

Development of a Human Centric Multidisciplinary Design Optimization Method Using Fuzzy Logic Systems and Controllers

by

Brian J. Cuneo

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Naval Architecture and Marine Engineering)
in The University of Michigan
2013

Doctoral Committee:

Assistant Professor David Jacob Singer, Chair
Assistant Professor Matthew David Collette
Professor Romesh Saigal
Professor Nickolas Vlahopoulos

© Brian J. Cuneo 2013

All Rights Reserved

To my family and friends

ACKNOWLEDGEMENTS

Throughout my time at the University of Michigan I have received great support from the people around me. Specifically I would like to thank:

My advisor Dr. David Singer, for his guidance, encouragement, and especially patience during the past five years. I will always appreciate the opportunities that Dr. Singer provided me.

Kelly Cooper at the Office of Naval Research for providing funding for this research.

My committee members for your help and thoughtful review of my work.

The members of the Advanced Naval Concepts Research (ANCR) lab. They provided both valuable feedback as well as needed distractions throughout my time pursuing my Ph.D.

Lastly, and most important, I thank my family. Without their support and sacrifices I would never have had the opportunity to reach this point.

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES	vii
LIST OF TABLES	x
LIST OF APPENDICES	xi
ABSTRACT	xii
CHAPTER	
I. Introduction	1
1.1 Background and Motivation	2
1.1.1 Multidisciplinary Design Optimization	2
1.1.2 Set-Based Design	4
1.2 Dissertation Contributions	5
1.3 Dissertation Overview	6
II. Multidisciplinary Design Optimization: Providing Structure to Design Analysis	7
2.1 Introduction	7
2.1.1 Optimization Introduction	7
2.1.2 Optimization Methodologies	10
2.2 Multi-objective Optimization	12
2.3 Multi-Disciplinary Design Optimization	13
2.3.1 MDO Introduction	13
2.3.2 MDO in Ship Design	14
2.3.3 MDO Classifications	16
2.3.4 MDO Notation	17
2.3.5 Overview of Current MDO Algorithms	19

2.4	Chapter Summary	27
III. Hierarchical Fuzzy Logic Controllers: Emulating Human Decision in Controllers		
29		
3.1	Introduction	29
3.1.1	Fuzzy Logic Terminology	30
3.2	Fuzzy Logic Systems	32
3.2.1	Fuzzy Logic System Example	33
3.3	Fuzzy Logic Systems in Set-Based Design	41
3.4	Fuzzy Control Engineering	43
3.4.1	Hierarchical Fuzzy Logic Controller	44
3.5	Chapter Summary	47
IV. Multidisciplinary Design Optimization Using Hierarchical Fuzzy Logic Controllers		
48		
4.1	Introduction	48
4.2	Design Method Overview	48
4.3	Design Intent Membership Function Creation	50
4.3.1	Function Value Preference	51
4.3.2	Gradient ℓ_2 Norm Preference	54
4.3.3	Distance From Global Minimum Preference	55
4.4	Discipline Fuzzy Logic System	58
4.4.1	Fuzzification Interface	62
4.4.2	Fuzzy Rule Bank	65
4.4.3	Fuzzy Inference	67
4.4.4	Defuzzification Interface	71
4.4.5	Discipline FLS Output	73
4.5	Fusion Logic	74
4.6	System Level Optimizer	80
4.7	Constraint Handling Method	80
4.8	Mathematical Example Problem	85
4.8.1	Discipline Analyzers	86
4.8.2	Discipline Fuzzy Logic System	90
4.8.3	Fusion Logic	98
4.8.4	System Level Decision Maker	100
4.8.5	Results Analysis	101
4.9	Chapter Summary	104
V. Ship Design Case Study		
106		
5.1	Introduction	106
5.2	Initial Design Point	106
5.3	Seakeeping Discipline	107

5.3.1	Seakeeping Analysis Tool	107
5.3.2	Seakeeping Design Intent	111
5.3.3	Seakeeping Input Membership Functions	113
5.3.4	Seakeeping Fuzzy Rule Bank	116
5.4	Resistance Discipline	118
5.4.1	Resistance Analysis Tool	118
5.4.2	Resistance Design Intent	119
5.4.3	Resistance Input Membership Functions	120
5.4.4	Resistance Fuzzy Rule Bank	123
5.5	Ship Design Case Study Problem Summary	123
5.6	Results	126
5.7	Chapter Summary	128
 VI. Conclusions and Recommendations for Future Work		130
6.1	Review of Contributions	130
6.2	Recommendations	132
 APPENDICES		134
 BIBLIOGRAPHY		142

LIST OF FIGURES

Figure

1.1	Examples of Different Types of Ship Resistance Models	4
2.1	Single Level MDO Flow Chart	16
2.2	Hierarchical MDO Flow Chart	17
3.1	Example “Tall” and “Not Tall” Fuzzy Sets	31
3.2	Example “Tall” and “Not Tall” Classic Sets	31
3.3	Fuzzy Logic System Flowchart	33
3.4	Example Height Fuzzy Sets	34
3.5	Example Weight Fuzzy Sets	34
3.6	Example Heart Disease Risk Fuzzy Sets	35
3.7	Example Height Input	36
3.8	Heart Disease Decision Matrix	37
3.9	Low Heart Disease Risk Membership Function with Trim Value	38
3.10	Low Heart Disease Risk Membership Function After Trim	38
3.11	Activated Rule #1 Fuzzy Set	39
3.12	Activated Rule #2 Fuzzy Set	39
3.13	Activated Rule #3 Fuzzy Set	39
3.14	Activated Rule #4 Fuzzy Set	40
3.15	Example Defuzzified Output	41
3.16	Modified Fuzzy Logic System Flowchart [72]	42
3.17	Transformation of SBD input preferences to Joint Preference Curve [72]	43
3.18	Fuzzy Logic Controller Flowchart	44
3.19	Hierarchical Fuzzy Logic Controller Flowchart	45
3.20	HFLC in Engineering Design Space	47
4.1	Discipline Preference Assignment Flowchart	49
4.2	System Level Discipline Combination	50
4.3	Objective Function Value Preference Fuzzy Sets	53
4.4	Gradient Norm Value Fuzzy Sets	55
4.5	Distance from Minimum Fuzzy Sets	57
4.6	HFLC MDO Flow Chart	58
4.7	Example Discipline Objective Function Values	59
4.8	Example Discipline $\ \nabla_{\mathbf{x}}f\ _2$ Values	59

4.9	Example Discipline Objective Function Preference Plot	60
4.10	Example Discipline $\ \nabla_{\mathbf{x}}f\ _2$ Preference Plot	61
4.11	Example Discipline Distance From Minimum Preference Plot	61
4.12	Example Discipline Objective Function Plot Section At $x_1 = 0$	63
4.13	Example Discipline $\ \nabla_{\mathbf{x}}f\ _2$ Preference Plot Section at $x_1 = 0$	64
4.14	Example Discipline Distance From Minimum Preference Plot Section at $x_1 = 0$	64
4.15	Example Decision Matrix When “Distance From Minimum Prefer- ence” is High	66
4.16	Example Decision Matrix When “Distance From Minimum Prefer- ence” is Med	66
4.17	Output Preference Fuzzy Sets	69
4.18	High Preference Output Membership Function With Rule #1 Trim Value	70
4.19	High Preference Output Membership Function Trimmed for Rule #1 Trim Value	70
4.20	Fuzzy Inference Output	71
4.21	Sample Point $\mathbf{x} = (0,1)$ Defuzzification	73
4.22	Example Discipline FLS Output	74
4.23	Example Discipline 2 Objective Function Values	76
4.24	Example Discipline 3 Objective Function Values	76
4.25	Example Discipline 2 FLS Output	77
4.26	Example Discipline 3 FLS Output	77
4.27	Fusion Logic Output Plot	78
4.28	Fusion Logic Output Plot Rotated 90 Degrees	79
4.29	Fusion Logic Output Contour Plot	79
4.30	Example Constraint Fuzzy Sets	82
4.31	Example Decision Matrix When All Constraints Are Satisfied	83
4.32	Example Decision Matrix When A Constraint Is Violated	83
4.33	Example Constraint Fuzzy Sets Plots	84
4.34	Example Discipline FLS Output Subject to Constraint	85
4.35	D1 Objective Function Plot	87
4.36	D1 Gradient ℓ_2 Norm Plot	87
4.37	D2 Objective Function Plot	88
4.38	D2 Gradient ℓ_2 Norm Plot	88
4.39	D3 Objective Function Plot	89
4.40	D3 Gradient ℓ_2 Norm Plot	89
4.41	D1 Objective Function Preference Plot	90
4.42	D2 Objective Function Preference Plot	91
4.43	D3 Objective Function Preference Plot	91
4.44	D1 $\ \nabla_{\mathbf{x}}f\ _2$ Preference	92
4.45	D2 $\ \nabla_{\mathbf{x}}f\ _2$ Preference	93
4.46	D3 $\ \nabla_{\mathbf{x}}f\ _2$ Preference	93
4.47	D1 Distance From Optimum Preference	94
4.48	D2 Distance From Optimum Preference	95

4.49	D3 Distance From Optimum Preference	95
4.50	Mathematical Example Constraint Fuzzy Sets Plot	96
4.51	D1 FLS Output	97
4.52	D2 FLS Output	97
4.53	D3 FLS Output	98
4.54	Fusion Logic Output Plot	99
4.55	Fusion Logic Output Plot Rotated 90 Degrees	99
4.56	Fusion Logic Output Contour Plot	100
4.57	Plot of Sampled Points Highlighting Pareto Front	102
4.58	Example Problem Pareto Front with MDO Results	102
4.59	Mathematical Example Distance to Utopia Plot	104
5.1	Parent Hull	107
5.2	Parent Hull Lines Drawing	109
5.3	ISO 2631/3 Severe Discomfort Boundaries [1]	112
5.4	Heave Acceleration Fuzzy Sets	114
5.5	Pitch Acceleration Fuzzy Sets	115
5.6	Roll Acceleration Fuzzy Sets	116
5.7	Seakeeping Fuzzy Rule Bank	117
5.8	Output Preference Fuzzy Sets	117
5.9	C_P Preference Fuzzy Sets	122
5.10	Resistance Rule Matrix	123
5.11	Case Study Design FLS Outputs	127

LIST OF TABLES

Table

2.1	Summary of Optimization/MDO Notation	27
3.1	Example FLS Defuzzification Input	41
4.1	Sample Point $\mathbf{x} = (0, 1)$ Fuzzy Set Membership Values	65
4.2	Activated Fuzzy Rules	67
4.3	Sample Point Output Linguistics and Membership Values	68
4.4	Sample Point Defuzzification Input Values	72
4.5	Mathematical Example Discipline Analyzers Output	90
4.6	Mathematical Example Results Comparison	101
4.7	MDO Method Results Comparison	103
5.1	Initial Design Variable Values	107
5.2	Resistance Constraints Goal and Threshold Values	123
5.3	Case Study Design Variables	124
5.4	Case Study Parameters	124
5.5	Ship Case Study Results	126

LIST OF APPENDICES

Appendix

A. Output of Seakeeping Prediction Program 135

B. Output of Power Prediction Program 139

ABSTRACT

Development of a Human Centric Multidisciplinary Design Optimization Method
Using Fuzzy Logic Systems and Controllers

by

Brian J. Cuneo

Chair: David Singer

During early stages of design of large scale engineering systems, such as ships, competing objectives of multiple discipline analyzers must be taken into account. Currently, tools such as Multidisciplinary Design Optimization (MDO) provide an automated method for analyzing trade-offs between competing discipline design objectives. While automated design methods are useful in the design world, automation removes the human ability to interpret results of analysis. Current design tools lack the ability to incorporate the intent of experienced designers in the communication between disciplines, so MDO results are dependent on the fidelity of the analysis models being used rather than human input. When evaluating designs, a team of experienced engineers communicates information about the design space to each other with respect to their own discipline's objectives and design constraints.

This thesis presents a new method of MDO which provides a framework to emulate the intent of an human designer in a MDO optimizer. The method uses the ideas of managing trade-offs and modeling discipline interactions from MDO. Typical MDO algorithms utilize a system level optimizer, which receives information from discipline

optimizers and moves toward a globally optimal solution. Generally, the information communicated to the system level optimizer is point data about the individual discipline's optimal point, which is then interpreted as the potential improvement from the current system design state. The proposed algorithm would communicate the discipline's preferred areas of the design space to be combined with the other discipline's preferences in an attempt to find a globally optimal solution.

This design intent modeling is introduced through fuzzy logic systems and fuzzy logic controllers. Fuzzy logic systems provide a method to interpret design analysis tools and glean more information to use in the decision making process. Fuzzy logic controllers are often used to emulate human decision models in the control of physical systems. These methods are extended to evaluation of designs to combine data from multiple disciplines to find the optimal system design while considering trade-offs between multiple disciplines. This dissertation will present the development of a novel method of MDO which incorporates fuzzy logic systems and controllers in the MDO optimizer. The method will be evaluated using a basic mathematical test case, then extended to a ship design case study to demonstrate the benefits of the design intent models.

CHAPTER I

Introduction

Ship design presents a unique set of challenges. During early stages of design, alternatives must be evaluated with relatively little information available. Early on, design are evaluated using low fidelity parametric models. In order to use more advanced analysis tools, designers are forced to make critical design decisions. Ideally, designers would be able to gather more data about the feasible designs before these decisions are made because making design decisions prematurely could inadvertently eliminate desirable solutions.

The motivation to evaluate feasible solutions before decisions are made has led to the development of design methods which explore feasible solutions. One widely used method in engineering design is multidisciplinary design optimization (MDO). MDO searches alternatives for a design that best satisfies the objectives of multiple discipline analyses. However, the approach used in the design selection process of MDO is purely objective and removes any human interpretation of information.

This dissertation will introduce a new MDO method which incorporates a human intent model into the MDO optimizer. The goal of the method presented is to increase the fidelity of the optimizer for decision making instead of relying on higher fidelity models in analyzing designs.

1.1 Background and Motivation

Throughout the design of a system, engineers search to find solutions to the design requirements. While searching alternatives, designs are evaluated by human designers and computer evaluation programs. Both humans and computers possess unique strengths in relation to the creation and evaluation of designs, such as a human’s intent and understanding of analysis models or a computer’s calculation speed and objectivity.

1.1.1 Multidisciplinary Design Optimization

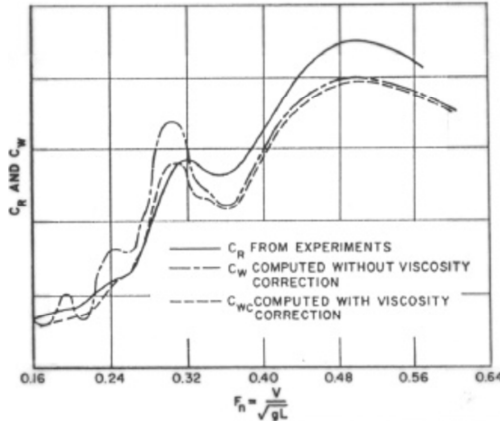
Design optimization is one automated design method used to search for the best design of an engineering system. Optimization algorithms rely on mathematical models representing the system’s performance with respect to a design objective. For small systems with low complexity, single objective function optimization algorithms may be sufficient, but when dealing with entire complex systems, these techniques lack the ability to model the interactions between disciplines with competing goals. For complex engineering systems, a multidisciplinary design optimization (MDO) algorithm that can model interactions between disciplines is necessary because “the performance of a multidisciplinary system is driven not only by the performance of the individual disciplines but also by their interactions” [48].

These algorithms provide a structured approach to considering design trade-offs. When dealing with complex systems with interacting disciplines, MDO algorithms offer an efficient method to explore a design space. As MDO gains acceptance in different industries, many different algorithms have been developed (outlined in [48]) allowing MDO to adapt to virtually any design problem. However, the overall goal of all MDO algorithms remains to “provide a more consistent, formalized method for complex system design than is found in traditional approaches” [40].

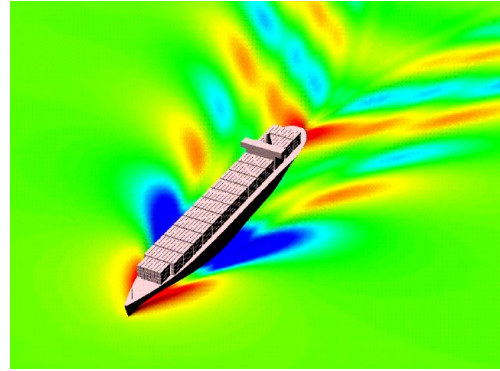
While multidisciplinary methods are successful in locating an optimum design

for a particular problem, these automated approaches remove the human designer from the creation and evaluation of design alternatives. When the human designer is removed, the benefits that humans bring to the design evaluation is limited. One of the main benefits of human design input is the knowledge of the intent of the system being designed and the understanding of the analysis models used. Knowing the goal of the overall system, a human designer can interpret the results of each discipline analysis and use this information to understand better the trade-offs that must be made.

Early stages of ship design rely heavily on the the use of low fidelity parametric models, which generally show trends compared to the highly accurate models that are used later in the design process. For example, different mathematical models would be used to evaluate the resistance characteristics of a ship during different design stages. Figure 1.1a, displays a parametric model used in early stages of design where the partial resistance of a ship can be estimated from basic parameters [46]. The results, when compared to a computational fluid dynamics (CFD) analysis (figure 1.1b), would likely be different even if the basic parameters of the ship being analyzed were the same; however, to use higher fidelity tools such as CFD, more information is required than what is available in the early stages of design. Therefore, to run a CFD code, designers would need to make critical decisions early in the design so that the higher fidelity models could be used and more precise calculations become possible.



(a) Example of parametric ship resistance analysis [46]



(b) Example CFD analysis of ship resistance [54]

Figure 1.1: Examples of Different Types of Ship Resistance Models

While analyzing parametric models, human designers glean more information from the low fidelity models than a MDO method that only analyzes the objective function values when evaluating trade-offs. A designer better understands the limitations of analysis models based on how the model was developed. Also, the data that is output from the models can relate back to the true intent of the design, even if robust formulas are not available.

1.1.2 Set-Based Design

Research in set-based design has shown strong results in early stages of ship design based on the input of human designers [72, 20, 73]. In SBD, a design team analyzes different disciplines of the ship design, and communicates preferences between design disciplines to locate globally optimal designs. Engineers analyzing different disciplines of the ship are able to interpret the results from analysis models without relying on automation to create and evaluate the feasible designs. This research is focused on emulating the design intent of the designers into a MDO algorithm to evaluate quickly early stage designs alternatives.

If the design intent and knowledge of experienced designers can be built into a

MDO process, then an automated design space exploration method may be able to make more intelligent decisions with the limited models used during early stage design. The remainder of this dissertation will describe the development of a MDO method that provides an adaptable approach to building designer intent into an automated design process. The structured approach to analyzing trade-offs derived from current MDO methods will be combined with Hierarchical Fuzzy Logic Controllers (HFLC). HFLC have been used in many applications to emulate human decision making in the control of a physical system, this process will be extended to making decisions in the engineering design process.

1.2 Dissertation Contributions

This dissertation presents the development of a new MDO method. The significant contributions to the field from this work are as follows:

- **Developed a new method of multidisciplinary design optimization**
- **Extension of set-based design fuzzy logic systems to MDO**
- **Extension of Set-Based Design Fuzzy Logic System using Hierarchical Fuzzy Logic Controllers to Combine Discipline Outputs**
- **Applied Hierarchical Fuzzy Logic Controllers to MDO**
- **Incorporated Expert Opinion into MDO Optimizer**

The contributions listed previously are the center of this research. Together they contribute to the overarching goal of aiding naval architects during early stage design evaluation.

1.3 Dissertation Overview

This dissertation is organized as follows. The following chapter will introduce multidisciplinary design optimization, beginning with a background on the development of MDO methods and describing how MDO is used in the current research. Chapter III will introduce fuzzy logic, fuzzy logic systems, and hierarchical fuzzy logic controllers. Chapter IV will provide a detailed description of the MDO method developed in this thesis and a simple example problem. Chapter V will develop a ship design case study as an example of the MDO method. Finally, chapter VI draws some conclusions and offers future research recommendations.

CHAPTER II

Multidisciplinary Design Optimization: Providing Structure to Design Analysis

2.1 Introduction

This chapter will describe various methods of MDO while providing background information and relevant literature. MDO is the basis of the method presented in this thesis, and this chapter will present information needed for the description of the method.

First, a general description of optimization will be provided. Then, MDO will be introduced, and previous methods of MDO will be discussed to present different ways that communication and trade-offs between disciplines are handled.

2.1.1 Optimization Introduction

During the development of a new product, engineers search design spaces for the best possible design out of all feasible designs. Models are developed to provide “an abstract description of the [product] using mathematical expressions of relevant natural laws, experience, and geometry” [56]. These models allow the engineer to make decisions based on quantitative measures of performance. The use of mathematical models to describe a product has led to the development of design optimization meth-

ods to generate and evaluate alternatives in search of the design that best achieves the design goal while satisfying any constraints placed on the final product.

Optimization methods provide an efficient method to converge on an optimal solution. The use of the word optimal should be taken in context, realizing that the “optimal” solution selected by an optimization algorithm is dependent on not only the mathematical models used, but also the specific goals and constraints in the problem’s formulation; the optimal design is thus subject to interpretation. In the following sections, basic optimization terminology will be presented for use in the remainder of this dissertation.

2.1.1.1 Optimization Notation

Basic Optimization Statement: Equation (2.1) shows a general optimization problem statement. This statement is in *Negative Null* form, [56], which requires all inequalities to be less than or equal to zero. In this statement, f is the *objective function*, \mathbf{x} is the vector of *design variables*, \mathbf{g} is the vector of *inequality constraints*, and \mathbf{h} is a vector of *equality constraints*. The goal of the optimization statement is to find the vector of design variables that produces the smallest value of the objective function while satisfying all constraints.

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}) \\ & \text{Subject To:} && \mathbf{g}(\mathbf{x}) \leq 0 \\ & && \mathbf{h}(\mathbf{x}) = 0 \end{aligned} \tag{2.1}$$

Objective Function: The objective function (f) of an optimization statement defines the performance metric that will be optimized. In a resistance optimization problem, the objective function would be whichever tool is used to calculate the resistance for a given set of design variables and parameters. This function could be a simple parametric model of resistance or a more expensive analysis tool, like a com-

putational fluid dynamics solver. When an optimization algorithm has converged on the minimum objective function value, that minimum is denoted f^* .

Design Variable: Design Variables are inputs to the system that the designer has control over. Examples of design variables (DV) in preliminary ship design could be the ship Length (L), Beam (B), and Draft (T). In standard optimization notation, the DVs are listed in a $I \times 1$ vector denoted \mathbf{x} , where I is the number of DVs in the current design problem. Generally, each DV is given an upper and lower bound, although some variables may be feasible throughout all real numbers. Like the objective function, when the minimum is found, the optimal DV values are denoted \mathbf{x}^* .

Constraint Function: A constraint function (g or h) in an optimization problem defines a boundary to the design space. In equation 2.1, \mathbf{g} is a $k \times 1$ vector of inequality constraints, where k is the number of inequality constraints and \mathbf{h} is the vector of equality constraints. These boundaries may be either a physical requirement of the model or an artifact of the design tools used. Simple examples of constraints include: restricting the dimensions of a ship such that it is able to fit through the Panama Canal or a design tool restriction such as a L/B restriction based on the resistance model. Generally, a constraint can be either an equality constraint or an inequality constraint. For example, restricting the beam to fit through the Panama Canal would be an inequality constraint, while restricting the beam to exactly 33.31 meters (the maximum value that can fit through the Panama Canal) would be an equality constraint. For many optimization architectures, only inequality constraints will be used. However, no modeling problems are introduced since equality constraint can be modeled as two inequality constraints.

Parameter: A parameter in a design optimization problem is an attribute of the system over which the designer has no control. Examples include physical constants, such as the acceleration due to gravity, or any dependent variables that are changed

by the DVs that the designer does not have direct control over, such as a ship displacement for the given DVs of L, B, T, as well as the ship's block coefficient (C_b).

2.1.2 Optimization Methodologies

Many methods exist for evaluating an optimization problem (equation (2.1)), with different strengths and weaknesses depending on the problem being addressed. The literature on optimization algorithms is vast and thorough, so specific algorithms will not be described here. Instead general classifications of algorithms and their approach to finding an optimal solution will be discussed.

2.1.2.1 Numerical Optimization Techniques

The classic approach to design optimization provides a closed form solution to an optimization problem by solving a system of equations. Using methods, as described in [4] and [56], optimal solutions can be found and verified as optimal. These methods are limited in the problems that may be solved because they are constrained by the requirement that the objective and constraint functions must be continuous in both the first and second derivative.

For problems with more complex objective functions, or where expensive analysis tools are desired, numerical techniques are used for optimization. Iterative numerical techniques provide an efficient method for moving through the design space when a closed form solution is difficult or impossible to calculate. These methods include gradient type methods [44] and Newton type methods [14]. Numerical techniques for solving optimization problems use local approximations of the objective and constraint functions' gradients and Hessian matrix to find a series of steps resulting in decreasing objective function values until a stopping criterion is met at a function's optima.

Gradient or steepest descent methods rely on the gradient of the objective function

to make a linear approximation while selecting the search direction. Newton type methods include information from the Hessian matrix or a modified Hessian matrix to give a quadratic approximation of the objective function for selecting the search direction. Gradient type methods, such as the steepest descent method, generally perform poorly when the algorithm is near the objective function minimum[4]. On the other hand, Newton type methods perform well when close to the minimum with faster convergence, but convergence is not guaranteed unless the starting point is near the optimal value [4].

2.1.2.2 Population Based Optimization Techniques

Another group of optimization techniques involves a more heuristic approach to evaluating the design space. These types of methods are generally used when the design space is extremely non-convex, resulting in many local optima. Examples of heuristic type optimization algorithms include genetic algorithms and particle swarm optimization. These methods vary greatly in the approach used to find an optimal solution in a design space. Nonetheless, both methods do not use a single path through the design space as compared to the numerical techniques described previously. Instead, these methods rely on a sampling of design points to find the optimal solution.

Genetic algorithms attempt to emulate the theory behind evolution and “survival of the fittest” [31]. A “population” is initialized within the design space with each “individual” in the population given a fitness value (generally the output of the optimization objective function). The fitness values of all individuals in the population are then compared. The individuals with the strongest fitness values are then allowed to “reproduce” to create a new generation of the population. As the generations continue this process, the population converges toward an optimal solution [5].

Particle Swarm Optimization (PSO) relies on the communication of many different

“particles” or possible solutions in a design space, and swarm intelligence to move to an optimal solution [15]. Like a genetic algorithm, a population of “particles” is initialized, and each individual in the population is given a fitness value. Based on the best fitness value found by the population, the particles move toward the best solution. The motion of the particles is dependent on the particular formulation of PSO, but the overall goal of all PSOs finds if a better solution is available in the design space because it will be found as other particles move toward the current particle with the greatest fitness value [64].

The techniques described in this section provide many ways to search a design space of an optimization problem with a single objective, but for design of more complex systems, multiple objectives must be examined in order to find the best compromise leading to an overall optimal design. The next sections will introduce different architectures that allow these techniques to be used in a system with competing design objectives.

2.2 Multi-objective Optimization

A brief introduction of multi-objective optimization (MOO) will be presented in this section to introduce elements used in analyzing solutions with a MDO structure. In the design of engineering systems, generally a single objective is not desired. A ship, for example, would be interested in minimizing the resistance to reduce fuel cost and also maximizing the amount of cargo that can be carried at a single time. These two objectives have competing goals as increasing the weight of a ship would cause more wetted surface leading to more resistance. As described in [47], the general multi-objective optimization problem is shown by equation (2.2).

$$\begin{aligned}
& \underset{\mathbf{x}}{\text{minimize}} && \mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}) f_2(\mathbf{x}), \dots, f_j(\mathbf{x})]^T \\
& \text{subject to:} && \mathbf{g}(\mathbf{x}) \leq 0 \\
& && \mathbf{h}(\mathbf{x}) = 0
\end{aligned} \tag{2.2}$$

Where, \mathbf{F} is a vector of all discipline objective functions, f_j is the objective function for discipline j , \mathbf{g} is the collection of all inequality constraints, and \mathbf{h} is the collection of all equality constraints.

In contrast to MDO, which attempts to find a single optimal point, MOO generally produces a set of solutions which cannot be shown as mathematically superior to each other. This set of points is called a Pareto front. The Pareto front includes all points that are non-dominated, or points where any improvement to one discipline would result in a reduction in performance in at least one other discipline [56].

One criterion reviewed when analyzing solutions on the Pareto front is the solutions' distance to the utopia point. The utopia point of a MOO problem is defined as the point where all objectives are at their optimum objective function value [56]. This point is usually infeasible, as it would indicate that all objectives obtain their optimal values at the same DV values with no negative impact on other objectives. While the results of MDO and MOO statements are different, understanding Pareto Frontiers and Utopia is helpful in analyzing the results of MDO problems.

2.3 Multi-Disciplinary Design Optimization

2.3.1 MDO Introduction

As products become more complex, it becomes difficult to model a system in a single mathematical model that accurately simulates the interactions of the entire system. Multidisciplinary Design Optimization (MDO) allows a system to be broken down into subsystems based on disciplines so that the interactions can be more

accurately modeled.

MDO originates from aerospace engineering in the optimization of wings which have coupled design disciplines with respect to aerodynamics and structural response [24, 71, 21, 22, 41]. MDO has since expanded to other engineering discipline including: other aspects of aerospace design [33, 53, 79, 30], automotive design [34, 39, 38, 26, 57], and civil engineering [3, 18, 9, 17].

2.3.2 MDO in Ship Design

Recently, there has been a push in ship design for tools capable of rapidly analyzing multidisciplinary systems to reduce costs. For example, the U.S. Navy has “expressed a desire for even more sophisticated design and analysis tools capable of facilitating advanced design methods” [19]. This push has led to the investigation of advanced methods of MDO and their uses in ship design. The following section will present examples of current applications of MDO in ship design.

The ideas from wing optimization, dealing with closely coupled aerodynamics and structures, has been extended to propeller optimization in [85] and [76], where fluid-structure interactions must be considered. These types of MDO problems produced strong results, which demonstrates there are readily available applications of MDO to ship design aspects.

The use of MDO in larger aspects of ship design has also been studied. An investigation of the multidisciplinary optimization of a ship hull based on resistance and hydrodynamic seakeeping was performed by Peri and Campana [63]. This application displayed the difficulties in modifying a ship hull, which cannot be easily modeled mathematically like an aircraft wing or propeller blade section. This difficulty leads to the use of high fidelity analysis tools in a MDO on ship hull design.

In response to the desire for advanced design tools in ship design, the U.S. Navy has began implementation of MDO methods in their Rapid Ship Design Environment

(RSDE). Gray et al. used MDO methods and U.S. Navy design analysis tools to optimize a frigate with respect to resistance operating characteristics and seakeeping performance [19]. Again, the results show the difficulties in varying a ship hull for use in higher fidelity, non-parametric design analysis tools.

The inclusion of uncertainty is one attempt to increase the effectiveness of applying MDO to the models that are available in early naval architecture design. The affects of uncertainty on cost in the multidisciplinary design optimization of a submarine were examined in [29]. Introducing uncertainty to a preliminary ship design MDO was explored in [27], where a container ship hull was optimized with respect to minimizing weight, maximizing cargo capacity, and minimizing transportation cost. Fuzzy membership functions were introduced into MDO ship design in [12] as another method of locating a robust design based on low fidelity models. Here, constraint relaxation was investigated by using fuzzy utility functions in the constraints to introduce goal and threshold constraint values.

These methods increase the robustness of the design selected with respect to the constraints, and are intended to reduce the probability of design failures in later stages of design. That being said, the results are still limited by the fidelity of the analysis tools with respect to the amount of information used in decision-making.

While there are many example of MDO being successfully used in naval architecture design, these methods do not address the issues presented in this thesis. Using high fidelity models in applications led to promising results, but suffered from modeling limitations. Methods of introducing uncertainty led to a more robust final result in early stages of design, but MDO results are still dependent on the models used without the added value of a human designer. The MDO architecture presented in Chapter IV will present a framework to emulate the strengths that a human designer can add to the process of creating and evaluating alternatives in an automated design algorithm.

2.3.3 MDO Classifications

Many types of MDO algorithms have been developed since the ideas were first introduced, and an overview of current algorithms will be introduced in section 2.3.5. MDO algorithms generally fall into two main categories: *monolithic* systems, with all disciplines at the same level and a single system optimizer (figure 2.1), and systems with a *hierarchical* design that generally includes a system optimizer and subsystem optimizers, which allow for local exploration within the subsystem (figure 2.2) [48, 78]. In a monolithic MDO, a single optimization algorithm controls the movement through the design space while disciplines are able to pass information between each other. In a hierarchical type MDO formulation, the system has its own objective function, and communicates the information between disciplines and facilitates movement through the design space while allowing the local computation of the subsystems.

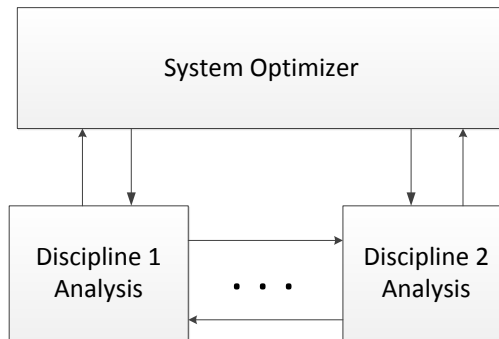


Figure 2.1: Single Level MDO Flow Chart

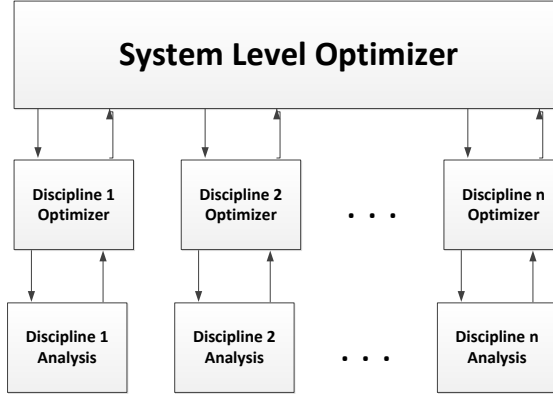


Figure 2.2: Hierarchical MDO Flow Chart

2.3.4 MDO Notation

Equations (2.3) and (2.4) show generalized MDO formulations to demonstrate notation that will be used throughout the remainder of this dissertation. Equation (2.3) is representative of a monolithic architecture where only a system level optimization statement is used.

$$\begin{aligned}
 & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}, \mathbf{y}) \\
 & \text{subject to:} && \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\
 & && \mathbf{h}(\mathbf{x}, \mathbf{y}) = \mathbf{0}
 \end{aligned} \tag{2.3}$$

Where, f is the system objective function. \mathbf{x} is the vector of DVs, the vector includes all DVs used in each discipline analysis. \mathbf{y} is the set of *coupling variables*, where $\mathbf{y} = [y_1, y_2, \dots, y_k]$ and k is the total number of coupling variables. The coupling variables are outputs of one discipline that are used by other disciplines. For example, the resistance model of a hull design, is dependent on the structural weight of the structures discipline. Each coupling variable, y_k , is dependent on the DVs, i.e. $y_k = y_k(\mathbf{x})$. \mathbf{g} and \mathbf{h} are vectors of inequality and equality constraints, respectively. The constraint functions are collections of constraints from all disciplines to ensure design feasibility for all disciplines when the system converges.

For a hierarchical architecture, there is an optimization statement for each of the different levels, as shown in equation 2.4.

System Level:

$$\begin{aligned} & \underset{\mathbf{x}_S}{\text{minimize}} && f_S(\mathbf{x}_S, \mathbf{y}) \\ & \text{subject to:} && \mathbf{g}_S(\mathbf{x}_S, \mathbf{y}) \leq \mathbf{0} \\ & && \mathbf{h}_S(\mathbf{x}_S, \mathbf{y}) = \mathbf{0} \end{aligned}$$

(2.4)

Discipline Level:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && f_j(\mathbf{x}_G, \mathbf{x}_j, \mathbf{y}) \\ & \text{subject to:} && \mathbf{g}_j(\mathbf{x}_G, \mathbf{x}_j, \mathbf{y}) \leq \mathbf{0} \\ & && \mathbf{h}_j(\mathbf{x}_G, \mathbf{x}_j, \mathbf{y}) = \mathbf{0} \end{aligned}$$

Here, f_S is the system level objective function, \mathbf{x}_S is the vector of system DVs, which is expanded to the form $\mathbf{x}_S = [\mathbf{x}_G, \mathbf{x}_1, \dots, \mathbf{x}_j]$ to include \mathbf{x}_G , the vector of global DVs used in two or more disciplines analyses, and \mathbf{x}_j which are DV vectors used only in discipline j , \mathbf{y} is the vector of coupling variables described previously, \mathbf{g}_S and \mathbf{h}_S are the system level inequality and equality constraints. At the discipline level, for each j discipline, f_j is the objective function, \mathbf{g}_j are the inequality constraints for the discipline, and \mathbf{h}_j are the equality constraints for the discipline. For the discipline objective and constraints, the inputs into the functions are the vector of global variables (\mathbf{x}_G), the vector of discipline variables (\mathbf{x}_j), and the coupling variables (\mathbf{y}). Again, in some formulations, no equality constraints will be used, but are easily replace by a pair of inequality constraints.

2.3.5 Overview of Current MDO Algorithms

The following sections will describe some highly utilized MDO approaches to demonstrate different methods of structuring multiple discipline analyses into a single system level design statement. This section is not meant to be inclusive of all MDO architectures, but to highlight methods of structuring trade-offs and certain aspects that pertain to this thesis.

2.3.5.1 Multidisciplinary Feasibility (MDF)

The first algorithm presented, MDF, falls into the category of monolithic, as shown in figure 2.1. Considered the “most common way of posing a MDO problem,” MDF relies on a full Multidisciplinary Analysis (MDA) at every DV location that is evaluated [11]. A MDA requires that all disciplines remain consistent for the DV location to be considered. Consistency, in this case, implies that all coupling variables must be consistent between design disciplines. While this type of formulation is more expensive than other types of MDO architectures [11], the resulting solutions from a MDF problem are guaranteed to be consistent. The general form of a MDF architecture is shown in equation (2.5)

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}, \mathbf{y}) \\ & \text{subject to:} && \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \end{aligned} \tag{2.5}$$

Here, the notation is equivalent to what is described in section 2.3.4. In MDF, feasibility is ensured because the constraints are evaluated at each DV location and are dependent on the current coupling variable values at that same location.

The MDF approach provides a method to optimize a complex system with respect to multiple discipline analyses. The approach ensures that disciplines which are highly coupled remain consistent by an iterative approach. An example of this consistency is in design of an aircraft wing. Aerodynamics are evaluated, calculating lift and drag

forces to the wing structure, thus deforming the wing [11]. This new deformed wing will have different aerodynamic properties. In MDF, these two discipline analyses will be evaluated iteratively until an equilibrium is reached and the result is returned to the optimizer.

This idea of ensuring consistency is handled by a MDA. The MDA analysis at each discipline results in high numbers of objective function evaluations, as each MDA requires multiple evaluations of each discipline objective. This presents issues when computationally expensive analysis models are used. Still, MDF has strong results for MDO problems where a single system goal is desired. However, the MDF approach does not allow for multiple goals within the system.

2.3.5.2 Simultaneous Analysis and Design (SAND) and All At Once (AAO)

Two closely related monolithic algorithms, the AAO and SAND architectures, both relax the requirements to conduct a full MDA at each DV evaluation. The AAO architecture is considered “the fundamental optimization problem from which all other problem statements are derived” [48]. There is a disagreement in the literature over the naming of methods, for example, the formulation in equation (2.6) is named AAO in [11] and SAND in [48, 78]. Martins et al. [48] list a formulation for AAO that is different than the AAO form in equation (2.6), which is presented by Cramer et al. [11]. However, these two definitions can be easily reconciled by removing the linking variables (\mathbf{y}) and adding set of consistency constraints [48].

$$\begin{aligned}
 & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}, \mathbf{y}) \\
 \text{Subject To:} &&& \mathbf{g}(\mathbf{x}, \mathbf{y}) \geq 0 \\
 &&& R_j(\mathbf{x}, M(\mathbf{x}, \mathbf{y}), \mathbf{y}) = 0
 \end{aligned} \tag{2.6}$$

Again, f is the system level objective function, \mathbf{g} is the vector of constraints,

\mathbf{x} is the vector of DVs, and \mathbf{y} is the vector of coupling variables. R_j is the set of “residual” equations for discipline j . The residuals are the governing equation of a discipline which are necessary for “feasibility,” as defined by [11], which is simply for a given set of inputs, the residual equations are solvable. The residuals for discipline j are evaluated using DVs (\mathbf{x}), and the coupling variables (\mathbf{y}). The function M is a mapping that may be necessary to put the DVs in a usable form for the discipline being analyzed (e.g. creating a mesh for a computational fluid dynamics code.)

In contrast to the MDF method, the coupling variables are not necessarily consistent at each DV location in the SAND and AAO methods. Instead, a set of residual constraints are created to reach consistency when the MDO is evaluated. When the optimization converges, the residual constraints must be satisfied, which leads to consistency at the optimal solution. The major drawback of the AAO and SAND formulations is that the residual constraints and coupling variables lead to high numbers of constraints and high numbers of DVs. This drawback leads to problems with the number of function iterations when dealing with gradient and Newton method solvers [11].

The need for solving the residual equations also presents problems when certain software is used for analyses. Some software used in optimization does not produce the residual equations needed for SAND and AAO constraints [48]. Thereby imposing limits on the problems that can be addressed using these methods.

Finally, the SAND and AAO methods still only allow for a single optimization statement. Thus, these methods do not allow for competing discipline objectives, but are desired when highly coupled disciplines affect an overall optimization goal.

2.3.5.3 Analytical Target Cascading (ATC)

The first hierarchical type MDO that is introduced is ATC. Adapted from a method of system engineering to fit a MDO framework, ATC produces target DV

and objective function values derived from individual discipline optimization to guide the MDO process [36, 2]. The formulation of ATC is formed as a tri-level problem, a top or super system level problem, an intermediate or system level problem, and a bottom or subsystem level problem. The formulation as presented in [37] is shown by level in equations (2.7) - (2.9). The subscript T is applied to all variables, objective functions, and constraints for the top level optimization statement, s is the same for the system level, and ss for the subsystem level.

Top Level

$$\begin{aligned}
& \underset{\mathbf{x}, \mathbf{y}_s, \epsilon_f, \epsilon_y}{\text{minimize}} && \|f_T - f_T^{target}\| + \epsilon_f + \epsilon_y \\
& \text{subject to:} && \sum_J \|f_{s,j} - f_{s,j}^{target}\| \leq \epsilon_f \\
& && \sum_J \|\mathbf{y}_{s,j} - \mathbf{y}_{s,j}^{target}\| \leq \epsilon_y \\
& && \mathbf{g}_T(\mathbf{x}_T), \mathbf{h}_T(\mathbf{x}_T)
\end{aligned} \tag{2.7}$$

For the top level problem, f_T is the top level objective evaluated for the current design and coupling variables, f_T^{target} is the target value of the top level objective function, $f_{s,j}$ and $f_{s,j}^{target}$ are the current and target values for each j th system level objective, $\mathbf{y}_{s,j}$ and $\mathbf{y}_{s,j}^{target}$ are the current and target values for the vector of coupling variables in discipline j , \mathbf{g}_T and \mathbf{h}_T are the vectors of constraints for the top level, and ϵ_f and ϵ_y are the consistency terms placed on the system level objective functions and coupling variables, respectively. These consistency terms are treated as extra DVs in the optimization.

System Level

$$\begin{aligned}
& \underset{\mathbf{x}_s, \mathbf{y}_s, \mathbf{y}_{ss}, \epsilon_f, \epsilon_y}{\text{minimize}} && \|f_s - f_s^{target}\| + \|\mathbf{y}_s - \mathbf{y}_s^{target}\| + \epsilon_f + \epsilon_y \\
& \text{subject to:} && \sum_J \|f_{ss,j} - f_{ss,j}^{target}\| \leq \epsilon_f \\
& && \sum_J \|\mathbf{y}_{ss,j} - \mathbf{y}_{ss,j}^{target}\| \leq \epsilon_y \\
& && \mathbf{g}_s(\mathbf{x}_s), \mathbf{h}_s(\mathbf{x}_s)
\end{aligned} \tag{2.8}$$

For the subsystem level problem, f_s is the objective function of the current system problem being analyzed, f_s^{target} is that target objective for the current system, \mathbf{y}_s and \mathbf{y}_s^{target} are the current and target values for the systems coupling variables, $f_{ss,j}$ and $f_{ss,j}^{target}$ are the current and target values for the objective function for each subsystem j , $\mathbf{y}_{ss,j}$ and $\mathbf{y}_{ss,j}^{target}$ are the current and target values for discipline j 's coupling variables, \mathbf{g}_s and \mathbf{h}_s are the constraints on the current system level problem, ϵ_f and ϵ_y are the consistency terms as described previously.

Subsystem Level

$$\begin{aligned}
& \underset{\mathbf{x}_{ss}, \mathbf{y}_{ss}}{\text{minimize}} && \|\mathbf{f}_{ss} - \mathbf{f}_{ss}^{target}\| + \|\mathbf{y}_{ss} - \mathbf{y}_{ss}^{target}\| \\
& \text{subject to:} && \mathbf{g}_{ss}(\mathbf{x}_{ss}), \mathbf{h}_{ss}(\mathbf{x}_{ss})
\end{aligned} \tag{2.9}$$

In the system level problem, f_{ss} is the objective function of the current subsystem problem being analyzed, f_{ss}^{target} is the target objective for the current subsystem, \mathbf{y}_{ss} and \mathbf{y}_{ss}^{target} are the current and target values for the subsystem coupling variables.

For the system level optimization statement, each system in the design problem would have its own optimization problem, and within each system, each subsystem would have a subsystem level optimization problem. The top level attempts to minimize the difference between its own objective function and the target value for that objective, and the consistency terms (ϵ_f and ϵ_y), which restrict the deviation from of

the system objective functions and coupling variables. The system levels act in the same manner but also try to minimize the difference between their own coupling variables. Finally, the subsystem level attempts to minimize the difference between its own objective function and coupling variables with no added constraints or variables.

ATC is the first method presented that allows for multiple optimization statements reflecting different goals. This architecture benchmarked well computationally when compared to other hierarchical MDO methods not discussed in this thesis [13]; however, all hierarchical MDO are more computationally expensive than monolithic architectures due to the local exploration of the lower level optimizers [48].

2.3.5.4 MDO with Target Values

Another hierarchical formulation is MDO with target values [28, 79]. This method involves a system level discipline, which may have its own objective function, and is optimized along with the lower discipline level objective functions. Equation (2.10) shows the formulation of MDO with target values.

System Level

$$\begin{aligned}
 & \underset{\mathbf{x}_S}{\text{minimize}} && f_S(\mathbf{x}_S) + (\|\mathbf{x}_S^{target} - \mathbf{x}_S\|_2)^2 \\
 & \text{subject to:} && \mathbf{g}_S(\mathbf{x}_S) \leq 0 \\
 & && \mathbf{g}_i(\mathbf{x}_G, \mathbf{x}_j) \leq 0
 \end{aligned} \tag{2.10}$$

Discipline Level

$$\begin{aligned}
 & \underset{\mathbf{x}_G, \mathbf{x}_j}{\text{minimize}} && f_j(\mathbf{x}_G, \mathbf{x}_j) \\
 & \text{subject to:} && \mathbf{g}_j(\mathbf{x}_G, \mathbf{x}_j) \leq 0
 \end{aligned}$$

Where f_S is the system level discipline objective function, f_j is discipline j 's objective function, \mathbf{g}_S is the vector of constraints for the top level discipline, and \mathbf{g}_j are the vectors of constraints for each lower level discipline. The system level DVs

(\mathbf{x}_S) consist of the vector of global DVs (\mathbf{x}_G) and each discipline level vector of DVs (\mathbf{x}_j). \mathbf{x}_S^{target} is the vector of all target variables defined by equation 2.11.

$$x_i^{target} = \frac{\sum_{j=1}^{N_j} x_{i,j}^{optimal} \left(f_j^{start} - f_j^{optimal} \right)^2}{\sum_{j=1}^{N_j} \left(f_j^{start} - f_j^{optimal} \right)^2} \quad (2.11)$$

Where x_i^{target} is the target value for the i th DV, $x_{i,j}^{optimal}$ is the value of DV i at the optimal location of discipline j , f_j^{start} is the value the j th discipline value at the start of the discipline optimization, $f_j^{optimal}$ is the optimal value found for discipline j , and N_j is the number of disciplines that use DV i .

In each iteration of the top level optimizer, the disciplines evaluate a local optimization where the DVs are given lower and upper bounds based on the starting value. From the discipline optimizations, the target values are calculated with equation 2.11. The target values are then used by the top level objective function. A step is taken in the DVs, and then the disciplines are locally optimized again. This process is continued until a minimum is found which minimizes the top level objective function as well as the distance between the target variables and the final DV values [28].

Like ATC, MDO with target values uses goals for the DVs; however, the addition of compatibility variables is removed in an attempt to improve efficiency [28]. Here, the target values which replace the compatibility variables are calculated using only the function values and DV locations of the disciplines objectives, so target values are dependent on the model's fidelity.

2.3.5.5 Multi-level System Optimization

Multi-level system optimization allows for weighting of each objective function while minimizing the distance to the utopia point on the Pareto Frontier [35]. Another example of a hierarchical form of MDO, the generalized form of the multi-level system

optimization is shown in equation (2.12).

System Level:

$$\begin{aligned} & \underset{\mathbf{x}_S}{\text{minimize}} && f_S(\mathbf{x}_S) = e \left(\sum_{j=1}^J w_j \left(\frac{f_j - f_j^{optimal}}{prr_j} \right)^2 \right) \\ & \text{subject to:} && \mathbf{g}_S(\mathbf{x}_S) \leq 0 \end{aligned} \quad (2.12)$$

Discipline Level:

$$\begin{aligned} & \underset{\mathbf{x}_G, \mathbf{x}_j}{\text{minimize}} && f_j(\mathbf{x}_G, \mathbf{x}_j) \\ & \text{subject to:} && \mathbf{g}_j(\mathbf{x}_G, \mathbf{x}_j) \leq 0 \end{aligned}$$

Where f_S is the system level discipline objective function, f_j is discipline j 's objective function, $f_j^{optimal}$ is the optimal value of discipline j 's objective function, \mathbf{g}_S is the vector of constraints for the top level discipline, and \mathbf{g}_j are the vectors of constraints for each lower level discipline. The system level DVs (\mathbf{x}_S) consist of all global DV (\mathbf{x}_G) and each discipline level vector of DVs (\mathbf{x}_j). The prr_j is the probable reduction range of discipline j , described in equation (2.13).

$$prr_j = \max (f_j(\mathbf{x}_{i \neq j}^*) - f_j(\mathbf{x}_j^*)) \quad (2.13)$$

The prr for discipline j value finds the maximum difference between discipline j 's optimal value ($f_j(x_j^*)$) and discipline j evaluated at all other disciplines' optimal DV locations ($f_j(x_{i \neq j}^*)$). The objective function therefore normalizes the difference between the current DV location and discipline j 's optimal value with the difference between the largest non-dominated point objective function value and the disciplines optimal value [35]. The idea of plausible reduction range will be used as part of the MDO architecture presented in chapter IV.

2.3.5.6 Optimization/MDO Notation Summary

For clarity and consistency, table 2.1 summarizes the notation that will be used throughout the remainder of this dissertation when describing optimization and MDO formulations.

Notation	Description
I	Number of Design Variables
x_i	Design Variable i
\mathbf{x}	Vector of Design Variables
J	Number of Design Disciplines
\mathbf{x}_j^*	Vector of Optimal DV Values for Discipline j
f_j	Objective Function of Discipline j
f_j^*	Optimal Objective Function Value of Discipline j
K	Number of Constraint Functions
g_k	Constraint Function k
\mathbf{g}	Vector of Constraints

Table 2.1: Summary of Optimization/MDO Notation

2.4 Chapter Summary

This chapter presents the basics of optimization and MDO. The techniques of MDO that were presented have all been successfully applied in system design. These methods were presented to show multiple ways that trade-offs have been addressed in automated system design, either by requiring consistency through the design pro-

cess with a single system optimization or breaking the system into subsystems. This literature review was presented to show that, while many methods of MDO are available to address complex system, the design method that this thesis presents provides a unique approach to automated system design. The results of all MDO methods presented are dependent on the fidelity of the models used. A human designer would be able to glean more knowledge from lower fidelity models to make more informed decision on the models available. The next chapter will present Hierarchical Fuzzy Logic Controllers, which emulate human decision-making in controlling physical systems. Using the structured methodology of handling system design taken from MDO, HFLC will be extended to the design space to create a MDO architecture that has the ability to model a human evaluation of design analysis tools.

CHAPTER III

Hierarchical Fuzzy Logic Controllers: Emulating Human Decision in Controllers

3.1 Introduction

This thesis relies heavily on fuzzy logic systems and fuzzy logic controllers. This chapter will describe general fuzzy logic theory. Fuzzy logic is a method of soft computing that allows linguistic variables to be represented in a mathematical fashion [86]. Fuzzy logic involves the use of fuzzy set theory. In classic set theory, an element must either be a member of a set, or not. Fuzzy set theory is an extension of classic set theory that allows an element to be a member of a set with varying degrees of membership.

Fuzzy Logic is important in computing because “humans employ mostly words in computing and reasoning, arriving at conclusions expressed as words from premises expressed in a natural language or having the form of mental perceptions” [88]. Computers, on the other hand, work by manipulating numbers and symbols. Fuzzy Logic provides a method to introduce “Linguistic Variables” by representing a word used in human communication as a fuzzy set. These fuzzy sets can then be treated in a mathematical framework.

When a complex engineering system is designed, communication between design

disciplines is important to find a compromise between each disciplines' goals. While MDO provides a method to structure this communication, the amount of information that is passed between disciplines is, in general, less than what would be examined by a human design team. Fuzzy logic provides a method where information can be shared between disciplines more similar to a design team's analysis. Fuzzy logic controllers, an extension of fuzzy logic, provide a method to blend discipline design space preferences in order to search for the best design compromise between multiple disciplines.

3.1.1 Fuzzy Logic Terminology

Fuzzy Set: Fuzzy Sets are extensions of classic sets. A fuzzy set has a linguistic term associated with it and allows elements to have membership in the set to varying degrees. Where classic set theory is useful in describing a binary decision, such as the position of a light switch, fuzzy sets are useful in defining sets that do not have clear boundaries.

For instance, describing a person's height as tall or not tall constitutes a fuzzy set. A "Tall" fuzzy set would have a boundary that is not crisp, meaning that the term "Tall" can imply different heights to different people, and it is this "fuzziness" that can be expressed using a fuzzy set. In contrast, modeling "Tall" as a classic set would require a crisp cutoff height, so modeling height in this way does not capture the imprecision in linguistics [87]. A detailed description of fuzzy set theory can be found in [84].

Example "Tall" and "Not Tall" fuzzy sets are displayed in figure 3.1, and figure 3.2 displays classic sets for "Tall" and "Not Tall." Where the classic sets require a person to be in only one set, fuzzy sets have a vague area where a person can be a member of both sets to varying degrees. In the classic sets, a person that is 5'11" would be considered not tall; however, in reality, to a person that is 5'4", the 5'11' person is

tall. This discretion illustrates the implicit uncertainty associated with linguistics that is missed in classic set theory. In fuzzy set theory, the person who is 5'11" would fall into both the “Tall” and “Not Tall” fuzzy sets.

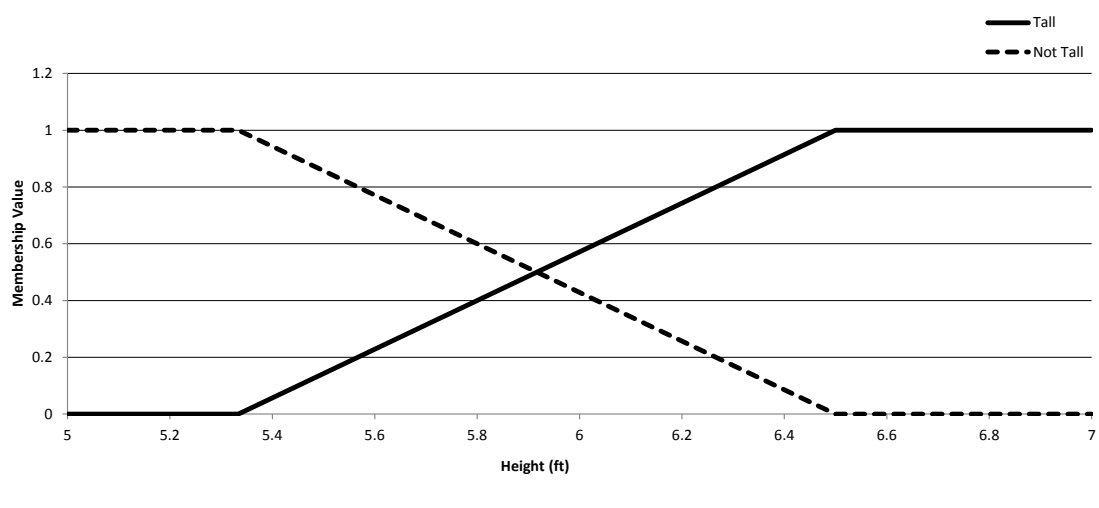


Figure 3.1: Example “Tall” and “Not Tall” Fuzzy Sets

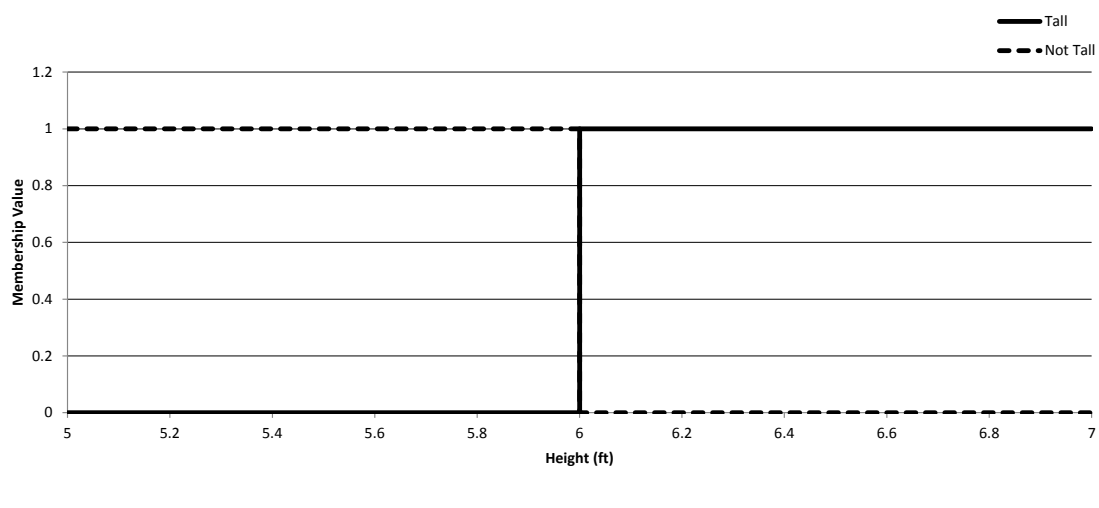


Figure 3.2: Example “Tall” and “Not Tall” Classic Sets

Fuzzy Membership Function: Any element that is a member of a fuzzy set is described by two labels, a fuzzy linguistic (in this example “Tall” or “Not Tall”), and a membership value assigned by a membership function. The boundaries of the fuzzy

sets in figure 3.1 are described by membership functions. Membership functions are typically denoted by μ . For a given crisp input value, a membership function returns that element's membership in the fuzzy set as defined by the membership function.

When looking at the “Tall” fuzzy set in figure 3.1, the associated membership function, $\mu^{Tall}(x)$, would take an input height x and return that height's membership value in the “Tall” fuzzy set. From the figure, any height below 5'3” would be assigned a zero membership value in the “Tall” fuzzy set while any value above 6'6” would be assigned a membership value of one. In between, the heights are assigned a membership value in the range $[0,1]$. As with most uses of fuzzy logic in engineering systems, for this work all membership values used for fuzzy sets are constrained to be between zero and one [10].

Fuzzy Rule: A fuzzy rule acts the same as rules in classic logic system, i.e. it maps inputs to outputs. Fuzzy rules are typically in the form of a standard “IF (antecedent) - Then (consequent)” statement. Depending on the setup of the rule, different operators from classic logic statements are implemented, such as AND and OR statements. These operators affect how the conclusions are drawn.

3.2 Fuzzy Logic Systems

An approach is needed to include fuzzy sets and rules in the design method. Fuzzy Logic Systems (FLS) provide a mechanism that accepts crisp inputs, applies fuzzy rules, and calculates crisp outputs. FLSs allow for the use of linguistic terms in a computing logical system and the inherent uncertainty of linguistic inputs [50, 81]. Figure 3.3 displays the flow chart of a standard FLS, which consists of four steps: fuzzification, application of a fuzzy rule bank, fuzzy inference, and defuzzification.

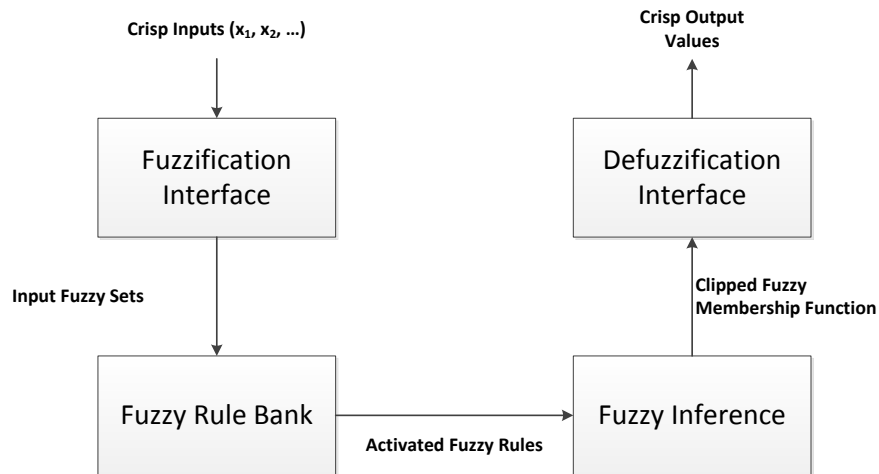


Figure 3.3: Fuzzy Logic System Flowchart

Inputs to the FLS are fuzzified by assigning each input to fuzzy sets with corresponding membership values. The fuzzy sets and membership values that are activated by the inputs are then passed to the fuzzy rule bank. The fuzzy rule bank contains rules which are used to map the linguistics of multiple input variables onto the linguistics of the desired output. The fuzzy inference then receives all rules that were activated in the fuzzy rule bank. This step applies logical methods to the multiple fuzzy rules that are activated. The outputs from the fuzzy inference are then sent to the defuzzification interface that produces a single crisp output from the FLS.

3.2.1 Fuzzy Logic System Example

To best describe a FLS, an example relating a person's heart disease risk to their height and weight is presented. The example is a modified version of the Heart Disease Risk FLS presented in Dr. David J. Singer's Ph.D dissertation [72]. This FLS will be used to calculate the heart disease risk of a person with a height of 63 inches and a weight of 135 pounds. This example will help in understanding fuzzy sets, fuzzy logic, and FLSs.

Prior to evaluating a FLS, the input and output fuzzy sets must be defined. For this example, input fuzzy sets must be created for the two input variables (height and weight). The height can be assigned to three fuzzy sets, with associated linguistics of “Short,” “Average,” and “Tall.” The input fuzzy sets for height are presented in figure 3.4. The weight can also be assigned to three fuzzy sets, “Light,” “Medium,” and “Heavy.” The weight input fuzzy sets are presented in figure 3.5

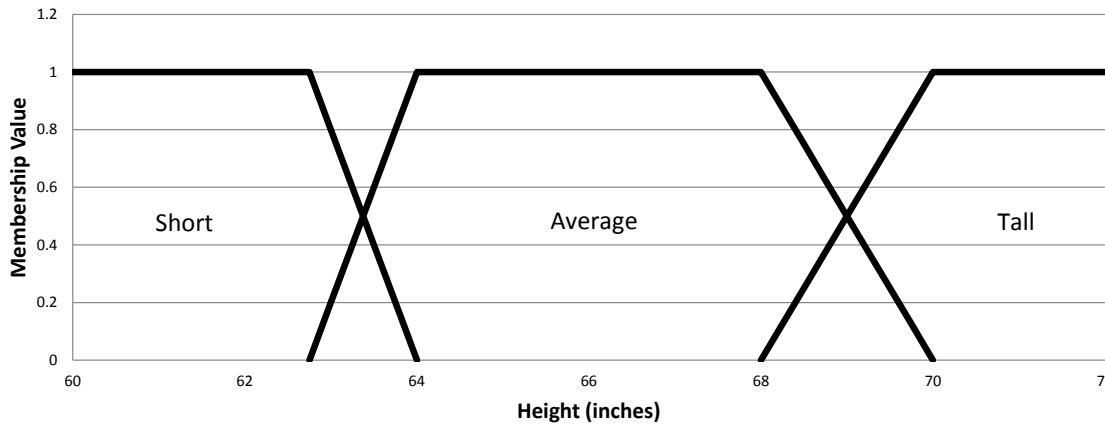


Figure 3.4: Example Height Fuzzy Sets

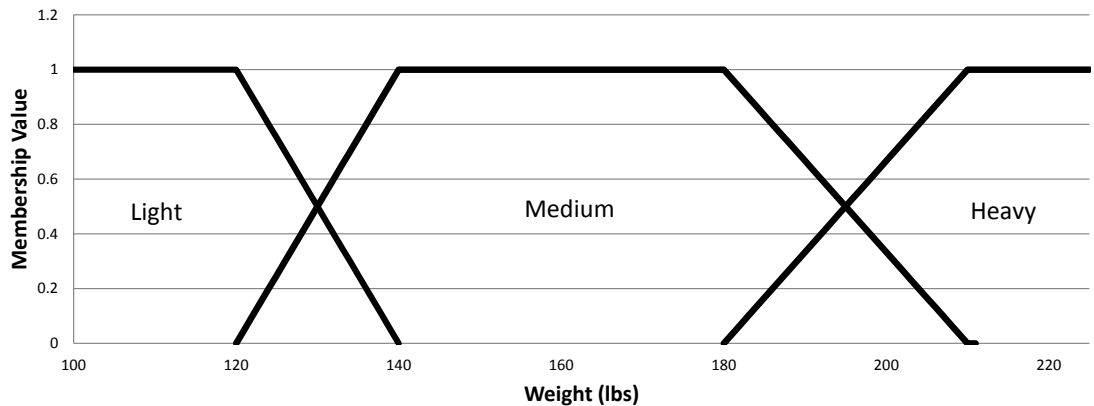


Figure 3.5: Example Weight Fuzzy Sets

The output fuzzy sets also need to be defined prior to evaluation of the FLS. The output fuzzy sets generate the risk of heart disease for a person. There are four levels of heart disease risk in this example, “Low,” “Average,” “Moderate,” and “High.”

The output fuzzy sets are displayed in figure 3.6.

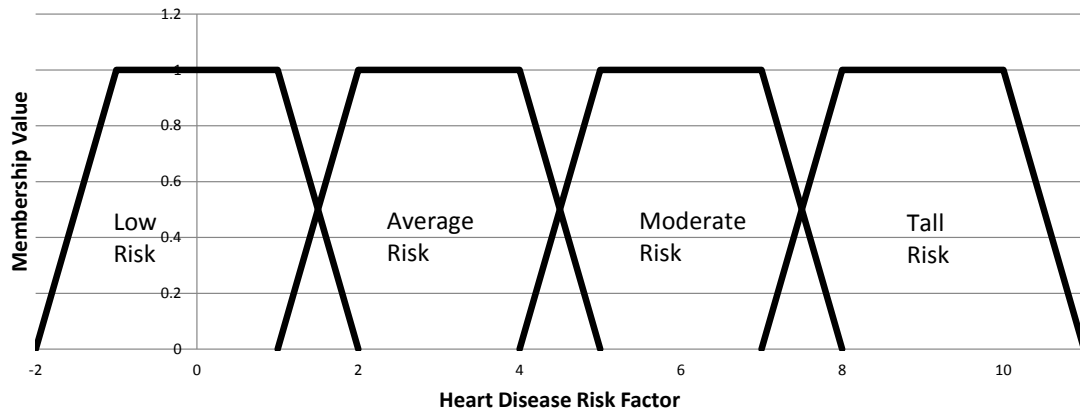


Figure 3.6: Example Heart Disease Risk Fuzzy Sets

3.2.1.1 Fuzzifier

The goal of this example is to determine a person’s risk of heart disease based on their height and weight. The subject’s height and weight are entered into the fuzzifier, which contains the six inputs from the fuzzy sets. For the hypothetical person with an input height of 63 inches, two fuzzy sets are activated. As shown in figure 3.7, the input height is a member of the “Short” fuzzy set with a membership value of $\mu^{Short}(63) = 0.8$, and a member of the “Average” fuzzy set with a membership value of $\mu^{Average}(63) = 0.2$. The same method is used for the weight input of 135 pounds, where the resulting fuzzy sets are “Light” with a membership value of $\mu^{Light}(135) = 0.25$, and “Medium” with a membership value of $\mu^{Medium}(135) = 0.75$.

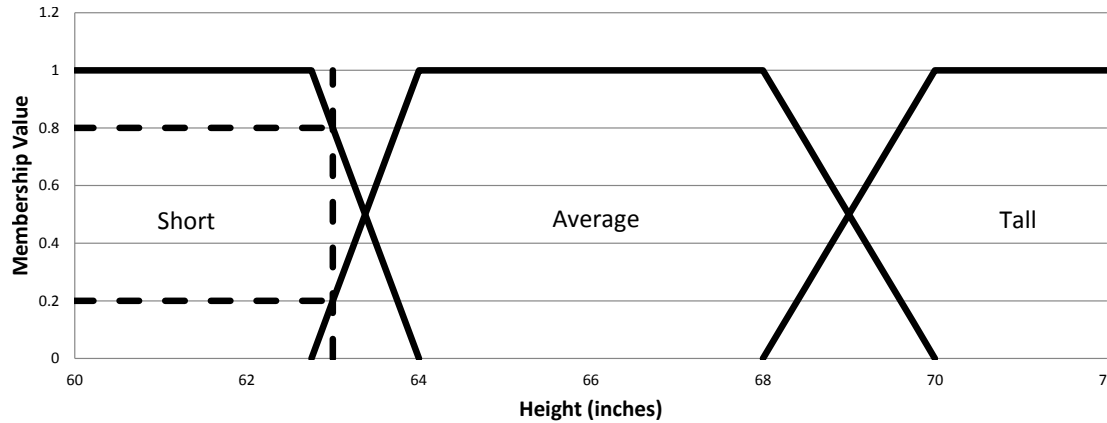


Figure 3.7: Example Height Input

3.2.1.2 Fuzzy Rule Bank

The activated fuzzy sets are evaluated by the fuzzy rule bank to map the input linguistics to the corresponding output linguistics. As noted in section 3.1.1, the rules in the fuzzy rule bank are in the form of IF-THEN rules. For example, “*IF Height is Short and Weight is Light, THEN Heart Disease Risk is Low Risk*”. The fuzzy rule bank of a FLS must have rules which map all possible inputs to an output.

A decision matrix is then used to display all of the rules in the fuzzy rule bank for the heart disease risk example (figure 3.8). From the four activated fuzzy sets (Short Height, Average Height, Light Weight, and Medium Weight), the decision matrix finds the activated rules. In figure 3.8, the rules activated by the inputs are highlighted in bold.

		Weight		
		Light	Medium	Heavy
Height	Short	Low Risk	Moderate Risk	High Risk
	Average	Low Risk	Average Risk	High Risk
	Tall	Low Risk	Low Risk	Moderate Risk

Figure 3.8: Heart Disease Decision Matrix

3.2.1.3 Fuzzy Inference

The fuzzy rule bank maps the input linguistics to output linguistics. From this point, a method assigns an activation level to the output fuzzy sets. Different methods of selecting an activation level and modifying the output fuzzy sets are used depending on the set-up of the FLS. An overview of methods can be found in [84].

The minimum inference method is typically applied when an “AND” operator is in the antecedent of the fuzzy rule. In this method, the minimum membership value from the input fuzzy sets is used as the activation level for the output fuzzy set, as presented in equation (3.1) where μ_{out} is the activation level for the output fuzzy set and μ_{in_k} is the membership value from input fuzzy set k .

$$\mu_{out} = \min(\mu_{in_1}, \dots, \mu_{in_k}) \quad (3.1)$$

The results of the minimum inference method for the example inputs are displayed in equations (3.2)-(3.5).

$$\min(\mu^{Short} = 0.8, \mu^{Light} = 0.25) \implies \mu^{LowRisk} = 0.25 \quad (3.2)$$

$$\min(\mu^{Short} = 0.8, \mu^{Medium} = 0.75) \implies \mu^{ModerateRisk} = 0.75 \quad (3.3)$$

$$\min(\mu^{Average} = 0.2, \mu^{Light} = 0.25) \implies \mu^{LowRisk} = 0.2 \quad (3.4)$$

$$\min(\mu^{Average} = 0.2, \mu^{Medium} = 0.75) \implies \mu^{AverageRisk} = 0.2 \quad (3.5)$$

A fuzzy implication method is used to modify the fuzzy sets corresponding to the output linguistics. For the remainder of this work, the trimming method is used. To illustrate the trimming method of fuzzy implication, figures 3.9 and 3.10 demonstrates the modification of the “Low Risk” fuzzy set. Using the membership value found in equation (3.2), the original “Low Risk” fuzzy set is trimmed to give the result of the “Short and Light” fuzzy rule. Figure 3.9 displays the original “Low Risk” membership function along with the trimming line found using equation (3.2). This line trims the “Low Risk” membership function to give the final output for the “Short and Light” activated rule displayed in figure 3.10.

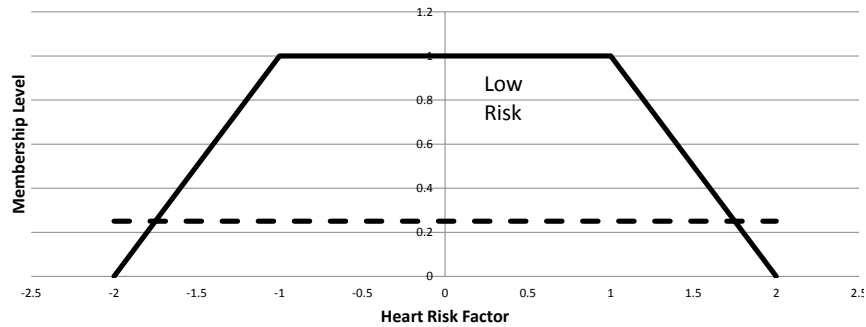


Figure 3.9: Low Heart Disease Risk Membership Function with Trim Value

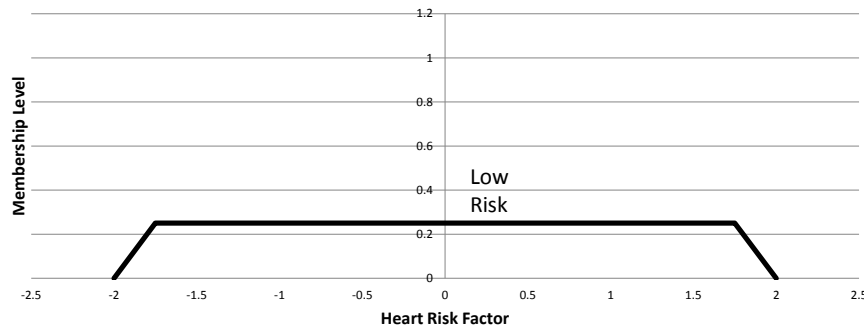


Figure 3.10: Low Heart Disease Risk Membership Function After Trim

The process is repeated for all activated rules. The four output fuzzy sets are

displayed in figures 3.11 - 3.14. These figures represents the output from the fuzzy inference module of the FLS. The four fuzzy sets created by the fuzzy inference are passed to the defuzzifier, which then outputs a single crisp heart disease risk value for the given height and weight.

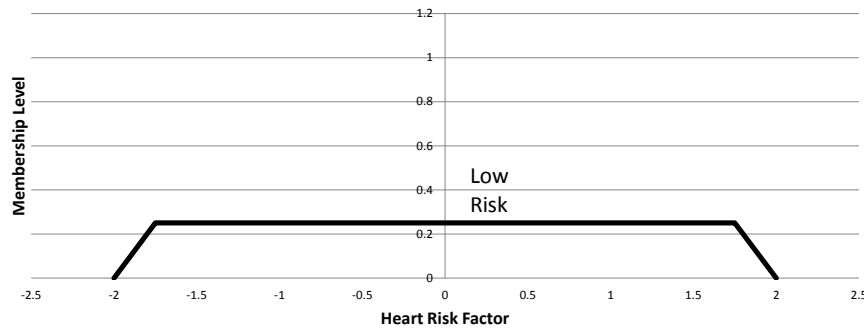


Figure 3.11: Activated Rule #1 Fuzzy Set

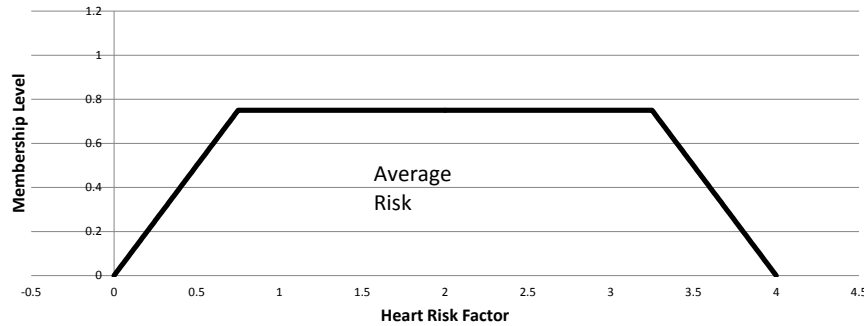


Figure 3.12: Activated Rule #2 Fuzzy Set

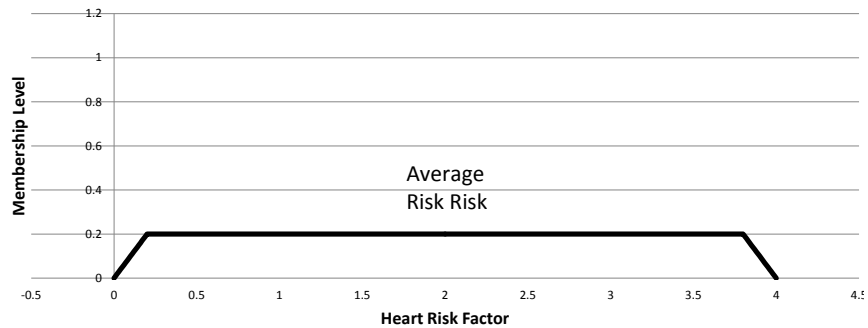


Figure 3.13: Activated Rule #3 Fuzzy Set

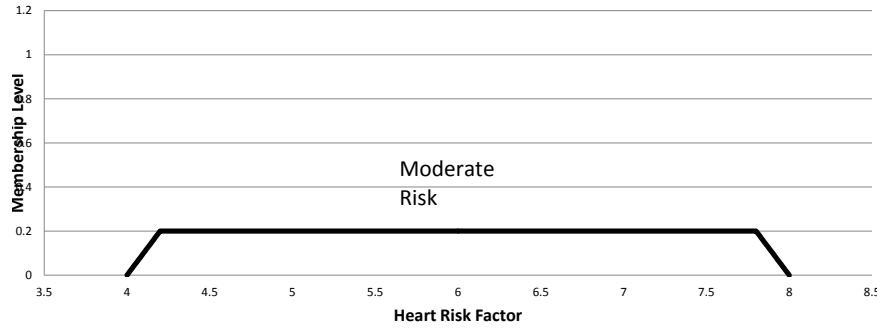


Figure 3.14: Activated Rule #4 Fuzzy Set

3.2.1.4 Defuzzification

As stated previously, depending on the problem being addressed, multiple methods for defuzzification exist [84]. For the purposes of this dissertation, the centroid defuzzification method is used. This method finds the defuzzified output by considering the area and centroid of each of the fuzzy sets input to the defuzzifier. Equation (3.6) shows the centroid defuzzification method, where I is the total number of output fuzzy sets, A_i is the area of output fuzzy set i , and x_i is the location of the centroid of output fuzzy set i .

$$\text{defuzzified output} = \frac{\sum_{i=1}^I A_i * x_i}{\sum_{i=1}^I A_i} \quad (3.6)$$

For the input height of 63 inches and input weight of 135 lbs, the values used in equation (3.6) are displayed in table 3.1. When entered into the centroid defuzzification equation, the resulting heart disease risk is approximately 2.44. When this value is compared to the original output fuzzy sets (figure 3.6), the crisp output value corresponds to an “Average Risk” of heart disease, as presented in figure 3.15.

Rule #	A_i	x_i
1	0.9375	0
2	2.56	3
3	1.14	0.76
4	1.14	0.76

Table 3.1: Example FLS Defuzzification Input

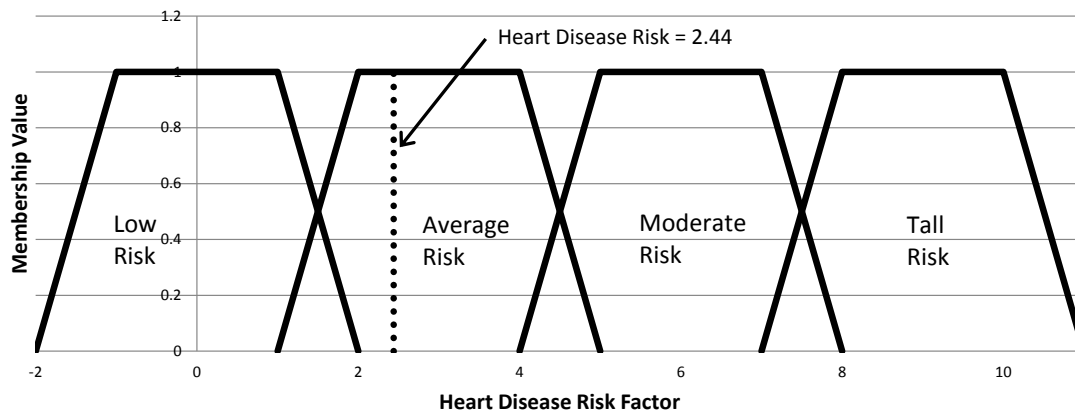


Figure 3.15: Example Defuzzified Output

The short example presented previously demonstrates how FLSs work, and how they can be used to represent linguistic uncertainty in a logical system. The next section will explain how a typical FLS is extended in a set-based design environment.

3.3 Fuzzy Logic Systems in Set-Based Design

The example presented demonstrates how a typical FLS operates. For this dissertation, a modified version of FLSs is used based on the work done in Set-Based Design (SBD). During a SBD analysis, sets of Design Variables (DV) are evaluated instead of individual points in a traditional point based design [73]. FLSs were utilized by Singer [72] and Gray [20] to facilitate communication between different design discipline agents. This process will be adapted for use in the MDO optimizer developed in chapter IV.

A flow chart for the modified FLS is displayed in figure 3.16. In the FLSs used in SBD, the fuzzifier does not receive a single crisp input, but instead is given an input membership function ($\mu_n(x)$) with the preference level over a DV set being analyzed. This input allows the preferences of areas of the design space to be evaluated instead of discrete points. Using the input membership functions, the fuzzifier sweeps through the entire DV range and passes the activated fuzzy sets and their associated membership values to the rule bank. The rest of the modified FLS system acts as a traditional FLS, but evaluates each DV location in the current set.

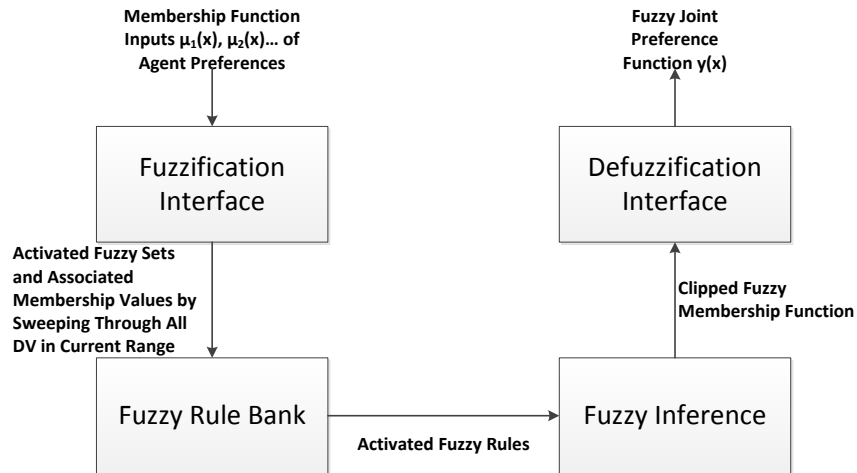


Figure 3.16: Modified Fuzzy Logic System Flowchart [72]

Figure 3.17 displays how the modified FLS transforms input preferences from different design agents and combines them into a single output preference. In the figure, the DV preferences of two different design agents are shown to the left and above a fuzzy rule matrix. The FLS sweeps through the DVs and assigns each location a Joint Output Preference based on the two agents input preferences with regard to the fuzzy rule matrix. In SBD practices, the lead designer uses this Joint Output Preference plot to evaluate where to place new upper and lower bounds on the DVs as the design progresses toward convergence.

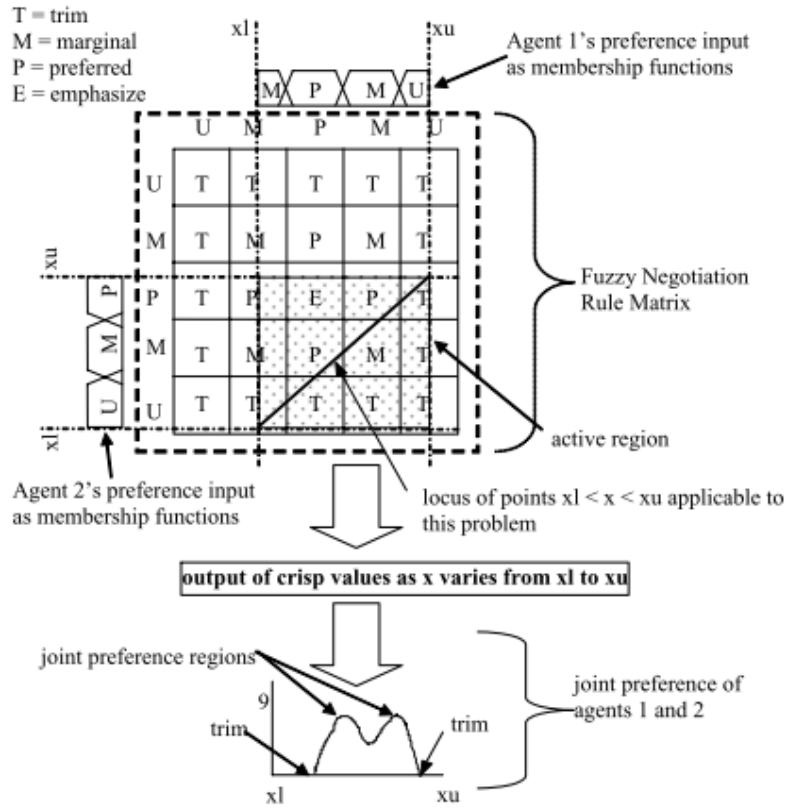


Figure 3.17: Transformation of SBD input preferences to Joint Preference Curve [72]

3.4 Fuzzy Control Engineering

The use of FLSs allow for the inclusion of linguistic uncertainty in a system which requires crisp inputs and outputs. By including FLSs in the controllers for physical systems, engineers emulate the decision of a human operator in an automated control process [42, 43]. The basic flow chart of a fuzzy logic controller is presented in figure 3.18.

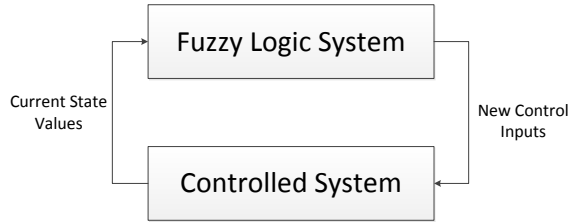


Figure 3.18: Fuzzy Logic Controller Flowchart

A simplified example of a fuzzy logic controller is a car’s cruise control, which takes into account the current speed of the car. The “current state value” would be the speed of the car, which would be entered into the controller’s FLS. The FLS would be evaluated in the same manner as described in the heart rate example, based on the input speed. The controller’s FLS would output a new throttle position, the “new control input,” which would be used to adjust the speed of the car to keep it at the target speed.

Due to the limitations of many controllers, Zadeh initially identified the need for fuzziness in controllers in 1972 [87]. Since then, numerous applications of fuzzy logic controllers have been successfully developed, demonstrating the “effective utilization of fuzzy control in the context of complex ill-defined processes that can be controlled by a skilled human operator without the knowledge of their underlying dynamics” [42]. Examples include the automation of train control [83, 70], oil drilling [52, 23], automobile control [51, 69, 66], and aerospace trajectory control [6, 82]. These applications show the ability of fuzzy controllers to incorporate human decisions in control of systems. In the design method presented in chapter IV, the structure of fuzzy logic controllers is extended to emulate human decisions in the design of complex systems.

3.4.1 Hierarchical Fuzzy Logic Controller

The MDO algorithm that will be described in the following chapter requires a hierarchical architecture. Hierarchical Fuzzy Logic Controllers (HFLC), an extension of

standard fuzzy logic controllers, allow for inputs to a system controller from multiple feedback sensors. HFLCs were originally developed for the control of gas turbine engines [7], and have since been used in other applications, such as autonomous robots [25], steam generation [67], and control of a milling process [62]. A flow chart of a HFLC is displayed in figure 3.19.

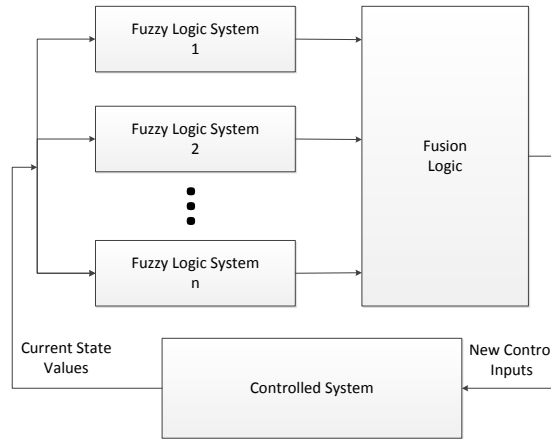


Figure 3.19: Hierarchical Fuzzy Logic Controller Flowchart

Extending the car cruise control example from section 3.4 provides a simple example of a HFLC. Instead of having a single sensor which reads the current state of the car, a HFLC would have multiple sensors. Example sensors include the speed, the current acceleration, and the current angle of the road that car is traveling on. Each of these sensors would have its own FLS system to transform the sensor output into a new throttle position. Each of the throttle positions would then be input to the fusion logic of the HFLC. The fusion logic contains methods for combining the different inputs from the FLSs. The output from the fusion logic would be a new throttle position for the car to maintain the current speed of the vehicle based on the information received from the sensors. HFLC have demonstrated better results than other types of controllers in emulating how a experienced operator would control a system when multiple inputs are considered [68].

Multiple methods of fusion logic exist, with examples provided in [74, 16]. For

this work, a minimum preference level method of fusion logic is used [8]. This method combines the outputs of the different sensors by taking the minimum output value from all sensors and applying these values as the new control inputs to the system. This method gives control of the system to the individual controller that is least comfortable at the current state values. Equation (3.7) presented the equation used by the fusion logic to blend the outputs of the individual controllers. Where $F_S(\mathbf{x})$ is the system level control input for a given set of variables \mathbf{x} , $F_j(\mathbf{x})$ is each lower level controller input, and \wedge is the minimum operator which selects the minimum of all controller outputs.

$$F_S(\mathbf{x}) = \left(\wedge_{j=1}^J F_j(\mathbf{x}) \right) \quad (3.7)$$

While fuzzy logic controllers and HFCLC have been successful in many applications in emulating human decision making, this process has not been extended to design space evaluation. However, these ideas are analogous to the design of a multidisciplinary system. Each discipline design space can be represented as a single FLS, which a human designer would evaluate. All of the information in the discipline design spaces must be combined for consideration in the final system design. This combination of information is evaluated in the fusion logic in the HFCLC analogy. This method of extending the tools of HFCLCs to the design space allows for a structured approach to design, while incorporating human decision making in an automated design process. An example flowchart of how a HFCLC would act in the engineering design space is shown in figure 3.20.

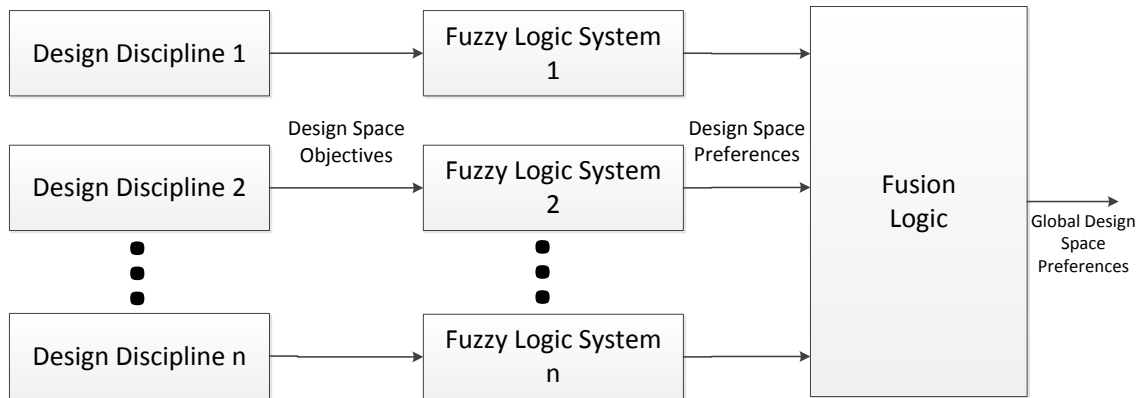


Figure 3.20: HFLC in Engineering Design Space

3.5 Chapter Summary

This chapter provides an introduction to fuzzy logic systems and fuzzy logic controllers. FLSs provide a method for interpreting data output from a system in a more human-like way than found in most computer interpretations of data. The methods used in FLS extend to controlling physical systems with fuzzy logic controllers and hierarchical fuzzy logic controllers. These controllers have shown to emulate better human decision making in the control of systems than typical non-fuzzy controllers. The theory presented in this chapter of emulating human decision making in system control will be extended to the decision making process in system design.

CHAPTER IV

Multidisciplinary Design Optimization Using Hierarchical Fuzzy Logic Controllers

4.1 Introduction

As described in chapter II, engineers have successfully used different formulations of MDO in many applications. However, these algorithms depend on the fidelity of the models used to represent the system, thereby removing the human designer from the creation and evaluation of design alternatives. In early stage ship design, most decisions are made based on low fidelity models. Many ship designs are successful because of an experienced designer's understanding the design intent of a ship and interpreting these models from that experience.

The following chapter will outline in detail the MDO method developed in this thesis. Chapter V will present a ship design case study that expands the algorithm to the evaluation of a more in depth design problem.

4.2 Design Method Overview

The method presented will allow for human design intent to be incorporated into the decision making process of the MDO algorithm. Beginning at a discipline level design space, as presented in figure 4.1, different preference levels are assigned to the

design space based on multiple human design intent models, which change depending on the problem in question. Using a fuzzy logic system, the areas of preferences from the different human intent models are then combined to give the discipline's preference level for each set of design variable (DV) values.

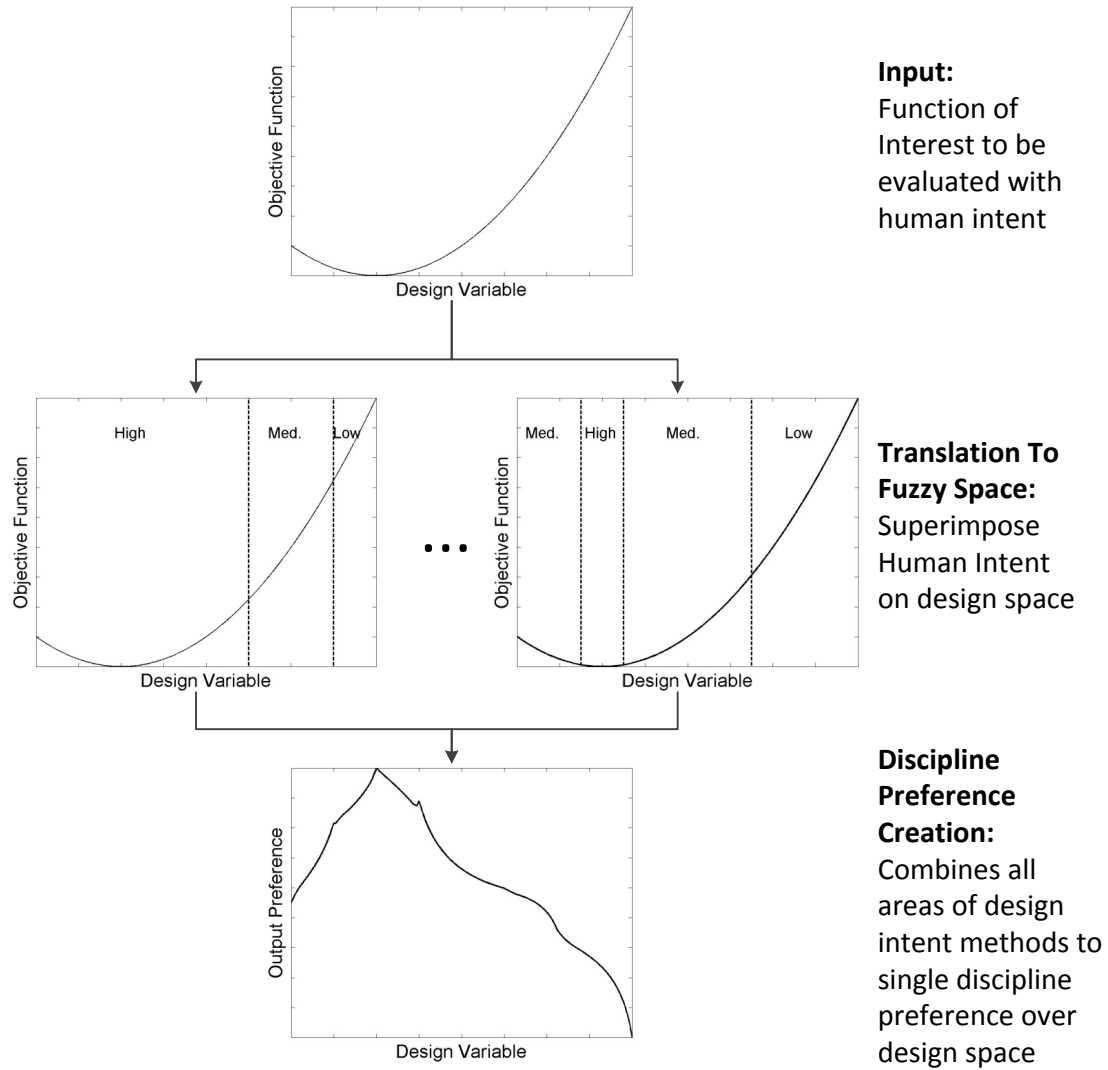


Figure 4.1: Discipline Preference Assignment Flowchart

This process of discipline evaluation is repeated for all disciplines. The resulting preferences at each DV location must then be combined to output one global set of preferences for the DVs. Figure 4.2 demonstrates multiple discipline preference curves combined into a single preference curve for the entire system. This system preference

curve identifies the most preferred DV location as well as other areas on which the designer should focus on as the design moves into later stages of development.

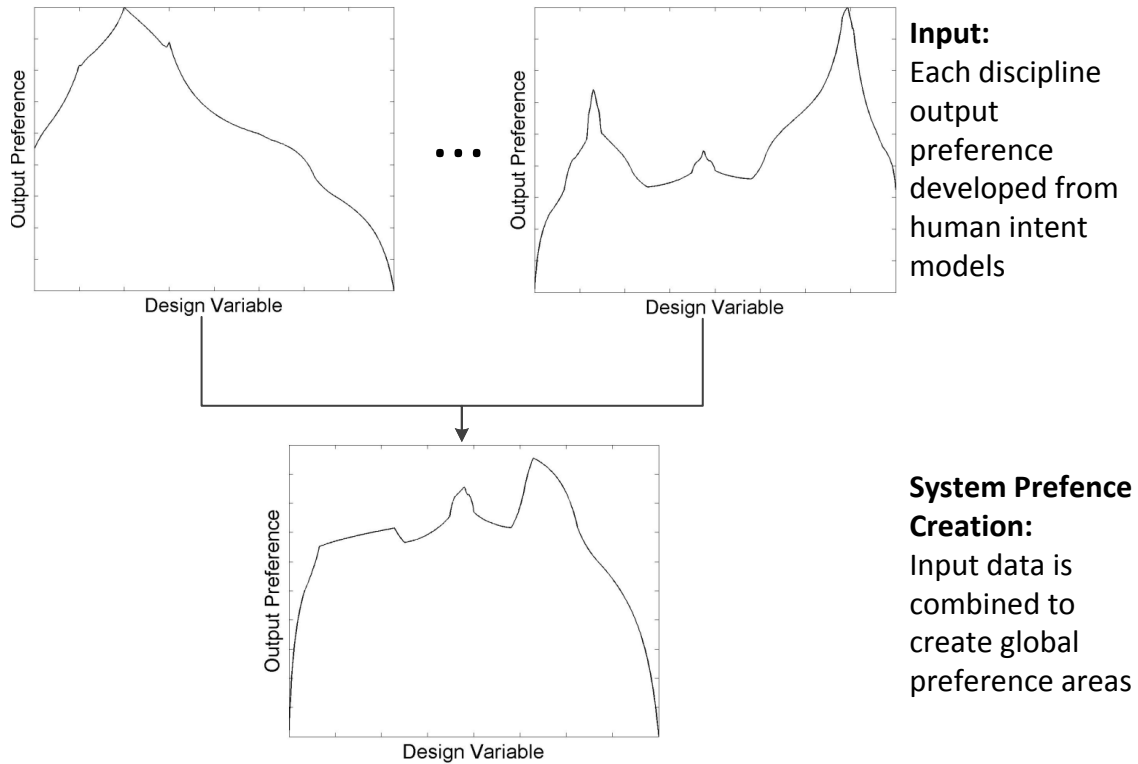


Figure 4.2: System Level Discipline Combination

The next sections will describe the design method framework in detail. While developing the framework for the MDO method, an example problem will be used for explanation.

4.3 Design Intent Membership Function Creation

As described in section 3.3, the design methodology will use modified fuzzy logic systems (FLS) based on work in set-based design (SBD). For the SBD approach to FLS, the system requires input membership functions. In typical SBD, these input membership functions are created by human design agents. Unlike the SBD approach, the input membership functions are created automatically for the MDO method. The design of the input membership functions is one area where the idea of designer intent

is integrated into the MDO algorithm. The input membership functions are created using on the intent of the discipline being analyzed and the understanding of the analysis tools used.

A mathematic example is presented where minimizing the three disciplines while considering trade-offs between disciplines is the designer intent. The membership functions give a mathematical discipline preference based on three metrics of design evaluation: the objective function value relative to the function maximum and minimum, the closeness of the ℓ_2 norm of the objective function gradient to zero, and the proximity of the design being evaluated to the discipline objective’s global minimum. The membership functions of these preference fuzzy sets are presented in sections 4.3.1-4.3.3. In the ship design case study presented in chapter V, more specialized membership functions will be developed for each individual discipline.

4.3.1 Function Value Preference

The first set of membership functions for the general mathematical disciplines focuses on the desire to have the lowest discipline objective function value. This set of functions assigns the DV values to preference fuzzy sets. The preference levels are assigned based on the objective function value at the current DV location relative to the discipline’s global minimum and the discipline’s “max” value. For this formulation, the discipline max is designated using the probable reduction range (PRR) developed by Kim [35]. The max is defined as the maximum objective function value at the DV location corresponding to the other disciplines’ global minimums. This maximum is used because it lowers the preferences on solutions that are “dominated” by other solutions. This leads to results that are located on the Pareto front of the feasible designs. The equation for the discipline maximum in this formulation is shown in (4.1), where: f_j^{max} is discipline j ’s maximum and $f_j(\mathbf{x}_k^*)$ is discipline j ’s objective function being evaluated at the optimum \mathbf{x}^* values for each other k th discipline. The

membership functions are also dependent on the midpoint of the objective function calculated using equation (4.2), where f_j^{mid} is the midpoint of the objective function values for discipline j , f_j^{max} is the function maximum, and f_j^* is the minimum value of the objective function.

$$f_j^{max} = \max(f_j(\mathbf{x}_k^*)) \quad \forall j \neq k \quad (4.1)$$

$$f_j^{mid} = \frac{1}{2}(f_j^{max} - f_j^*) + f_j^* \quad (4.2)$$

Three membership functions map each DV location to a preference fuzzy set. For the objective function value preference, all membership functions will be denoted with a subscript 1. In equation (4.3), μ_1^{high} assigns membership values to the “High Objective Function Preference” fuzzy set. Objective function values between the function minimum and the function midpoint have a membership value between zero and one. Any value greater than the midpoint of the objective function is not a member of the “High Objective Function Preference” fuzzy set, and therefore is given a membership value of zero.

In equation (4.4), μ_1^{med} assigns membership values to the “Medium Objective Function Preference” fuzzy set. The DV location of the objective function midpoint is assigned a membership value of one, and the values decrease linearly as the objective function value moves toward the maximum or minimum.

Finally, in equation (4.5), μ_1^{low} gives any DV location with a value greater than the midpoint of objective function a positive membership function value in the “Low Objective Function Preference” fuzzy set. Any objective function value greater than the function maximum, as described previously, is assigned a membership value of one in the “Low Objective Function Preference” fuzzy set. In equations (4.3) - (4.5), f_j^* is the minimum value of discipline j 's objective function, f_j^{max} is discipline j 's

objective function maximum, f_j^{mid} is discipline j 's objective function midpoint, and $f_j(\mathbf{x})$ is the objective function value being examined. A graphical representation of the objective function fuzzy sets are displayed in figure 4.3.

$$\mu_1^{high}(f_j(\mathbf{x})) = \begin{cases} \frac{f_j^* - f_j(\mathbf{x})}{\frac{1}{2}(f_j^{max} - f_j^*)} + 1 & f_j(\mathbf{x}) \leq f_j^{mid} \\ 0 & f_j(\mathbf{x}) > f_j^{mid} \end{cases} \quad (4.3)$$

$$\mu_1^{med}(f_j(\mathbf{x})) = \begin{cases} \frac{f_j(\mathbf{x}) - f_j^*}{\frac{1}{2}(f_j^{max} - f_j^*)} & f_j(\mathbf{x}) \leq f_j^{mid} \\ \frac{f_j^{max} - f_j(\mathbf{x})}{\frac{1}{2}(f_j^{max} - f_j^*)} & f_j(\mathbf{x}) > f_j^{mid} \end{cases} \quad (4.4)$$

$$\mu_1^{low}(f_j(\mathbf{x})) = \begin{cases} 0 & f_j(\mathbf{x}) \leq f_j^{mid} \\ \frac{f_j(\mathbf{x}) - f_j^{max}}{\frac{1}{2}(f_j^{max} - f_j^*)} + 1 & f_j^{max} > f_j(\mathbf{x}) > f_j^{mid} \\ 1 & f_j(\mathbf{x}) \geq f_j^{max} \end{cases} \quad (4.5)$$

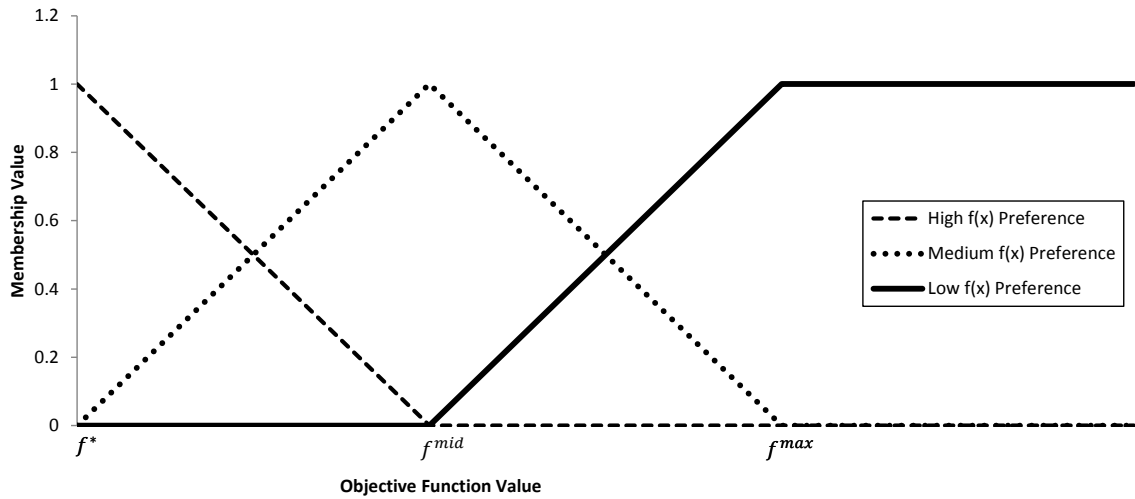


Figure 4.3: Objective Function Value Preference Fuzzy Sets

4.3.2 Gradient ℓ_2 Norm Preference

The next set of membership functions for the mathematical disciplines gives higher preferences to areas with low gradient ℓ_2 norms. At each DV location, the ℓ_2 gradient norm ($\|\nabla_{\mathbf{x}}f\|_2$) is calculated using equation (4.6) where: I is the number of DVs and $\left.\frac{\partial f}{\partial x_i}\right|_{\mathbf{x}}$ is the partial derivative with respect to DV i evaluated at \mathbf{x} . $\|\nabla_{\mathbf{x}}f\|_2$ gives no preference to local maximums, local minimums, or saddle points; however, the author believes that while analyzing trade-offs, any point where variable changes have little effect on the objective function value would be preferable to a solution with the same objective value but a steep gradient.

$$\|\nabla_{\mathbf{x}}f\|_2 = \sqrt{\sum_{i=1}^I \left(\left.\frac{\partial f}{\partial x_i}\right|_{\mathbf{x}}\right)^2} \quad (4.6)$$

Again, like the objective function preference equations, three equations are used to assign DV locations to preference level fuzzy sets. For the gradient norm membership functions, a subscript 2 will be assigned. In equation (4.7), μ_2^{high} assigns membership value to the “High Gradient Norm Preference” fuzzy set if $\|\nabla_{\mathbf{x}}f\|_2$ is between zero and one (from equation (4.6) it can be observed that $\|\nabla_{\mathbf{x}}f\|_2 \geq 0$.) The value of $\|\nabla_{\mathbf{x}}f\|_2 = 1$ was chosen as a transition point between gradient norm fuzzy sets, this value is merely for demonstration purposes and could be adjusted depending on the problem being addressed. In equation (4.8), μ_2^{med} assigns membership values for the “Medium Gradient Norm Preference” fuzzy set, with a maximum membership value of one at the location of the transition point. The “Low Gradient Norm Preference” fuzzy set is defined by equation (4.9), where any gradient greater than the transition point is assigned a membership value (μ_2^{low}) between zero and one. In equations (4.7) - (4.9) $\|\nabla_{\mathbf{x}}f\|_2$ is the ℓ_2 norm of the gradient evaluated at the current DV values and $\|\nabla_{\mathbf{x}}f\|_2^{max}$ is the maximum gradient norm found over the range of DVs evaluated. Figure 4.4 represents the gradient norm preference fuzzy sets.

$$\mu_2^{high}(\|\nabla_{\mathbf{x}}f\|_2) = \begin{cases} -\|\nabla_{\mathbf{x}}f\|_2 + 1 & \|\nabla_{\mathbf{x}}f\|_2 \leq 1 \\ 0 & \|\nabla_{\mathbf{x}}f\|_2 > 1 \end{cases} \quad (4.7)$$

$$\mu_2^{med}(\|\nabla_{\mathbf{x}}f\|_2) = \begin{cases} \|\nabla_{\mathbf{x}}f\|_2 & \|\nabla_{\mathbf{x}}f\|_2 \leq 1 \\ \frac{1-\|\nabla_{\mathbf{x}}f\|_2}{\|\nabla_{\mathbf{x}}f\|_2^{max}-1} + 1 & \|\nabla_{\mathbf{x}}f\|_2 > 1 \end{cases} \quad (4.8)$$

$$\mu_2^{low}(\|\nabla_{\mathbf{x}}f\|_2) = \begin{cases} 0 & \|\nabla_{\mathbf{x}}f\|_2 \leq 1 \\ \frac{\|\nabla_{\mathbf{x}}f\|_2-1}{\|\nabla_{\mathbf{x}}f\|_2^{max}-1} & \|\nabla_{\mathbf{x}}f\|_2 > 1 \end{cases} \quad (4.9)$$

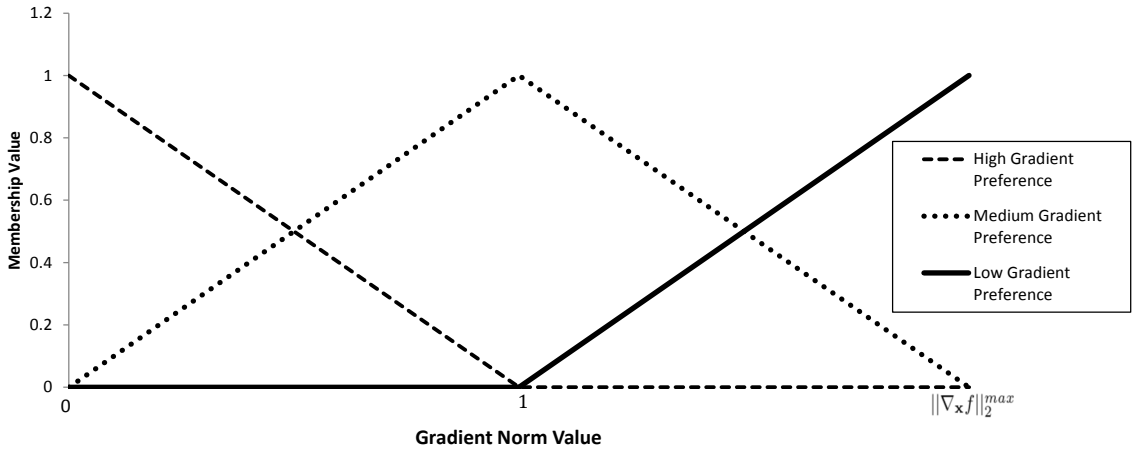


Figure 4.4: Gradient Norm Value Fuzzy Sets

4.3.3 Distance From Global Minimum Preference

The final set of membership functions used for mathematic disciplines give preferences based on the proximity of the DV location to the discipline's objective function global minimum. Two membership functions are used to give each DV location a high or medium preference with the highest preference located at the objective function minimum and the lowest preference at the furthest distance from optimum. In the distance from minimum preference design intent model, no low membership function

is used because locations are not excluded due to their proximity to the optimum. The distance from minimum membership functions are all denoted with a subscript of 3. In equation (4.10), μ_3^{high} gives the membership value for the “High Distance From Optimum Preference” fuzzy set. This equation gives a membership value of one at the optimal location and a membership value of zero at the furthest distance from the optimum.

Equation (4.11) is the equation for μ_3^{med} , which defines the “Medium Distance From Optimum Preference” fuzzy set, which has a membership value of zero at the optimum location and a membership value of one at the greatest distance from the optimum. In equations (4.10) and (4.11), I is the total number of DVs, x_i is the current value of DV i , x_i^* is the optimal value for DV i , x_i^{ub} is the upper bound of DV i , and x_i^{lb} is the lower bound of DV i . The distance from minimum fuzzy sets are displayed graphically in figure 4.5

$$\mu_3^{high}(\mathbf{x}) = 1 - \frac{\sum_{i=1}^I (x_i - x_i^*)^2}{\sum_{i=1}^I \max((x_i^* - x_i^{ub})^2, (x_i^* - x_i^{lb})^2)} \quad (4.10)$$

$$\mu_3^{med}(\mathbf{x}) = \frac{\sum_{i=1}^I (x_i - x_i^*)^2}{\sum_{i=1}^I \max((x_i^* - x_i^{ub})^2, (x_i^* - x_i^{lb})^2)} \quad (4.11)$$

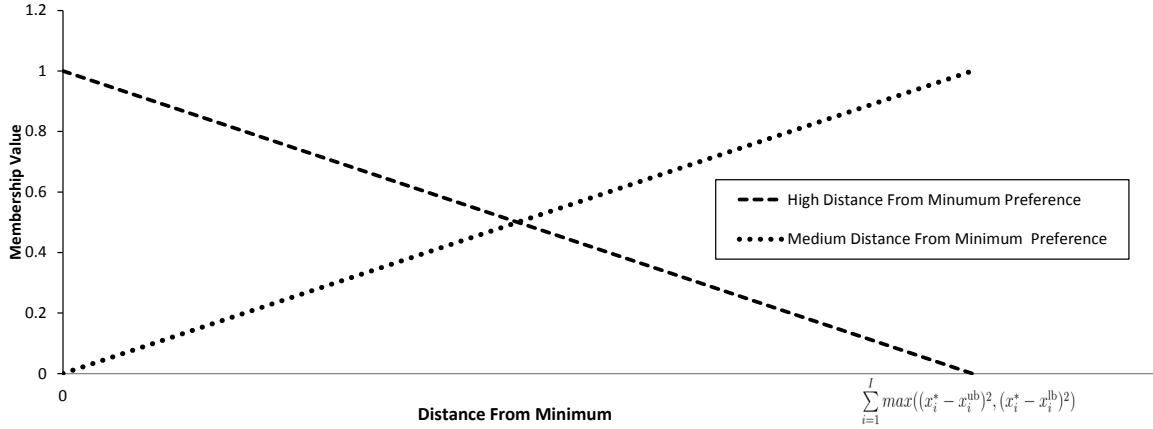


Figure 4.5: Distance from Minimum Fuzzy Sets

The three groups of input fuzzy sets described previously were developed to demonstrate how using Hierarchical Fuzzy Logic Controllers (HFCLC) to input designer intent into a MDO formulation can be adapted for a generalized mathematical problem. In the following sections, three example disciplines will describe the MDO method.

A flow chart of the MDO algorithm using HFCLC is in figure 4.6. The algorithm begins at level (a), the discipline analyzer which evaluates the discipline objective function, objective function gradient, and any constraints. The information is passed to the discipline fuzzy logic system, level (b), where the “designer intent” is built into the algorithm. The discipline FLS develops preferences for each DV location based on the designer intent defined prior to executing the algorithm. These DV preferences are developed for each discipline separately, and the outputs of the discipline FLSs are passed to level (c), the fusion logic. The fusion logic combines the output from each discipline’s FLS into one system level preference for each DV location, which is passed to the system level decision-maker, level (d). The system level decision-maker chooses the point with the highest combined preference as the system optimum.

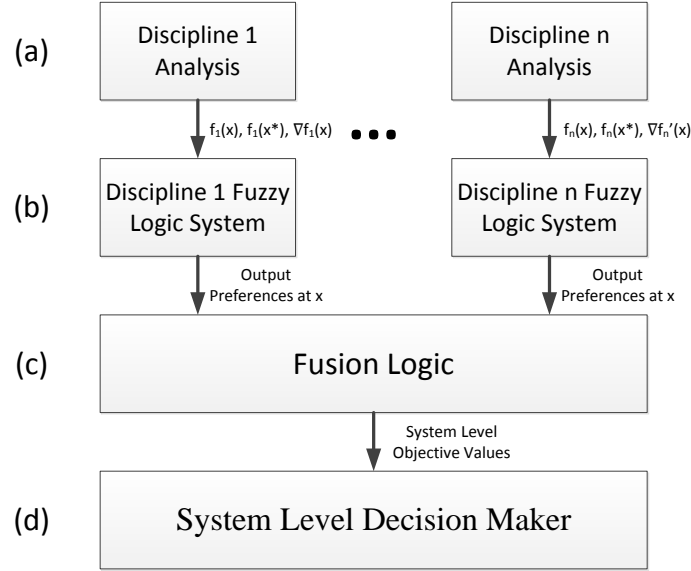


Figure 4.6: HFLC MDO Flow Chart

4.4 Discipline Fuzzy Logic System

As described in chapter III, Hierarchical Fuzzy Logic Controllers (HFLC) use fuzzy logic systems to transform the system inputs into the desired outputs. Again, for demonstration the fuzzy logic system is adapted to generalized mathematic functions with the design intent of finding a global minimum for all disciplines. As the fuzzy logic system is defined, a simple discipline will be used to walk through the steps and help visualize how the system works. This discipline's objective function is presented in equation (4.12) where $\mathbf{x} = (x_1, x_2)$ and $x_1, x_2 \in [-2, 2]$. Figure 4.7 displays the discipline's objective function values over the DV ranges and Figure 4.8 displays the $\|\nabla_{\mathbf{x}} f\|_2$ values over the DV ranges.

$$f_1(\mathbf{x}) = x_1^2 + x_2^2 \quad (4.12)$$

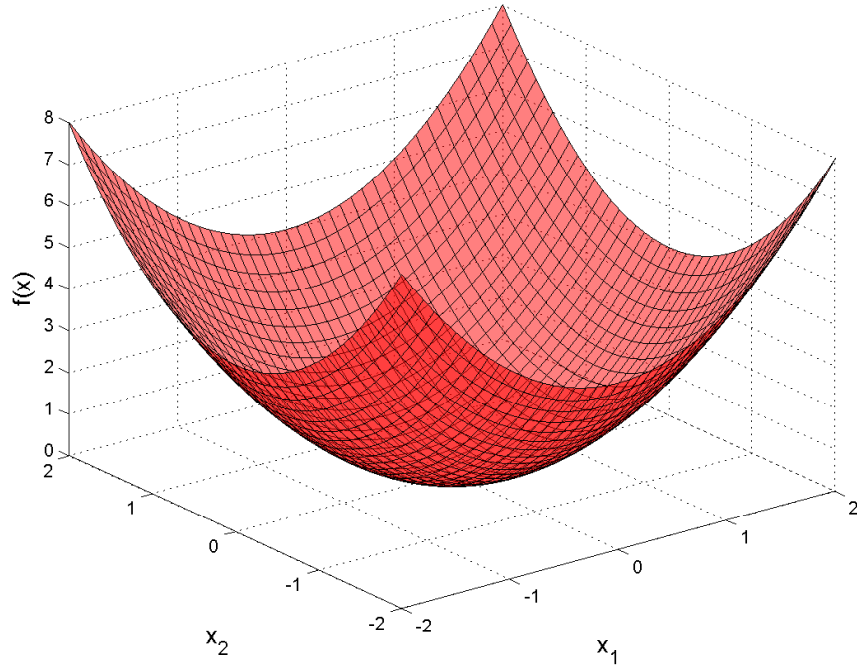


Figure 4.7: Example Discipline Objective Function Values

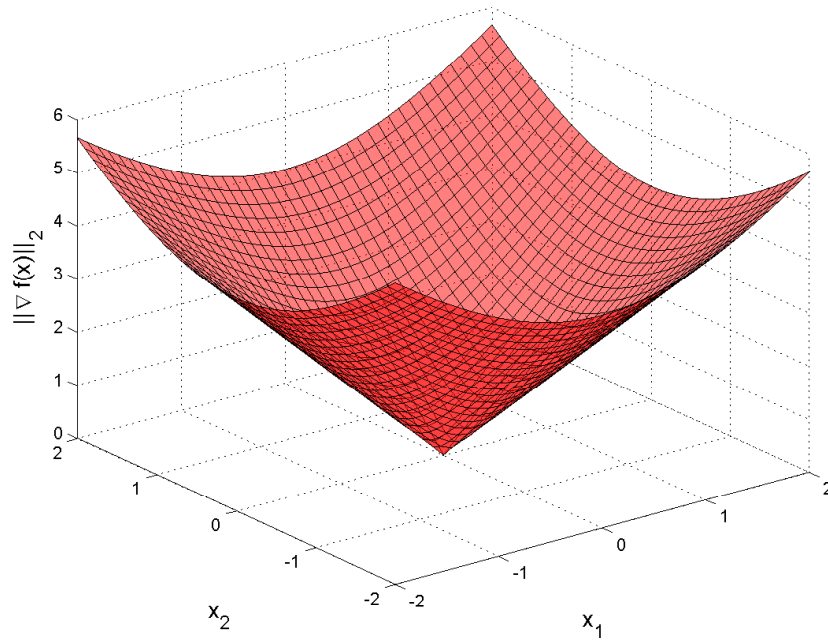


Figure 4.8: Example Discipline $\|\nabla_{\mathbf{x}} f\|_2$ Values

Using the membership functions described in (4.3) - (4.5), (4.7) - (4.9), and (4.10) - (4.11) each DV location is assigned to the fuzzy sets for the three design aspects

described previously. These preferences become the input membership functions for the FLS. Using the functions described in section 4.3.1, the value of the objective function is given high, medium, or low preferences relative the the function minimum and maximum (because only one discipline is considered, the functions actual maximum over the DV ranges is used as f^{max}). Figure 4.9 displays the results of evaluating the membership functions for the example discipline. Note that the “High Objective Function Preference” surface is at its maximum membership value at the location of the minimum in figure 4.7, and, as the DV values move away from the function minimum, the preferences transition from high to medium to low.

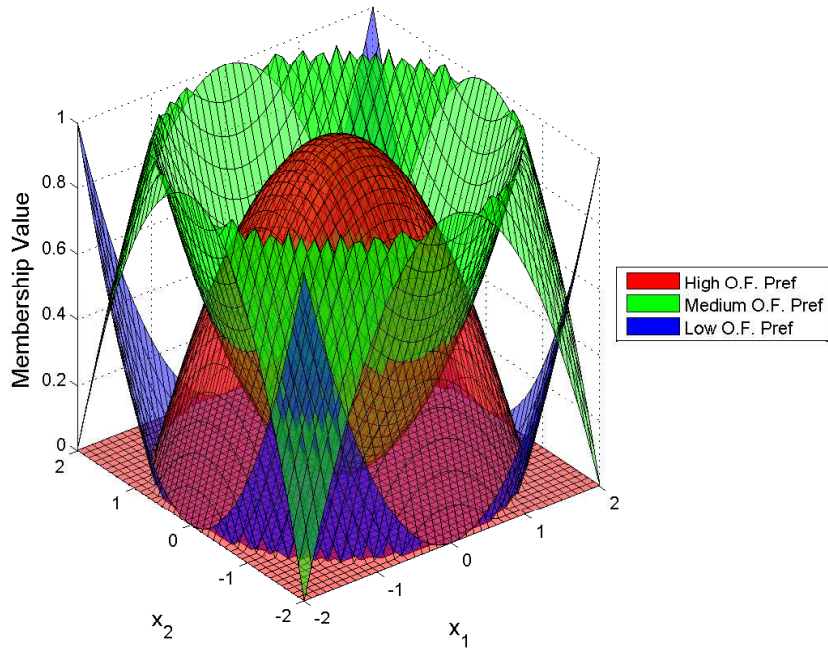


Figure 4.9: Example Discipline Objective Function Preference Plot

Figures 4.10 and 4.11 display the input membership functions that are created for the gradient norm preference and distance from optimum preference, respectively. For this example discipline, at the location of the true optimum, both of the design aspects are at their highest preference. This may not always be the case, take for example, a discipline which has a boundary optima and a local optima which has a higher objective function value. At the boundary optima, the objective function value

and distance from minimum preferences would both be at their highest preference, but the gradient norm preference would have its highest preference value at the location of the local optima.

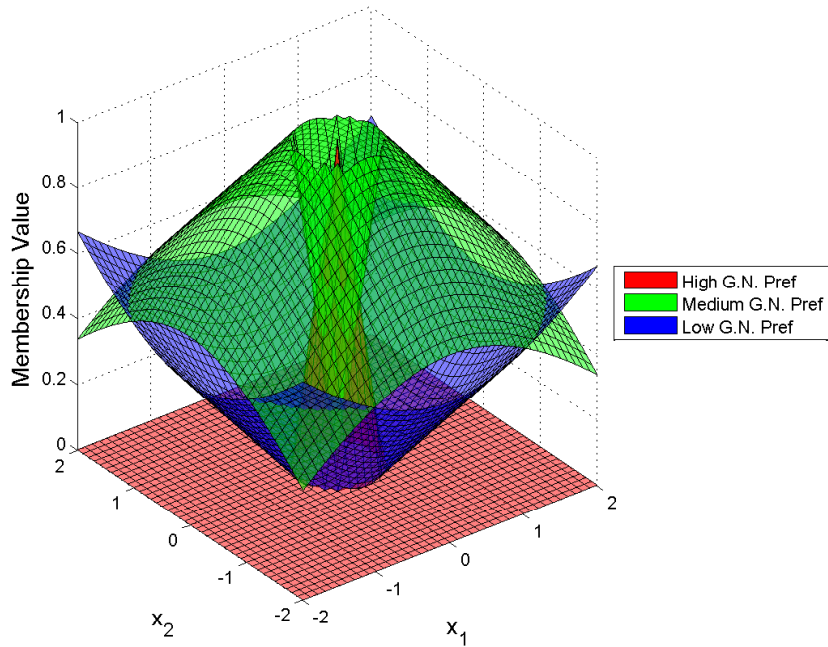


Figure 4.10: Example Discipline $\|\nabla_{\mathbf{x}} f\|_2$ Preference Plot

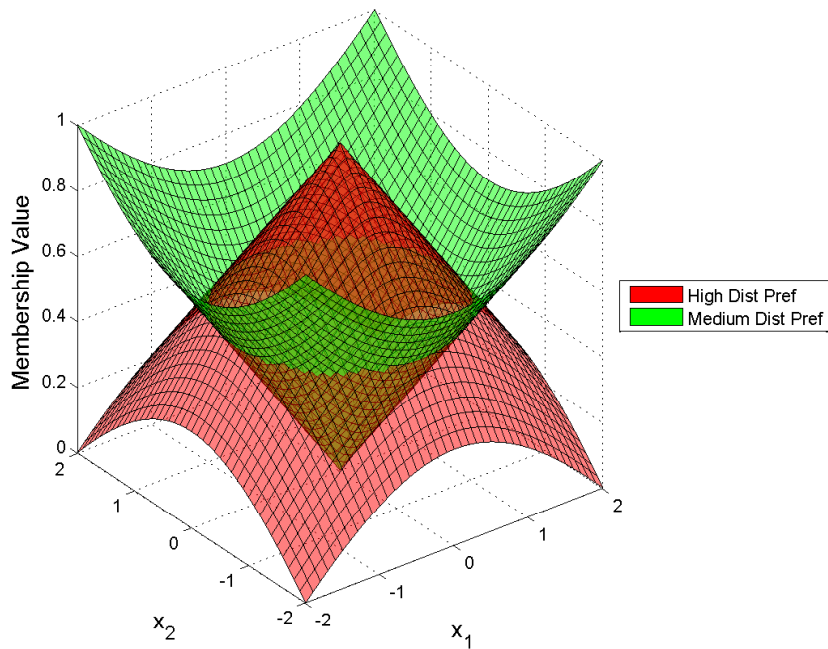


Figure 4.11: Example Discipline Distance From Minimum Preference Plot

In short, figures 4.9 - 4.11 display the input membership functions of the fuzzy logic system for this example discipline. Using these membership functions, the fuzzification interface of the FLS can transform the design space into fuzzy sets as described in the next section.

4.4.1 Fuzzification Interface

Using the input membership functions for the discipline, each DV location is assigned to fuzzy sets and given an associated membership value. This process will be demonstrated by using a test point at $\mathbf{x} = (x_1 = 0, x_2 = 1)$ where $f_1(0, 1) = 1$, and $\|\nabla_{\mathbf{x}} f\|_2|_{(0,1)} = 2$. For each designer intent aspect used, two possibilities exists for fuzzy set assignment. First, the location belongs to one fuzzy set with membership value $\mu = 1$ and all other fuzzy sets have a membership value of $\mu = 0$. Otherwise, the DV location is a member of two fuzzy sets with membership value $\mu \in (0, 1)$ and the sum of the two membership values is equal to one.

For the example discipline, the first design aspect evaluated is the objective function preference. Figure 4.12 displays a section of the example discipline objective function preference plot (figure 4.9) where $x_1 = 0$. A vertical line is added at the location of $x_2 = 1$ to highlight the value of the two non-zero fuzzy sets of which $\mathbf{x} = (0, 1)$ belongs to. From both figure 4.12 and equation (4.3), when $\mathbf{x} = (0, 1)$, the DV values are members of the fuzzy sets “High Objective Function Preference” (with a membership value $\mu_1^{high} = 0.75$) and “Medium Objective Function Preference” (with a membership value $\mu_1^{med} = 0.25$.)

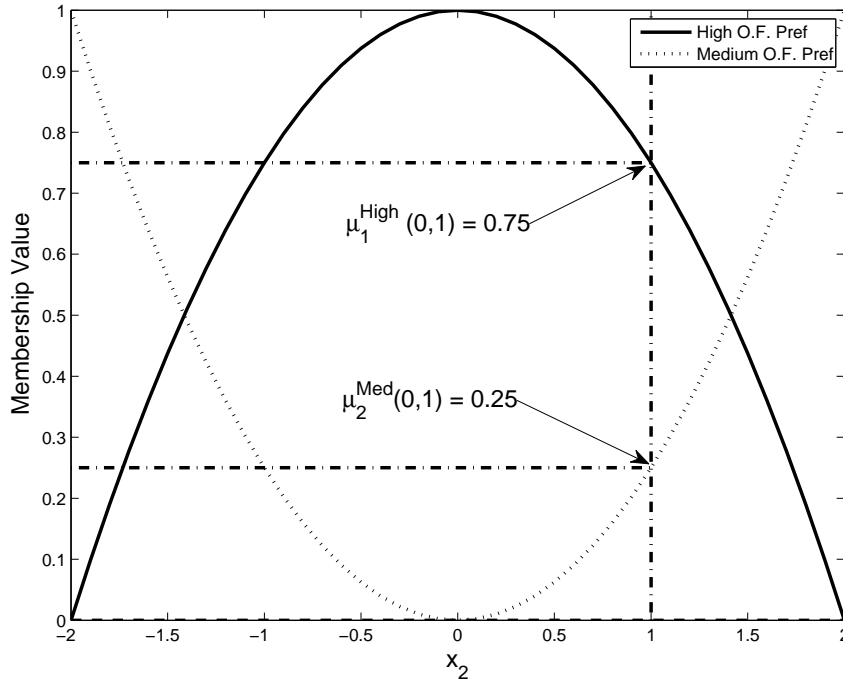


Figure 4.12: Example Discipline Objective Function Plot Section At $x_1 = 0$

The sections of figures 4.13 and 4.14 at $x_1 = 0$ are displayed in figures 4.10 and 4.11, respectively. The example discipline gradient norm preference is in figure 4.13. At the sample point $\mathbf{x} = (0,1)$ the two nonzero membership functions are $\mu_2^{\text{med}} = 0.787$ and $\mu_2^{\text{low}} = 0.213$. Figure 4.14 represents the distance from minimum preference with the sample point having membership values of $\mu_3^{\text{high}} = 0.646$ and $\mu_3^{\text{med}} = 0.354$.

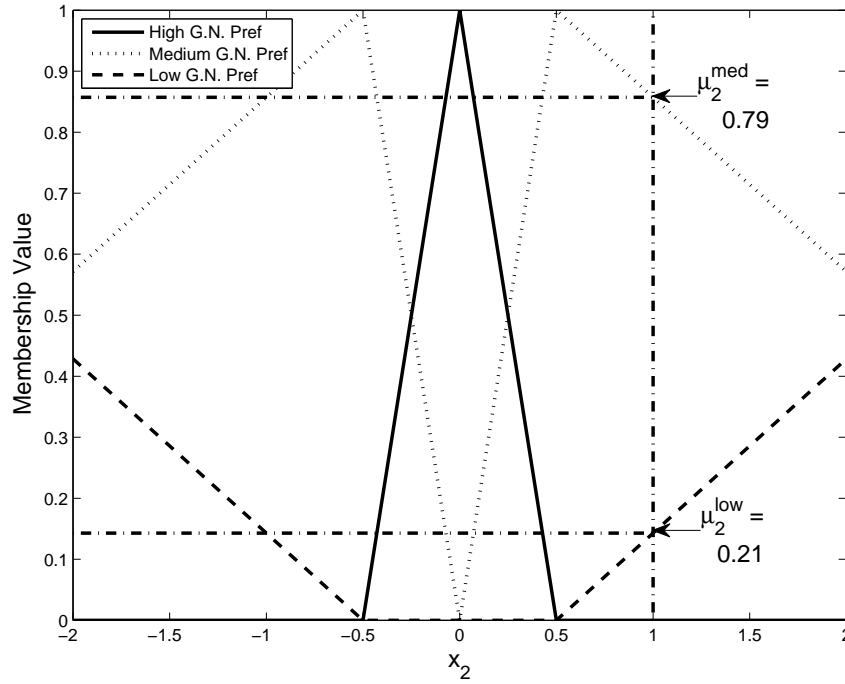


Figure 4.13: Example Discipline $\|\nabla_{\mathbf{x}} f\|_2$ Preference Plot Section at $x_1 = 0$

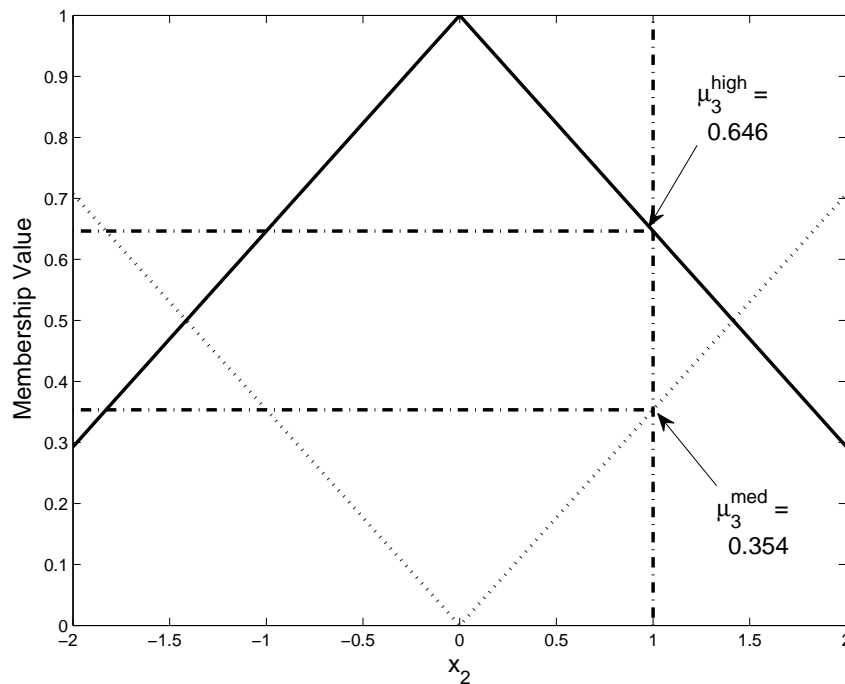


Figure 4.14: Example Discipline Distance From Minimum Preference Plot Section at $x_1 = 0$

To summarize, at the sample point $\mathbf{x} = (0, 1)$, table 4.1 lists the point's member-

ship in each of the eight possible fuzzy sets. The values of all membership functions correspond to the evaluation of the membership functions described at the beginning of this chapter with inputs being: $x_1 = 0$, $x_2 = 1$, $f(0, 1) = 1$, and $\|\nabla_{\mathbf{x}}f\|_2 = 2$.

Fuzzy Set	Membership Value
μ_1^{high}	0.75
μ_1^{med}	0.25
μ_1^{low}	0.0
μ_2^{high}	0.0
μ_2^{med}	0.787
μ_2^{low}	0.213
μ_3^{high}	0.646
μ_3^{med}	0.354

Table 4.1: Sample Point $\mathbf{x} = (0, 1)$ Fuzzy Set Membership Values

4.4.2 Fuzzy Rule Bank

The output from the fuzzification interface are sent to the fuzzy rule bank to map the input linguistic to the corresponding output linguistic. Again, as will be demonstrated in chapter V, depending on the intent of the design, the rule bank can be modified to fit the problem being addressed. To demonstrate the algorithm for the example discipline, with objective function in equation (4.12), a decision matrix has been developed for mathematics disciplines. This matrix gives an output discipline preference level based on the combination of the three input design intent models.

Figures 4.15 and 4.16 display the decision matrix developed for the example problem. The figure is divided into two matrices according to the distance from minimum preference. In this decision matrix, more emphasis is given to the actual value of the objective function over other aspects. This is apparent from the asymmetry in the matrices.

High Gradient Preference	Low Preference	Medium Preference	High Preference
Medium Gradient Preference	Low Preference	Medium Preference	High Preference
Low Gradient Preference	Low Preference	Low Preference	Medium Preference
	Low Objective Function Preference	Medium Objective Function Preference	High Objective Function Preference

Figure 4.15: Example Decision Matrix When “Distance From Minimum Preference” is High

High Gradient Preference	Low Preference	Medium Preference	Medium Preference
Medium Gradient Preference	Low Preference	Low Preference	Medium Preference
Low Gradient Preference	Unpreferred	Low Preference	Low Preference
	Low Objective Function Preference	Medium Objective Function Preference	High Objective Function Preference

Figure 4.16: Example Decision Matrix When “Distance From Minimum Preference” is Med

The fuzzy rule bank examines all combinations of the input fuzzy sets. Each possible combination that contains one non-zero set from all three of the input preference models is considered an activated fuzzy rule. In this example, the fuzzy rules are in

the form of equation (4.13), where each of the three parts of the antecedent correspond to a nonzero preference in one of the three input design intent models, and the consequent is the resultant output discipline preference linguistics found using the decision matrix.

$$\begin{aligned}
 & \mathbf{IF} \mu_1 \text{ is high preference, AND} \\
 & \quad \mu_2 \text{ is medium preference, AND} \\
 & \quad \mu_3 \text{ is high preference,} \\
 & \mathbf{THEN} \text{ Output is High preference}
 \end{aligned} \tag{4.13}$$

At the sample point $\mathbf{x} = (0, 1)$ the input linguistics to the fuzzy rule bank are listed in table 4.1. These inputs are translated to the output preferences using the decision matrix. The result is eight activated fuzzy rules which are in table 4.2.

Rule #	μ_1	μ_2	μ_3	Output
1	High	Med	High	High
2	Med	Med	High	Medium
3	High	Low	High	Medium
4	Med	Low	High	Low
5	High	Med	Med	Medium
6	Med	Med	Med	Low
7	High	Low	Med	Low
8	Med	Low	Med	Low

Table 4.2: Activated Fuzzy Rules

4.4.3 Fuzzy Inference

After the fuzzy rule bank is applied to the input linguistics, an activation level must be selected to complete the fuzzy sets that will be the output from the fuzzy inference module. As discussed in chapter III, the minimum inference method will be used to select an activation level.

To describe this process, the same example discipline and sample point described previously will be used. The minimum inference method is used on each activated rule to create an output fuzzy set for that rule. For example, in table 4.2, rule #1 is activated because the membership values for “High Objective Function Preference”, “Medium Gradient Norm Preference”, and “High Distance to Minimum Preference” are all greater than zero. From the fuzzy rule bank, this combination of non-zero inputs results in a “High Preference,” which needs an activation level, found using equation (3.1).

For the sample point rule #1, the input membership values are: $\mu_1^{high} = 0.75$, $\mu_2^{med} = 0.787$, and $\mu_3^{high} = 0.646$. Using equation (3.1), the activation level associated with the “High Preference” linguistic found from the fuzzy rule bank would be $\mu_{out}^{high} = 0.646$. This same process repeats for each of the activated rules in table 4.2, while the resultant rules and membership values are in table 4.3.

Rule #	Linguistic	Membership Value
1	High	0.646
2	Medium	0.250
3	Medium	0.213
4	Low	0.213
5	Medium	0.354
6	Low	0.250
7	Low	0.213
8	Low	0.213

Table 4.3: Sample Point Output Linguistics and Membership Values

Once both a linguistic and an activation level have been assigned, the full fuzzy sets must be defined for use in the defuzzification interface. The linguistic terms that are assigned by the rule bank correspond to the output fuzzy sets displayed in figure 4.17. Four output linguistics are used: “Unpreferred”, “Low Preference”, “Medium Preference”, and “High Preference”. These output linguistic correspond to how satisfied the designer intent models are for the discipline being analyzed. For example, at the discipline’s true optimum, the location would have high preferences in

each of the input areas described previously’ which would lead to the highest output preference value. This method of assigning DV preferences was used by Singer [72] and Gray [20] in their work on set-based design and communicating preferences between design agents.

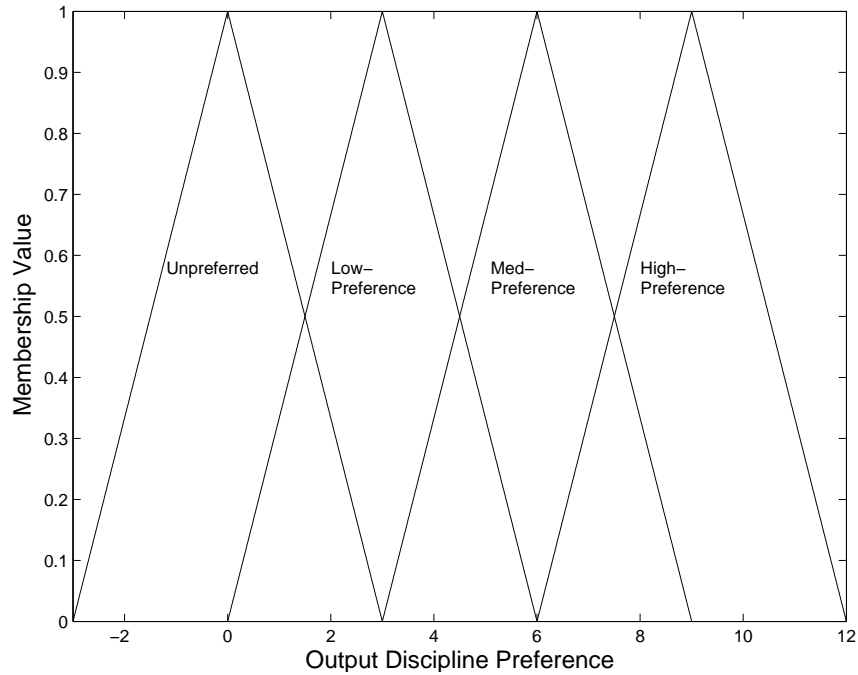


Figure 4.17: Output Preference Fuzzy Sets

The output fuzzy sets assigned by the rule bank are modified by the activation level. As in section 3.2, the clipping method of modifying the output fuzzy sets will be used. In this method, the parent output fuzzy set is trimmed based on the activation level found during the fuzzy inference. For rule #1, the activated rule gives an output of “High Preference,” which corresponds to the triangular membership function labeled by the the same name with a centroid located at nine. Figure 4.18 displays the high output preference membership function along with the trim value from table 4.3 for rule #1, this value is then used to modify the parent membership function by trimming all values higher than the trim value. The modified output membership function created for rule #1 is in figure 4.19.

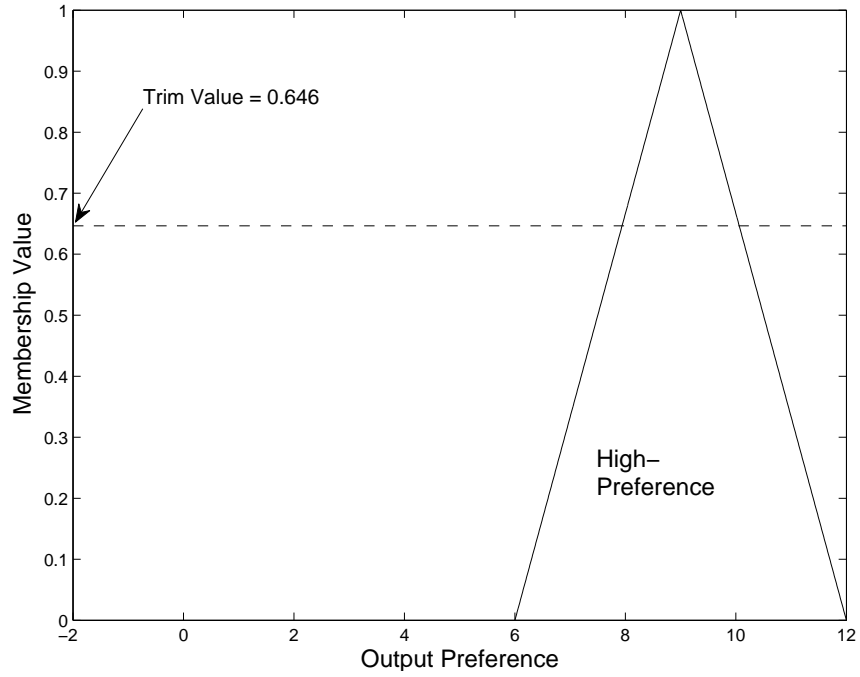


Figure 4.18: High Preference Output Membership Function With Rule #1 Trim Value

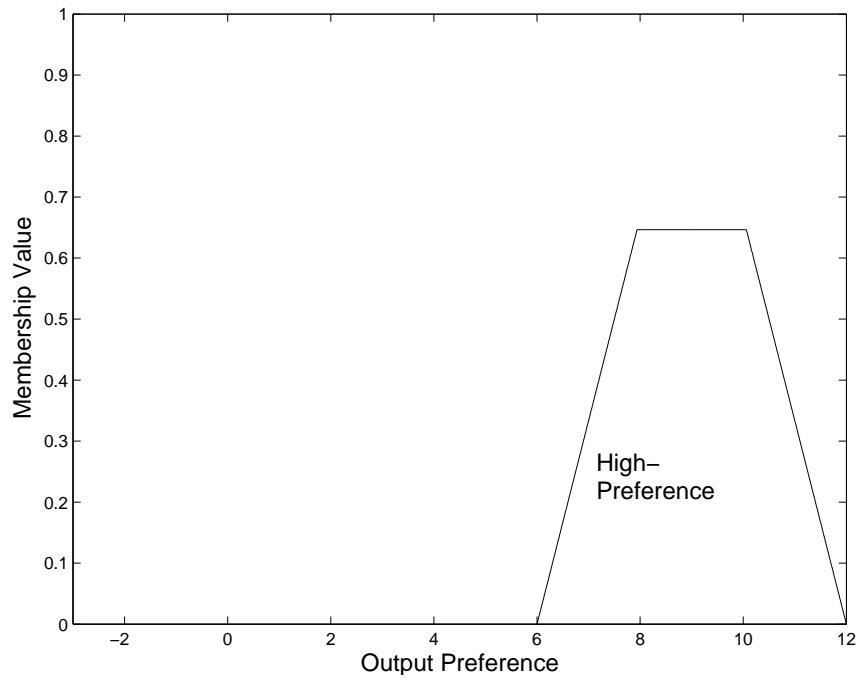


Figure 4.19: High Preference Output Membership Function Trimmed for Rule #1 Trim Value

The modified output membership function in figure 4.19 is passed to the defuzzi-

fication interface along with modified output membership functions derived using the other seven rules in table 4.3. The full set of output fuzzy sets is displayed in figure 4.20. These output fuzzy sets correspond to the sample point ($\mathbf{x} = (0, 1)$) used from the example discipline.

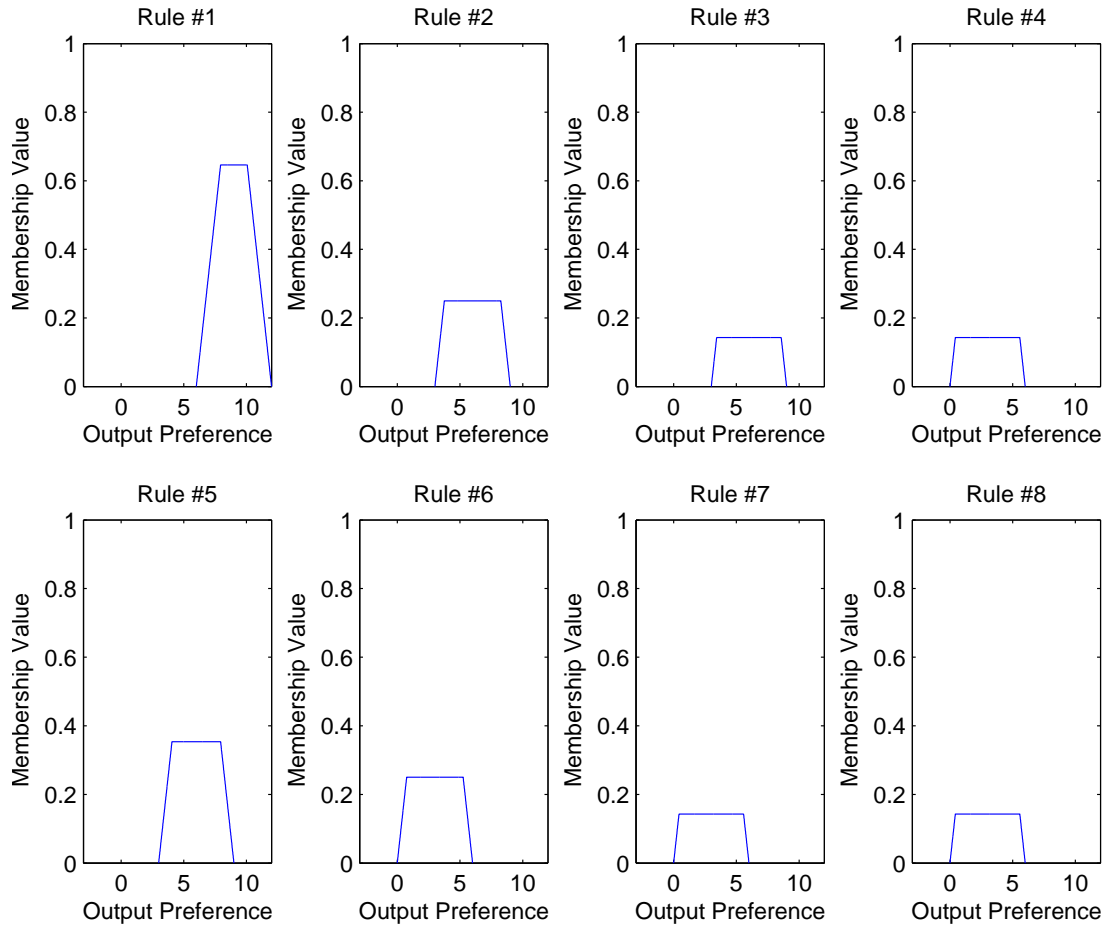


Figure 4.20: Fuzzy Inference Output

4.4.4 Defuzzification Interface

After the fuzzy inference has created output fuzzy sets, such as those shown for the example point (figure 4.20), the information passes to the defuzzification interface which will transform the data back into a crisp output used in the fusion logic. In

this dissertation the centroid method of defuzzification will be used. The centroid defuzzification method uses equation (3.6) to create the crisp, defuzzified output. In equation (3.6), A_i is the area of the fuzzy set for rule i , x_i is the location of the centroid for rule i , and I is the total number of rules activated for the current DV location.

For the sample point $\mathbf{x} = (0, 1)$, the inputs into equation 3.6 are listed in table 4.4. The discipline fuzzy logic system relies on these areas and centroids to find the final defuzzified output. The output of the defuzzifier for the sample point is represented graphically in figure 4.21. In the figure, rules #4, #7, and #8 all overlap, so only six rules are visible. When equation 3.6 is evaluated using the inputs from table 4.4, the defuzzified output is 5.46

Rule #	A_i	x_i
1	2.63	9
2	1.31	6
3	1.14	6
4	1.14	3
5	1.75	6
6	1.31	3
7	1.14	3
8	1.14	3

Table 4.4: Sample Point Defuzzification Input Values

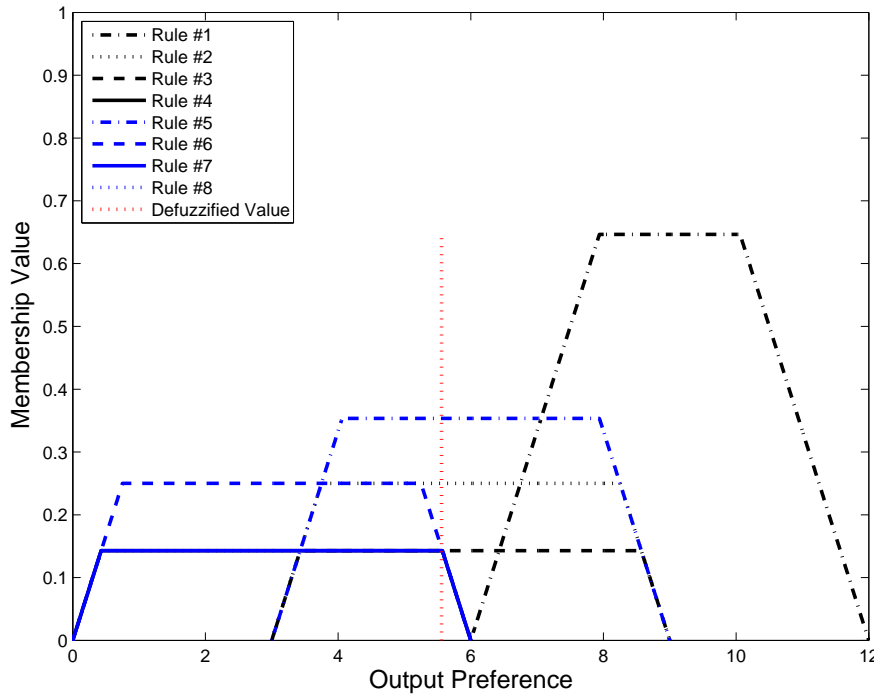


Figure 4.21: Sample Point $x = (0,1)$ Defuzzification

4.4.5 Discipline FLS Output

The fuzzification, fuzzy rules, fuzzy inference, and defuzzification process described previously are carried out for each point being analyzed. This process maps the design space to discipline preferences for each DV location based on the design intent rules used for inputs of the discipline's fuzzy logic system. The output of the example discipline's FLS over the range of DVs is in figure 4.22. While observing the output with respect to the single discipline, the objective function's true minimum is the area with the highest preference. At this location, the objective function is at its minimum value, the gradient is zero, and the location is at the global minimum. At the points $x = (2, 2)$, $(2, -2)$, $(-2, 2)$, $(-2, -2)$, the output discipline preference of the FLS is zero. Here the objective function value is at its highest point, the gradient norm is at its maximum value, and the four points are all at the maximum distance

from the global minimum.

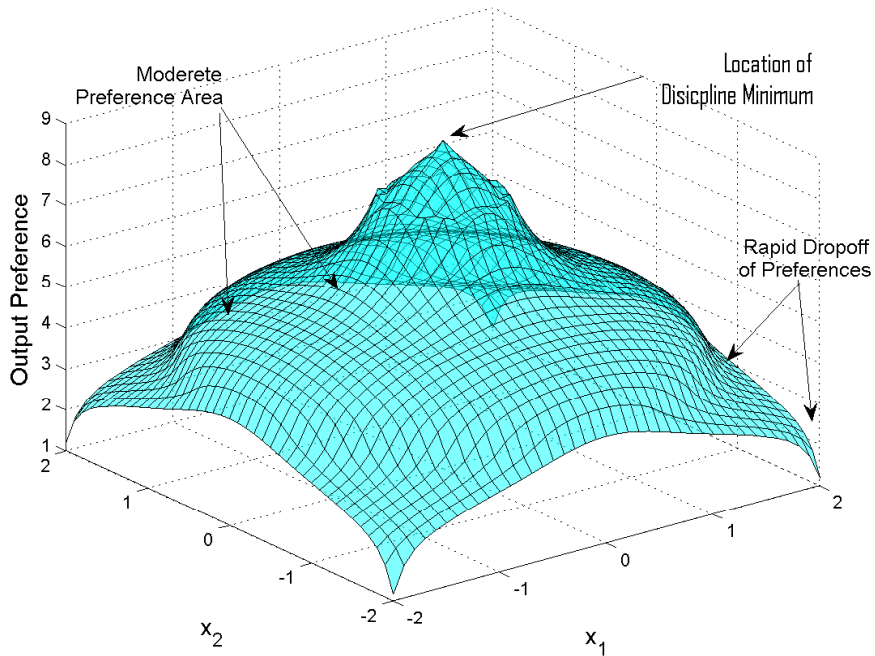


Figure 4.22: Example Discipline FLS Output

Figure 4.22 also presents how mapping design intent can highlight the important areas in the design space. Near to the optimum, where the gradient norm of the objective function is small, is much more preferred than the other areas. Also, there is a flat spot, as the objective function value and gradient norm are still moderately preferred compared to other areas followed by a steep drop off in preferences at the corners where the DVs have high objective function values, very steep gradients, and are the furthest points from the minimum.

4.5 Fusion Logic

For a single discipline, it is apparent from figure 4.22 that the design method would lead to the same optimal design point, as an optimization routine that only considers the objective function value of the discipline during analysis. However, the mapping of the design space based on design intent features can lead to different

results when the discipline is considered as part of a system that has multiple goals and when trade-offs must be considered.

The fuzzy logic system used to analyze a discipline, described for the previous example, would represent only one discipline analysis in a MDO problem. In a real world problem, several disciplines would be analyzed and a method is needed to decide on the final trade-offs between disciplines. The concept of fusion logic from hierarchical fuzzy logic controllers (HFLLC) will be described in the next section. For the description, three sample disciplines are used, including the example discipline used previously.

The other two example disciplines are described by equations (4.14) and (4.15). As previously described, $\mathbf{x} = (x_1, x_2)$, and $x_1, x_2 \in [-2, 2]$. The plots for the objective functions of the two sample disciplines are displayed in figures 4.23 and 4.24; the outputs of each disciplines fuzzy logic systems are shown in figure 4.25 and 4.26. The plots in figures 4.22, 4.25, and 4.26 represent the input into the fusion logic.

$$f_2(\mathbf{x}) = x_1^4 - 3 * x_1^2 + x_1 + x_2^4 - 3 * x_2^2 + x_2 \quad (4.14)$$

$$f_3(\mathbf{x}) = x_1 + x_2 \quad (4.15)$$

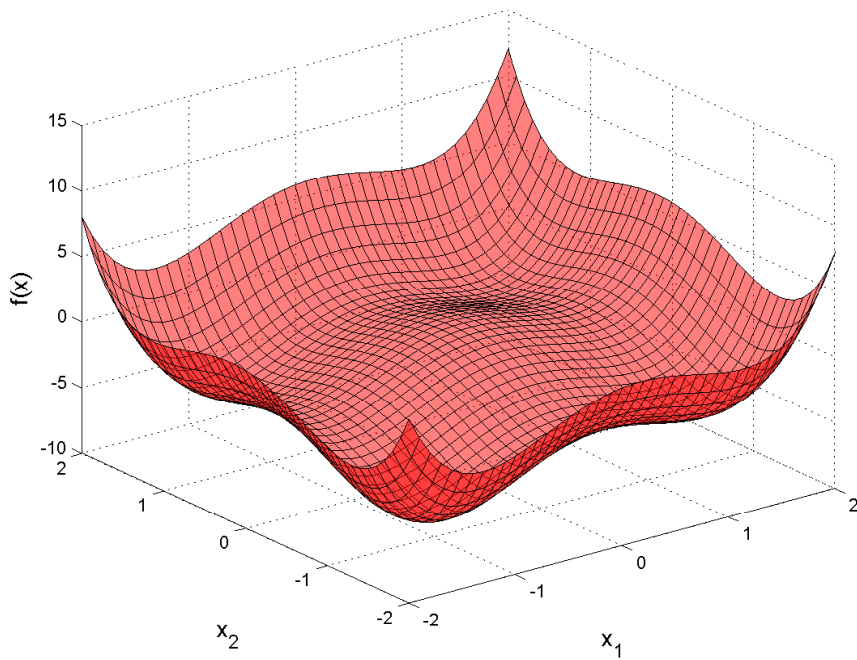


Figure 4.23: Example Discipline 2 Objective Function Values

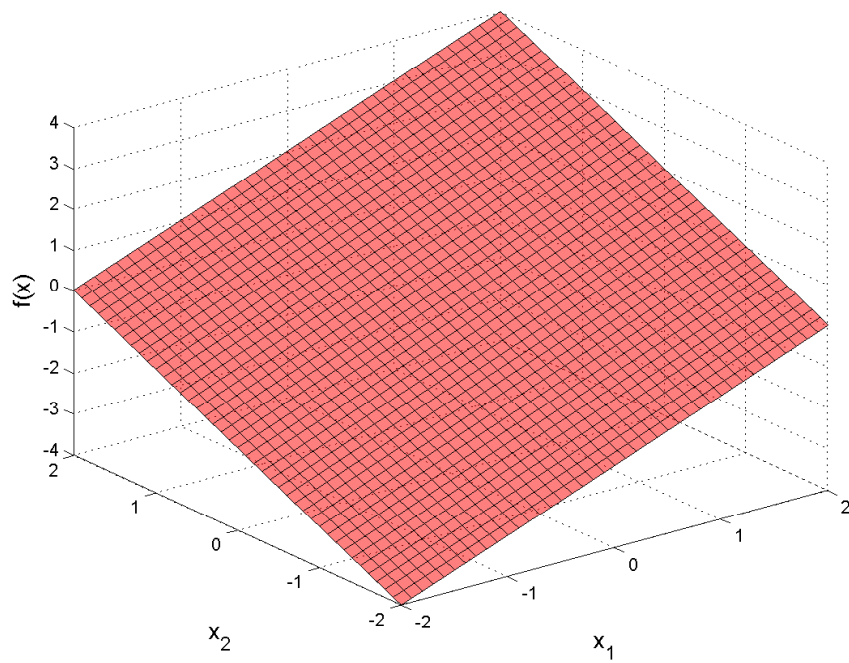


Figure 4.24: Example Discipline 3 Objective Function Values

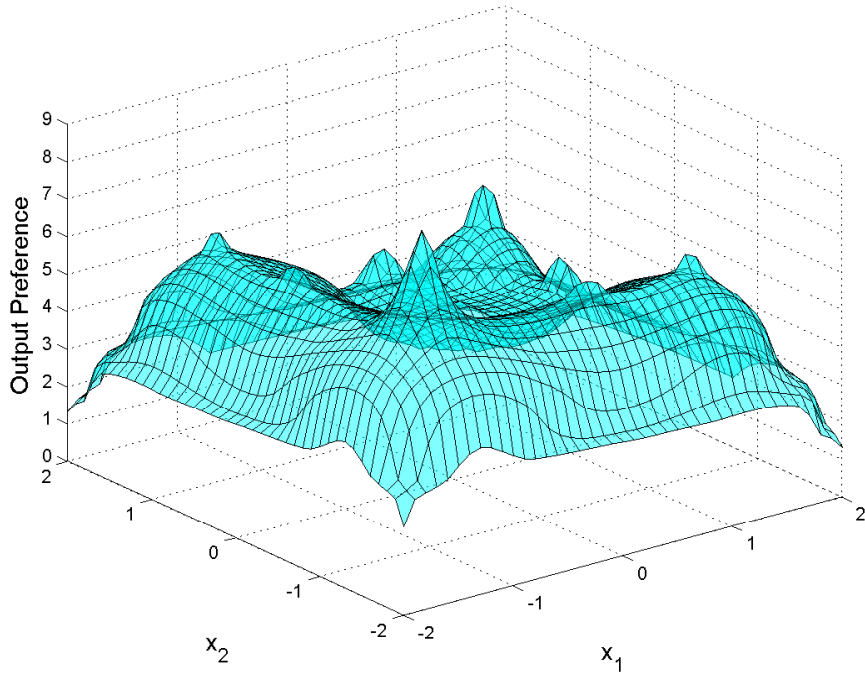


Figure 4.25: Example Discipline 2 FLS Output

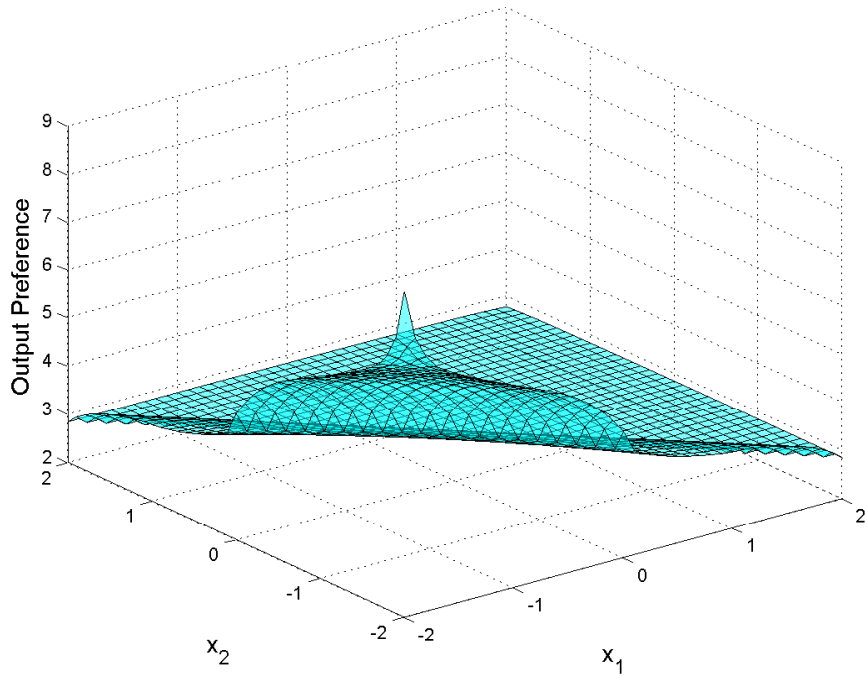


Figure 4.26: Example Discipline 3 FLS Output

The fusion logic in a HFCLC takes the inputs from all sensors and contains logic to

transform the inputs from the sensors into an output that can be used by the system. When considering a controller in the design space, the inputs to the fusion logic would be each disciplines preference for the current DV location. The fusion logic uses the inputs from each of the disciplines to create the system level preference for the DV location.

As described in chapter III, the minimum input method is used for selecting the fusion logic output, where the discipline controller that has the lowest preference level at each DV location is set as the system level preference. The fusion logic uses equation 3.7 to determine the system level preference.

The disciplines in figures 4.22, 4.25, and 4.26 will be used to demonstrate the output of the fusion logic. Equation (3.7) is applied at each of the DV locations tested for each discipline. Figures 4.27 and 4.28 show the evaluation of the fusion logic at each point, and in figure 4.29 the contour plot of the fusion logic graph is displayed for improved clarity.

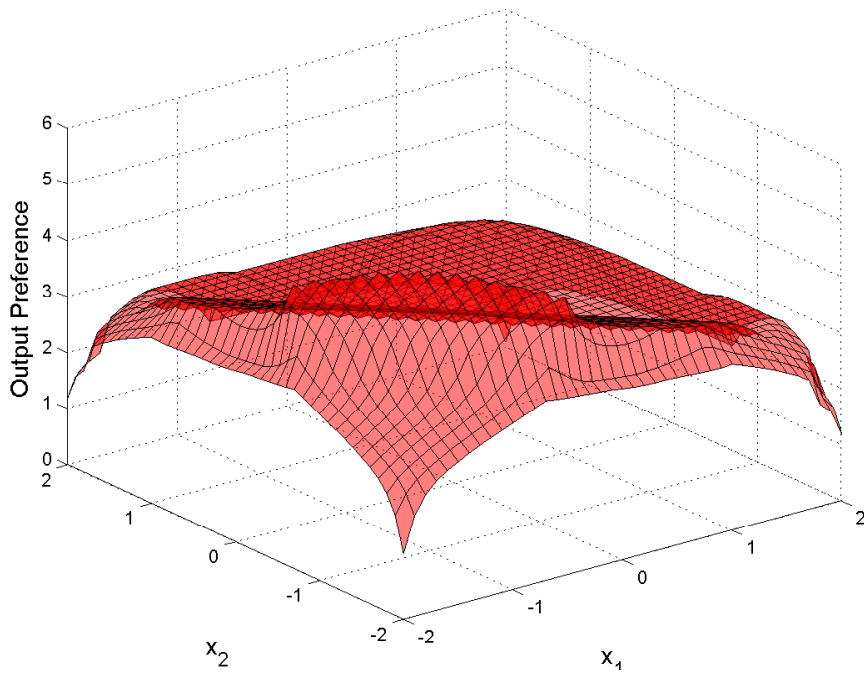


Figure 4.27: Fusion Logic Output Plot

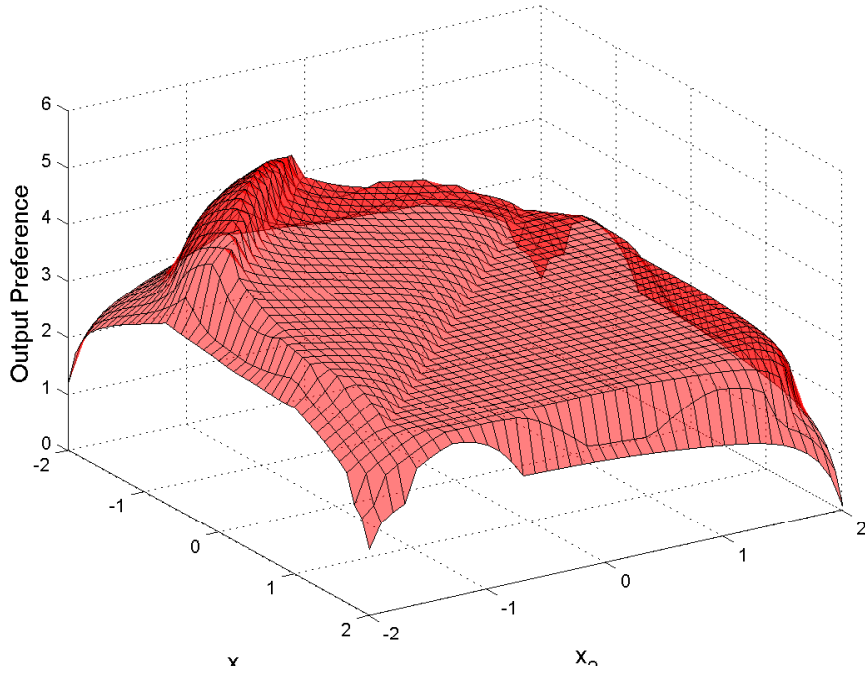


Figure 4.28: Fusion Logic Output Plot Rotated 90 Degrees

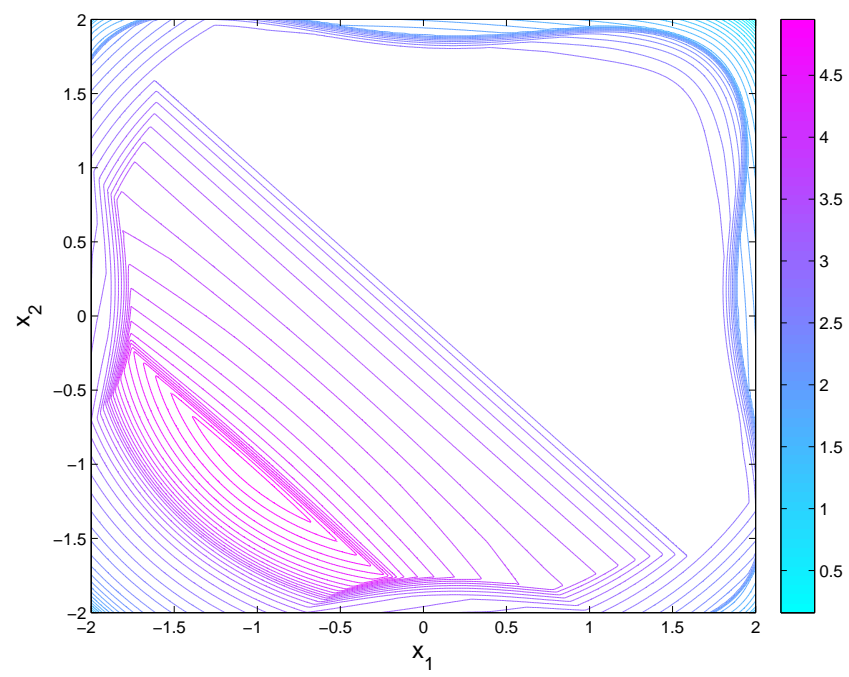


Figure 4.29: Fusion Logic Output Contour Plot

4.6 System Level Optimizer

When the system level optimizer receives the output data from the fusion logic, the design space has been mapped to a set of system level preferences for each DV location. The system level optimizer takes this information and selects an optimum point. The point selected reflects the maximum preference over the design space. Because of the fusion logic method, this point corresponds to the point where the discipline with the lowest preference level is at its highest value.

For the example problem developed throughout this chapter, the “optimal” point is located at $\mathbf{x} = (-1.05, -1.05)$. The maximum system level preference for this example problem is 5.05.

4.7 Constraint Handling Method

During optimization algorithms, constraints are used to ensure the final design meets certain requirements, such as having a required minimum top speed. In a multidisciplinary design space, requirements of other disciplines may push the design away from an optimal top speed, but these requirements must still be satisfied. For the method developed, the constraints are treated as another set of rules in the fuzzy rule bank.

Defining constraints as fuzzy sets allows for flexibility in the constraints. As demonstrated in [12], by defining constraints based on a goal and threshold, constraint values can be weighted allowing for trade-offs between an optimal design and proximity to the constraint threshold. Using the speed example, a ship’s requirements may state that the final design must be able to travel faster than 30 knots, while a speed greater than 35 knots is highly preferred. In this example, the goal speed is 35 knots while the threshold speed is 30 knots.

Equation (4.16) formulates how membership functions for a satisfied constraint are

calculated, while equation (4.17) expresses the membership function when a constraint is violated. In equations (4.16) and (4.17), $\mu_C^{Satisfied}$ is the membership value for the “Constraint Satisfied” fuzzy set, $\mu_C^{Violated}$ is the membership value for the “Constraint Violated” fuzzy set, $g_k(\mathbf{x})$ is the value of constraint k at DV values \mathbf{x} , g_k^G is the goal value of the constraint, and g_k^T is the threshold value of the constraint. Figure 4.30 shows an example of constraint fuzzy sets where there is a transition area from the constraint being fully satisfied and fully violated. All constraints can be handled in this manner, but if the engineer desires a constraint to have a crisp value, the threshold and goal values are equal.

$$\mu_C^{Satisfied}(g_k(\mathbf{x})) = \begin{cases} 0 & g_k(\mathbf{x}) \geq g_k^T \\ \frac{g_k(\mathbf{x})}{g_k^G - g_k^T} - \frac{g_k^T}{g_k^G - g_k^T} & g_k^G \leq g_k(\mathbf{x}) \leq g_k^T \\ 1 & g_k(\mathbf{x}) \leq g_k^G \end{cases} \quad (4.16)$$

$$\mu_C^{Violated}(g_k(\mathbf{x})) = \begin{cases} 0 & g_k(\mathbf{x}) \leq g_k^G \\ \frac{g_k(\mathbf{x})}{g_k^T - g_k^G} - \frac{g_k^G}{g_k^T - g_k^G} & g_k^G \leq g_k(\mathbf{x}) \leq g_k^T \\ 1 & g_k(\mathbf{x}) \geq g_k^T \end{cases} \quad (4.17)$$

$$(4.18)$$

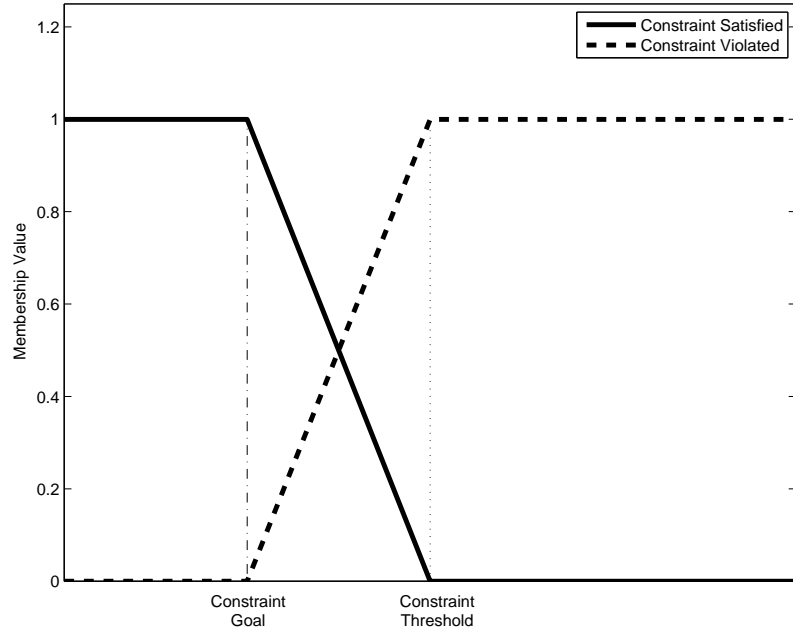


Figure 4.30: Example Constraint Fuzzy Sets

Adding constraints adds new input fuzzy sets to the FLS, so new rules must be derived in the fuzzy rule bank. The number of rules that are necessary does not become a problem of dimensionality because the concept of “meta rules” is used [72, 20]. If all constraints are totally satisfied, the output preference is identical to the original rule bank. On the other hand, if any constraint is totally violated then the DV location would have an unpreferred output preference. This outcome is displayed in figures 4.31 and 4.32 where figure 4.31 is an example decision matrix with all constraints satisfied and figure 4.32 is the decision matrix with a constraint violated.

High Gradient Preference	Low Preference	Medium Preference	High Preference
Medium Gradient Preference	Low Preference	Medium Preference	High Preference
Low Gradient Preference	Low Preference	Low Preference	Medium Preference
	Low Objective Function Preference	Medium Objective Function Preference	High Objective Function Preference

Figure 4.31: Example Decision Matrix When All Constraints Are Satisfied

High Gradient Preference	Unpreferred	Unpreferred	Unpreferred
Medium Gradient Preference	Unpreferred	Unpreferred	Unpreferred
Low Gradient Preference	Unpreferred	Unpreferred	Unpreferred
	Low Objective Function Preference	Medium Objective Function Preference	High Objective Function Preference

Figure 4.32: Example Decision Matrix When A Constraint Is Violated

A constraint, equations (4.19) and (4.20), is added to the example MDO problem that has been developed throughout this chapter to demonstrate the constraint handling method. The threshold value for the example constraint is $g_1^{threshold} \leq 0$, but an arbitrary goal value is assigned where the constraint is fully satisfied if the constraint

value is $g_1^{goal} \leq -10$.

$$g_1^{threshold}(\mathbf{x}) : 0 \geq -16x_1 - 3x_2 - 26 \quad (4.19)$$

$$g_1^{goal}(\mathbf{x}) : -10 \geq -16x_1 - 3x_2 - 26 \quad (4.20)$$

When analyzed over the design space, the constraints' membership functions are displayed in figure 4.33. This figure shows how the constraint values ($g_1(\mathbf{x})$) transition from satisfied to violated.

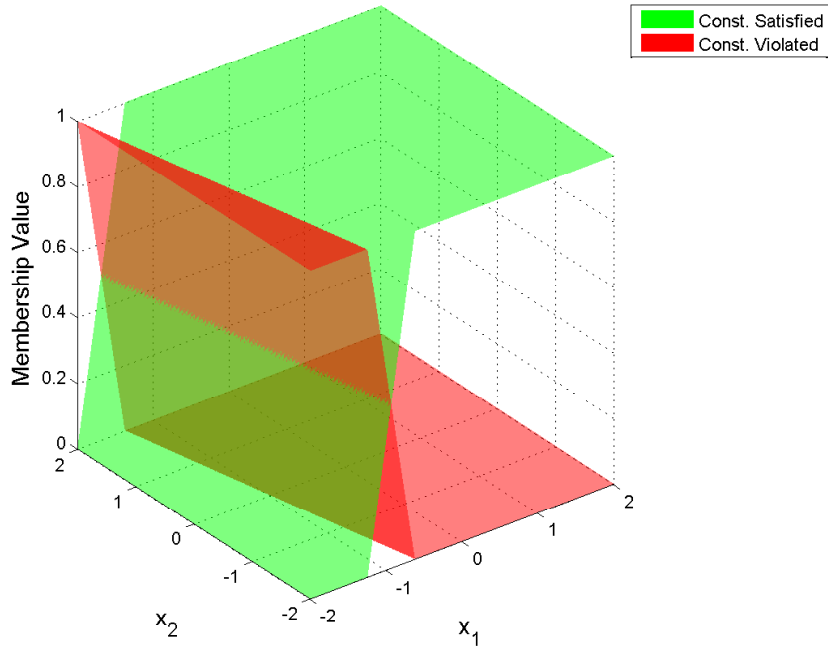


Figure 4.33: Example Constraint Fuzzy Sets Plots

Using the rules in figures 4.31 and 4.31, the example discipline with objective function in equation (4.12) is analyzed with the fuzzy logic system using the constraint described in equations (4.19) and (4.20). The output preferences from the discipline, subject to the constraint, are displayed in figure 4.34. This figure presents how, in the areas where the constraint is fully satisfied, the output preference values are not affected by the introduction of the constraint. While the constraint is violated,

the preference is zero. In the area between the goal and threshold the preference is penalized from the unconstrained problem.

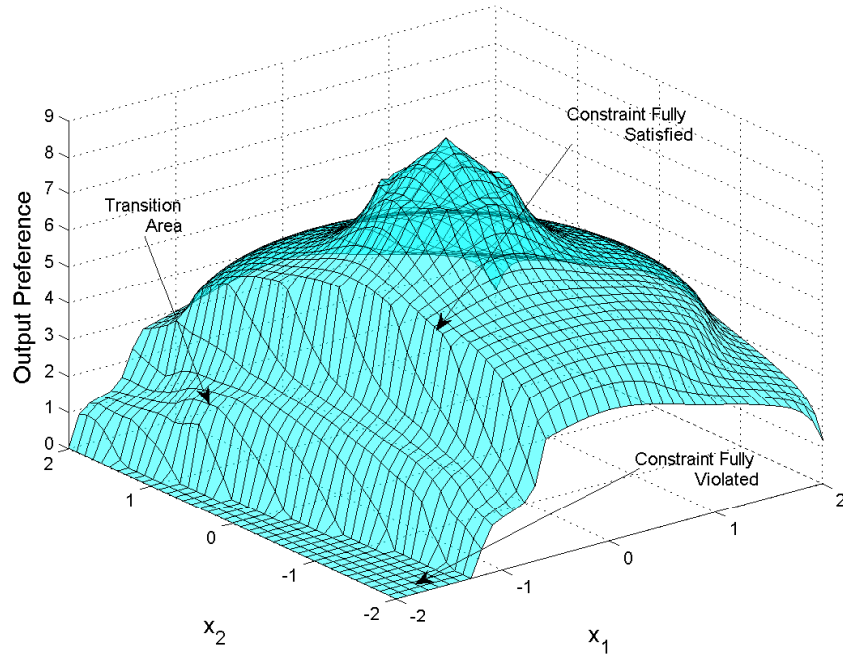


Figure 4.34: Example Discipline FLS Output Subject to Constraint

For a single discipline, the constraint output appears to rule out any values between the constraint goal and threshold; however, this is not the case when trade-offs are considered. If only one discipline is considered, it would be optimal to reach the goal of the constraint; however, when other disciplines are being taken into account, it becomes necessary to look into options that are in the area between the goal and threshold values.

4.8 Mathematical Example Problem

An example problem was created to compare the HFLC MDO to existing methods of MDO. This problem uses the design intent found in equations (4.3) - (4.5), (4.7) - (4.9), (4.10) - (4.11), and (4.19) - (4.20), and the fuzzy rule bank described in sections 4.4.2 and 4.7. The MDO problem statement is outlined in equation (4.21).

System Level:

$$\underset{\mathbf{x}}{\text{maximize}} \quad F(\mathbf{x})$$

Discipline 1:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f_1(\mathbf{x}) = x_1^2 + x_2^2$$

$$\text{Subject To: } g_1(\mathbf{x}) : 0 \geq -16x_1 - 3x_2 - 26 \quad (4.21)$$

Discipline 2:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f_2(\mathbf{x}) = x_1^4 - 3x_1^2 + x_1 + x_2^4 - 3x_2^2 + x_2$$

Discipline 3:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f_3(\mathbf{x}) = x_1 + x_2$$

In equation (4.21), $F(\mathbf{x})$ is the output of the fusion logic at DV values \mathbf{x} , f_1 is the objective function for discipline 1 (D1), f_2 is the objective function for discipline 2 (D2), and f_3 is the objective function for discipline 3 (D3) and \mathbf{x} is the vector of design variables where $\mathbf{x} = (x_1, x_2)$. The constraint $g_1(\mathbf{x})$ is applied to D1. For this example, the constraint is applied as a crisp constraint, so that the HFLC MDO method can be compared to existing methods of MDO.

4.8.1 Discipline Analyzers

Each discipline begins at a discipline analyzer in figure 4.6. The results of the discipline analyzer are in figures 4.35 - 4.40 and table 4.5. Table 4.5 also lists each objective function value when evaluated at the optimum DV locations for the other

two disciplines. This information is used in the objective function preference membership functions for calculations of the objective function maximum.

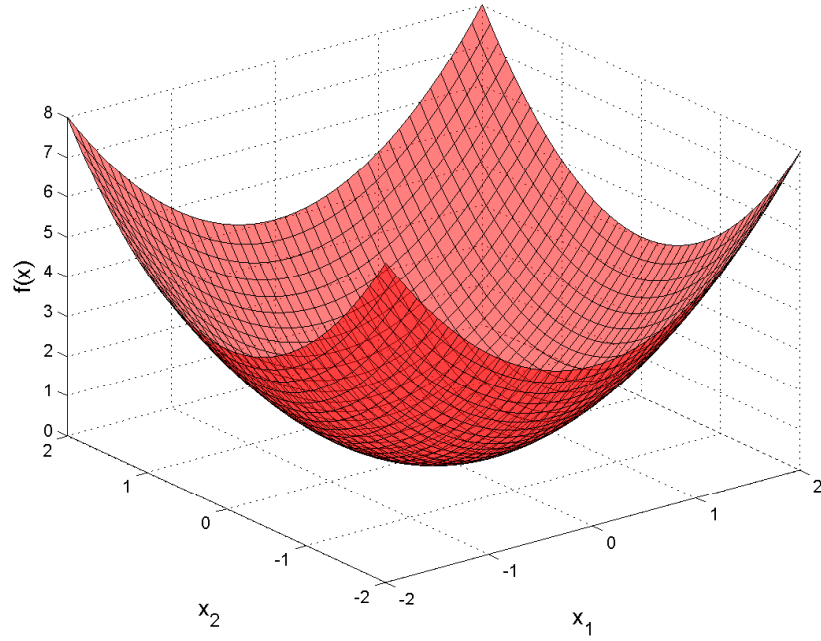


Figure 4.35: D1 Objective Function Plot

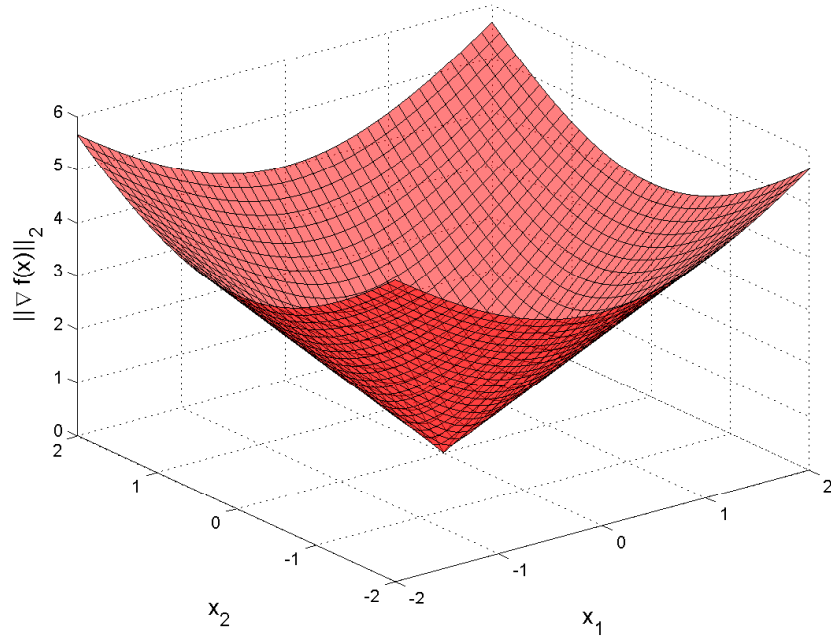


Figure 4.36: D1 Gradient ℓ_2 Norm Plot

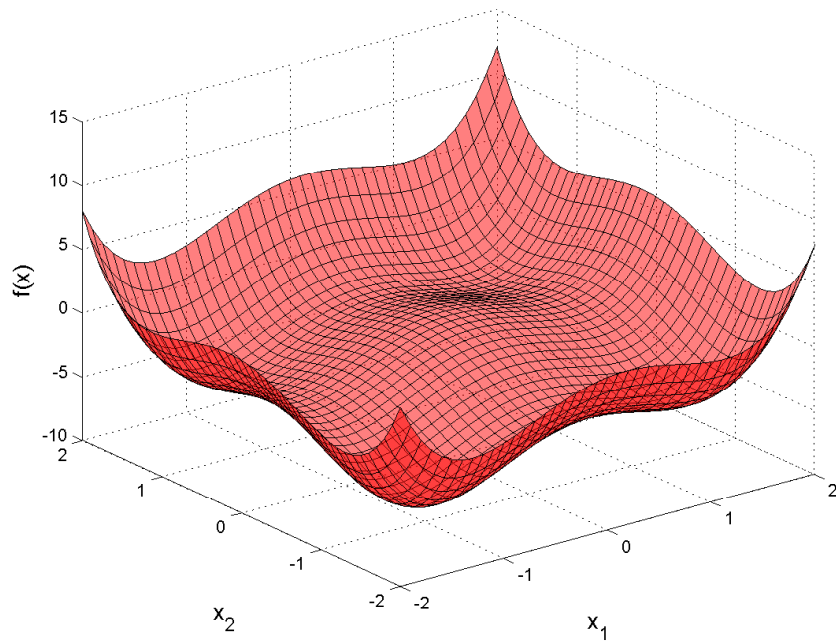


Figure 4.37: D2 Objective Function Plot

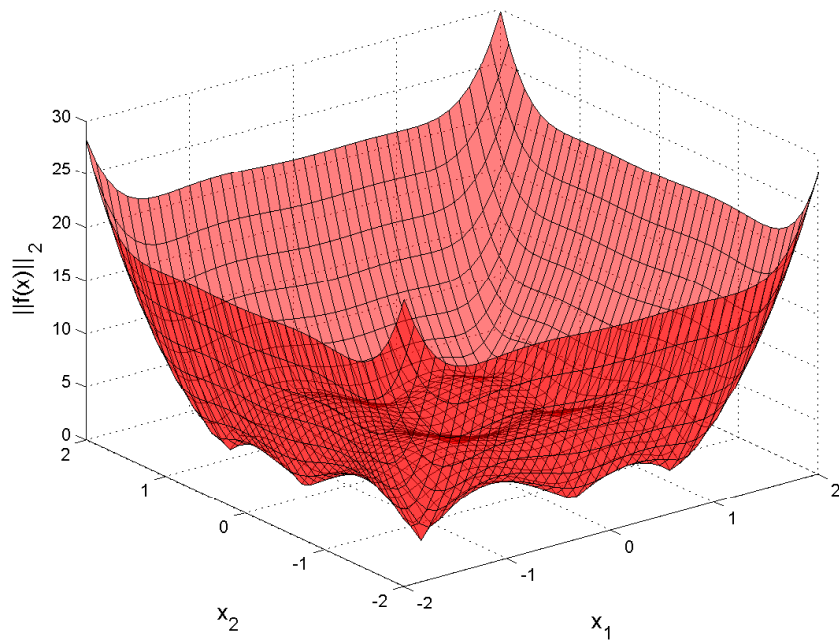


Figure 4.38: D2 Gradient ℓ_2 Norm Plot

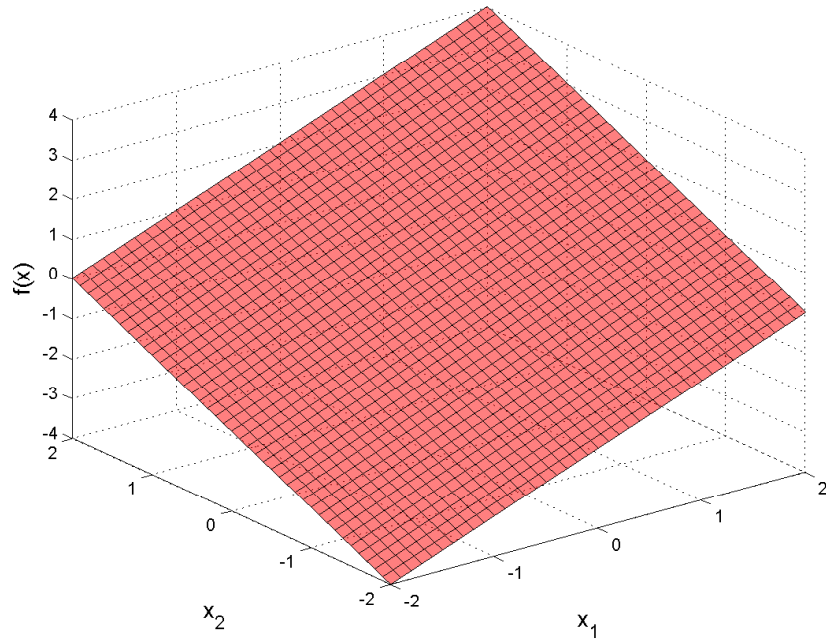


Figure 4.39: D3 Objective Function Plot

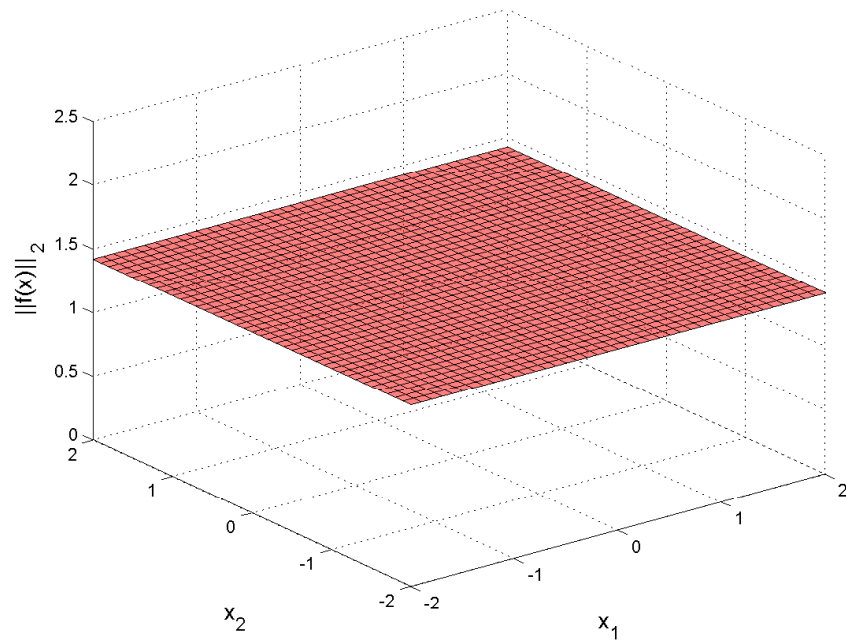


Figure 4.40: D3 Gradient ℓ_2 Norm Plot

	D1	D2	D3
x_1^*	0	-1.3	-2
x_2^*	0	-1.3	-2
f_i^*	0.00	-7.03	-4.00
$f_i(\mathbf{x}_1^*)$	—	0.00	0.00
$f_i(\mathbf{x}_2^*)$	3.38	—	-2.60
$f_i(\mathbf{x}_3^*)$	8.00	4.00	—

Table 4.5: Mathematical Example Discipline Analyzers Output

4.8.2 Discipline Fuzzy Logic System

The outputs from the discipline analyzers are then passed to the discipline FLS, which implies the three models of design intent: Objective Function Value, Gradient ℓ_2 Norm Value, and Distance From Global Minimum. Using the membership functions developed in section 4.3 each discipline is analyzed with respect to the three design intent models. Figures 4.41 - 4.43 shows the membership functions created for the objective function preferences.

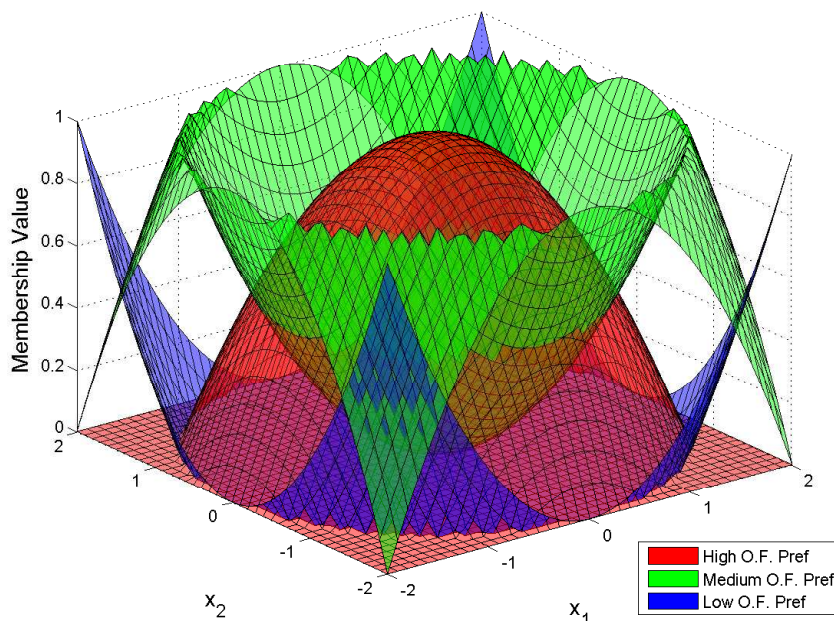


Figure 4.41: D1 Objective Function Preference Plot

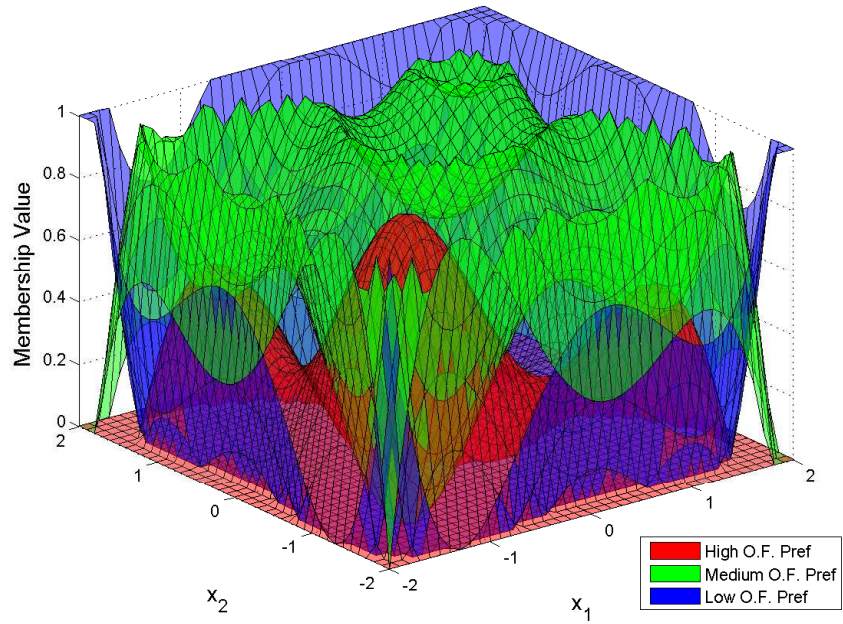


Figure 4.42: D2 Objective Function Preference Plot

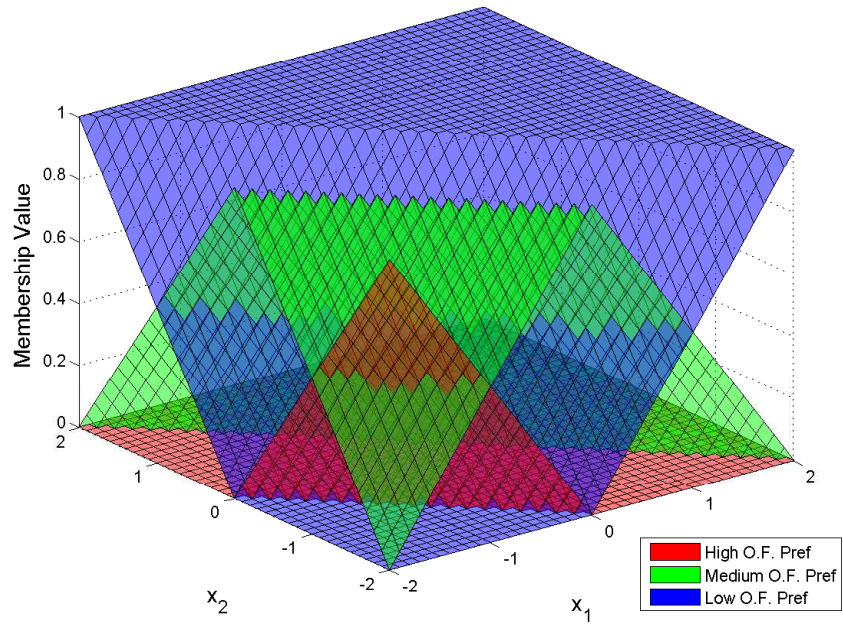


Figure 4.43: D3 Objective Function Preference Plot

In each plot, at the location of the true global minimum, the “High Objective Function Preference” surface is at its maximum value. As the objective function values increase, the preferences at the DV locations decline from high to low. In D3

(figure 4.43), the use of equation (4.1) to find the function maximum is evident and any point that satisfies $0 \leq x_1 + x_2$ is given a low preference. These points correspond to dominated solutions.

The second area of design intent is the value of the ℓ_2 gradient norm. The results of the equations developed in section 4.3.2 are evaluated over the design space as presented in figures 4.44 - 4.46. These figures indicate that at all local minima and maxima the “High Gradient Norm Preference” surface reaches a preference value of one, and preferences are lowered as the gradient norm increases. Discipline 3 has a constant gradient leading to a constant preference over the design space.

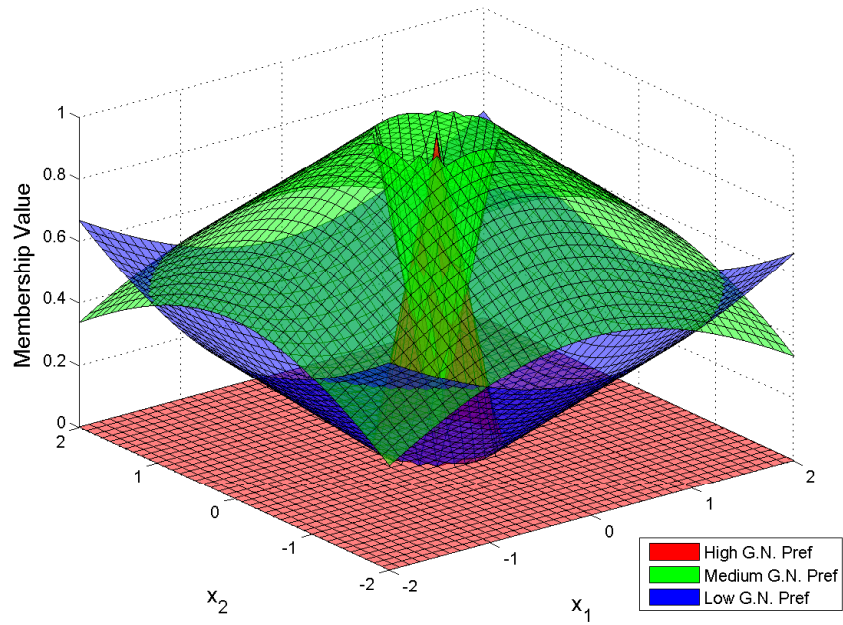


Figure 4.44: D1 $\|\nabla_{\mathbf{x}}f\|_2$ Preference

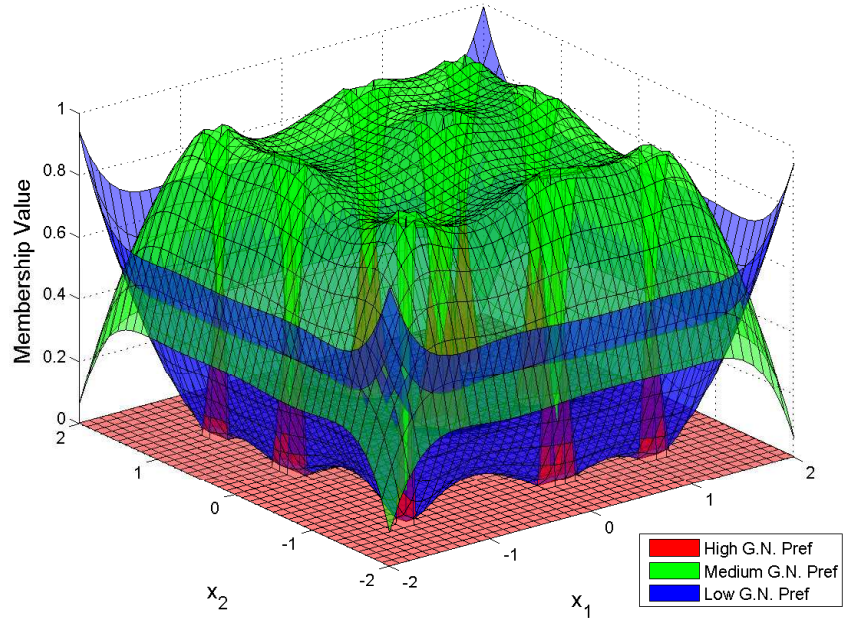


Figure 4.45: D2 $\|\nabla_{\mathbf{x}} f\|_2$ Preference

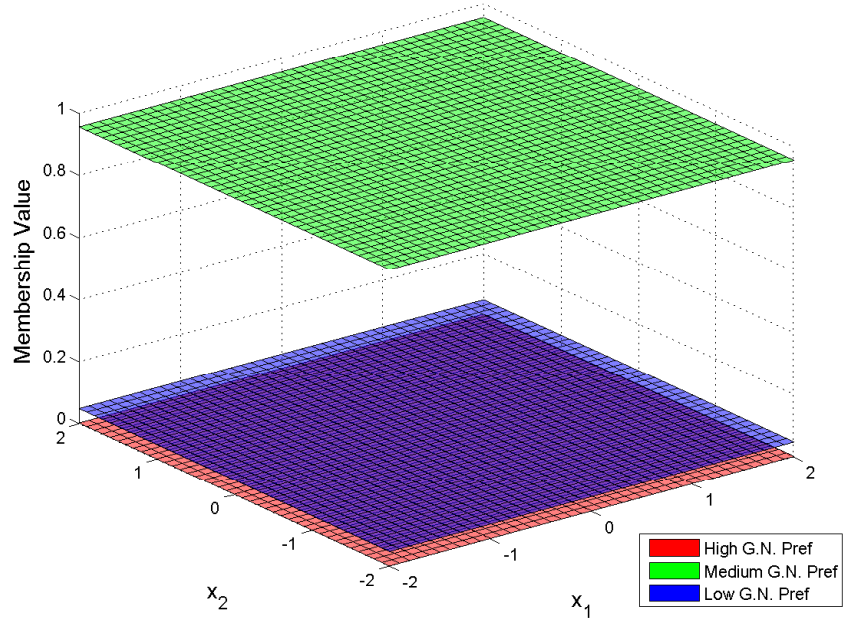


Figure 4.46: D3 $\|\nabla_{\mathbf{x}} f\|_2$ Preference

The final area of design intent for the general mathematic disciplines in this example problem is the distance from the disciplines true optimum. The locations of the optima can be found in table 4.5. Figures 4.47 - 4.49 shows the membership values

over the design space resulting from evaluation of the functions in section 4.3.3. For the discipline distance from optimum preference, the objective function value at the location does not have an effect on the preference. Preferences decline linearly as the DV location moves away from the discipline's optimum with no dependence on the objective function value.

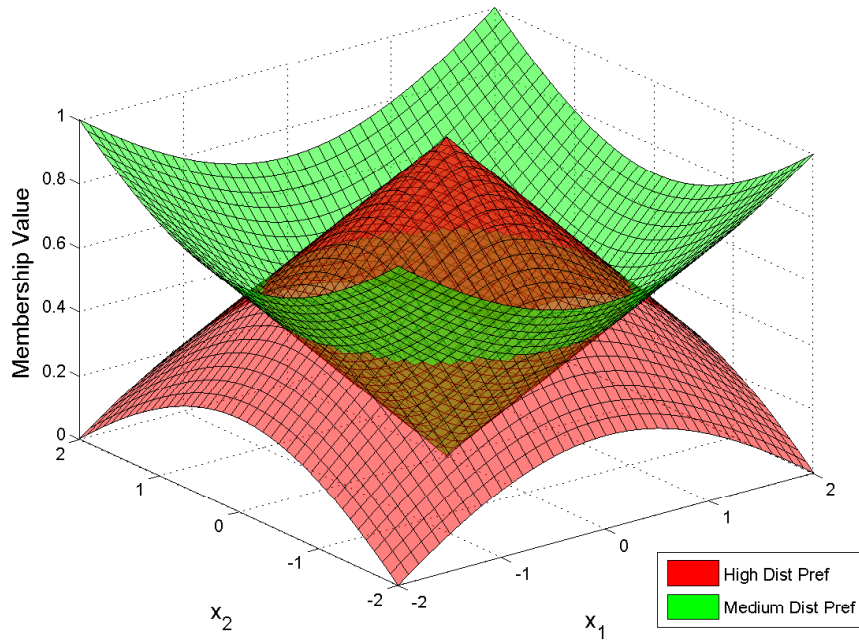


Figure 4.47: D1 Distance From Optimum Preference

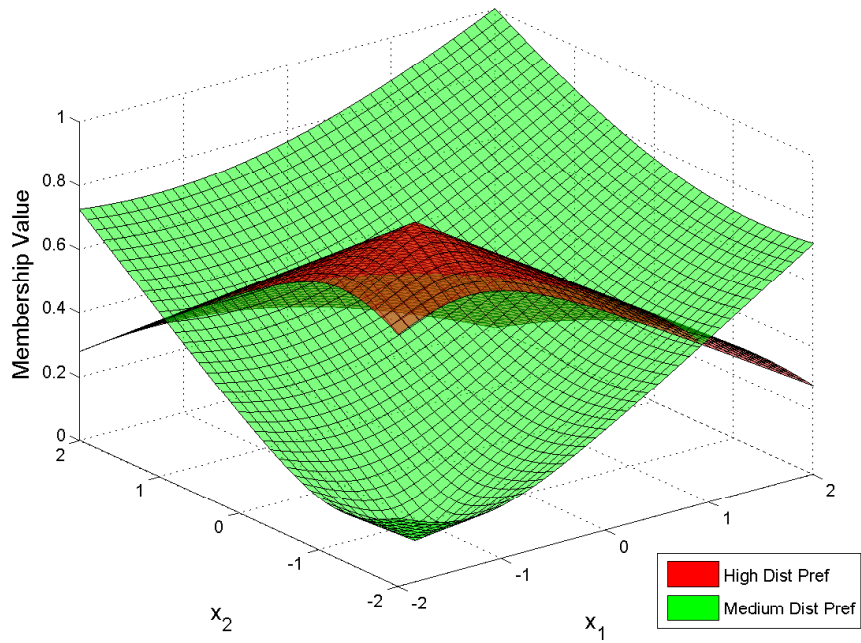


Figure 4.48: D2 Distance From Optimum Preference

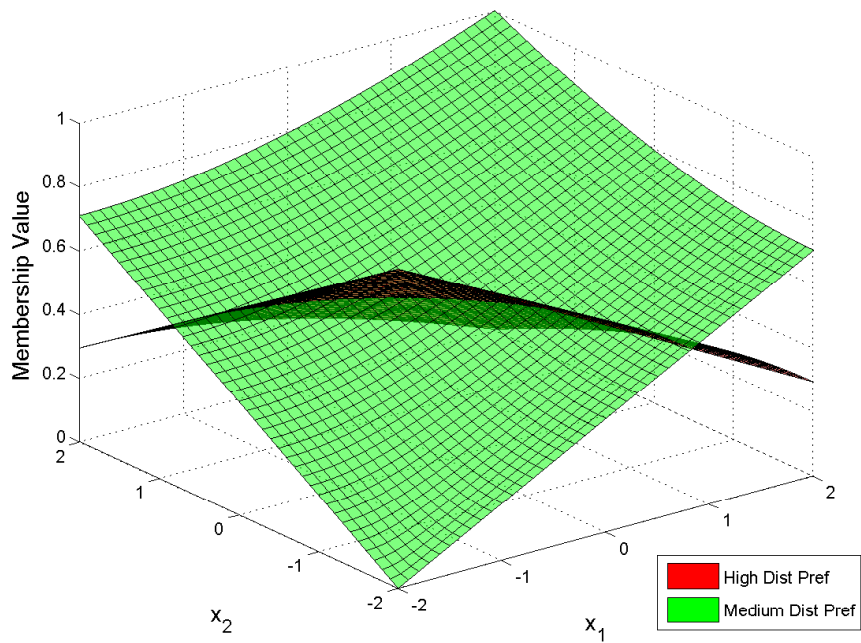


Figure 4.49: D3 Distance From Optimum Preference

After the input membership functions for each design intent model are created, the fuzzification interface assigns each DV location preferences, as described in sec-

tion 4.4.1. This information is then passed to the fuzzy rule bank, which applies the fuzzy rules developed for the problem being addressed. This example uses the fuzzy rules described in section 4.4.2 with discipline 1 utilizing a rule bank similar to what is described in section 4.7 due to the inclusion of a constraint. The fuzzy sets for the constraint are in figure 4.50. Because a crisp constraint is used, there is no transition area. As soon as $g_1(x)$ becomes greater than zero, the input DV values become a member of the constraint violated fuzzy set.

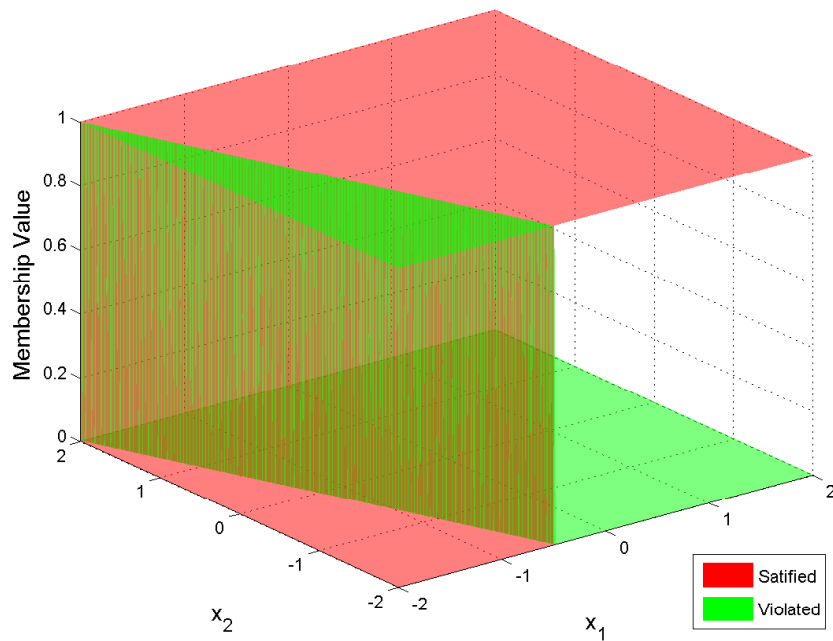


Figure 4.50: Mathematical Example Constraint Fuzzy Sets Plot

Each discipline is run through its respective FLS to get the output preferences over the design space. Figures 4.51 - 4.53 presents the output of each discipline FLS. These figures display how the FLSs highlight areas of the design space based on the design intent rules decided on before running the process.

Figure 4.51 also highlights the effect the constraint applied to D1 has on the output preference. The area where the constraint is satisfied is unaffected, but, as soon as the constraint is violated the output preference drops to zero, or fully a member of the “Unpreferred” output fuzzy set in figure 4.17.

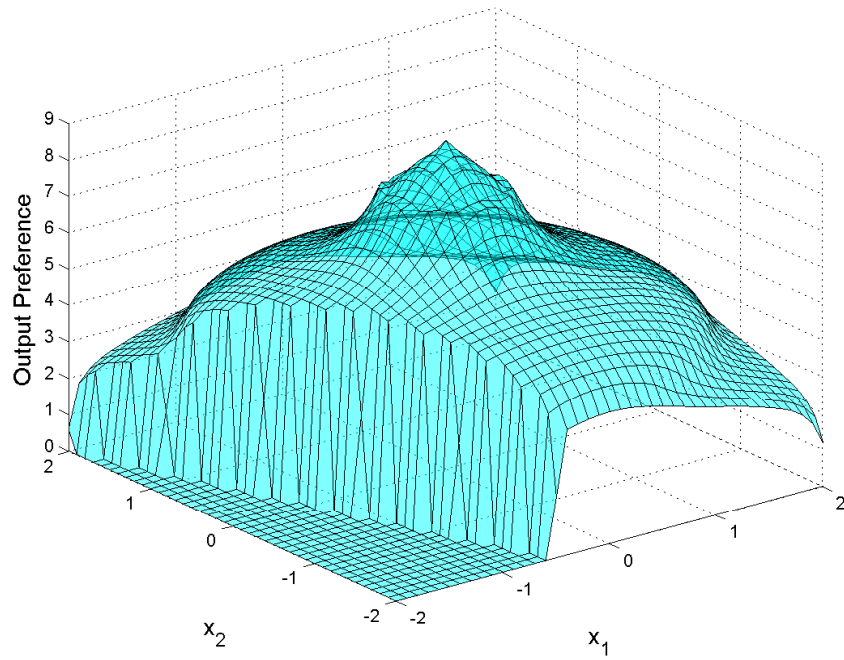


Figure 4.51: D1 FLS Output

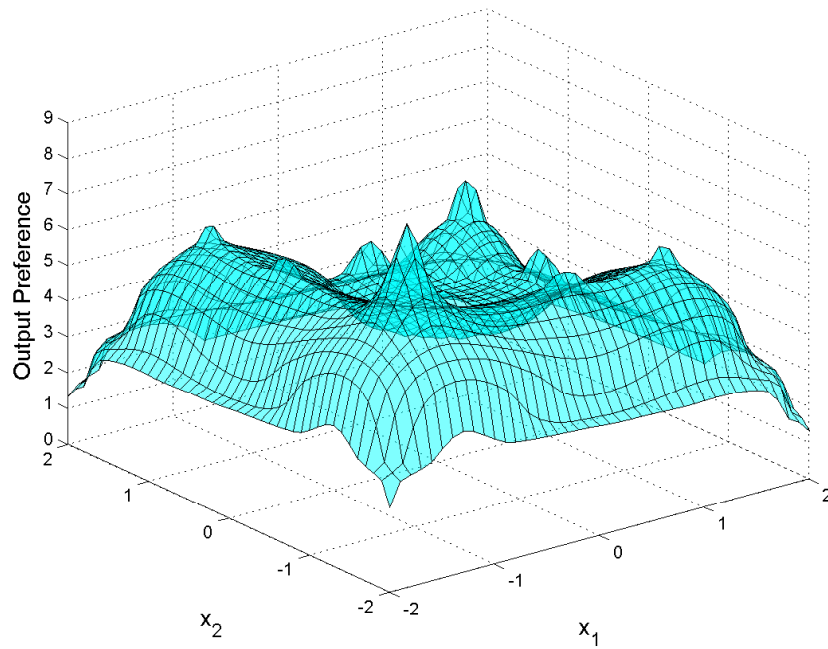


Figure 4.52: D2 FLS Output

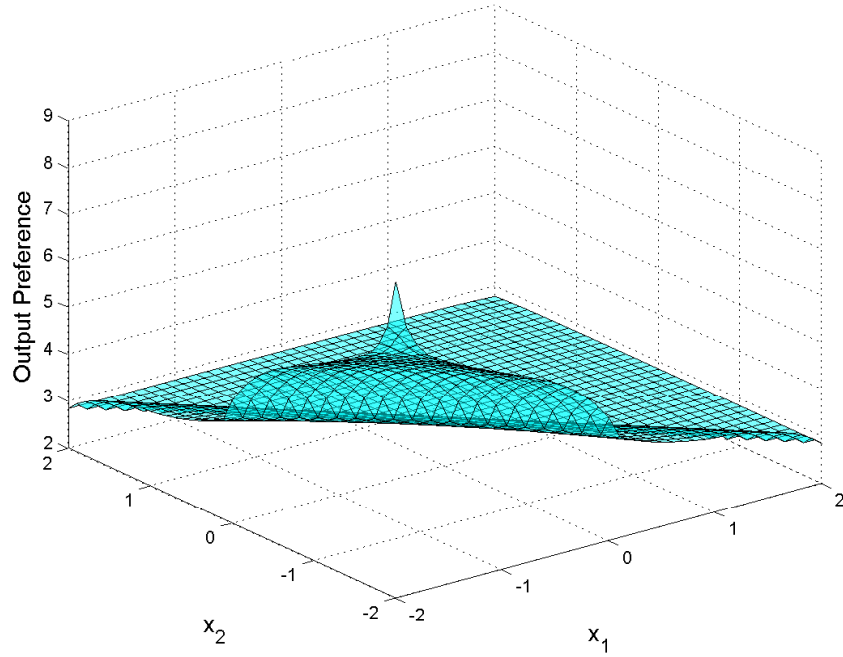


Figure 4.53: D3 FLS Output

4.8.3 Fusion Logic

When all disciplines have been analyzed by their respective FLS, the output data is combined by the fusion logic to create global preferences. The output from each discipline is combined using equation (3.7), where the discipline with the lowest preference at each DV value is the output preference from the Fusion Logic. Figures 4.54 and 4.55 show the plot of the fusion logic output over the design space, and figure 4.56 is the contour plot of the same data.

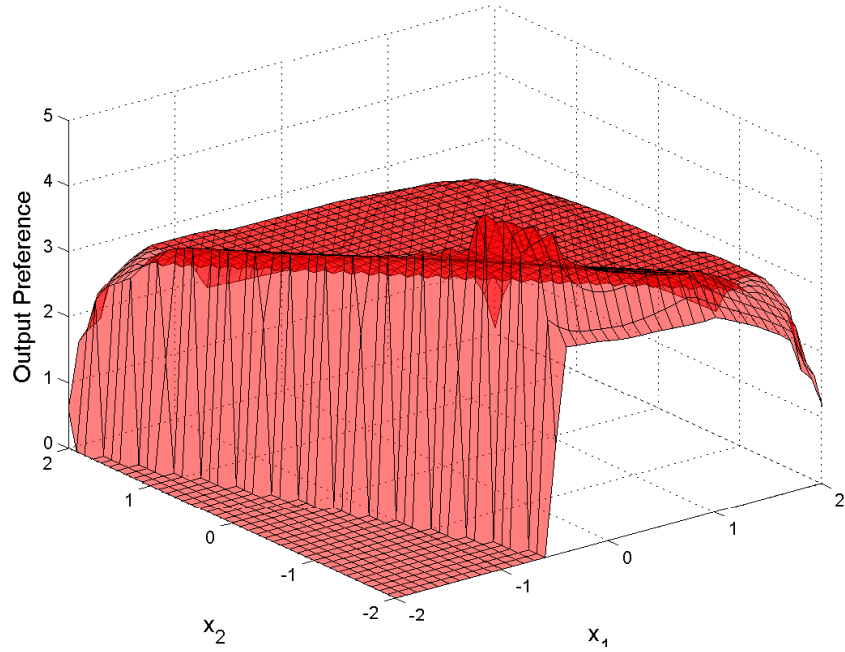


Figure 4.54: Fusion Logic Output Plot

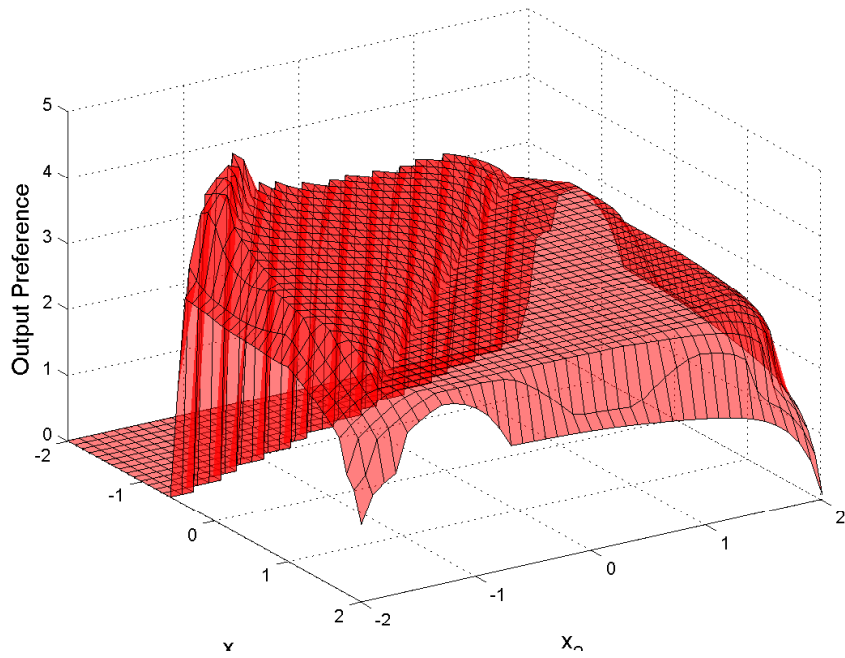


Figure 4.55: Fusion Logic Output Plot Rotated 90 Degrees

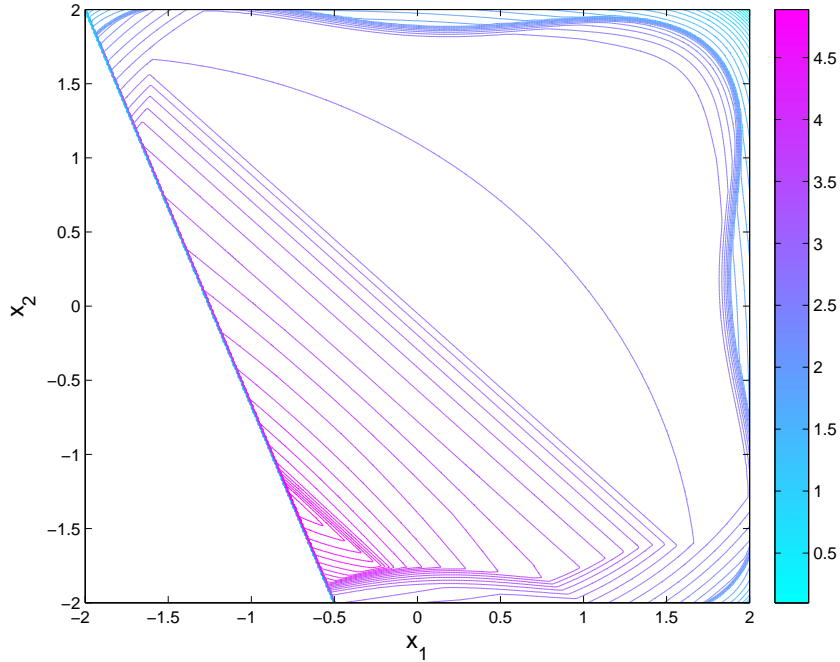


Figure 4.56: Fusion Logic Output Contour Plot

The figures display the effects of the constraint on the output from the fusion logic when compared to the unconstrained fusion logic in figure 4.27. The constraint on the single discipline removes the area of highest preference from the design space because if one discipline becomes infeasible then the output of the fusion logic for the entire system becomes unpreferred. The areas of the design space that are feasible in this example are unchanged from the unconstrained fusion logic output displayed in figure 4.27 and figure 4.29.

4.8.4 System Level Decision Maker

The global preference levels outputted from the fusion logic are passed to the system level decision maker, which analyzes the resultant to select the optimal point. The optimal point is chosen based on the DV value with the highest global preference. For this example, the highest combined preference is 4.98. This maximum value occurs at the point $\mathbf{x} = (-0.75, -1.33)$. At this point, the disciplines objective functions

are $f_1((-0.75, -1.33)) = 2.33$, $f_2((-0.75, -1.33)) = -5.63$, and $f_3((-0.75, -1.33)) = -2.08$.

4.8.5 Results Analysis

The results of the example problem are compared to other methods of MDO. Table 4.6 shows the optimal location and objective function values for the HFLC MDO, as well as other methods of MDO. “MDO target” is outlined in [28] and section 2.3.5.4, “MDO MLS” is outlined in [35] and section 2.3.5.5.

	HFLCMDO	MDO Target	MDO MLS
x_1^*	-0.75	-0.87	-0.78
x_2^*	-1.33	-1.02	-1.25
$f_1(\mathbf{x}^*)$	2.33	1.80	2.17
$f_2(\mathbf{x}^*)$	-5.63	-5.63	-5.73
$f_3(\mathbf{x}^*)$	-2.08	-1.89	-2.03

Table 4.6: Mathematical Example Results Comparison

The results of the MDO example are presented in figure 4.57 and 4.58. Figure 4.57 shows all tested feasible points with the Pareto frontier highlighted. Each value is plotted based on the its $f_1(\mathbf{x})$, $f_2(\mathbf{x})$, and $f_3(\mathbf{x})$ at each \mathbf{x} value tested. The design space for each discipline was tested over the DV ranges $x_1 \in [-2, 2]$ and $x_2 \in [-2, 2]$ with a point spacing for each DV of 0.1. Each of the non-mathematically dominated points of the Pareto front are highlighted. The location of the utopia point is also highlighted on figure 4.57.

Figure 4.58 represents the Pareto front of the mathematical example problem. Each of the designs in table 4.6 are highlighted with all designs falling on the Pareto front.

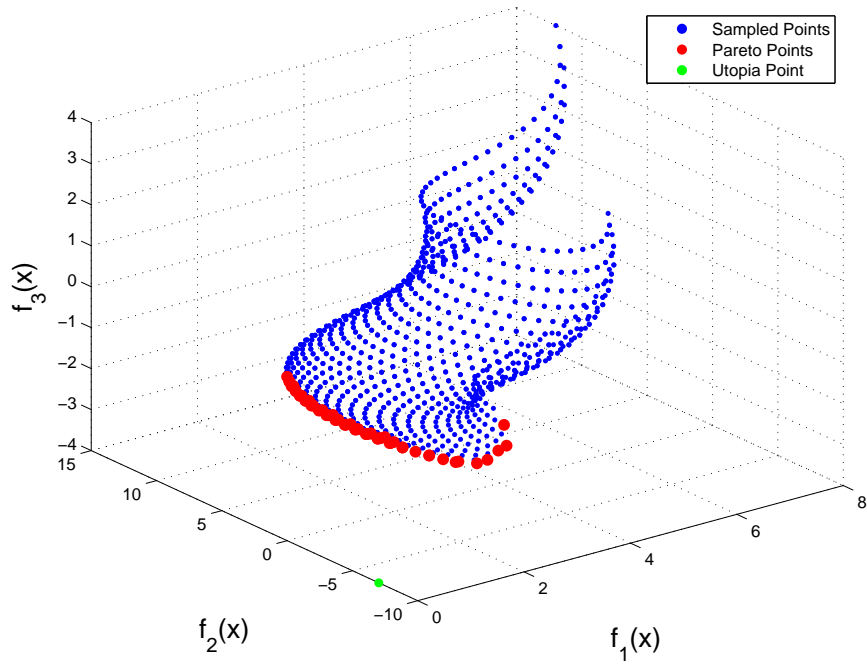


Figure 4.57: Plot of Sampled Points Highlighting Pareto Front

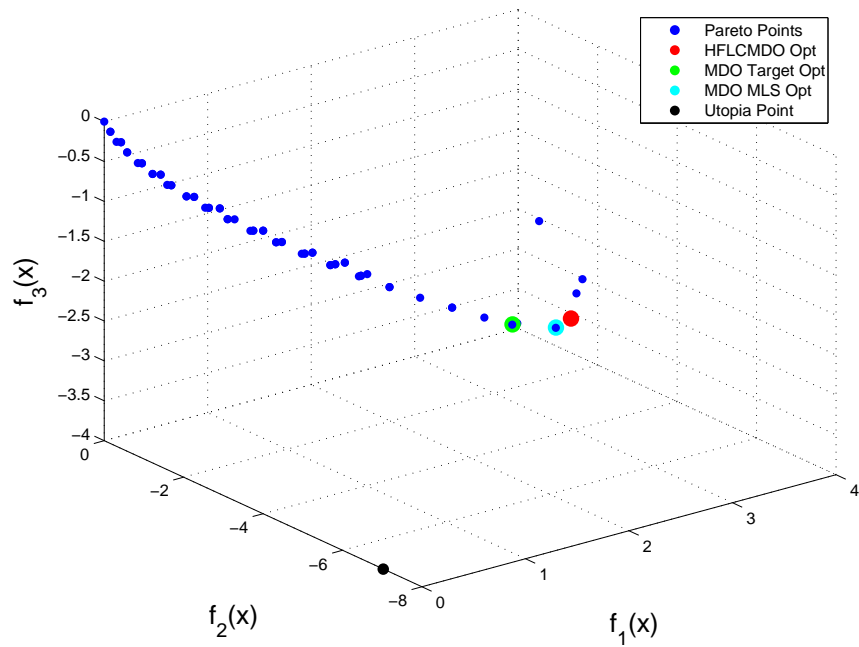


Figure 4.58: Example Problem Pareto Front with MDO Results

For a quick comparison of the results, the distance to the utopia point (DTU) of each optimal result was calculated using equation (4.22). Because all the optimal

points fell on the Pareto front, none of the solutions can be mathematically proven superior to any others. However, this does provide a metric to demonstrate that the HFLC MDO method developed preforms well when compared to other algorithms.

$$DTU = \sqrt{\sum_{i=1}^I (f_i^* - f_i(\mathbf{x}))} \quad (4.22)$$

Method	DTU
HFLCMDO	3.32
MDO Target	3.11
MDO MLS	3.20

Table 4.7: MDO Method Results Comparison

Figure 4.59 displays all feasible sampled points distance to utopia. The results of the different MDO formulations are highlighted. Each of the final designs from the MDO problems have distances to the utopia point which are comparable. All MDO values are in the 99th percentile in terms of the distance to utopia metric when compared to all other sampled points.

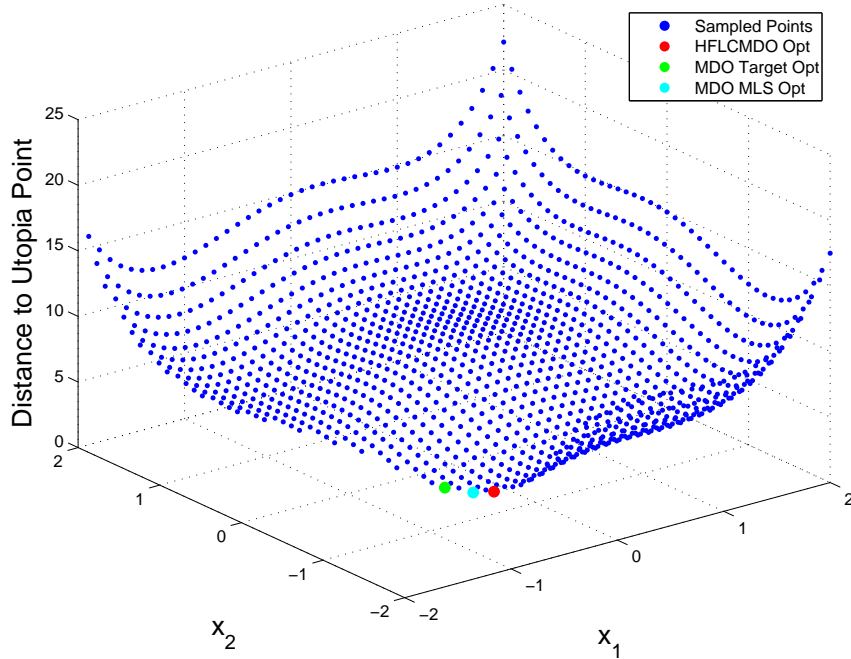


Figure 4.59: Mathematical Example Distance to Utopia Plot

This example demonstrated how the presented MDO method can be adapted for a mathematical problem. The optimal point selected by the HFLC MDO landed on the Pareto front of the feasible designs. This result verifies that if the design intent models are create correctly, the HFLC MDO method produces optimal results comparable to other methods of MDO. The results of the HFCL MDO test case demonstrate the method of building designer intent into the optimizer and using HFLC methods to blend the outputs of disciplines locates optimal designs.

4.9 Chapter Summary

The framework for the HFLC MDO design method was described using an example problem. Areas of design intent were first developed and used to examine the disciplines. Using the fuzzy logic system, the different design intent rules were combined to create discipline preferences for the multiple disciplines in the example problem. The discipline preferences were then combined using the fusion logic of a

hierarchical fuzzy logic controller to create system level preferences for the design space.

When the method was fully developed, the HFLC MDO method was used to optimize an example problem. The optimal point selected by the method presented landed on the Pareto front of the design alternatives. The method also compared favorably with other methods of MDO. This demonstrates that if the design intent models are created within the right context, the HFLC MDO will produce optimal designs.

CHAPTER V

Ship Design Case Study

5.1 Introduction

A ship design case study has been developed to demonstrate how the previously described algorithm can include designer intent into a MDO optimizer. A ship hull will be designed subject to two disciplines: seakeeping and resistance. The two disciplines will each have their own fuzzy logic system designed specifically for the individual discipline and the understanding of the models that are used in analysis.

5.2 Initial Design Point

The hull is simplified to three design variables: the ship's waterline length (L), the ship's beam (B) and the ship's draft (T). The initial ship hull is based on a very large crude carrier (VLCC) tanker that is available through the commercial ship design software MaxSurf [75]. The initial design point uses the average values for a Suezmax tanker found in [55]. The bounds of the design variables were selected based on the minimum and maximum values for a Suezmax tanker [55].

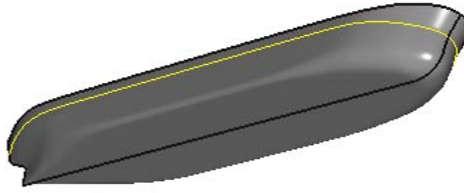


Figure 5.1: Parent Hull

Designer Variable	Units	Value
L	[m]	260.8
B	[m]	45.8
T	[m]	15.9

Table 5.1: Initial Design Variable Values

The actual hull geometry is not used in the analysis, instead the original hull is used to estimate parameters that are modified by the design variables. Figure 5.1 displays the initial hullform used in the design process and the initial design variable values are displayed in table 5.1. The displacement (Δ) of the vessel is set as a constraint in the optimization problem, with all ship designs having a required displacement of 162,000 tonnes.

5.3 Seakeeping Discipline

5.3.1 Seakeeping Analysis Tool

For the seakeeping discipline, the “Seakeeping Prediction Program” (SPP), developed by Parsons at the University of Michigan [61], is used. The program is based on a five degree of freedom seakeeping analysis algorithm written by Raff and Kaplan [65].

SPP requires the most hull information of the analysis tools used. The inputs include: the design variable L, the ship’s vertical center of gravity (VCG) which is

simplified to be 110% of T (equation (5.1)), and the ship's roll radius of gyration (k_{11}) which is estimated based on the design variables L , B , and T using equation (5.2) from Parsons [58].

$$VCG = 1.1 * T \quad (5.1)$$

$$k_{11} = 0.50 * \left(0.746 + 0.046 \frac{B}{T} - 0.086 \left(\frac{L}{100} \right) \right) * B \quad (5.2)$$

SPP also requires hull station information. For these inputs the hull lines drawing, displayed in figure 5.2, is used as a reference to define relationships based on the design variables (DV). The first aspect of the ship that must be defined at each station is the ship draft. Because of the long parallel mid-body of the parent hull, the drafts at stations 1 through 9 are equal to the design variable T . At station 0, the draft is set to be zero. At station 10, the draft is set to be 10% of T based on the parent hull.

The next values that are necessary for inputs into SPP are the hull beams and the hull section areas at station. These values are found using a spreadsheet developed by Parsons at University of Michigan and based on [59] and [77]. This spreadsheet estimates the beam and section area at each station based on the ship's L , B , T , underwater volume (∇), block coefficient (C_B) found using equation (5.3), the mid-ship coefficient (C_M) which is assumed constant with a value of 0.996, the prismatic coefficient (C_P) found using equation (5.4), and the waterplane coefficient (C_{WP}) estimated using equation (5.5) from Parsons [58]

$$C_b = \frac{\nabla}{L * B * T} \quad (5.3)$$

$$C_p = \frac{C_b}{C_M} \quad (5.4)$$

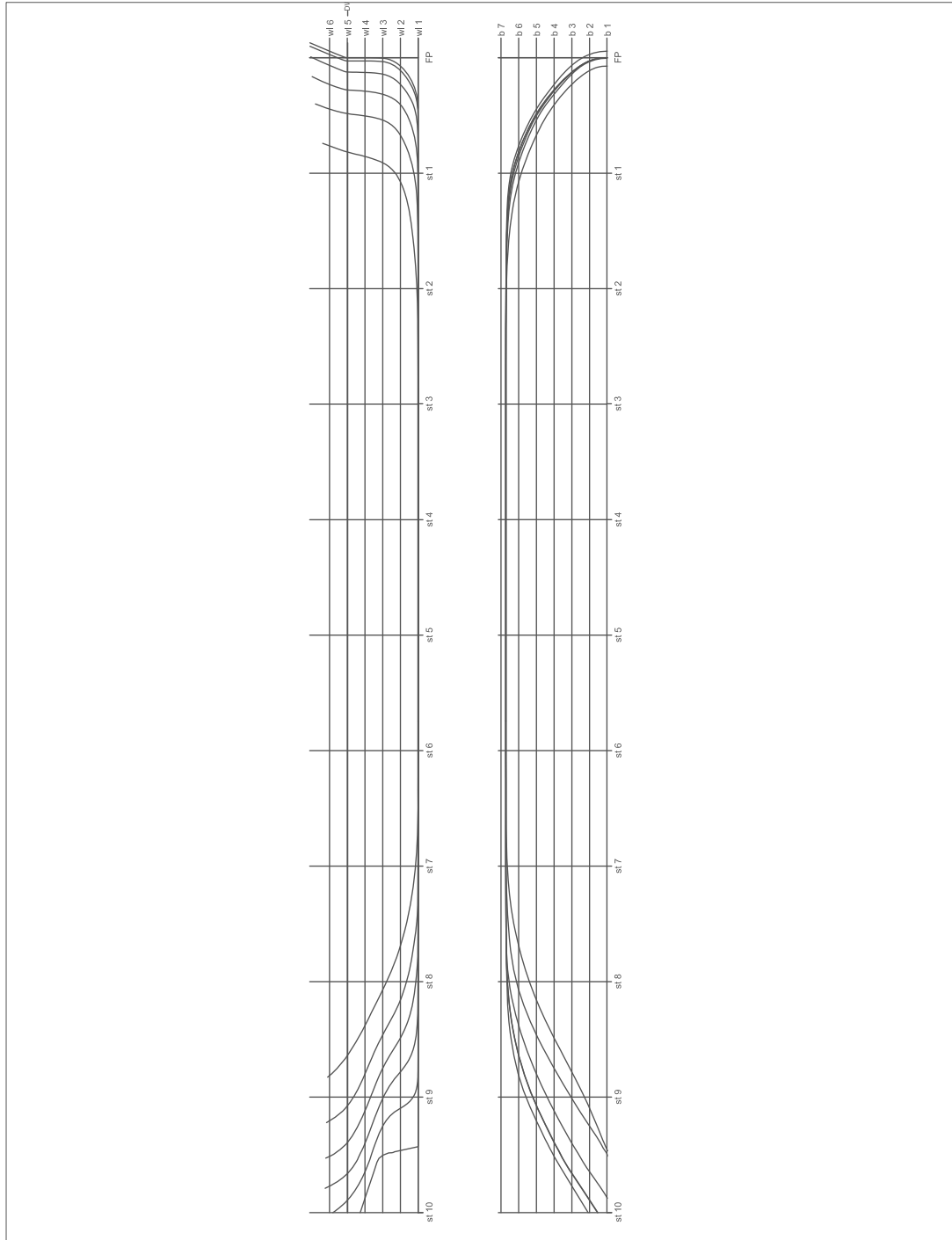


Figure 5.2: Parent Hull Lines Drawing

$$C_{wp} = \frac{C_b}{0.471 + 0.551 * C_b} \quad (5.5)$$

The final input to SPP is the ship's weight at each station. The weight at each station is found using a constant percentage of the total displacement. The percentage of weight at each station leads to a longitudinal center of gravity (LCG) at midship. Evaluation of SPP requires the ship LCB to be within 0.5% of the ship L to the ship LCG. This leads to two constraints on the design shown in equation (5.6) and (5.7). The LCB is the centroid of the underwater volume, found using the section areas as described in [89].

$$LCG - (LCB + (0.005L)) \leq 0 \quad (5.6)$$

$$(LCB - (0.005L)) - LCG \leq 0 \quad (5.7)$$

SPP calculates the ship motions based on the ship speed, heading with respect to incoming waves, and the sea state. The ship speed is a constant at the design speed (V_k) of 15 knots. The heading chosen is at an angle of 45 degrees. This heading is used so that the coupling effects of the three ship motions could be examined. The sea state for the optimization run is based on the two input ITTC spectrum with a selected wave height of 5 meters and a period of 10.8 seconds.

SPP analyzes the ship seakeeping characteristics and outputs the ship natural frequencies, motion response amplitude operator (ROA), moment ROA, and wave input amplitude spectra. Statistics of the amplitude responses are also output. A full output of SPP is found in appendix A.

The seakeeping discipline in this case study is dependent on the ship accelerations. The ship accelerations were calculated using the ship ROAs and the wave input spectra. The method to calculate the accelerations from the output of SPP is outlined

in [45]. The variance of the ship accelerations is found using equation (5.8). Where, $m_{4,j}$ is the variance of the acceleration for each ship motion j , ω_e is the encounter frequency, and S_j is the input amplitude spectra for the j th motion response.

$$m_{4,j} = \int_0^{\infty} \omega_e^4 * S_j(\omega_e) d\omega_e \quad (5.8)$$

Equation (5.8) is evaluated using Simpson's rule for integration, and the motion response spectra outputs from SPP. The root mean square (RMS) accelerations of the three motions, heave, pitch, and roll, are used to find the discipline preferences for the seakeeping discipline. The RMS acceleration for each motion is calculated using equation (5.9), where $RMS_{acc,j}$ is the RMS acceleration for response j .

$$RMS_{acc,j} = \sqrt{m_{4,j}} \quad (5.9)$$

The three motions of the ship are sampled at the ship LCG. This implies that only the heave accelerations are considered as vertical accelerations, and the roll and pitch motions are purely rotational accelerations.

5.3.2 Seakeeping Design Intent

When considering seakeeping optimization, many different objective functions have been used. Most optimization algorithms attempt to minimize the displacements or accelerations of the various degrees of freedoms [28, 27, 63, 19].

For this case study, the same data that is available in standard optimization is processed through the fuzzy logic system with the intent of minimizing the effects of seasickness on the crew of the ship. The data on ship motions is readily available, but relating that information to seasickness effects does not have clear formulas that are easily adapted to traditional optimization techniques.

One major issue in the analysis of seakeeping motions with respect to seasickness

is the lack of hard correlation between vertical motions and rotational motion. Studies have been conducted on the effects of individual motions on seasickness [49], indicating vertical accelerations have the greatest effect on seasickness. However, there is very little data available on what happens when motions are coupled.

A guideline for the effects of vertical acceleration on seasickness is found in figure 5.3. This gives guidelines for exposure limits based on the frequency and vertical accelerations experienced.

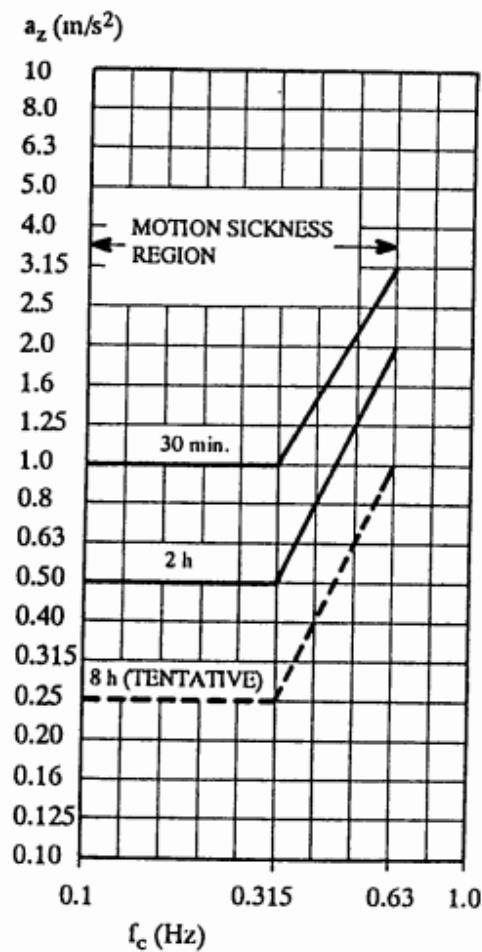


Figure 5.3: ISO 2631/3 Severe Discomfort Boundaries [1]

Decoupled pitch and roll exhibit very little effect on seasickness as shown by Mc-

Cauley et al [49]. When the rotational motions were coupled with heave motions, the rate of seasickness increased compared to a decoupled heave motion [80]. Though there is no standard model for how coupled motions affect seasickness, this is a consideration that should be taken into account when the motions of a ship are being analyzed and optimized. A fuzzy logic system can provide the means for this consideration.

5.3.3 Seakeeping Input Membership Functions

Based on the output of SPP, membership values are derived to analyze the ship motions with respect to the effects of seasickness. The heave RMS acceleration is evaluated with respect to the ISO guidelines shown in figure 5.3. The membership functions that assign inputs to the heave RMS acceleration fuzzy sets are formulated in equations (5.10)-(5.12), where μ_{heave}^{high} is the membership value in the “High Heave Preference” fuzzy set, μ_{heave}^{med} is the membership value in the “Medium Heave Preference” fuzzy set, μ_{heave}^{low} is the membership value in the “Low Heave Preference” fuzzy set, and $f_{heave}(\mathbf{x})$ is the RMS heave acceleration (in m/s^2) output from SPP based on the current DV values where $\mathbf{x} = [L, B, T]$. The membership functions are displayed with the ISO seasickness guidelines in figure 5.3. In figure 5.3, the three membership functions correspond to the left y-axis, and the ISO regulations correspond to the right y-axis. As with most FLS, the membership functions are subjective and tuning may be necessary.

$$\mu_{heave}^{high}(f_{heave}(\mathbf{x})) = \begin{cases} -2f_{heave}(\mathbf{x}) + 1 & 0 \leq f_{heave}(\mathbf{x}) \leq 0.5 \\ 0 & f_{heave}(\mathbf{x}) > 0.5 \end{cases} \quad (5.10)$$

$$\mu_{heave}^{med}(f_{heave}(\mathbf{x})) = \begin{cases} 2f_{heave}(\mathbf{x}) & 0 \leq f_{heave}(\mathbf{x}) \leq 0.5 \\ -2f_{heave}(\mathbf{x}) + 2 & 0.5 < f_{heave}(\mathbf{x}) \leq 1.0 \\ 0 & f_{heave}(\mathbf{x}) > 1.0 \end{cases} \quad (5.11)$$

$$\mu_{heave}^{low}(f_{heave}(\mathbf{x})) = \begin{cases} 0 & 0 \leq f_{heave}(\mathbf{x}) \leq 0.5 \\ 2f_{heave}(\mathbf{x}) - 1 & 0.5 < f_{heave}(\mathbf{x}) \leq 1.0 \\ 1 & f_{heave}(\mathbf{x}) > 1.0 \end{cases} \quad (5.12)$$

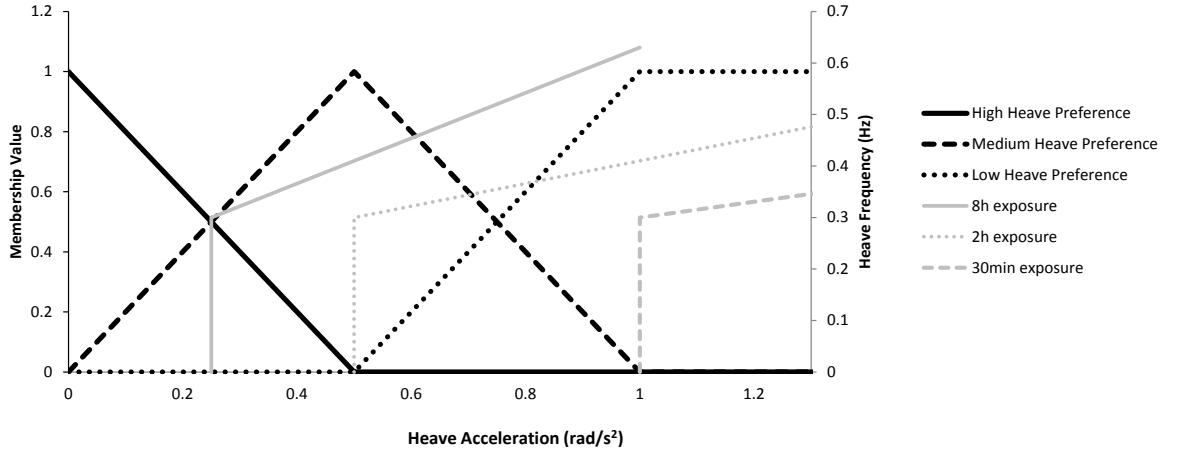


Figure 5.4: Heave Acceleration Fuzzy Sets

For this case study, the heave accelerations were based on the guidelines where the two hour exposure limit is the maximum acceptable heave to be in the “High Heave Preference” fuzzy set. Any acceleration over the 30 minute exposure limit is fully a member of the “Low Heave Preference” fuzzy set.

Although there are no hard guidelines for coupled heave with pitch and roll, the motions used in experiments by Wertheim et al [80] are applied to assign the acceler-

ation from pitch and roll motions to fuzzy sets. Based on the lab test, the rotational motion membership functions transition linearly from high to low preference using the test acceleration as the 50% high 50% low preferences. In equations (5.13) and (5.14), μ_{pitch}^{high} assigns pitch acceleration values to the “High Pitch Preference” fuzzy set, μ_{pitch}^{low} assigns pitch acceleration values to the “Low Pitch Preference” fuzzy set, and $f_{pitch}(\mathbf{x})$ is the pitch RMS acceleration (in *radians/s²*) found from SPP. The membership functions for pitch acceleration are presented in figure 5.5.

$$\mu_{pitch}^{high}(f_{pitch}(\mathbf{x})) = \begin{cases} -\frac{2}{3}f_{pitch}(\mathbf{x}) + 1 & 0 \leq f_{pitch}(\mathbf{x}) \leq 1.5 \\ 0 & f_{pitch}(\mathbf{x}) > 1.5 \end{cases} \quad (5.13)$$

$$\mu_{pitch}^{low}(f_{pitch}(\mathbf{x})) = \begin{cases} \frac{2}{3}f_{pitch}(\mathbf{x}) & 0 \leq f_{pitch}(\mathbf{x}) \leq 1.5 \\ 1 & f_{pitch}(\mathbf{x}) > 1.5 \end{cases} \quad (5.14)$$

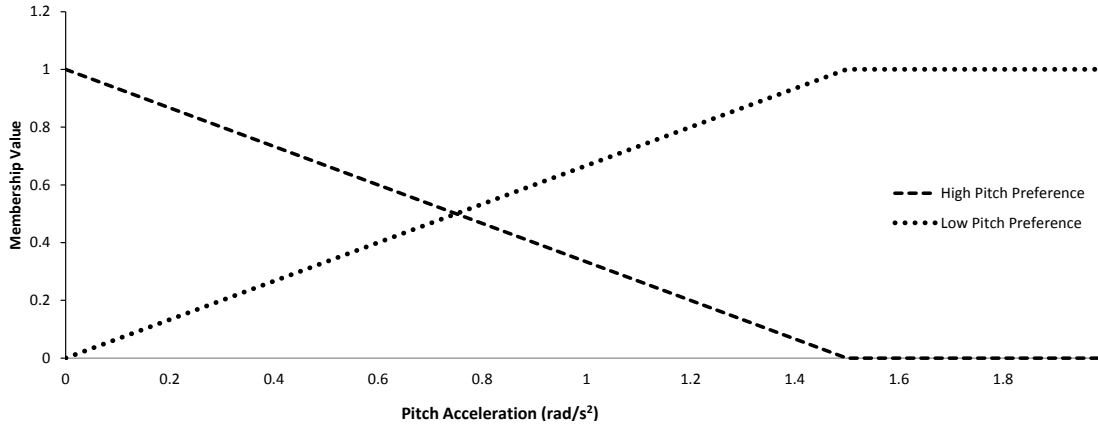


Figure 5.5: Pitch Acceleration Fuzzy Sets

In equations (5.15) and (5.16), μ_{roll}^{high} assigns roll acceleration values to the “High Roll Preference” fuzzy set, μ_{roll}^{low} assigns roll acceleration values to the “Low Roll Preference” fuzzy set, and $f_{roll}(\mathbf{x})$ is the roll RMS acceleration (in *radians/s²*) found from SPP. The membership functions for roll acceleration are shown in figure 5.6.

$$\mu_{roll}^{high}(f_{roll}(\mathbf{x})) = \begin{cases} -\frac{2}{3}f_{roll}(\mathbf{x}) + 1 & 0 \leq f_{roll}(\mathbf{x}) \leq 1.5 \\ 0 & f_{roll}(\mathbf{x}) > 1.5 \end{cases} \quad (5.15)$$

$$\mu_{roll}^{low}(f_{roll}(\mathbf{x})) = \begin{cases} \frac{2}{3}f_{roll}(\mathbf{x}) & 0 \leq f_{roll}(\mathbf{x}) \leq 1.5 \\ 1 & f_{roll}(\mathbf{x}) > 1.5 \end{cases} \quad (5.16)$$

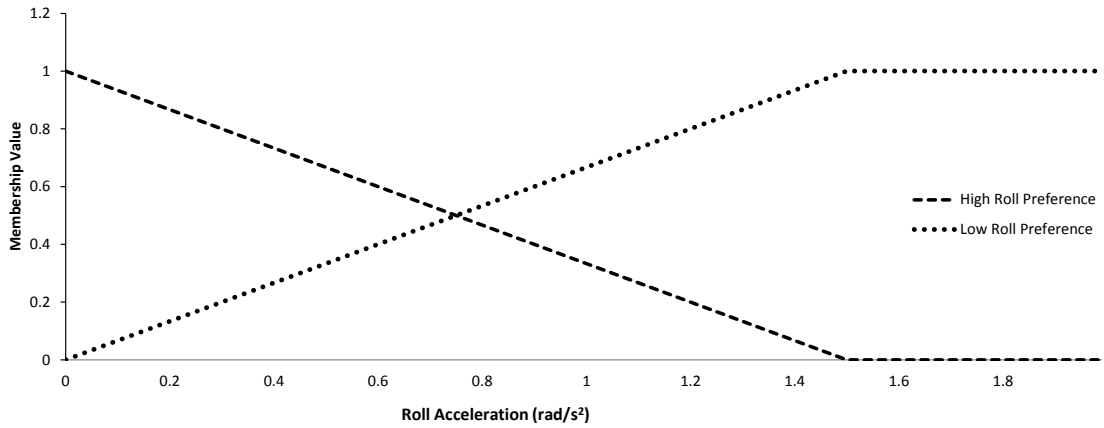


Figure 5.6: Roll Acceleration Fuzzy Sets

5.3.4 Seakeeping Fuzzy Rule Bank

The rules for the seakeeping discipline were developed from the conclusions made by [80]. The heave RMS acceleration was considered the most important component affecting seasickness, but motion coupling effects were taken into consideration. The fuzzy rule matrix is in figure 5.7. The output linguistics correspond to the fuzzy sets in figure 5.8.

High Heave Preference	High Preference	High Preference
Medium Heave Preference	Medium Preference	Low Preference
Low Heave Preference	Low Preference	Unpreferred
	High Roll or Pitch Preference	Low Roll or Pitch Preference

Figure 5.7: Seakeeping Fuzzy Rule Bank

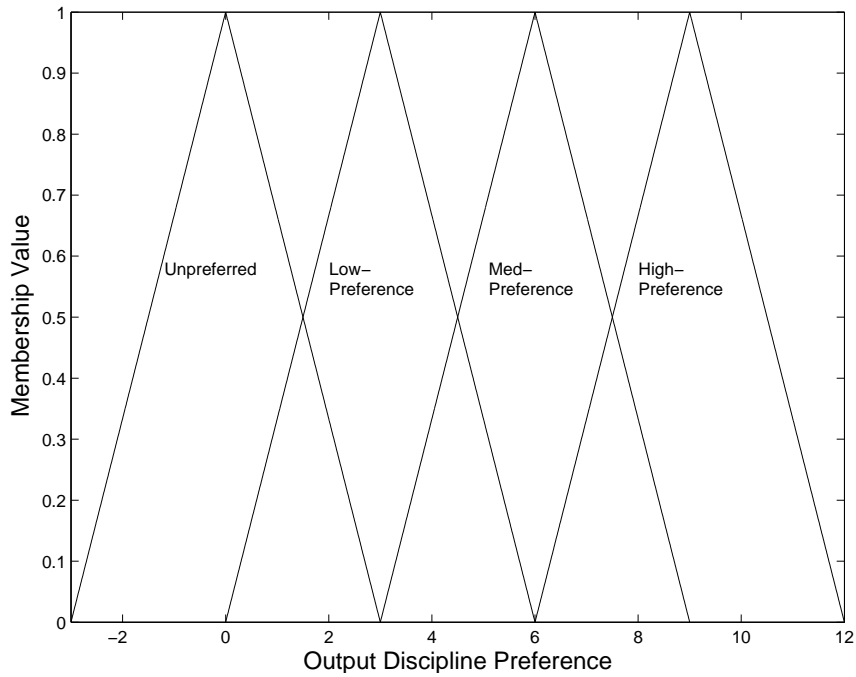


Figure 5.8: Output Preference Fuzzy Sets

The rule bank shows that if the heave RMS accelerations are very low, leading to a high preference, the output is always a high discipline preference. As the heave accelerations increase, the output preference decreases. When high accelerations of either pitch, roll, or both are coupled with heave, the output preference is downgraded.

For the seakeeping rules, the rules are in the form of equation (5.17). Here, the rules include an “OR” statement between the pitch and roll preferences. This leads

to a different method of selecting an activation level.

$$\begin{aligned}
 & \mathbf{IF} \text{ Heave is High Preference, } \mathbf{AND} \\
 & \quad (\text{Pitch is High Preference, } \mathbf{OR} \\
 & \quad \text{Roll is High Preference})
 \end{aligned} \tag{5.17}$$

THEN Output is High preference

The equation for selecting the activation level (equation (3.1)) must be modified. In FLSs when an “OR” statement is used, the maximum of the input membership values is set as the activation level. For this set of rules, equation (5.18) is used. First the maximum of the pitch and roll input membership values is calculated (corresponding to the “OR” statement). The minimum is then taken between the heave and the result from pitch and roll (corresponding to the “AND” statement).

$$\mu_{out} = \min(\mu_{\text{heave acc.}}, \max(\mu_{\text{pitch acc.}}, \mu_{\text{roll acc.}})) \tag{5.18}$$

5.4 Resistance Discipline

5.4.1 Resistance Analysis Tool

For resistance analysis, the “Power Prediction Program” (PPP) developed at the University of Michigan by Parsons [60], is used. This program calculates the ship required installed horsepower for a given range of speeds. PPP is based on Holtrop and Mennen’s method of power prediction, which uses statistical analysis of model test and full scale ship data to produce coefficients for calculations of ship resistance and powering [32]. PPP does not require any hull data points and is evaluated based only on the ship’s principal dimensions and coefficients.

The ship parameters input to PPP are the design variables (L, B, and T), C_B

(equation (5.3)), C_M , C_{WP} (equation (5.5)), and the ship's LCB. The LCB is found from the same method as in SPP.

For PPP evaluation, there are certain constraints placed on the ship hull coefficients to ensure the ship fits in the regression model. Equations (5.19)-(5.22) show the constraints on the C_P , the length to beam ratio (L/B), the beam to draft ratio (B/T), and the Froude number (Fn).

$$0.55 \leq C_p \leq 0.85 \quad (5.19)$$

$$3.90 \leq \frac{L}{B} \leq 14.9 \quad (5.20)$$

$$2.10 \leq \frac{B}{T} \leq 4.00 \quad (5.21)$$

$$0.05 \leq Fn \leq 1.00 \quad (5.22)$$

An example output from PPP is shown in appendix B. The program outputs the ship's resistance coefficients, partial resistance factors, the total resistance, and the ship's required power. For the objective function of the optimization, only the total resistance (R_T) at 15 knots is used.

5.4.2 Resistance Design Intent

For the resistance discipline, the design intent is to reduce the total resistance at the design speed, 15 knots, like many other optimization algorithms. Using the FLS evaluation of the designs created, the methods behind the resistance tool used can be taken into account. The Holtrop and Mennen method of calculating ship resistance used by PPP is generally a strong method for early stage prediction of ship

resistance. However, from equations (5.19) - (5.22) the method is only effective over certain ranges of coefficients because of the regression analysis used.

To attempt to obtain more robust results from the design tool, the discipline fuzzy logic system is used to create weighted constraints on the coefficients to put more emphasis on the results further from the model bounds. This effect is implemented in the method by using the constraint weighting method show in section 4.7. As the resistance model reaches the extents of the limits in equations (5.19)-(5.22), the preferences of the area are reduced. For the solutions near the boundaries to be considered as the design progresses, they must have significantly superior solutions to areas more central to the model boundaries.

5.4.3 Resistance Input Membership Functions

The input fuzzy sets for the resistance discipline are defined based on the ship's total resistance at 15 knots and the proximity to the limits of the model. For the resistance fuzzy sets, the same method is used as the objective function value preference shown in section 4.3.1. The resistance optimization was run to find the maximum and minimum resistance values over the designs being evaluated. These values are entered into equations (5.24) through (5.26) to create the total resistance preference fuzzy sets. Where $\mu_{R_T}^{high}$ assigns membership values to the "High Resistance Preference" fuzzy set, $\mu_{R_T}^{med}$ assigns membership values to the "Medium Resistance Preference" fuzzy set, $\mu_{R_T}^{low}$ assigns membership values to the "Low Resistance Preference" fuzzy set, R_T^* is the minimum total resistance, R_T^{max} is the maximum total resistance, R_T^{mid} is the midpoint of the resistance values (equation (5.23)), and $R_T(\mathbf{x})$ is the resistance at the current DV values.

$$R_T^{mid} = \frac{1}{2}(R_T^{max} - R_T^*) + R_T^* \quad (5.23)$$

$$\mu_{R_T}^{high}(R_T(\mathbf{x})) = \begin{cases} \frac{R_T^* - R_T(\mathbf{x})}{\frac{1}{2}(R_T^{max} - R_T^*)} + 1 & R_T(\mathbf{x}) \leq R_T^{mid} \\ 0 & R_T(\mathbf{x}) > R_T^{mid} \end{cases} \quad (5.24)$$

$$\mu_{R_T}^{med}(R_T(\mathbf{x})) = \begin{cases} \frac{R_T(\mathbf{x}) - R_T^*}{\frac{1}{2}(R_T^{max} - R_T^*)} & R_T(\mathbf{x}) \leq R_T^{mid} \\ \frac{R_T^{max} - R_T(\mathbf{x})}{\frac{1}{2}(R_T^{max} - R_T^*)} & R_T(\mathbf{x}) > R_T^{mid} \end{cases} \quad (5.25)$$

$$\mu_{R_T}^{low}(R_T(\mathbf{x})) = \begin{cases} 0 & R_T(\mathbf{x}) \leq R_T^{mid} \\ \frac{R_T(\mathbf{x}) - R_T^{max}}{\frac{1}{2}(R_T^{max} - R_T^*)} + 1 & R_T(\mathbf{x}) > R_T^{mid} \end{cases} \quad (5.26)$$

The input fuzzy sets for the resistance discipline also include fuzzy sets based on the model constraints in equations (5.19) through (5.22). The transitions from high to low fuzzy sets were selected just for demonstration, and could be adjusted for other experiments. The constraint fuzzy sets for C_P are shown in equations (5.27) and (5.28), which were developed based off the constraint membership functions (equations (4.16) and (4.17)) with a goal value of 0.60 and a threshold value of 0.55 on the lower bound for C_P and a goal of 0.80 and threshold of 0.85 on the upper bound of C_P . In equations (5.27) and (5.28), $\mu_{C_P}^{high}$ is the membership in the ‘‘High C_P Preference’’ fuzzy set, $\mu_{C_P}^{low}$ is the membership value in the ‘‘Low C_P Preference’’ fuzzy set, and C_P is the prismatic coefficient for the current design. The graphic representation of the C_P membership functions are shown in figure 5.9

$$\mu_{C_P}^{high}(C_P) = \begin{cases} 0 & C_P < 0.55, C_P > 0.85 \\ 20C_P - 11 & 0.55 \leq C_P \leq 0.6 \\ 1 & 0.6 < C_P < 0.8 \\ -20C_P + 17 & 0.8 \leq C_P \leq 0.85 \end{cases} \quad (5.27)$$

$$\mu_{C_P}^{low}(C_P) = \begin{cases} 1 & C_P < 0.55, C_P > 0.85 \\ -20C_P + 12 & 0.55 \leq C_P \leq 0.6 \\ 0 & 0.6 < C_P < 0.8 \\ 20C_P - 16 & 0.8 \leq C_P \leq 0.85 \end{cases} \quad (5.28)$$

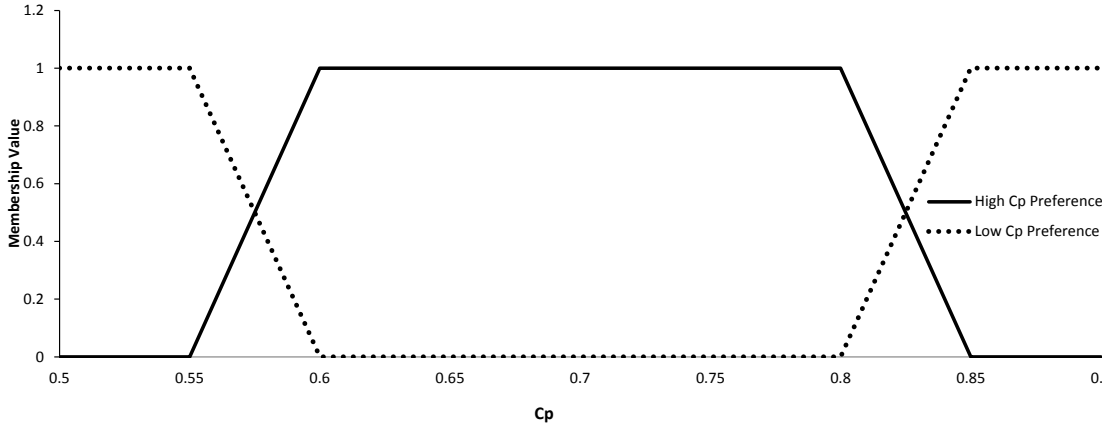


Figure 5.9: C_P Preference Fuzzy Sets

The remaining constraint fuzzy sets are constructed in the same manner with the goal and threshold values of each listed in table 5.2. Where LB and UB are the lower bound and upper bound of the coefficient constraints, respectively.

	Goal	Threshold
L/B LB	4.40	3.90
L/B UB	14.40	14.90
B/T LB	2.35	2.10
B/T UB	3.75	4.00
Fn LB	0.10	0.05
Fn UB	0.95	1.00

Table 5.2: Resistance Constraints Goal and Threshold Values

5.4.4 Resistance Fuzzy Rule Bank

A rule bank was developed to give higher preferences to the solutions that are not near the model constraints. The preferences for designs created between the goal and threshold values described previously will be degraded based on their closeness to the threshold value. The rule antecedents contain all AND statements so the lowest value of input resistance fuzzy set and all constraints is used for the activation level.

High Resistance Preference	High Preference	Unpreferred
Medium Resistance Preference	Medium Preference	Unpreferred
Low Resistance Preference	Low Preference	Unpreferred
	High Constraints Preference	Low Constraints Preference

Figure 5.10: Resistance Rule Matrix

5.5 Ship Design Case Study Problem Summary

The case study's goal is the optimization of a ship hull with respect to the resistance and seakeeping performances. The design variables for the case study are listed in table 5.3, with their lower and upper bound values. Table 5.4 lists all the parameters that are held constant throughout the optimization.

Design Variable	Units	Initial Value	Lower Bound	Upper Bound
L	[m]	260.8	245	270
B	[m]	45.8	40	50
T	[m]	15.9	15	20

Table 5.3: Case Study Design Variables

Constant	Units	Value
Δ	[tonnes]	162000
C_M	[-]	0.996
V_k	[knots]	15
Water Density	[tonnes/m ³]	1.025
Shell Appendage Allowance	[-]	1.005
Roll Damping Constant	[-]	0.2

Table 5.4: Case Study Parameters

Equation (5.29) outlines the problem statement for the case study MDO with all constraints listed as the crisp constraints that *must* be satisfied, where $F(\mathbf{x})$ is the system level preference function, f_{SK} is the seakeeping motions, and R_T is the total resistance.

System Level:

$$\underset{\mathbf{x}=[L,B,T]}{\text{maximize}} \quad F(\mathbf{x})$$

Seakeeping Discipline:

$$\underset{\mathbf{x}=[L,B,T]}{\text{minimize}} \quad f_{SK}(\mathbf{x})$$

$$\text{Subject To: } g_1(\mathbf{x}) : 0 \geq LCG - (LCB + (0.005L))$$

$$g_2(\mathbf{x}) : 0 \geq (LCB + (0.005L)) - LCG$$

Resistance Discipline:

$$\underset{\mathbf{x}=[L,B,T]}{\text{minimize}} \quad R_T(\mathbf{x})$$

$$\text{Subject To: } g_1(\mathbf{x}) : 0 \geq 0.55 - C_p$$

$$g_2(\mathbf{x}) : 0 \geq C_P - 0.85$$

$$g_3(\mathbf{x}) : 0 \geq 3.9 - \frac{L}{B}$$

$$g_4(\mathbf{x}) : 0 \geq \frac{L}{B} - 14.9$$

$$g_5(\mathbf{x}) : 0 \geq 2.10 - \frac{B}{T}$$

$$g_6(\mathbf{x}) : 0 \geq \frac{B}{T} - 4.00$$

$$g_7(\mathbf{x}) : 0 \geq 0.05 - Fn$$

$$g_8(\mathbf{x}) : 0 \geq Fn - 1.00$$

(5.29)

5.6 Results

The results of the HFLC MDO are shown in table 5.5. The table lists the objective function and key parameter values for the parent hullform, the resistance optimum, the seakeeping optimum, and the MDO optimum. The results of the seakeeping and resistance optimums are found when the individual disciplines are optimized with no consideration of the other disciplines.

		Initial Design	Res. Opt	Seasick. Opt	MDO Opt
L	[m]	260.8	270	263.8	270
B	[m]	45.8	45.1	43.4	46.5
T	[m]	15.9	17.4	16.4	16.1
L/B	[-]	5.69	5.98	6.08	5.80
B/T	[-]	2.88	2.88	2.65	2.89
C_b	[-]	0.832	0.746	0.841	0.782
C_P	[-]	0.836	0.749	0.845	0.786
R_T	[N]	1342180	1179320	1351970	1269840
Heave Acc.	[m/s ²]	0.117	0.114	0.112	0.113
Pitch Acc.	[rad/s ²]	1.176	1.185	1.128	1.156
Roll Acc.	[rad/s ²]	0.550	0.339	0.192	0.726
Seasick. Rating	[-]	6.848	6.805	6.956	6.930

Table 5.5: Ship Case Study Results

The table shows that the seakeeping discipline, had the lowest motion accelerations of any of the designs. The resistance optimum had a significantly lower resistance than other designs in the table. The MDO result demonstrates the trade offs that are made by the MDO algorithm.

The MDO resulted in objective function values less than optimal for each of the individual disciplines, but improved overall design. The MDO results showed improvements in the non-optimized objectives from the individual discipline optimizations. The total resistance for the MDO showed an approximate 6% improvement over the original hull and seakeeping optimal hull, and the ship motions improved for heave and pitch accelerations over the parent design and resistance optimum.

This case study demonstrated how the algorithm can evaluate design disciplines

with objectives on very different scales. When the design analysis tool outputs are entered into the discipline's FLS the results are all scaled by the membership functions so every discipline is on the same order of magnitude. This ensures that when trade-offs are considered, a single discipline will not have more weight than others simply because the discipline's objective function is orders of magnitude higher than the other disciplines.

Figure 5.11 presents the sampled points of the design space based on the discipline FLS outputs. The optimal designs and the initial design from table 5.5 are highlighted and the utopia point is also shown. In the figure, the trade offs made by the MDO optimizer are shown. While the solution from the MDO is not optimal for either individual discipline objective, the location is favorable in both disciplines.

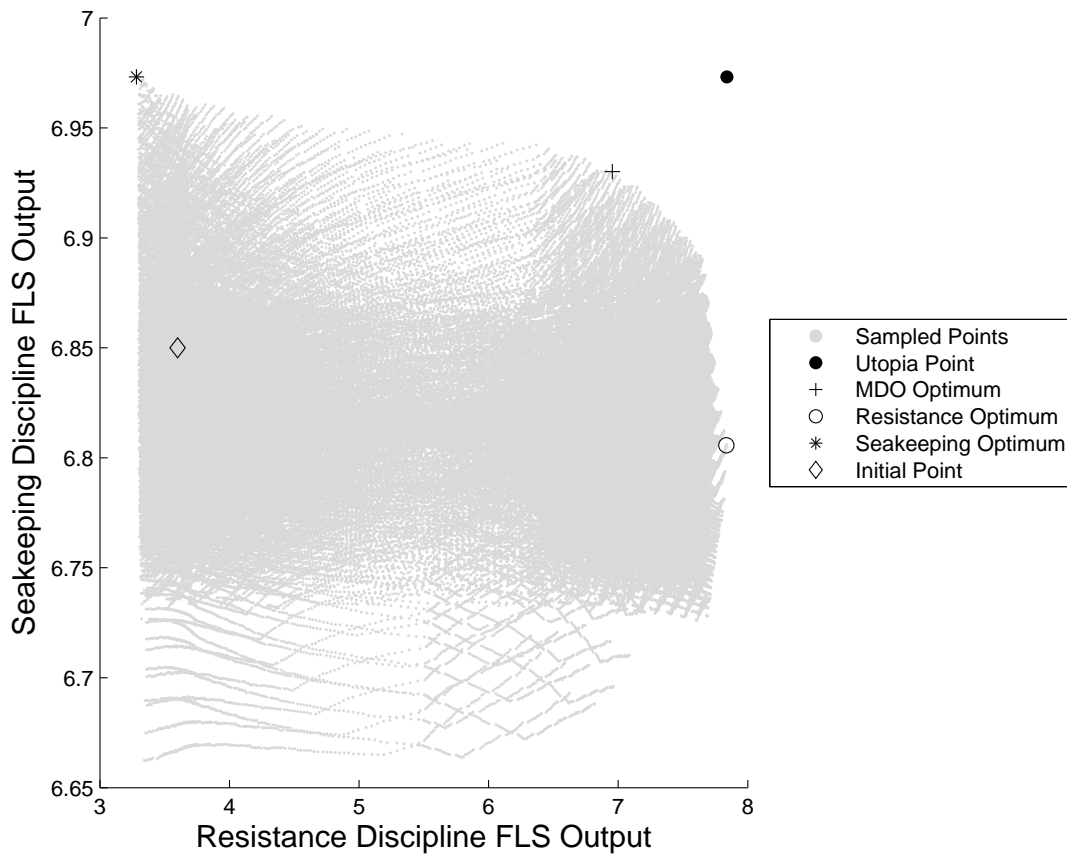


Figure 5.11: Case Study Design FLS Outputs

The figure displays the flatness of the design space with respect to the seakeeping discipline. All seakeeping outputs fell between 6.65 and 7.0 on the preference scale used. This is a result of the heave accelerations which had a very narrow range over the evaluated designs. This limited the range of the outputs due to the fact that the design intent had the most emphasis on the heave acceleration over other motions.

The resistance discipline outputs fell between 3 and 8 on the preference scale. The lower range of resistance preferences were limited by the upper bound of the C_P constraint, as shown by the seakeeping optimum which is nearly in full violation of the C_P constraint.

This case study also demonstrated the information that can be captured by including designer intent. The importance put on heave accelerations is shown by the seakeeping having the lowest heave followed by the MDO optimum. When just the seakeeping discipline are examined, the coupling effects added by the design intent are visible by the reduction in pitch and roll accelerations. When the trade-offs were evaluated by the MDO, the pitch and roll accelerations were not reduced to the level of the seakeeping optimum, but the highest of the values were reduced from the resistance optimum and parent hull.

5.7 Chapter Summary

To demonstrate the algorithm, a ship design case study is presented. The case study optimized a hullform based on a VLCC tanker, subject to the restrictions of the Suez Canal. The hullform optimization was subject to two discipline level optimizations: resistance and seakeeping. Two low-fidelity design tools were used to evaluate the resistance and seakeeping characteristics.

Fuzzy logic systems were built for the low-fidelity analysis models used. These systems modeled input the designer intent into the MDO optimizer FLS to extract more information from the models than what is available in standard MDO formu-

lations. Instead of simply examining the accelerations or displacements of the ship motions, the seakeeping discipline was intended to reduce the effects of seasickness on the ship's crew. By including the FLS analysis in the seakeeping optimization, aspects of the design were captured that are missed by traditional optimizers. The effects of seasickness could be modeled, even though no hard rules or mathematical models exist. This example demonstrates the potential benefits of including designer intent in the MDO optimizer.

The HFLC MDO was run, and an optimal design was found that reduced the resistance and ship motions that affect seasickness. The case study presented the ability of the framework to adapt to different disciplines, by modifying the discipline FLSs. Also, the HFLC MDO algorithm demonstrated the ability to handle trade-offs of disciplines with objective functions on very different scales.

CHAPTER VI

Conclusions and Recommendations for Future Work

This dissertation developed a novel method for multidisciplinary design optimization. The HFLC MDO method showed strong results in both the mathematical example and ship design case study presented. The development of the method presented in this dissertation provided several significant contributions.

6.1 Review of Contributions

Developed a New Method of Multidisciplinary Design Optimization

As discussed in chapter II, many methods of MDO exist and have been successfully used. The algorithm presented provides a novel approach to MDO by introducing designer intent models to the optimizer. The HFLC MDO method enables the optimizer to utilize more information from low-fidelity models in early stages of complex design.

When the algorithm was applied to mathematical disciplines (section 4.8), comparisons to other methods of MDO resulted favorably. Modeling design intent and minimizing the disciplines with considerations of trade-offs, the results directly compared to other MDO algorithms. In this example, the final design selected by the

HFLC MDO algorithm fell on the Pareto front of the evaluated designs. When compared to the other MDO algorithms, the HFLC MDO results yielded similar evaluation metrics.

Extension of Set-Based Design (SBD) Fuzzy Logic Systems to MDO

The methods from SBD of using fuzzy logic systems to communicate preferences between multiple analyses were adapted for use in a MDO optimizer. The sections 4.3, 5.3.3 and 5.4.3 presented disciplines with different design analysis aspects that are combined using fuzzy logic systems to create discipline preference curves. This allows for human design intent to be incorporated into the optimizer.

Extension of Set-Based Design Fuzzy Logic System using Hierarchical Fuzzy Logic Controllers to Combine Discipline Outputs

The FLS from SBD provided a means to communicate preferences of design aspects within a discipline, however a method was needed to blend different discipline FLS outputs into a single output for the entire system. In section 4.5, HFLC were outlined as a method to combined the outputs of multiple disciplines. The method presented successfully combined outputs from the disciplines that are scaled so trade-offs can be examined without concern for objective functions on extreme scales, as shown in section 5.6.

Applied Hierarchical Fuzzy Logic Controllers to MDO

HFLC methods were introduced to MDO as a automated method of analyzing systems with competing goals. As described previously, HFLC provide a method to evaluate trade-offs when analyzing FLS outputs. This method was incorporated in the HFLC MDO method to select optimal designs. The use of HFLC MDO to find optimal points was compared to other methods of MDO in section 4.8.5, and the

results of a ship design case study were presented in section 5.6. In both cases the HFLC MDO method successfully selected an optimal design.

Incorporated Expert Opinion into MDO Optimizer

The case studies in this dissertation also showed how the algorithm can be used as a framework for many different types of optimization problems. The designer intent membership functions can be changed depending on the problem being addressed. In the ship design case study, the use of FLS and HFLC allowed for information to be captured that does not have reliable mathematical models for use in a traditional design optimizer. As demonstrated in the seakeeping discipline (section 5.3), outputs from standard models available during early stages of design can be analyzed with a specific design intent, in this case the affects of seasickness on a ship's crew.

6.2 Recommendations

During the development of the HFLC MDO method presented, multiple research directions presented themselves. The use of fusion logic from HFLC provided a method to combine the preferences of multiple design disciplines. This presents a significant contribution to set-based design community. SBD methods have established methods for communication of preferences between disciplines, but a hierarchical method for blending the outputs from disciplines has not been formally established. The extension of fusion logic methods to the set-based design process is a potential area of investigation.

The use of fuzzy logic controllers in the design creation and evaluation process showed promising results throughout the example problems provided. The research on fuzzy logic controllers is extensive in different design disciplines. The use of different methods of fusion logic needs to be investigated as methods of assessing trade offs. Different types of fusion logic have been used in fuzzy controllers, such as weighted

average methods, which could give flexibility to the algorithm.

Another area that could be examined is the inclusion of uncertainty in the MDO process. Because fuzzy logic systems are used in the MDO algorithm, a clear direction for including uncertainty is the use of Type-2 FLS. Type-2 FLS allow for random uncertainty to be included in a FLS, this could lead to a HFLLC MDO method for optimization under uncertainty. Type-2 FLSs have been used in fuzzy logic controllers for system control under uncertainty, this could be extended to the evaluation of designs using the framework presented in this thesis.

A third direction that could be taken for future research is the use of classic optimization algorithms to control the search direction. The method presented used a set-based design approach to evaluation of the design space. This method produced optimal results with the case studies described. If a more efficient method of evaluating the design space is necessary, the framework presented provides a method to input the designer intent and consider trade offs which could be adapted to use optimization algorithms.

APPENDICES

APPENDIX A

Output of Seakeeping Prediction Program

Univeristy of Michigan
Department of Naval Architecture and Marine Engineering

Seakeeping Prediction Program (SPP-1.5) by M.G. Parsons

Reference: Raff, A. I., "Program SCORES - Ship Structural Response
in Waves", Ship Structures Committee Report SSC-230, 1972

Hull Data Identification: AutoCallSPP

Run Identification: AutoRun

Input Verification:

Length of Waterline LWL (m)	=	349.00
Vessel Displacement (tonnes)	=	223592.1
Vertical Center of Gravity VCG (m)	=	10.00
Roll Radius of Gyration k11 (m)	=	22.12
Fraction of Critical Roll Damping	=	0.1000
Ship Speed (knots)	=	15.00
Ship Heading Relative to Waves (deg)	=	45.00
Water Type	=	Salt@15C
Water Density Rho (kg/m ³)	=	1025.00
ISSC Two Parameter Spectrum Excitation		
Significant Wave Height (m)	=	8.00
Characteristic Wave Period (s)	=	15.00

Lower Freq. Integration Limit (rad/s) = 0.30

Upper Freq. Integration Limit (rad/s) = 1.20

Sta.	Beam[m]	Area[m ²]	Draft[m]	Weight[t]
0	0.00	0.00	0.00	13415.5
1	37.15	398.66	15.00	15651.5
2	56.00	780.28	15.00	21241.3
3	56.00	836.64	15.00	26831.1
4	56.00	836.64	15.00	26831.1
5	56.00	836.64	15.00	26831.1
6	56.00	836.64	15.00	26831.1
7	56.00	807.61	15.00	26831.1
8	56.00	593.17	15.00	17887.4
9	36.03	260.51	15.00	13415.5
10	5.49	4.20	0.75	7825.7

Univeristy of Michigan
Department of Naval Architecture and Marine Engineering

Seakeeping Prediction Program (SPP-1.5) by M.G. Parsons

Reference: Raff, A. I., "Program SCORES - Ship Structural Response
in Waves", Ship Structures Committee Report SSC-230, 1972

Hull Data Identification: AutoCallSPP

Run Identification: AutoRun

Motion Natural Frequencies and Periods:

Heave Natural Frequency = 0.575 rad/s Heave Natural Period = 10.92 sec.

Pitch Natural Frequency = 0.575 rad/s Pitch Natural Period = 10.93 sec.

Roll Natural Frequency = 0.508 rad/s Roll Natural Period = 12.38 sec.

Roll Damping Results:

Roll Wave Damping = 0.307E+06

Added Viscous Roll Damping = 0.108E+07

Seakeeping Response Results:

Ship Speed = 15.0 knots = 7.72 m/s

Wave Angle [with Head Seas 180 deg.] = 45.0 deg.

ISSC Two Parameter Spectrum - Sign.Height = 8.00 m Char. Period = 15.00 s

Nondimensional Motion Response Amplitude Operators:

Wave Freq. r/s	Encount. Freq. r/s	Wave Length m	Heave Ampl. n.d.	Heave Phase deg.	Pitch Ampl. n.d.	Pitch Phase deg.	Roll Ampl. n.d.	Roll Phase deg.
0.300	0.250	684.6	0.839E+00	177.6	0.652E+00	-85.9	0.645E+00	-91.4
0.350	0.282	503.0	0.717E+00	175.6	0.595E+00	-84.3	0.575E+00	-90.7
0.400	0.311	385.1	0.552E+00	171.9	0.512E+00	-82.5	0.466E+00	-89.5
0.450	0.337	304.3	0.358E+00	164.3	0.403E+00	-80.1	0.316E+00	-86.6
0.500	0.361	246.5	0.171E+00	142.1	0.276E+00	-76.5	0.137E+00	-74.7
0.550	0.382	203.7	0.104E+00	67.1	0.147E+00	-70.0	0.791E-01	46.5
0.600	0.400	171.2	0.156E+00	25.6	0.412E-01	-44.0	0.221E+00	71.9
0.650	0.415	145.8	0.146E+00	10.6	0.400E-01	80.6	0.288E+00	76.2
0.700	0.427	125.7	0.698E-01	-7.5	0.578E-01	102.4	0.244E+00	79.9
0.750	0.437	109.5	0.337E-01	-130.1	0.389E-01	121.3	0.113E+00	94.4
0.800	0.444	96.3	0.677E-01	-162.4	0.145E-01	-172.2	0.598E-01	-153.5
0.850	0.448	85.3	0.412E-01	-171.3	0.198E-01	-92.0	0.122E+00	-121.2
0.900	0.449	76.1	0.644E-02	27.8	0.148E-01	-51.7	0.956E-01	-112.5
0.950	0.448	68.3	0.197E-01	20.5	0.850E-02	33.6	0.206E-01	-108.7
1.000	0.444	61.6	0.132E-01	97.7	0.700E-02	109.8	0.240E-01	81.3
1.050	0.437	55.9	0.157E-01	96.8	0.797E-02	-144.3	0.117E-01	46.6
1.100	0.427	50.9	0.343E-01	-4.8	0.698E-02	-100.7	0.247E-01	-45.3
1.150	0.414	46.6	0.506E-01	-40.8	0.688E-02	60.7	0.133E-01	-31.8
1.200	0.399	42.8	0.453E-01	-135.6	0.110E-01	82.6	0.193E-01	123.8

Seakeeping Prediction Program (SPP-1.5) by M.G. Parsons

Run Identification: AutoRun

Nondimensional Moment Response Amplitude Operators:

Wave Freq. r/s	Encount. Freq. r/s	Wave Length m	Vertical Moment Ampl. n.d.	Vertical Moment Phase deg.	Trans. Moment Ampl. n.d.	Trans. Moment Phase deg.	Tors. Moment Ampl. n.d.	Tors. Moment Phase deg.
0.300	0.250	684.6	0.402E-02	25.3	0.173E-02	87.8	0.603E-03	-176.1
0.350	0.282	503.0	0.736E-02	29.2	0.284E-02	87.4	0.995E-03	-174.9
0.400	0.311	385.1	0.115E-01	33.6	0.449E-02	87.1	0.145E-02	-173.4
0.450	0.337	304.3	0.157E-01	38.5	0.667E-02	86.9	0.186E-02	-171.8
0.500	0.361	246.5	0.186E-01	44.0	0.907E-02	87.1	0.210E-02	-170.1
0.550	0.382	203.7	0.189E-01	50.4	0.110E-01	87.8	0.203E-02	-168.1
0.600	0.400	171.2	0.156E-01	59.1	0.117E-01	89.2	0.159E-02	-166.1

0.650	0.415	145.8	0.936E-02	74.7	0.104E-01	91.8	0.868E-03	-164.3
0.700	0.427	125.7	0.358E-02	129.2	0.718E-02	96.3	0.101E-03	-175.7
0.750	0.437	109.5	0.465E-02	-147.5	0.277E-02	106.3	0.403E-03	29.3
0.800	0.444	96.3	0.454E-02	-117.8	0.139E-02	-93.6	0.448E-03	36.2
0.850	0.448	85.3	0.172E-02	-58.4	0.322E-02	-73.3	0.133E-03	82.2
0.900	0.449	76.1	0.268E-02	48.9	0.171E-02	-49.8	0.366E-03	-163.9
0.950	0.448	68.3	0.259E-02	100.7	0.273E-02	99.0	0.497E-03	-154.9
1.000	0.444	61.6	0.265E-02	-179.7	0.560E-02	116.5	0.264E-03	-164.9
1.050	0.437	55.9	0.188E-02	-124.6	0.293E-02	133.0	0.143E-03	95.5
1.100	0.427	50.9	0.266E-02	7.3	0.408E-02	-61.5	0.100E-03	63.0
1.150	0.414	46.6	0.300E-02	35.4	0.610E-02	-45.8	0.213E-03	-101.4
1.200	0.399	42.8	0.263E-02	-158.2	0.194E-02	94.2	0.302E-03	-103.1

Wave Input and Response Amplitude Spectra:

Freq.	Wave Amp.	Heave	Pitch	Roll	Vert. Mom.	Lat. Mom.	Tors. Mom.
r/s	m ² s	m ² s	deg. ² s	deg. ² s	(t-m) ² s	(t-m) ² s	(t-m) ² s
0.300	16.608	11.691	1.953	1.908	0.131E+11	0.243E+10	0.296E+09
0.350	16.752	8.602	3.040	2.842	0.444E+11	0.658E+10	0.811E+09
0.400	12.542	3.817	2.873	2.384	0.813E+11	0.124E+11	0.128E+10
0.450	8.512	1.088	1.937	1.188	0.102E+12	0.185E+11	0.144E+10
0.500	5.639	0.165	0.918	0.225	0.957E+11	0.227E+11	0.122E+10
0.550	3.754	0.041	0.255	0.073	0.655E+11	0.223E+11	0.758E+09
0.600	2.539	0.062	0.019	0.550	0.302E+11	0.170E+11	0.315E+09
0.650	1.752	0.037	0.017	0.884	0.750E+10	0.931E+10	0.645E+08
0.700	1.233	0.006	0.034	0.600	0.771E+09	0.311E+10	0.618E+06
0.750	0.886	0.001	0.014	0.122	0.938E+09	0.333E+09	0.705E+07
0.800	0.648	0.003	0.002	0.032	0.651E+09	0.615E+08	0.635E+07
0.850	0.482	0.001	0.003	0.128	0.699E+08	0.244E+09	0.417E+06
0.900	0.364	0.000	0.002	0.074	0.127E+09	0.521E+08	0.239E+07
0.950	0.279	0.000	0.001	0.003	0.917E+08	0.101E+09	0.337E+07
1.000	0.216	0.000	0.000	0.004	0.742E+08	0.332E+09	0.739E+06
1.050	0.170	0.000	0.000	0.001	0.293E+08	0.714E+08	0.169E+06
1.100	0.135	0.000	0.000	0.004	0.466E+08	0.110E+09	0.662E+05
1.150	0.108	0.000	0.000	0.001	0.478E+08	0.197E+09	0.240E+06
1.200	0.088	0.000	0.001	0.002	0.297E+08	0.161E+08	0.392E+06

Wave Input and Response Amplitude Statistics:

	m	m	deg.	deg.	t-m	t-m	t-m
R.M.S.	1.794	0.992	0.710	0.710	0.148E+06	0.757E+05	0.174E+05
Ave.	2.242	1.240	0.888	0.887	0.185E+06	0.946E+05	0.218E+05
Signif.	3.588	1.983	1.421	1.419	0.295E+06	0.151E+06	0.348E+05
Ave 1/10	4.574	2.529	1.812	1.810	0.377E+06	0.193E+06	0.444E+05
Design Value with N=1000 and alpha=0.01					0.709E+06	0.363E+06	0.835E+05

APPENDIX B

Output of Power Prediction Program

University of Michigan
Department of Naval Architecture and Marine Engineering

Power Prediction Program (PPP-1.8) by M. G. Parsons

Source: 1. Holtrop, J., & Mennen, G.G.J., "An Approximate Power Prediction Method," International Shipbuilding Progress, Vol.29, No.335, July, 1982.
2. Holtrop, J., "A Statistical Reanalysis of Resistance and Propulsion Data," International Shipbuilding Progress, Vol.31, No.363, Nov., 1984.

Run Identification: A

Input Verification:

Length of Waterline LWL (m)	=	315.00
Maximum Beam on LWL (m)	=	50.80
Depth at the Bow (m)	=	0.00
Mean Draft (m)	=	16.90
Draft Forward (m)	=	16.90
Draft Aft (m)	=	16.90
Block Coefficient on LWL CB	=	0.8230
Prismatic Coefficient on LWL CP	=	0.8263
Midship Coefficient to LWL CM=CX	=	0.9960
Waterplane Coefficient on LWL CWP	=	0.8912
Center of Buoyancy LCB (% LWL; + Fwd)	=	0.5000
Center of Buoyancy LCB (m from FP)	=	155.93
Molded Volume (m ³)	=	222567.0
Deck House/Cargo Frontal Area (m ²)	=	0.00
Water Type	=	Salt@15C
Water Density (kg/m ³)	=	1025.87
Kinematic Viscosity (m ² /s)	=	0.118831E-05
Appen. Drag (% Bare Hull Resistance)	=	5.00

Bulb Section Area at Station 0 (m²) = 0.00
 Vertical Center of Bulb Area (m) = 0.00

 Transom Immersed Area (m²) = 0.00
 Stern Type = Normally Shaped
 Design Margin on RT,PE,REQ.THR (%) = 0.00
 Propulsion Type = SS, Conv.

 Propeller Diameter (m) = 10.14
 Propeller Expanded Area Ratio Ae/Ao = 0.8500

 Wetted Surface (m²) = 22635.26
 Half Angle of Entrance (deg) = 45.53

University of Michigan
 Department of Naval Architecture and Marine Engineering

Power Prediction Program (PPP-1.8) by M. G. Parsons

Source:1. Holtrop,J., & Mennen, G.G.J., "An Approximate
 Power Prediction Method," International Shipbuilding
 Progress, Vol.29, No.335, July, 1982.
 2. Holtrop,J., "A Statistical Reanalysis of
 Resistance and Propulsion Data," International
 Shipbuilding Progress, Vol.31, No.363, Nov., 1984.

Run Identification: A

Speed, Resistance Coefficients and Frictional Resistance RF(N):

V(kts)	V(m/s)	FN	SLRATIO	CF	CR	CA	RF
5.00	2.57	0.0463	0.1555	0.001606	0.000509	0.000237	123369.7
7.00	3.60	0.0648	0.2177	0.001539	0.000488	0.000237	231785.8
9.00	4.63	0.0833	0.2800	0.001492	0.000473	0.000237	371448.6
11.00	5.66	0.1018	0.3422	0.001456	0.000465	0.000237	541484.4
13.00	6.69	0.1203	0.4044	0.001427	0.000476	0.000237	741224.6
15.00	7.72	0.1388	0.4666	0.001403	0.000534	0.000237	970131.0
17.00	8.75	0.1574	0.5288	0.001383	0.000683	0.000237	1227754.1
19.00	9.77	0.1759	0.5910	0.001365	0.000968	0.000237	1513710.0

Remaining Resistance Components (N):

V(kts)	Form RF*K1	Appendage RAPP	Wave RW	Bulb RB	Transom RTR	Correlation RA	Air Drag RAIR
5.00	39110.2	8124.0	0.0	0.0	0.0	18204.6	0.0
7.00	73479.9	15263.3	0.6	0.0	0.0	35681.0	0.0
9.00	117755.2	24463.4	63.5	0.0	0.0	58982.8	0.0
11.00	171659.3	35726.7	1390.2	0.0	0.0	88110.1	0.0
13.00	234980.3	49422.8	12251.6	0.0	0.0	123062.9	0.0
15.00	307547.3	66982.1	61964.4	0.0	0.0	163841.2	0.0
17.00	389218.0	91730.8	217643.6	0.0	0.0	210444.8	0.0
19.00	479870.7	129367.3	593765.5	0.0	0.0	262874.0	0.0

Resistance, Effective Power, Propulsion Factors and Required Thrust

V(kts)	RT(N)	PE(kW)	w	t	REQ.THR(N)	etaH	etaRR
5.00	188808.5	485.65	0.5081	0.2007	236219.1	1.6249	1.0025
7.00	356210.6	1282.74	0.5044	0.2007	445656.6	1.6127	1.0025
9.00	572713.5	2651.64	0.5018	0.2007	716524.4	1.6043	1.0025
11.00	838370.8	4744.21	0.4998	0.2007	1048889.4	1.5980	1.0025
13.00	1160942.3	7764.06	0.4983	0.2007	1452459.9	1.5931	1.0025
15.00	1570466.0	12118.66	0.4970	0.2007	1964816.9	1.5890	1.0025
17.00	2136791.5	18687.27	0.4959	0.2007	2673349.0	1.5856	1.0025
19.00	2979587.5	29123.56	0.4949	0.2007	3727774.8	1.5826	1.0025

Design Margin Has Been Included in RT, PE, and REQ.THR = RT/(1-t).

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] ISO 2631-1.2: Mechanical vibration and shock - guide to experiments with people - Part 1: Exposure to mechanical vibration and repeated shock. Technical report, International Organization for Standardisation, Geneva, 1997.
- [2] James Allison, Michael Kokkolaras, Marc Zawislak, and Panos Y Papalambros. On the Use of Analytical Target Cascading and Collaborative Optimization for Complex System Design. In *6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, 2005.
- [3] R. Balling and M.R. Rawlings. Collaborative optimization with disciplinary conceptual design. *Structural and Multidisciplinary Optimization*, 20(3):232–241, November 2000.
- [4] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley-Interscience, New Jersey, 3 edition, 2006.
- [5] David Beasley, RR Martin, and DR Bull. An overview of genetic algorithms: Part 1. Fundamentals. *University computing*, 15:1–16, 1993.
- [6] H.R. Berenji, S. Saraf, and S.R. Swanson. Pitch control of the space shuttle training aircraft. *IEEE Transactions on Control Systems Technology*, 9(3):542–551, May 2001.
- [7] Piero P Bonissone, Hsin-Chao Chiang, Mark E Dausch, and James B Comly. Fuzzy Hierarchical Controller for a Turboshaft Engine, 1998.
- [8] P.P. Bonissone, V. Badami, K.H. Chiang, P.S. Khedkar, K.W. Marcelle, and M.J. Schutten. Industrial applications of fuzzy logic at General Electric. *Proceedings of the IEEE*, 83(3):450–465, March 1995.
- [9] R. Choudhary, a. Malkawi, and P.Y. Papalambros. Analytic target cascading in simulation-based building design. *Automation in Construction*, 14(4):551–568, August 2005.
- [10] E. Cox. *The Fuzzy Systems Handbook: Practitioner’s Guide to Building, Using and Maintaining Fuzzy Systems*. Academic Press, San Diego, 2nd edition, 1999.
- [11] Evin J Cramer, J E Dennis, Paul D Frankt, Robert Michael Lewis, and Gregory R Shubin. PROBLEM FORMULATION FOR MULTIDISCIPLINARY. *SIAM Journal of Optimization*, 4(4):754–776, 1994.

- [12] Brian J Cuneo, Kevin J Maki, and David J Singer. Applying Soft Constraints to Multi-Disciplinary Design Optimization using Fuzzy Logic. In *11th International Conference on Fast Sea Transportation (FAST 2011)*, Honolulu, Hawaii, 2011. American Society of Naval Engineers.
- [13] AJ de Wit and F Van Keulen. Numerical comparison of multi-level optimization techniques. *48th AIAA Structures, Structural Dynamics, and Materials Conference*, (April), 2007.
- [14] Peter Deuffhard. *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms*. Springer-Berlag, Berlin, Germany, 2004.
- [15] R Eberhart and Y Shi. Particle swarm optimization: developments, applications and resources. *Proceedings of the Evolutionary Computation*, pages 81–86, 2001.
- [16] P.J. Escamilla-Ambrosio and N. Mort. A hybrid Kalman filter-fuzzy logic architecture for multisensor data fusion. *Proceeding of the 2001 IEEE International Symposium on Intelligent Control (ISIC '01) (Cat. No.01CH37206)*, (1):364–369, 2001.
- [17] Forest Flager, Benjamin Welle, and Prasun Bansal. Multidisciplinary process integration and design optimization of a classroom building. *Journal of Information . . .*, 14(August):595–612, 2009.
- [18] P. Fuglsang, C. Bak, J. G. Schepers, B. Bulder, T. T. Cockerill, P. Claiden, a. Olesen, and R. van Rossen. Site-specific Design Optimization of Wind Turbines. *Wind Energy*, 5(4):261–279, October 2002.
- [19] Alexander W. Gray, Brian J. Cuneo, Nickolas Vlahopoulos, and David J. Singer. The Rapid Ship Design Environment Multi - Disciplinary Optimization of a U . S . Navy Frigate. In *ASNE Day 2013 Proceedings*, Crystal City, 2013. American Society of Naval Engineers.
- [20] Alexander Westley Gray. *Enhancement of Set-Based Design Practices Via Introduction of Uncertainty Through the Use of Interval Type-2 Modeling and General Type-2 Fuzzy Logic Agent Based Methods*. Ph.d, University of Michigan, 2012.
- [21] John a. Green. Aeroelastic tailoring of aft-swept high-aspect-ratio composite wings. *Journal of Aircraft*, 24(11):812–819, November 1987.
- [22] B. Grossman, Z. Gurdal, R. T. Haftka, G. J. Strauch, and W. M. Eppard. Integrated aerodynamic/structural design of a sailplane wing. *Journal of Aircraft*, 25(9):855–860, September 1988.
- [23] Rodolfo E. Haber, Rodolfo Haber-Haber, Agustín Jiménez, and Ramón Galán. An optimal fuzzy control system in a network environment based on simulated annealing. An application to a drilling process. *Applied Soft Computing*, 9(3):889–895, June 2009.

- [24] R. T. Haftka. Optimization of flexible wing structures subject to strength and induced drag constraints. *AIAA Journal*, 15(8):1101–1106, August 1977.
- [25] H.a. Hagrass. A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots. *IEEE Transactions on Fuzzy Systems*, 12(4):524–539, August 2004.
- [26] Jeongwoo Han. *Sequential Linear Programming Coordination Strategy for Deterministic and Probabilistic Analytical Target Cascading*. Ph.d., University of Michigan, 2008.
- [27] Shari Hannapel and Nickolas Vlahopoulos. Introducing Uncertainty in Multidiscipline Ship Design. *Naval Engineers Journal*, 122(2), June 2010.
- [28] Shari E. Hannapel. *DEVELOPMENT OF MULTIDISCIPLINARY DESIGN OPTIMIZATION ALGORITHMS FOR SHIP DESIGN UNDER UNCERTAINTY*. PhD thesis, University of Michigan, 2012.
- [29] Christopher Gregory Hart. *Multidisciplinary Design of Complex Engineering Systems for Cost Assessment Under Uncertainty*. PhD thesis, University of Michigan, 2010.
- [30] RP Henderson, J Martins, and RE Perez. Aircraft conceptual design for optimal environmental performance. *Aeronautical Journal*, (3712):1–22, 2012.
- [31] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan, 1975.
- [32] J. Holtrop and G. G. J. Mennen. An Approximate Power Prediction Method. *International Shipbuilding Progress*, 29(335), 1982.
- [33] PW Jansen. *Aerostructural Optimization of NonPlanar Lifting Surfaces*. M.s., University of Toronto, 2009.
- [34] H.M. Kim, Michael Kokkolaras, L.S. Louca, G.J. Delagrammatikas, N.F. Michelena, Z.S. Filipi, Panos Y Papalambros, J.L. Stein, and D.N. Assanis. Target Cascading in vehicle redesign: a class VI truck study. *International Journal of Vehicle Design*, 29(3):199–225, 2002.
- [35] Hong Yoon Kim and Nickolas Vlahopoulos. A Multi-Level Optimization Algorithm and a Ship Design Application. *The American Institute of Aeronautics and Astronautics*, pages 1–16, 2012.
- [36] Hyung Min Kim. *TARGET CASCADING IN OPTIMAL SYSTEM DESIGN*. Ph.d, University of Michigan, 2001.
- [37] Hyung Min Kim, Nestor F. Michelena, Panos Y. Papalambros, and Tao Jiang. Target Cascading in Optimal System Design. *Journal of Mechanical Design*, 125(3):474, 2003.

- [38] S. Kodiyalam, R.J. Yang, L. Gu, and C.-H. Tho. Multidisciplinary design optimization of a vehicle system in a scalable, high performance computing environment. *Structural and Multidisciplinary Optimization*, 26(3-4):256–263, February 2004.
- [39] Michael Kokkolaras, L.S. Louca, G.J. Delagrammatikas, N.F. Michelena, Z.S. Filipi, Panos Y Papalambros, J.L. Stein, and D.N. Assanis. Simulation-Based Optimal Design of Heavy Trucks by Model-Based Decomposition: An Extensive Analytical Target Cascading Case Study. *International Journal of Heavy Vehicle Systems*, 11:402–432, 2004.
- [40] Ilan Kroo. MDO for Large-Scale Design. In *Multidisciplinary Design Optimization: State of the Art*, pages 22–44. Society for Industrial and Applied Mathematics, Hampton, 1997.
- [41] Ilan Kroo, Steve Altus, Robert Braun, Peter Gage, and Ian Sobieski. MULTIDISCIPLINARY OPTIMIZATION METHODS FOR AIRCRAFT PRELIMINARY DESIGN. In *5th AIAA/USF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, pages 1–19. AIAA, 1994.
- [42] C.C. Lee. Fuzzy logic in control systems: fuzzy logic controller. I. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):404–418, 1990.
- [43] C.C. Lee. Fuzzy logic in control systems: fuzzy logic controller. II. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):419–435, 1990.
- [44] Claude Lemar. Cauchy and the Gradient Method. *Documenta Mathematica*, Extra Volu:251–254, 2012.
- [45] E Lewis, editor. *Principles of Naval Architecture: Volume III Motion in Waves and Controllability*. The Society of Naval Architects and Marine Engineers, Jersey City, 1988.
- [46] Edward V. Lewis, editor. *Principles of Naval Architecture: Volume II Resistance, Propulsion, and Vibration*. The Society of Naval Architects and Marine Engineers, Jersey City, 1988.
- [47] R.T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395, April 2004.
- [48] Joaquim R R A Martins and Andrew B. Lambe. Multidisciplinary Design Optimization : A Survey of Architectures. *AIAA Journal*, pages 1–53, 2012.
- [49] M.E. McCauley, J.W. Royal, C.D. Wylie, J.F. O’Hanlon, and R.R. Mackie. Motion Sickness Incidence: Exploratory Studies of Habituation, Pitch and Roll, and the Refinement of a Mathematical Model. Technical Report April, Human Factors Research Inc, 1976.

- [50] J.M. Mendel. Introduction to uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems. *10th IEEE International Conference on Fuzzy Systems. (Cat. No.01CH37297)*, pages 662–665, 2001.
- [51] Ahmad Mirzaei and Mehdi Moallem. Design of an optimal fuzzy controller for antilock braking systems. *Vehicle Power and . . .*, 55(6):1725–1730, 2005.
- [52] Arup Kumar Nandi and J. Paulo Davim. A study of drilling performances with minimum quantity of lubricant using fuzzy logic rules. *Mechatronics*, 19(2):218–232, March 2009.
- [53] S. Andrew Ning and Ilan Kroo. Multidisciplinary Considerations in the Design of Wings and Wing Tip Devices. *Journal of Aircraft*, 47(2):534–543, March 2010.
- [54] K Nozawa and Y Kodama. Hydrodynamic Performance and Turbulent Flow Problems of Ships. In *27th Symposium on Turbulence*, 1995.
- [55] Michael Osborne. Oil Tankers. In T Lamb, editor, *Ship Design and Construction: Volume II*, chapter 29. Society Of Naval Architects and Marine Engineers, 2004.
- [56] Panos Y Papalambros and Douglass J. Wilde. *Principles of Optimal Design*. Cambridge University Press, Cambridge, 2nd edition, 2000.
- [57] Edward J. Park, Luis Falcão da Luz, and Afzal Suleman. Multidisciplinary design optimization of an automotive magnetorheological brake design. *Computers & Structures*, 86(3-5):207–216, February 2008.
- [58] Parsons. NA470 Foundations of Ship Design. Technical report, University of Michigan, Ann Arbor, MI, 2007.
- [59] Michael G. Parsons. Parametric Design. In Thomas Lamb, editor, *Ship Design and Construction: Volume I*, chapter 11. Society Of Naval Architects and Marine Engineers, 2003.
- [60] Michael G. Parsons, Jun Li, and David J Singer. Power Prediction Program (PPP1.8), 1998.
- [61] Michael G. Parsons, Jun Li, and David J. Singer. Seakeeping Prediction Program (SPP1,5), 1998.
- [62] Clodeinir Ronei Peres, Rodolfo Elias Haber Guerra, Rodolfo Haber Haber, Angel Alique, and Salvador Ros. Fuzzy model and hierarchical fuzzy control integration: an approach for milling process optimization. *Computers in Industry*, 39(3):199–207, July 1999.
- [63] Daniele Peri and Emilio F Campana. Multidisciplinary Design Optimization of a Naval Surface Combatant. *Journal of Ship Research*, 47(1):1–12, 2003.
- [64] Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization. *Swarm Intelligence*, 1(1):33–57, August 2007.

- [65] Alfred I Raff. SCORES-SHIP RESPONSE IN STRUCTURAL. Technical report, Ship Structures Committee Report SSC-230 1972, Washington DC, 1972.
- [66] Francesco M. Raimondi and Maurizio Melluso. Fuzzy motion control strategy for cooperation of multiple automated vehicles with passengers comfort. *Automatica*, 44(11):2804–2816, November 2008.
- [67] G V S Raju, J U N Zhou, and Roger A Kisner. Hierarchical fuzzy control. *International Journal of Control*, 54(5):37–41, 1991.
- [68] G.V.S. Raju. Adaptive hierarchical fuzzy controller. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(4):973–980, 1993.
- [69] Farzad Rajaei Salmasi. Control Strategies for Hybrid Electric Vehicles: Evolution, Classification, Comparison, and Future Trends. *IEEE Transactions on Vehicular Technology*, 56(5):2393–2404, September 2007.
- [70] MA Sandidzadeh and Babak Shamszadeh. Improvement of Automatic Train Operation Using Enhanced Predictive Fuzzy Control Method. *Reliability and Safety in Railway*, 2012.
- [71] Lucien A. Schmit. Structural Design by Systematic Synthesis. In *ASCE Conference on Electronic Computation*, 1960.
- [72] David J Singer. *A Hybrid Agent Approach for Set-Based Conceptual Ship Design through the Use of a Fuzzy Logic Agent to Facilitate Communications and Negotiation*. Ph.d., University of Michigan, 2003.
- [73] David J. Singer, Norbert Doerry, and Michael E. Buckley. What Is Set-Based Design? *Naval Engineers Journal*, 121(4):31–43, October 2009.
- [74] J.a. Stover, D.L. Hall, and R.E. Gibson. A fuzzy-logic architecture for autonomous multisensor data fusion. *IEEE Transactions on Industrial Electronics*, 43(3):403–410, June 1996.
- [75] Bentley Systems. MaxSurf V8i, 2012.
- [76] Yoshihisa Takekoshi, Takafumi Kawamura, Hajime Yamaguchi, Masatsugu Maeda, Norio Ishii, Koyu Kimura, Tadashi Taketani, and Akihiko Fujii. Study on the design of propeller blade sections using the optimization algorithm. *Journal of Marine Science and Technology*, 10(2):70–81, June 2005.
- [77] D.W. Taylor. Calculations for ships’ forms and the light thrown by modern experiments upon resistance, propulsions and rolling of ships. In *Transactions of the International Engineering Congress: Naval Architecture and Marine Engineering*, San Francisco, 1915.
- [78] Nathan P. Tedford and Joaquim R. R. a. Martins. Benchmarking multi-disciplinary design optimization algorithms. *Optimization and Engineering*, 11(1):159–183, March 2009.

- [79] Nickolas Vlahopoulos, Geng Zhang, and Ricardo Sbragio. Design of Rotorcraft Gearbox Foundation for Reduced Vibration and Increased Crashworthiness Characteristics. In *SAE 2011 Noise and Vibration Conference and Exhibition*, Grand Rapids, 2011. SAE.
- [80] A H Wertheim, J E Bos, and W Bles. Contributions of roll and pitch to sea sickness. *Brain research bulletin*, 47(5):517–24, November 1998.
- [81] Dongrui Wu, Woei Wan Tan, and Rule Base. Computationally Efficient Type-Reduction Strategies for a Type-2 Fuzzy Logic Controller. *The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05.*, pages 353–358, 2005.
- [82] SF Wu, CJH Engelen, QP Chu, R. Babuska, J.A. Mulder, and G. Ortega. Fuzzy logic based attitude control of the spacecraft X-38 along a nominal re-entry trajectory. *Control Engineering . . .*, 9:699–707, 2001.
- [83] Seiji Yasunobu and Shoji Miyamoto. Automatic train operation system by predictive fuzzy control. . . . *Applications of Fuzzy Control*, 1, 1985.
- [84] John Yen and Reza Langari. *Fuzzy Logic: Intelligence, Control, and Information*. Prentice-Hall, Inc, Upper Saddle River, 1999.
- [85] Y.L. Young, J.W. Baker, and M.R. Motley. Reliability-based design and optimization of adaptive marine structures. *Composite Structures*, 92(2):244–253, January 2010.
- [86] L Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3):199–249, 1975.
- [87] LA Zadeh. A RATIONALE FOR FUZZY CONTROL. *Journal of Dynamic Systems, Measurement and Control*, (March 1972):3–4, 1972.
- [88] LA Zadeh. Fuzzy logic = computing with words. *Fuzzy Systems, IEEE Transactions on*, 4(2):103–111, 1996.
- [89] R.B. Zubaly. *Applied Naval Architecture*. Schiffer Publishing, Atglen, PA, 2nd edition, 1996.