated within the deterministic framework. That is, uncertainties in the volume and drag due to uncertainties in the shape were neglected. It should be mentioned that no surrogate model was used in this work since the MNIT solver is able to compute the drag in approximately one to two seconds. The optimization was performed in a "master-slave" parallel arrangement where the master performs all the recombination and selection whereas the slaves compute the objective functions for the designs in parallel.

Figure 7.2 shows the Pareto front for the deterministic problem. The shape of the Pareto front is as expected, where a larger volume design leads to more drag. It can be seen that the Pareto front obtained by MOHO is slightly higher than some of the other constitutive algorithms such as NSGA-III and SPEA-R. The performance of MOHO in this problem is comparable to that of NSDE-R1B and NSDE-D3. It should be mentioned that MOEA-DD performed the worst and failed to obtain the complete Pareto front for this problem, resulting in all of the Pareto designs to be concentrated in the high-drag-high-volume region of the front. The designs obtained by MOEA-DD in that region are still similar to those obtained by the other algorithms. Despite having algorithms that under performed for this case, the MOHO algorithm was able to converge to the Pareto front.



Figure 7.2: Pareto front, obtained by each algorithm, within the deterministic framework, for the hypersonic shape optimization problem showing the: a) objective function space and b) normalized objective function space

Figure 7.3 shows the Pareto fronts obtained by MOHO at different generations. It can be seen that in just 25 generations, MOHO is able to sufficiently converge to the final converged Pareto front at 300^{th} generation. The Pareto fronts obtained by MOHO for 50 and 50+ generations are almost indistinguishable. It can also be seen that the hypervolume at higher generations is drastically greater than the hypervolume due to the initial population. This shows that MOHO is able to converge to the Pareto front in fewer function evaluations.



Figure 7.3: Pareto front obtained by MOHO at different number of generations

Two designs were selected from the Pareto front; one from a low drag region and one from the optimal region. Here, the term "optimal" again refers to "preferred" designs. The region on the Pareto front where the two designs were selected from is shown in Fig. 7.2, and the corresponding geometry is shown in Fig. 7.4. In can be seen in both cases that the two geometries have a pointed nose cone, and the majority of the cross-section remains constant along the length. This is because the local coefficient of pressure is proportional to the local panel inclination angle. This constant cross-section along the length gives a zero panel inclination angle and therefore creates minimum drag for the body.



Figure 7.4: Pareto designs, obtained within the deterministic framework by MOHO, selected from the: a) minimum drag region and b) optimal region

The objective function values of the two geometries are given in Table 7.1. It can be seen that both geometries performed better in both objectives than the starting point.

Table 7.1: Objective functions for the starting shape and the two optimized hypersonic designs obtained within the deterministic framework

| Geometry | Coefficient of Drag | Volume (m^3) | |
|--------------|---------------------|----------------|--|
| Cylinder | 1.8077~(0%) | 50.8 (0%) | |
| Minimum Drag | 0.0549~(-97%) | 65.8 (+30%) | |
| Optimal | $0.1560 \ (-91\%)$ | 87.4 (+72%) | |

7.3 Optimization of Hypersonic Bodies Under Uncertainty

Although optimization under the deterministic framework can yield well converged results, it often becomes necessary to account for the uncertainty. In the previous section, Section 7.2, the uncertainty in volume and drag due to the uncertainties in geometry were neglected. In this section, they are taken into account. The uncertainty in geometry is incorporated through the uncertainty in FFD control points movement. In this case, the uncertainties in control points movement are considered additive and Gaussian with zero mean and a specified standard deviation.

The uncertainty associated with the volume and drag of the designs obtained using deterministic optimization is computed using various approaches. The uncertainties in control points, with $\sigma = 0.2m$, about their converged values, are propagated using Polynomial Chaos Expansion (PCE) and Monte Carlo (MC) sampling. Figure 7.5 shows the probability distribution functions (PDF) of coefficient of drag and volume, computed using different approaches. The Monte Carlo method is used here as a benchmark and it can be seen that it converges to a similar PDF with both 1,400 and 14,000 sampling points. The PDF obtained using Polynomial Chaos Expansions of first and second order are also shown, where the Linear and Quadratic represent the standard first and second order PCE respectively, while the Linear_{SP} and Quadratic_{SP} represent the sparse PCE of first and second order respectively.

It can be seen that although the standard PCE captures the mean of the PDF well, the standard deviation is typically larger than that obtained using MC. It is mainly the case for the PDF of coefficient of drag. The standard PCE using second order expansion is able to model the PDF of volume well for all three geometries, whereas the linear expansion only provides accurate results for one case. The standard PCE, both linear and quadratic expansions, is unable to capture the PDF of drag for any case, whereas the sparse PCE, of both first and second order, is able to converge to the PDF obtained using MC well. This is due to the problem of over-fitting encountered by many polynomial interpolation techniques. The sparse PCE does not suffer from this since it only considers the most dominant coefficients. Therefore, for the case of hypersonic shape design under uncertainty, the sparse PCE of second order will be used to compute the stochastic moments of the objectives.



0.00

46

48

50

(d)

Volume (m³)

52

54

56

0

0.05

0.10

0.15 0.20 Coefficient of Drag

(c)

0.25

0.30



Figure 7.5: The probability density functions, obtained using different uncertainty quantification methods, of the coefficient of drag for: a) starting cylinder, b) low drag design and c) optimal trade-off design; volume for: d) starting cylinder, e) low drag design and f) optimal trade-off design

Figure 7.6 shows the probability distribution functions for drag and volume with different levels of uncertainty, σ , in the control points. As expected, the standard deviation of the volume and drag both drastically increases with increasing σ . The mean of the volume is relatively unaffected while the mean of the coefficient of drag is greatly affected. Due to the accuracy of modern manufacturing processes with high precision, a 5% and 10% relative error ($\sigma = 0.05m$ and $\sigma = 0.1m$), to the cylindrical radius, are used to model the uncertainty in the control point locations.





Figure 7.6: The probability density functions, due to varying degree of uncertainty in the control point, of the coefficient of drag for: a) starting cylinder, b) low drag design and c) optimal trade-off design; volume for: d) starting cylinder, e) low drag design and f) optimal trade-off design

Table 7.2 and Table 7.3 show the mean and standard deviation of drag and volume for the designs obtained using deterministic optimization for an uncertainty in control points of $\sigma = 0.05m$ and $\sigma = 0.1m$ respectively. The robust Pareto designs will be compared to the stochastic moments of the design obtained using a deterministic framework.

| | Coefficient of Drag | | Volume (m^3) | |
|--------------|---------------------|----------|----------------|----------|
| Geometry | μ | σ | μ | σ |
| Cylinder | 1.745 | 0.078 | 50.891 | 0.433 |
| Mimimum Drag | 0.057 | 0.004 | 65.867 | 0.272 |
| Optimal | 0.157 | 0.005 | 87.420 | 0.413 |

Table 7.2: Mean and standard deviation of the objective function, under control point uncertainty of $\sigma = 0.05m$, for the hypersonic designs obtained within the deterministic framework

Table 7.3: Mean and standard deviation of the objective function, under control point uncertainty of $\sigma = 0.1m$, for the hypersonic designs obtained within the deterministic framework

| | Coefficient of Drag | | Volume (m^3) | |
|--------------|---------------------|----------|----------------|----------|
| Geometry | μ | σ | μ | σ |
| Cylinder | 1.673 | 0.157 | 50.894 | 0.904 |
| Mimimum Drag | 0.062 | 0.012 | 65.875 | 0.564 |
| Optimal | 0.161 | 0.011 | 87.423 | 0.858 |

Figure 7.7 shows the Pareto fronts obtained by each algorithm for a relative error of 5% where the mean and the error bars represent the standard deviation of the objective. It can be seen that MOEA-DD again fails to preserve diversity in the Pareto front. Each algorithm is able to find a Pareto front in a different region of the objective function space. It can also be seen that the Pareto front is sparse, with regions where no Pareto points are obtained. This is due to the relatively small population used in the optimization, where not enough members were available to be distributed on the Pareto front.







Coefficient of Drag

Figure 7.8 shows the Pareto fronts obtained by each algorithm for a relative error of 5% and 10%. A similar trend can be seen between the Pareto front obtained using robust optimization and that of the deterministic optimization, Fig. 7.2. It is also evident that a larger uncertainty in the control points leads to larger uncertainty in the output, as can be seen by the increased width of the error bars. It can be seen that MOHO and SPEA-R were able to find designs with lower uncertainty in the optimum region of the Pareto front. The designs selected from this optimal region are shown in Fig. 7.8.



Figure 7.8: Pareto front, obtained by each algorithm, within the probabilistic framework, for the hypersonic shape optimization problem under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$

Figure 7.9 shows the designs selected from the optimal region of the Pareto fronts. It can be seen that the relative shape is similar to that of the designs obtained using deterministic optimization. Although designs were taken from similar regions of the Pareto fronts, their overall shape is slightly different.



Figure 7.9: Pareto designs, obtained within the probabilistic framework by MOHO, under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$

Table 7.4 shows the mean and standard deviation of the objective functions for different levels of uncertainty and their relative changes with respect to the mean and standard deviation of the deterministic designs as shown in Table 7.2 and Table 7.3. It can be seen that in most cases, both the mean and the standard deviation has decreased. This is with respect to an already optimized geometry. The relative change is greater for cases with greater uncertainty in the the control point location. Again, MOHO and the probabilistic framework are able to identify designs that are both optimal and stable under uncertainty.

Table 7.4: Mean and standard deviation of the objective function, under varying degrees of control point uncertainty, for the hypersonic designs obtained within the probabilistic framework

| | Coefficier | nt of Drag | Volume (m^3) | | |
|----------------------------|---------------|--------------|----------------|-------------|--|
| Geometry | μ | σ | μ | σ | |
| Optimal (σ =0.05m) | 0.1477 (-6%) | 0.010 (+10%) | 82.1 (-6%) | 0.400 (-2%) | |
| Optimal (σ =0.10m) | 0.1201 (-25%) | 0.010 (-7%) | 79.6 (-9%) | 0.7~(-18%) | |

Figure 7.10 shows the probability distributions for the optimal design obtained within the deterministic framework and the ones obtained using robust optimization.

In each case the mean drag and volume of the robust optimum is less than that of the deterministic optimization. This is expected, as the deterministic approach yields the best possible design since it does not account for random deviations from the designs.



Figure 7.10: Probability density functions for each design, obtained within the probabilistic framework, of coefficient of drag under a control point uncertainty level of: a) $\sigma = 0.05m$ and b) $\sigma = 0.1m$; volume under a control point uncertainty level of: c) $\sigma = 0.05m$ and d) $\sigma = 0.1m$

7.4 Optimization of Hypersonic Bodies: Summary

This section presented the results of MOHO on a hypersonic shape optimization problem. The freeform deformation toolbox was used to parameterize the projectile geometry. The modified Newton impact theory solver was used to obtain the wave drag. The two simultaneous objectives were to maximize the volume of the projectile and minimize the coefficients of drag. Starting from an inefficient shape, the optimum designs obtained within the deterministic framework featured a 72% increase in volume and a 91% decrease in drag. The designs obtained within the probabilistic framework featured a 9% decrease in volume and a 25% decrease in drag relative to the deterministic optimum. The probabilistic optimization framework was also able to find designs more robust than those found within the deterministic framework. A comparison of the Pareto fronts showed that MOHO is able to converge to the Pareto front with only relatively low number of function evaluations. This proved that the MOHO algorithm drastically decreases the time and function evaluations needed for convergence.

CHAPTER 8

CONCLUSION

In this section the main findings of the dissertations are presented and some areas for further improvement and future research suggested.

8.1 Summary of Thesis

In this work, a novel robust optimization algorithm was developed for solving optimization problems with several objectives. The objective of this research was to develop an algorithm that is robust, requires little user input, can solve many-objective optimization problems and account for uncertainty. Since the objective function characteristics are usually not known, a robust method that can adapt is required to arrive at the optimum. This robust method was developed by combining five different algorithms, all performing under different principles, into a single optimizer, MOHO. MOHO learns and adapts to the objective function topology by actively switching between the five constitutive algorithms.

The MOHO suite consists of the NSGA-III, MOEA-DD, SPEA-R, NSDE-R1B and NSDE-D3 algorithms, which can be ran individually or hybridized. The MOHO algorithm monitors the hypervolume of non-dominated solutions. A larger value of hypervolume indicates a better converged and diverse Pareto front. If the hypervolume obtained by an algorithm increases from the previous generation, the current algorithm is allowed to continue. If the hypervolume does not improve, the algorithm with the largest probability of success is selected for the next generation. This gives the algorithm that performs best for the current test problem to have a higher probability to be selected.

It was shown that MOHO performed better than the other five algorithms in approximately 50% of the analytical test cases and no worse for the other test cases. It was shown that some algorithms in the MOHO suite under perform for certain test cases, yet MOHO itself is able to select the best performing algorithm for the current test problem. This was seen by the fact that MOHO never severely under performed on any test problem. MOHO was shown to perform well for problems with constraints and several objectives.

The inevitable uncertainty that is present in real-world design problems is also modeled in this work. An efficient framework based on sparse polynomial chaos is used to propagate the uncertainty in the objectives due to the uncertainty in the design variables. The sparse approach was verified against results obtained using Monte Carlo sampling and was shown to be more accurate than the standard polynomial chaos expansion. This is due to the over-fitting and the oscillations that are present in several polynomial interpolation techniques. The sparse polynomial chaos expansion was used in the robust optimization of subsonic and hypersonic bodies.

The single objective version of MOHO, named SOHO, was used to non-intrusively estimate 44 parameters in a highly non-linear electrochemical model of a Lithium-Ion battery. It was shown that the converged parameters resulted in a mean absolute error of 6.47 mV. The SOHO algorithm also reduced the time required to estimate the parameters in the model from three-weeks, using traditional methods, to under one day.

The developed MOHO algorithm and the sparse polynomial chaos package developed was applied to the robust optimization of subsonic and hypersonic aerodynamic shapes. An accelerated potential flow solver was developed by solving the boundary integral equation using the fast multipole method (FMM). The computational time for the FMM solver was shown to increase linearly with the number of elements while the computational time increases cubically for the standard boundary element solver. The FMM solver was also verified against an analytical solution of flow around an ellipsoid. A subsonic aircraft was optimized for maximum lift, minimum drag and minimum moment. The wing was parameterized using the freeform deformation approach (FFD). It was shown that the lift and drag increased by 20% while the coefficient of moment decreased by 96%. It was also shown that the robust optimization framework was able to identify designs with less uncertainty in the performance than the design arrived at using the deterministic framework.

The robust optimization framework was also used to design a robust hypersonic body for minimum drag and maximum volume. An analysis code based on modified Newton impact theory (MNIT) was used to obtain the coefficient of pressure and compute the drag on the hypersonic body. The MNIT solver was validated against experimental data obtained for hypersonic flow around a sphere and a sphere-cone body. The hypersonic body was parameterized using the freeform deformation approach where the underlying starting geometry was a simple cylinder. The deterministic optimization was able to find designs which reduced drag by 91% and increased volume by 71%. The design obtained using robust optimization featured similar objective function values as the designs obtained using the deterministic framework but with greater certainty in the values.

It was shown that MOHO coupled with the sparse-polynomial chaos approach is able to solve several types of optimization problems of high dimensionality. It is able to solve single, multi-objective and many-objective, constrained and unconstrained, deterministic and probabilistic optimization problems. The toolbox is written using object-oriented programming and can be easily extended to include more algorithms and other switching criteria.

117

8.2 Future Work

The work on Many-Objective Hybrid Optimizer (MOHO) can be further expanded. Some interesting areas of research might include the following:

- 1. Use more constitutive algorithms: The MOHO suite can include additional algorithms that operate under different principles. Increasing the number of algorithms that are available for selection can increase the number of different problems MOHO is able to solve.
- 2. Metrics for evaluating Pareto superiority: Currently only the hypervolume, since it measures both convergence and diversity, is used to compare two Pareto fronts. In the future, it might be beneficial to monitor the convergence and diversity separately to gain more insight into the performance of the constitutive algorithms.
- 3. Additions to switching criteria: Currently, the algorithm that has the highest probability of success is selected when an improvement in the Pareto front is not made. If the convergence and diversity are monitored separately, an algorithm that performs best at improving convergence or diversity (whichever is lacking) can be selected.
- 4. Additions to local search algorithm: Currently, the MOHO suite only includes global search algorithms. Although the distribution indices for simulated binary crossover and polynomial mutation can be used to create designs closer to its parents and implicitly perform a local search, a true local search algorithm should also be added. This can include a multi-objective gradient based algorithm or simulated annealing.

REFERENCES

- E. Talbi, "A taxonomy of hybrid metaheuristics," *Journal of Heuristics*, vol. 8, pp. 541–564, Sep 2002.
- [2] F. Lin, C. Kao, and C. Hsu, "Incorporating genetic algorithms into simulated annealing," in *Proceedings of the 4th International Symposium on Artificial Intelligence*, pp. 290–297, 1991.
- [3] E. Talbi, T. Muntean, and I. Samarandache, "Hybridation des algorithmes genetiques avecla recherche tabou," in *Evolution Artificielle*, vol. 2012, (Toulouse, France), pp. 1–19, 1994.
- [4] S. W. Mahfoud and D. E. Goldberg, "Parallel recombinative simulated annealing: A genetic algorithm," *Parallel Computing*, vol. 21, pp. 1–28, 1995.
- [5] N. F. Foster, G. S. Dulikravich, and J. Bowles, "Three-dimensional aerodynamic shape optimization using genetic evolution and gradient search algorithms," in 34th Aerospace Sciences Meeting and Exhibit, (Reno, NV, USA), pp. 1–16, 1996.
- [6] G. S. Dulikravich, T. J. Martin, and N. F. Foster, "Multidisciplinary hybrid constrained ga optimization," in EUROGEN'99-Evolutionary Algorithms in Engineering andComputer Science: Recent Advances and Industrial Applications, pp. 1709–1716, May 1999.
- [7] C. Poloni, A. Giugevich, L. Onesti, and V. Pediroda, "Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics," *Computer Method in Applied Mechanics and Engineering*, vol. 186, pp. 403–420, 2000.
- [8] M. Emmerich, A. Deutz, and N. Beume, "Gradient-based/evolutionary relay hybrid for computing pareto front approximations maximizing the s-metric," in *Hybrid Metaheuristics* (T. Bartz-Beielstein, M. J. B. Aguilera, C. Blum, B. Naujoks, A. Roli, G. Rudolph, and M. Sampels, eds.), (Berlin, Heidelberg), pp. 140–156, Springer Berlin Heidelberg, 2007.
- [9] H. Satoru, H. Tomoyuki, and M. Mitsunori, "Hybrid optimization using direct, ga, and sqp for global exploration," in 2007 IEEE Congress on Evolutionary Computation, pp. 1709–1716, Sep. 2007.

- [10] E. Balsa-Canto, M. Peifer, J. R. Banga, J. Timmer, and C. Fleck, "Hybrid optimization method with general switching strategy for parameter estimation," *BMC Systems Biology*, vol. 2, p. 26, Mar 2008.
- [11] V. Kelner, F. Capitanescu, O. Lonard, and L. Wehenkel, "A hybrid optimization technique coupling an evolutionary and a local search algorithm," *Journal of Computational and Applied Mathematics*, vol. 215, no. 2, pp. 448 – 456, 2008. Proceedings of the Third International Conference on Advanced Computational Methods in Engineering (ACOMEN 2005).
- [12] R. J. Moral and G. S. Dulikravich, "Multi-objective hybrid evolutionary optimization with automatic switching among constituent algorithms," AIAA Journal, vol. 46, no. 3, pp. 673–681, 2008.
- [13] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," in *TIK-Report 103, Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH)*, pp. 1–21, May 2001.
- [14] R. Eberhart, Y. Shi, and J. Kennedy, Swarm Intelligence. Amsterdamn, Netherlands: Elsevier, 1 ed., March 2001.
- [15] D. Sahoo, "Wavelet-based neural networks for response surace in multiobjective evolutionary optimization," Master's thesis, Florida International University, Miami, FL, 2005.
- [16] M. J.Colao, G. S. Dulikravich, and D. Sahoo, "A response surface methodbased hybrid optimizer," *Inverse Problems in Science and Engineering*, vol. 16, no. 6, pp. 717–741, 2008.
- [17] G. S. Dulikravich, T. J. Martin, M. J. Colaco, and E. J. Inclan, "Automatic switching algorithms in hybrid single-objective optimization," *FME Transactions*, vol. 41, pp. 167–179, July 2013.
- [18] K. Sindhya, K. Miettinen, and K. Deb, "A hybrid framework for evolutionary multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, pp. 495–511, Aug. 2013.
- [19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, Apr. 2002.

- [20] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, pp. 712–731, Dec. 2007.
- [21] Y. Wang, Z. Cai, G. Guo, and Y. Zhou, "Multiobjective optimization and hybrid evolutionary algorithm to solve constrained optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B Cybernetics*, vol. 17, pp. 560–575, June 2007.
- [22] P. A. Bosman, "On gradients and hybrid evolutionary algorithms for realvalued multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, pp. 51–69, Oct. 2012.
- [23] A. H. Sekhar and A. L. Devi, "Hybrid optimization algorithms for analyzing the performance of transmission system incorporating advanced svc model," in 2017 International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology (ICEEIMT), pp. 11–18, Feb 2017.
- [24] B. Xin, J. Chen, J. Zhang, H. Fang, and Z.-H. Peng, "Hybridizing differential evolution and particle swarm optimization to design powerful optimizers: A review and taxonomy," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 744–767, Sep. 2012.
- [25] A. Liu and M.-T. Yang, "A new hybrid nelder-mead particle swarm optimization for coordination optimization of directional overcurrent relays," *Mathematical Problems in Engineering*, vol. 2012, pp. 1–19, 2012.
- [26] S. R. Reddy, M. K. Scharrer, F. Pichler, D. Watzenig, and G. S. Dulikravich, "Accelerating parameter estimation in doyle-fuller-newman model for lithiumion batteries," in 18th International IGTE Symposium on Numerical Field Calculation in Electrical Engineering, Sep. 2018.
- [27] S. R. Reddy, M. K. Scharrer, F. Pichler, D. Watzenig, and G. S. Dulikravich, "Accelerating parameter estimation in doyle-fuller-newman model for lithiumion batteries," *COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronics Engineering*, pp. 1–9, 2019.
- [28] J. Lepagnot, L. Idoumghar, M. Brevilliers, and M. Idrissi-Aouad, "A new high-level relay hybrid metaheuristic for black-box optimization problems," in *Artificial Evolution* (E. Lutton, P. Legrand, P. Parrend, N. Monmarche, and M. Schoenauer, eds.), (Cham), pp. 115–128, Springer International Publishing, 2018.

- [29] S. Cheng, J. Zhou, and M. Li, "A new hybrid algorithm for multi-objective robust optimization with interval uncertainty," ASME Journal of Mechanical Design, vol. 137, no. 2, pp. 021401–021401–9, 2015.
- [30] K. Zhang, Z. Wang, L. Zhang, J. Yao, and X. Yan, "A hybrid optimization method for solving bayesian inverse problems under uncertainty," *PLoS One*, vol. 10, no. 8, pp. 1–24, 2015.
- [31] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms. New York, NY, USA: John Wiley and Sons, Inc., 2001.
- [32] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, pp. 577–601, Aug. 2014.
- [33] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 67–82, Apr. 1997.
- [34] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, pp. 115–148, 1995.
- [35] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, pp. 694–716, Oct. 2015.
- [36] S. Jiang and S. Yang, "A strength pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, pp. 329–346, June 2017.
- [37] S. R. Reddy and G. S. Dulikravich, "Many-objective differential evolution optimization based on reference points: Nsde-r," *Structural and Multidisciplinary Optimization*, pp. 1 – 19, 2019.
- [38] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, pp. 257–271, Nov. 1999.
- [39] K. Deb, K. Miettinen, and S. Chaudhuri, "Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches," *IEEE Transactions on Evolutionary Computation*, vol. 14, pp. 821–841, 2010.

- [40] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, Scalable Test Problems for Evolutionary Multiobjective Optimization, pp. 105–145. London: Springer London, 2005.
- [41] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, pp. 477–506, Oct. 2006.
- [42] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part ii: Handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, pp. 602–622, Aug. 2014.
- [43] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 7, pp. 174–188, Apr. 2003.
- [44] L. While, L. Bradstreet, and L. Barone, "A fast way of calculating exact hypervolumes," *IEEE Transactions on Evolutionary Computation*, vol. 16, pp. 86–95, Feb. 2012.
- [45] J. Bader and E. Zitzler, "Hype: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [46] A. Shapiro, "Monte carlo sampling methods," in Stochastic Programming, vol. 10 of Handbooks in Operations Research and Management Science, pp. 353 – 425, Elsevier, 2003.
- [47] R. G. Chanem and P. D. Spanos, Stochastic Finite Elements: A Spectral Approach. New York, NY, USA: Springer-Verlag, 1991.
- [48] S. Hosder, R. W. Walters, and M. Balch, "Point-collocation nonintrusive polynomial chaos method for stochastic computational fluid dynamics," AIAA Journal, vol. 48, no. 12, pp. 2721–2730, 2010.
- [49] D. Xiu and G. E. Karniadakis, "The wiener-askey polynomial chaos for stochastic differential equations," SIAM Journal on Scientific Computing, vol. 24, no. 2, pp. 619–644, 2002.

- [50] A. Kaintura, T. Dhaene, and D. Spina, "Review of polynomial chaos-based methods for uncertainty quantification in modern integrated circuits," *MDPI Electronics*, vol. 7, no. 30, pp. 1–21, 2018.
- [51] G. Blatman and B. Sudret, "Adaptive sparse polynomial chaos expansion based on least angle regression," *Journal of Computational Physics*, vol. 230, no. 6, pp. 2345 – 2367, 2011.
- [52] A. Doostan and H. Owhadi, "A non-adapted sparse approximation of pdes with stochastic inputs," *Journal of Computational Physics*, vol. 230, no. 8, pp. 3015 – 3034, 2011.
- [53] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.
- [54] T. W. Sederberg and S. R. Parry, "Free-form deformation of solid geometric models," ACM SIGGRAPH Computer Graphics, vol. 20, pp. 151–160, Aug 1986.
- [55] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *Journal of Electrochemical Soceity*, vol. 166, pp. 1526–1533, 1993.
- [56] D. K. Karthikeyan, G. Sikha, and R. E. White, "Thermodynamic model development for lithium intercalation electrodes," *Journal of Power Sources*, vol. 185, p. 13981407, 2008.
- [57] F. Pichler, "Derivation of a multi-scale battery model and its high-performance computing implementation," Master's thesis, University of Graz, Graz, Austria, 2018.
- [58] G. Alfonsi, "Reynolds-averaged navierstokes equations for turbulence modeling," ASME Applied Mechanics Reviews, vol. 62, pp. 040802–1–040802– 20, 2009.
- [59] A. Jameson, "Numerical solution of the euler equations for compressible inviscid fluids," in *Numerical Methods for the Euler Equations of Fluid Dynamics* (F. Angrand, A. Dervieux, J. D. side ri, and R. Glowinski, eds.), pp. 199–245, Philadelphia, USA: Society of Industrial and Applied Mathematics (SIAM), 1985.

- [60] L. L. Erickson, "Panel methods an introduction," in NASA Technical Paper NASA TP-2995, pp. 1–68, 1990.
- [61] J. Anderson, *Modern Compressible Flow*. New York, NY, USA: McGraw-Hill Education, 2002.
- [62] A. M. Kuethe and C.-Y. Chow, Foundations of Aerodynamics. Hoboken, NJ, USA: John Wiley and Sons, Inc., 1997.
- [63] H. Glauert, "The effect of compressibility on the lift of an aerofoil," Proceedings of the Royal Society London, vol. 118, 1928.
- [64] B. Gothert, "Plane and three dimensional flow at high subsonic speeds," in Technical Memorandum, National Advisory Committee for Aeronautics, 1946.
- [65] L. Morino, L.-T. Chen, and E. O. Suciu, "Steady and oscillatory subsonic and supersonic aerodynamics around complex configurations," *AIAA Journal*, vol. 13, pp. 368–374, Mar. 1975.
- [66] C. Brebbia, *The Boundary Element Method for Engineers*. Hoboken, NJ, USA: John Wiley and Sons, Inc., 1980.
- [67] L. Greengard and V. Rokhlin, "A new version of the fast multipole method for the laplace equation in three dimensions," *Acta Numerica*, vol. 6, pp. 229–269, Jan. 1997.
- [68] Z. Gimbutas and L. Greengard, "A fast multipole method for the evaluation of elastostatic fields in a half-space with zero normal stress," Advances in Computational Mathematics, vol. 42, pp. 175–198, 2016.
- [69] Y. Han, Y. Nie, and H. Dong, "A fast multipole algorithm for radiative heat transfer in 3d semitransparent media," *Journal of Quantitative Spectroscopy* and Radiative Transfer, vol. 221, pp. 8–17, 2018.
- [70] W. Qu, W. Chen, Z. Fu, and Y. Gu, "Fast multipole singular boundary method for stokes flow problems," *Mathematics and Computers in Simulation*, vol. 146, pp. 57–69, 2018.
- [71] W. R. Wolf and S. K. Lele, "Aeroacoustic integrals accelerated by fast multipole method," AIAA Journal, vol. 49, pp. 1466–1477, 2011.

- [72] Y. Liu, Fast Multipole Boundary Element Method Theory and Applications in Engineering. Cambridge, UK: Cambridge University Press, 1 ed., 2009.
- [73] H. Samet, "An overview of quadtrees, octrees and related hierarchical data structures," in *Theoretical Foundations of Computer Graphics and CAD*, vol. F40, pp. 51–58, 1988.
- [74] Q. Ren and C. L. Chan, "Analytical evaluation of the bem singular integral for 3d laplace and stokes flow equations using coordinate transformation," *Engineering Analysis with Boundary Elements*, vol. 53, pp. 1–8, 2015.
- [75] Y. Saad and M. H. Schultz, "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems," SIAM Journal on Scientific and Statistical Computing, vol. 7, pp. 856–869, July 1986.
- [76] C.-Y. Huang, S. R. Kennon, and G. S. Dulikravich, "Generalized nonlinear minimal residual (gnlmr) method for iterative algorithms," *Journal of Computational and Applied Mathematics*, vol. 16, pp. 215–232, 1986.
- [77] C.-Y. Huang and G. S. Dulikravich, "Fast iterative algorithms based on optimized explicit time stepping," *Computer Methods in Applied Mechanics* and Engineering, vol. 63, pp. 15–36, 1987.
- [78] M. M. Munk, "Remarks on the pressure distribution over the surface of an ellipsoid, moving translationally through a perfect fluid," in *Technical Report*, *National Advisory Committee for Aeronautics*, pp. 1–9, 1924.
- [79] L. Lees, "Hypersonic flow," in 5th International Aeronautical Conference, (Los Angeles, CA, USA), pp. 241–276, 1955.
- [80] W. Heybey, "Newtonian aerodynamics for general body shapes with several applications," in NASA Technical Memorandum X-53391, pp. 1–51, Feb. 1966.
- [81] M. D. Maughmer, L. N. Long, N. Guilmette, and P. Pagano, "Prediction of forces and moments for hypersonic flight vehicle control effectors," in NASA Technical Report NASA-CR-188954, pp. 1–48, 1993.
- [82] D. Masson, D. Morris, and D. E. Bloxsom, "Measurements of sphere drag from hypersonic continuum to free-molecule flow," in U.S. Air Force Project Rand RM-2678, pp. 1–46, Nov. 1960.

- [83] R. E. Geiger, "Experimental lift and drag of a series of glide configurations at mach numbers 12.6 and 17.5," *Journal of the Aerospace Sciences*, vol. 29, no. 4, pp. 410–419, 1962.
- [84] S. Santhanagopalan, Q. Guo, and R. E. White, "Parameter estimation and model discrimination for a lithium-ion cell," *Journal of The Electrochemical Society*, vol. 154, pp. 198–206, 2007.
- [85] M. K. Scharrera, B. Suhrb, , and D. Watzenig, "A new space mapping parameter surrogate optimization for lithium-ion cell models," in 4th Inverse Problems, Design and Optimization Symposium, (Albi, France), pp. 1–10, 2013.
- [86] J. C. Formana, S. J. Mourab, J. L. Steina, and H. K. Fathy, "Genetic identification and fisher identifiability analysis of the doylefullernewman model from experimental cycling of a lifepo4 cell," *Journal of Power Sources*, vol. 210, pp. 263–265, 2012.
- [87] N. Jin, D. L. Danilov, P. M. V. den Hof, and M. Donkers, "Parameter estimation of an electrochemistry-basedlithium-ion battery model using a twostep procedure and aparameter sensitivity analysis," *International Journal of Energy Research*, vol. 42, pp. 2417–2430, 2018.
- [88] L. Zhang, C. Lyu, L. Wang, J. Zheng, W. Luo, and K. Ma, "Parallelized genetic identification of the thermal-electrochemical model for lithium-ion battery," *Advances in Mechanical Engineering*, vol. 2013, pp. 1–12, 2013.
- [89] K. Uddin, S. Perera, W. D. Widanage, L. Somerville, and J. Marco, "Characterising lithium-ion battery degradation through the identification and tracking of electrochemical battery model parameters," *MDPI Batteries*, vol. 2, pp. 1–17, 2016.
- [90] A. Wirgin, "The inverse crime." https://arxiv.org/pdf/math-ph/0401050.pdf. Accessed: 2018-12-15.
- [91] G. S. Dulikravich, "Aerodynamic shape design and optimization: Status and trends," AIAA Journal of Aircraft, vol. 29, pp. 1020–1026, 1992.
- [92] G. S. Dulikravich, "Shape inverse design and optimization for threedimensional aerodynamics," in 33rd Aerospace Sciences Meeting and Exhibit, (Reno, NV, USA), pp. 1–11, 1995.

- [93] S. Skinner and H. Zare-Behtash, "State-of-the-art in aerodynamic shape optimisation methods," *Applied Soft Computing*, vol. 62, pp. 933 962, 2018.
- [94] Z. Lyu, G. K. W. Kenway, and J. R. Martins, "Aerodynamic shape optimization investigations of the common research model wing benchmark," *AIAA Journal*, vol. 53, no. 4, pp. 968–985, 2015.
- [95] H. Gagnon and D. W. Zingg, "High-fidelity aerodynamic shape optimization of unconventional aircraft through axial deformation," in 52nd Aerospace Sciences Meeting, (National Harbor, Maryland), pp. 1–18, 2014.
- [96] P. W. Jansen, R. E. Perez, and J. R. R. A. Martins, "Aerostructural optimization of nonplanar lifting surfaces," *Journal of Aircraft*, vol. 47, no. 5, pp. 1490–1503, 2010.
- [97] A. Ning and I. Kroo, "Multidisciplinary considerations in the design of wings and wing tip devices," *Journal of Aircraft*, vol. 47, no. 2, pp. 534–543, 2010.
- [98] P. D. Vecchia and F. Nicolosi, "Aerodynamic guidelines in the design and optimization of new regional turboprop aircraft," Aerospace Science and Technology, vol. 38, pp. 88 – 104, 2014.
- [99] S. N. Skinner and H. Zare-Behtash, "Aerodynamic optimisation of non-planar lifting surfaces," in 57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, (San Diego, California), pp. 1–21, 2016.
- [100] D. Willis, D. Peraire, and J. White, "Fastaero a precorrected fft fast multipole tree steady and unsteady potential flow solver," in SMA Symposium, (Singapore), pp. 1–9, 2005.
- [101] J. P. More, "An arbitrarily high-order, unstructured free-wake panel solver," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, 2013.
- [102] R. L. Hardy, "Multiquadric equations of topography and other irregular surfaces," *Journal of Geophysical Research*, vol. 76, no. 8, pp. 1905–1915, 1972.
- [103] S. Rippa, "An algorithm for selecting a good value for the parameter c in radial basis function interpolation," Advances in Computational Mathematics, vol. 11, no. 2, pp. 193–210, 1999.

- [104] I. M. Sobol, "Distribution of point in a cube and approximate evaluation of integrals," U.S.S.R. Computational Mathematics and Mathematical Physics, vol. 7, pp. 86–112, 1967.
- [105] G. Elliot and W. Hankey, "Hypersonic lifting body optimization.," Journal of Spacecraft and Rockets, vol. 5, no. 12, pp. 1463–1467, 1968.
- [106] N. Takashima and M. J. Lewis, "Optimization of waverider-based hypersonic cruise vehicles with off-design considerations," *Journal of Aircraft*, vol. 36, no. 1, pp. 235–245, 1999.
- [107] K. Cui and G.-W. Yang, "Shape optimization for hypersonic arc-wing missiles," *Journal of Spacecraft and Rockets*, vol. 47, no. 4, pp. 694–700, 2010.
- [108] D. Lesieutre, M. Dillenius, and T. Lesieutre, "Multidisciplinary design optimization of missile configurations and fin planforms for improved performance," in 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, (St. Louis, MO), pp. 1373–1384, 1998.
- [109] K. G. Bowcutt, "Multidisciplinary optimization of airbreathing hypersonic vehicles," *Journal of Propulsion and Power*, vol. 17, no. 6, pp. 1184–1190, 2001.
- [110] J. E. Theisinger and R. D. Braun, "Multi-objective hypersonic entry aeroshell shape optimization," *Journal of Spacecraft and Rockets*, vol. 46, no. 5, pp. 957– 966, 2009.
- [111] N. F. Foster and G. S. Dulikravich, "Three-dimensional aerodynamic shape optimization using genetic and gradient search algorithms," *Journal of Spacecraft and Rockets*, vol. 34, no. 1, pp. 36–42, 1997.
- [112] S. G. Sheffer and G. S. Dulikravich, "Constrained optimization of threedimensional hypersonic vehicle configurations," in 31st Aerospace Sciences Meeting and Exhibit, (Reno, NV, USA), pp. 1–7, 1993.
APPENDIX A

APPENDICES

Performance of MOHO on DTLZ Test Problem

The IGD values obtained by each algorithm on the DTLZ test problem are shown in Table A.1, where the best performing algorithm for each test problem is highlighted in red.

| Ce | | |
|----------------------|--------|--|
| nar | | |
| orr | | |
| erf | | |
| st p | | |
| Be | | |
| IS. | | |
| len | | |
| op | | |
| 7 bī | | |
| Ę | | |
| Б | | |
| he | | |
| or t | | |
| n fe | | |
| thr | | |
| ĵori | | |
| alg | | |
| ach | | |
| y ea | | |
| þ. H | | |
| inec | | |
| otai | | |
| of O | | |
| g | | |
| le I | | |
| f th | | |
| n o | | |
| tio | | |
| via | | |
| l de | | |
| ard | | |
| nd | | |
| sta | | |
| 'nd | | |
| 'na | eq | |
| Iea | n r. | |
| \sim | d i | |
| A.1 | ure | |
| le . | olo | |
| Tat | C S | |
| - | | |

| $\operatorname{Problem}$ | Μ | III-ADSN | MOEA-DD | SPEA-R | NSDE-R1B | NSDE-D3 | ОНОМ |
|--------------------------|----|-------------------------|---|-------------------------------|---|-------------------------------|-------------------------|
| DTLZ1 | 3 | 2.29E-03 (2.72E-03) | 1.09 E-02 (3.96 E-04) | 4.07E-01 (4.74E-01) | 1.83E-03(1.68E-03) | 6.05E-01 (3.74E-01) | 1.72E-04(7.45E-05) |
| | ŋ | 1.99E-03 (1.86E-03) | 3.33E-02 (3.62E-04) | 2.21E+00(1.28E+00) | 1.20E-03 (6.82E-04) | 7.96E-02(1.22E-01) | 2.37E-05 (1.61E-05) |
| | x | 6.28E-03 (2.25E-03) | 7.30E-02 (7.82E-04) | 1.12E + 01 (5.85E + 00) | 5.57E-03(1.47E-03) | 2.51E-01 (1.92E-01) | 3.42E-03(1.50E-03) |
| | 10 | 5.97E-03 (2.83E-03) | 9.44E-02(9.01E-04) | 9.34E+00(3.51E+00) | 5.47E-03 (1.08E-03) | 2.06E-01 (2.34E-01) | 3.39E-03 (8.23E-04) |
| | 15 | 9.19E-03 (7.44E-03) | 1.64E-01 (2.47E-03) | $1.32E \pm 01 (4.00E \pm 00)$ | 8.87E-03 (5.18E-03) | 4.61E-01(2.69E-01) | 3.46E-03 (1.54E-03) |
| DTLZ2 | e | 3.27E-03 (1.51E-03) | 3.59E-02 (4.47E-04) | 1.60E-02 (3.98E-03) | 3.24E-03(2.24E-03) | 3.73E-03 (2.22E-04) | 4.41E-03(2.45E-04) |
| | ŋ | 9.56E-03 (1.79E-03) | 5.29 E-02 (4.63 E-04) | 4.45E-02 (6.50E-03) | 9.12E-03(1.33E-03) | 8.55E-03 (4.04E-04) | 8.25E-03 (6.00E-04) |
| | x | 2.50E-02 (2.11E-03) | $3.75 \text{E} - 02 \ (1.09 \text{E} - 03)$ | 8.19E-02 (8.15E-03) | 2.56E-02 ($2.10E-03$) | 1.94E-02(9.80E-04) | 1.92E-02(1.46E-03) |
| | 10 | 2.89E-02 (2.44E-03) | 3.32E-02(5.84E-04) | 7.38E-02 (8.28E-03) | 3.02E-02 (3.11E-03) | 1.80E-02 (6.94E-04) | 1.76E-02 ($1.11E-03$) |
| | 15 | 3.64E-02 (3.43E-03) | 1.09E-02(8.73E-04) | 1.28E-01 (1.40E-02) | $3.64 \text{E} - 02 \ (2.69 \text{E} - 03)$ | 1.16E-02(4.96E-04) | 1.68E-02 (1.23E-03) |
| DTLZ3 | e | 5.99E-03 ($2.98E-03$) | 3.67E-02 (8.10E-04) | 1.04E+00(1.16E+00) | 6.22E-03 (2.55E-03) | $8.46E \pm 00 (5.49E \pm 00)$ | 4.21E-01 (5.37E-01) |
| | ю | 2.47E-02 (2.68E-02) | 5.36E-02(5.71E-04) | 5.05E+01 (1.95E+01) | 2.53E-02(2.46E-02) | 3.71E+00(2.15E+00) | 3.41E-03 ($3.20E-04$) |
| | x | 6.45E-02 (2.10E-02) | 5.07E-02 ($5.07E-03$) | 1.12E + 02 (3.68E + 01) | 6.45E-02(1.66E-02) | 4.01E+00(1.84E+00) | 1.49E-02(9.23E-04) |
| | 10 | 4.28E-02 (1.33E-02) | 3.71E-02(1.86E-03) | 1.00E + 02 (2.88E + 01) | 4.14E-02 ($6.48E-03$) | 2.99E+00(1.44E+00) | 1.32E-02 (5.64E-04) |
| | 15 | 5.98E-02(2.94E-02) | 1.48E-02(1.46E-03) | 8.23E+01 (2.64E+01) | 1.46E-01(1.45E-01) | 7.47E+00 (3.53E+00) | 1.41E-02(9.99E-04) |
| DTLZ4 | e | 5.27E-04 (3.23E-04) | 3.60E-02 (8.49E-03) | 9.93E-03 (1.83E-03) | 1.06E-01 (2.15E-01) | 2.56E-03 (1.92E-04) | 3.00E-03 (1.78E-04) |
| | ю | 1.81E-03(4.78E-04) | 5.24 E - 02 (3.76 E - 04) | 2.09E-02 (3.31E-03) | 1.83E-03(4.53E-04) | 5.29 E-03 (4.50 E-04) | 5.89E-03 (5.58E-04) |
| | × | 7.93E-03(1.02E-03) | 3.27E-02(7.92E-04) | 4.65E-02(7.12E-03) | 8.10E-03(1.15E-03) | 1.65E-02(9.98E-04) | 2.33E-02 (1.20E-03) |
| | 10 | 9.78E-03 (8.33E-04) | 2.98E-02(5.20E-04) | 5.60E-02 (6.41E-03) | 9.71E-03 ($8.53E-04$) | 1.54E-02 (6.76E-04) | 2.03E-02 (8.41E-04) |
| | 15 | 1.30E-02 (1.41E-03) | 8.82E-03 (9.96E-04) | 9.49E-02(7.12E-03) | 1.35E-02 (1.29E-03) | 1.26E-02 (6.60E-04) | 1.35E-02 (6.32E-04) |

Performance of MOHO on WFG Test Problem

The hypervolume values obtained by each algorithm on the WFG test problem are shown in Table A.2, where the best performing algorithm for each test problem is highlighted in red.

Table A.2: Mean and standard deviation of the hypervolume obtained by each algorithm for the WFG problems. Best performance is coloured in red

| Problem | Μ | NSGA-III | MOEA-DD | SPEA-R | NSDE-R1B | NSDE-D3 | МОНО |
|---------|--------|------------------------------------|------------------------------------|------------------------------------|------------------------------------|------------------------------------|------------------------------------|
| WFG1 | 3 | 0.4821 (0.0028) | 0.4612 (0.0037) | 0.4584 (0.0018) | 0.4695 (0.0021) | 0.4824 (0.0033) | 0.4979 (0.0022) |
| | 5 | 0.4245 (0.0019) | $0.4164 \ (0.0018)$ | 0.4017 (0.0019) | $0.4005 \ (0.0013)$ | $0.4235\ (0.0024)$ | $0.4208 \ (0.0018)$ |
| | 8 | 0.3517 (0.0025) | $0.3440\ (0.0037)$ | $0.3365\ (0.0031)$ | $0.3252 \ (0.0059)$ | $0.3521 \ (0.0021)$ | $0.3613 \ (0.0068)$ |
| | 10 | 0.3227 (0.0036) | $0.3362 \ (0.0046)$ | $0.3075 \ (0.0021)$ | $0.2951 \ (0.0019)$ | $0.3232 \ (0.0033)$ | $0.3379 \ (0.0060)$ |
| | 15 | 0.2714 (0.0090) | 0.2856 (0.0052) | 0.2698 (0.0022) | 0.2528 (0.0054) | 0.2665 (0.0085) | 0.2913 (0.0131) |
| WFG2 | 3 | 0.9134 (0.0580) | 0.9314(0.0049) | 0.8643 (0.0634) | 0.9512(0.0021) | 0.9001 (0.0662) | 0.9514(0.0038) |
| | 5 | 0.9777 (0.0016) | 0.9214 (0.0036) | 0.9565 (0.0048) | 0.9814 (0.0026) | 0.9779 (0.0020) | 0.9894 (0.0030) |
| | 10 | 0.9752 (0.0030) 0.9792 (0.0027) | 0.9270 (0.0071) 0.9445 (0.0048) | 0.9481 (0.0401) 0.9672 (0.0042) | 0.9890 (0.0150) 0.9997 (0.0006) | 0.9582 (0.0554) 0.9793 (0.0032) | 0.9397 (0.0408) 0.9755 (0.0050) |
| | 15 | 0.9192(0.0027) 0.9101(0.0515) | 0.9101 (0.0402) | 0.9758 (0.0042) | 0.9289 (0.0247) | 0.9302 (0.0032) 0.9302 (0.0131) | 0.9133(0.0417) |
| WFG3 | 3 | 0.6869 (0.0057) | 0.6780 (0.0132) | 0.6606 (0.0048) | 0.6752 (0.0063) | 0.6895 (0.0042) | 0.7007 (0.0067) |
| | 5 | 0.6479 (0.0096) | 0.6502(0.0051) | 0.6365 (0.0115) | 0.5941 (0.0102) | 0.6478 (0.0073) | 0.6239 (0.0093) |
| | 8 | 0.4942 (0.0222) | 0.5566 (0.0122) | $0.5641 \ (0.0189)$ | 0.4600 (0.0150) | 0.4896(0.0264) | 0.4797(0.0202) |
| | 10 | 0.4732 (0.0455) | 0.5459(0.0087) | $0.5542 \ (0.0165)$ | 0.4500 (0.0160) | 0.4687 (0.0403) | $0.4316\ (0.0418)$ |
| | 15 | 0.3371 (0.0442) | $0.2104 \ (0.0250)$ | $0.4760\ (0.0144)$ | $0.3085\ (0.0401)$ | $0.3179\ (0.0392)$ | 0.3419(0.0474) |
| WFG4 | 3 | 0.7027 (0.0026) | $0.7043 \ (0.0031)$ | $0.6938\ (0.0040)$ | $0.7031 \ (0.0038)$ | $0.7046\ (0.0044)$ | $0.7058 \ (0.0039)$ |
| | 5 | 0.8291 (0.0029) | 0.8211 (0.0036) | $0.8381 \ (0.0036)$ | 0.8435(0.0028) | 0.8283(0.0042) | $0.8521 \ (0.0029)$ |
| | 8 | 0.8421 (0.0062) | 0.7769 (0.0093) | 0.8707 (0.0064) | 0.8772(0.0033) | 0.8441 (0.0078) | 0.8821 (0.0200) |
| | 10 | 0.8733 (0.0064) | 0.7729 (0.0077) | 0.9141 (0.0059) | 0.9011 (0.0041) | 0.8761 (0.0053) | 0.9054 (0.0173) |
| WECE | 15 | 0.8554 (0.0106) | 0.6599(0.0437) | 0.9205 (0.0039) | 0.9308 (0.0042) | 0.8489 (0.0123) | 0.8400(0.0373) |
| WFG5 | 5 | 0.6782(0.0050) 0.8077(0.0040) | 0.6759(0.0046) 0.7877(0.0037) | 0.6709(0.0040) 0.8035(0.0028) | 0.6826 (0.0047) 0.8155 (0.0035) | 0.6785(0.0037) 0.8076(0.0034) | 0.6831 (0.0036) 0.8106 (0.0057) |
| | 8 | 0.8077(0.0040) 0.8291(0.0056) | 0.7033(0.0094) | 0.8033(0.0023) 0.8230(0.0047) | 0.3133 (0.0033) 0.7601 (0.0153) | 0.8070 (0.0054) 0.8288 (0.0050) | 0.3100(0.0037) 0.7484(0.0139) |
| | 10 | 0.8496 (0.0053) | 0.6898 (0.0128) | 0.8608 (0.0041) | 0.7412 (0.0176) | 0.8483 (0.0046) | 0.7574 (0.0212) |
| | 15 | 0.8053 (0.0241) | 0.4431(0.0435) | 0.8401(0.0044) | 0.7076(0.0194) | 0.8117 (0.0180) | 0.6549(0.0425) |
| WFG6 | 3 | 0.6777 (0.0041) | 0.6724(0.0096) | 0.6636 (0.0043) | 0.6781 (0.0126) | 0.6794 (0.0041) | 0.7161 (0.0047) |
| | 5 | 0.8079 (0.0069) | $0.7849\ (0.0068)$ | $0.8070 \ (0.0035)$ | $0.7797 \ (0.0180)$ | $0.8097 \ (0.0062)$ | $0.8680\ (0.0038)$ |
| | 8 | 0.8505 (0.0102) | $0.7280 \ (0.0153)$ | $0.8460 \ (0.0053)$ | $0.8323 \ (0.0386)$ | $0.8490 \ (0.0093)$ | $0.9106\ (0.0072)$ |
| | 10 | 0.8774 (0.0059) | 0.7199(0.0157) | 0.8814(0.0040) | 0.8718 (0.0015) | 0.8750 (0.0070) | 0.9423 (0.0044) |
| | 15 | 0.8879 (0.0112) | 0.4973 (0.0684) | 0.8844 (0.0060) | 0.8682 (0.0005) | 0.8898 (0.0103) | 0.9152 (0.0319) |
| WFG7 | 3 | 0.7145 (0.0037) | 0.7032(0.0156) | 0.6831 (0.0160) | 0.7239(0.0030) | 0.7147 (0.0034) | 0.7272(0.0039) |
| | о 8 | 0.8558(0.0031) 0.8897(0.0040) | 0.8338(0.0036) 0.8004(0.0063) | 0.8425 (0.0048) 0.8990 (0.0041) | 0.8593 (0.0038) 0.8555 (0.0078) | 0.8552 (0.0033) 0.8907 (0.0048) | 0.8721 (0.0044) 0.8965 (0.0131) |
| | 10 | 0.8397 (0.0040) 0.9195 (0.0034) | 0.8004 (0.0003) 0.8025 (0.0079) | 0.8330(0.0041) 0.9460(0.0022) | 0.8333 (0.0073) 0.8884 (0.0062) | 0.8307 (0.0043) 0.9208 (0.0037) | 0.8303(0.0131) 0.9439(0.0039) |
| | 15 | 0.9348 (0.0036) | 0.7094 (0.0336) | 0.9636 (0.0024) | 0.9812 (0.0031) | 0.9348 (0.0034) | 0.9556 (0.0101) |
| WFG8 | 3 | 0.6659 (0.0043) | 0.6614 (0.0179) | 0.6594 (0.0061) | 0.6725 (0.0036) | 0.6662 (0.0039) | 0.6740 (0.0041) |
| | 5 | 0.7784 (0.0046) | 0.7815(0.0052) | 0.7887 (0.0044) | 0.7670(0.0027) | 0.7781(0.0034) | 0.7857(0.0057) |
| | 8 | 0.7660 (0.0098) | 0.7205(0.0287) | $0.8074 \ (0.0046)$ | 0.7198(0.0082) | $0.7665 \ (0.0072)$ | $0.7641 \ (0.0144)$ |
| | 10 | 0.7902 (0.0093) | $0.7045\ (0.0395)$ | $0.8582 \ (0.0048)$ | $0.7219\ (0.0070)$ | $0.7918\ (0.0105)$ | $0.8027 \ (0.0155)$ |
| | 15 | 0.8202 (0.0082) | 0.5248(0.0572) | $0.8588 \ (0.0106)$ | 0.7738(0.0217) | 0.8210(0.0097) | 0.7770(0.0203) |
| WFG9 | 3 | 0.6596 (0.0171) | 0.6571 (0.0211) | 0.6313 (0.0038) | 0.6397 (0.0029) | 0.6468 (0.0161) | $0.6724 \ (0.0215)$ |
| | 5 | 0.7395 (0.0096) | 0.7532(0.0051) | 0.7278(0.0093) | 0.7428 (0.0050) | 0.7360 (0.0107) | 0.7583 (0.0237) |
| | 8 | 0.7244 (0.0118) | 0.6352 (0.0187) | 0.6815(0.0230) | 0.7231 (0.0094) | 0.7332 (0.0216) | 0.7231 (0.0156) |
| | 10 | 0.7499 (0.0207) | 0.6250 (0.0123) | 0.7131 (0.0194) | 0.7248 (0.0054) | 0.7428 (0.0184) | 0.7389 (0.0143) |
| | 10 | 0.0914 (0.0190) | 0.4780 (0.0361) | 0.0717 (0.0221) | 0.0015 (0.0112) | 0.7029 (0.0190) | 0.0749 (0.0239) |

VITA

SOHAIL R. REDDY

| May 2019 - Aug. 2019 | Research and Development Intern Vanderplaats Research and Development Inc. Novi, Michigan, USA |
|-----------------------|--|
| May 2018 - Aug. 2018 | Visiting Researcher Virtual Vehicle Research Center Graz, Austria |
| Aug. 2014 - Aug. 2015 | M.Sc., Mechanical Engineering Florida International University Miami, Florida, USA |
| Aug. 2010 - Aug. 2014 | B.Sc., Mechanical Engineering Florida International University Miami, Florida, USA |

PUBLICATIONS AND PRESENTATIONS

G.S. Dulikravich, S.R. Reddy, M.A. Pasqualette, M.J. Colaco, H.R.B. Orlande, J. Coverston, (2016). *Inverse Determination of Spatially Varying Material Coefficients in Solid Objects*, Journal of Inverse and Ill-Posed Problems, Vol 22, pp. 181-194.

S.R. Reddy, and G.S. Dulikravich, (2017). Inverse Design of Cooling of Electronic Chips Subject to Specified Hot Spot Temperature and Coolant Inlet Temperature, Heat Transfer Engineering, Vol 38, pp. 1147-1156

S.R. Reddy, A. Abdoli, G.S. Dulikravich, C.C. Pacheco, G. Vasquez, R. Jha, M.J. Colaco and H.R.B. Orlande, (2017). *Multi-Objective Optimization of Micro Pin-Fin Arrays for Cooling of High Heat Flux Electronics With a Hot Spot*, Heat Transfer Engineering, Vol 38, pp. 1235-1246

A. Abdoli, S.R. Reddy, G.S. Dulikravich, S.M.J. Zeidi, (2017). Effects of Cooling Fluids on High Frequency Electric and Magnetic Fields in Microelectronic Systems with Integrated TSVs, Microelectronics Journal, Vol. 64, pp. 19-28

S.R. Reddy, B. Freno, P.G.A. Cizmas, S. Gokaltun, D. McDaniel and G.S. Dulikravich, (2017). *Constrained Reduced Order Model Based on Proper Orthogonal Decomposition*, Computer Methods in Applied Mechanics and

Engineering, Vol. 321, pp. 18-34

S.R. Reddy, G.S. Dulikravich and S.M.J. Zeidi, (2017). Non-Destructive Estimation of Spatially Varying Material Properties in 3D Objects Using Boundary Thermal Measurements, International Journal of Thermal Sciences, Vol. 118, pp. 488-496

S.R. Reddy and G.S. Dulikravich, (2019). Simultaneous Determination of Spatially Varying Thermal Conductivity and Specific Heat Using Boundary Temperature Measurements, Inverse Problems in Science and Engineering, Vol. 27, pp. 1635-1649

S.R. Reddy and G.S. Dulikravich, (2019). Many-Objectives Differential Evolution Optimization Based on Reference Points: NSDE-R, Structural and Multidisciplinary Optimization, Vol. 60, pp. 1455-1473

S.R. Reddy, H. Sobeiczky, G.S. Dulikravich and M. Gonzalez, (2019). *Bladelets Winglets on Blades of Wind Turbines: A Design Optimization Study*, ASME Journal of Solar Energy Engineering, Vol. 141, pp. 061003-1 061003-6

S.R. Reddy, M.K. Scharrer, F. Pichler, D. Watzenig and G.S. Dulikravich, (2019). Accelerating Parameter Estimation in Doyle-Fuller-Newman Model for Lithium-Ion Batteries, COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronics Engineering, Vol. 38, pp. 1533-1544

S.R. Reddy, G. S. Dulikravich and J. M. S. Zeidi, (2017). Inverse Determination of Spatially Varying Thermal Capacity and Thermal Conductivity in Arbitrary 2D Objects, 7th International Symposium on Advances in Computational Heat Transfer CHT-17, Napoli, Italy, May 28 June 02

S.R. Reddy and G.S. Dulikravich, (2017). Analysis of Anisotropic Graphene Platelet Heat Spreader to Reduce Hot Spot Temperature and Temperature Non-Uniformity, IEEE ITherm2017, Orlando, FL, USA, May 30-June 2.

F.L. Carpenter, P.G.A Cizmas, S.R. Reddy and G.S. Dulikravich, (2019). Controlling Sonic Boom Loudness Through Outer Mold Line Modification: A Sensitivity Study, AIAA Scitech 2019 Forum, San Diego, CA, USA, Jan. 7-11

S.R. Reddy, G.S. Dulikravich , F.L. Carpenter, and P.G.A Cizmas, (2019). Achieving Quiter Supersonic Flight Through Outer-Mold Line Modifications: An Optimization Study, AIAA Aviation 2019 Forum, Dallas, TX, USA, June 17-21