
This dissertation introduces methods for optimal engineering design and for exploration of the design space. The applications explored involve the design of electromechanical devices, in addition to benchmark problems. Two methods are introduced: a computationally efficient variation of firefly algorithm (FA), tuned to find multimodal objective function minima; and a new method for programmatically and visually identifying locally optimal solutions. The visualization method, a plot of distances between cost-sorted designs (the cost-sorted distance or CSD plot), is shown to reveal clusters of designs at minima of the objective function. An investigation is made of the methods’ capability to address uncertainty in the design process. Correlation between the percent of fireflies in clusters and robustness measures for associated designs is considered. Finally, a new type of magnetorheological fluid device (MRFD) is described and its design is optimized using FA and CSD. Performance predictions made for the optimized design are compared to results from finite element analysis and experimental testing.
© Copyright 2017 by Christopher Mark Elliott

All Rights Reserved
Methods for Streamlined Firefly Optimization and Interpretation: Applications to Electromechanical Systems Design

by

Christopher Mark Elliott

A dissertation submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Mechanical Engineering

Raleigh, North Carolina 2017

APPROVED BY:

__________________________________________________________
Dr. Gregory D. Buckner
Committee Chair

__________________________________________________________
Dr. Larry M. Silverberg

__________________________________________________________
Dr. Scott M. Ferguson

__________________________________________________________
Dr. Ralph C. Smith
DEDICATION

To Pam, Gus, and Eva.
BIOGRAPHY

Chris Elliott was born in 1973 in Asheville, North Carolina. He graduated from Enka High School in 1991 and then attended North Carolina State University (NCSU) where he majored in mechanical engineering and was a varsity wrestler. He received a bachelor’s degree from NCSU in mechanical engineering in 1996, and a master’s degree there in 1998, with a minor in mathematics. Chris and his wife Pam were married in 1996 and moved to Peoria, Illinois in 1998 where Chris worked for Caterpillar – first as a noise control engineer, then as a simulation and control system specialist. In 2003, their daughter, Eva was born in Peoria. In 2004 they moved back to North Carolina where Chris continued working for Caterpillar, specializing in simulation, control system development, and optimization. In 2005, their son, Gus was born in Raleigh. Chris completed a doctoral degree at NCSU in 2017.
ACKNOWLEDGMENTS

My family has been supportive through what has in recent years been a challenging combination of work and education. My advisor, Dr. Gregory Buckner, has helped to tailor a flexible learning experience for me and I appreciate his active involvement in my research. I am grateful to my committee and other NCSU faculty who have always provided friendly, helpful feedback and advice. Caterpillar has provided for me a flexible work arrangement, paid for prototypes, and enabled my research.
TABLE OF CONTENTS

LIST OF TABLES ...................................................................................................................VIII
LIST OF FIGURES ............................................................................................................... IX
LIST OF ALGORITHMS ...................................................................................................... XIII

CHAPTER 1 INTRODUCTION ............................................................................................... 1
  1.1 Background and Motivation ..................................................................................... 1
  1.2 Research Objectives ............................................................................................... 4
  1.3 Organization ............................................................................................................ 5

CHAPTER 2 METHODS FOR STREAMLINED FIREFLY OPTIMIZATION AND
INTERPRETATION: APPLICATION TO ENGINEERING DESIGN ........................................ 6
  2.1 Introduction .............................................................................................................. 6
  2.2 Methods .................................................................................................................. 10
    2.2.1 Firefly Algorithm Implementation ................................................................... 10
    2.2.2 The Cost-Sorted Distance Method ................................................................. 14
    2.2.3 Identifying CSD Cluster Champions ............................................................... 16
    2.2.4 Configurable Benchmark Function ................................................................. 18
    2.2.5 FA Parameter Tuning ...................................................................................... 21
    Benchmark Optimization: 1D, 2D, 3D, and 4D .......................................................... 26
  2.3 Example: Electromechanical Design Optimization ................................................ 33
    2.3.1 Active Magnetic Bearing and Controller ...................................................... 33
    2.3.2 FA Optimization of AMB+PID Design Parameters ................................... 36
  2.4 Discussion/Conclusions ......................................................................................... 39

CHAPTER 3 ROBUSTNESS CONSIDERATIONS ASSOCIATED WITH FIREFLY
AND THE COST-SORTED DISTANCE METHOD .............................................................. 41
3.1 Introduction

3.2 Methods for Quantifying Robustness

3.2.1 A Deterministic Approach

3.2.2 Monte Carlo Approach

3.2.3 Probability of Dominance

3.2.4 Correlation Between Robustness Measures and the Fireflies Clustered

3.3 Results

3.4 Discussion/Conclusions

CHAPTER 4  DESIGN OPTIMIZATION: ELASTOMERIC BAFFLE MRFD

4.1 Introduction

4.2 Methods

4.2.1 Magnetic Circuit Model

4.2.2 Electrical Circuit Model

4.2.3 Mechanical System Properties

4.2.4 Material Properties

4.2.5 Design Optimization

4.2.6 FEA Validation

4.2.7 Prototype Development

4.2.8 Experimental Validation

4.3 Results

4.4 Discussion/Conclusions

CHAPTER 5  CONCLUSIONS

5.1 Future Research Opportunities

REFERENCES

APPENDICES

APPENDIX A: ACTIVE MAGNETIC BEARING MODEL

APPENDIX B: PARAMETRIC LENGTH AND AREA EXPRESSIONS
APPENDIX C: EXPERIMENTS TO MEASURE B VS H................................. 116
APPENDIX D: MRFD PERFORMANCE EXPERIMENTS.............................. 118
APPENDIX E: MRFD PROTOTYPE PARTS..................................................... 120
LIST OF TABLES

Table 2.1  The 10 lowest cost results from a full-factorial FA parameter search. ........................................ 26
Table 2.2  Summary of 1D-4D benchmark optimization results. Optimization was completed for 1000 each
of 1D, 2D, 3D, and 4D randomized benchmark problems. ........................................................................ 32
Table 2.3  FA optimization champions and associated of the engineering design example. ...................... 37
Table 3.1  Robustness measures, (3.7) minima ................................................................................................. 51
Table 3.2  linear correlation measures for $PFF$ vs $PR$. .................................................................................. 56
Table 3.3  linear correlation measures for $PFF$ vs PoD. .................................................................................. 59
Table 4.1  Parameters of (4.2), corresponding to Figure 4.8.b. ....................................................................... 70
Table 4.2  Optimization variables are a partially modified version of the design parameters with the
addition of magnetic flux density ........................................................................................................ 76
Table 4.3  FEMM settings for solving the magnetic field problem using FEA ............................................. 82
Table 4.4  The optimization variables, their units, firefly lower and upper limits, and optimal results after
conjugate gradient optimization for each of the cluster champions of Figure 4.18. .......................... 90
Table 4.5  Important results for the designs of Figure 4.19. .......................................................................... 91
Table 4.6  Predicted $|B|$ for the optimal design. ............................................................................................ 93
Table A.1  Parameters of (4.2), corresponding to Figure 4.8.a. ................................................................... 114
Table A.2  Parameters of (4.2), corresponding to Figure 4.8.c. ................................................................. 115
LIST OF FIGURES

Figure 1.1  a. A drive-by-wire construction machine.  b. The operator station with adjustable, seat-mounted, electronic joysticks. Images courtesy of Caterpillar................................................................. 2

Figure 2.1  Equation (2.4) for five different values of $\alpha$ ................................................................. 13

Figure 2.2  Histograms indicating distributions for the output of (2.4) ......................................................... 14

Figure 2.3  a. A typical 2D cost function contour plot with fireflies (white data points) in their final migration positions – fireflies have clustered near three local minima; b. CSD plot ..................... 15

Figure 2.4  CSD plot views ................................................................................................................................. 17

Figure 2.5  Examples of a. 1D, b. 2D, and c. 3D cost functions from (2.5) with randomly generated parameters ............................................................................................................................... 20

Figure 2.6  Averages and standard deviations from 25 batches of $\text{cost}_x$ as functions of the number of optimizations in each batch for fixed FA tuning parameters ................................................................. 25

Figure 2.7  1D cost function (2.5) with initial and final firefly positions and corresponding CSD plots.... 27

Figure 2.8  2D cost function (2.5) with initial and final firefly positions and corresponding CSD plots.... 29

Figure 2.9  3D cost function (2.5) with initial and final migration firefly positions and corresponding CSD plots ................................................................................................................................. 30

Figure 2.10  4D cost function (2.5) with initial and final firefly positions and corresponding CSD plots... 31

Figure 2.11  Schematic of a longitudinal AMB, where the separation between stator and mover is maintained by a PID controlled coil voltage ......................................................................................... 34

Figure 2.12  AMB design example with initial and final CSD plots and designs resulting from firefly optimization. Table 2.3 provides detail regarding the champions and related performance measures ........................................................................................................................................ 36

Figure 2.13  Transient simulation results ......................................................................................................... 38

Figure 3.1  An example of the simplest possible multimodal objective function........................................... 47

Figure 3.2  Uncertainty propagation for each of two minima of a typical 1D benchmark function .......... 49

Figure 3.3  Output distributions overlap somewhat .......................................................................................... 50
Figure 3.4  Typical 2D optimization results. In each plot, champions are labeled 1 through 4 from lowest to highest objective function value.

Figure 3.5  PFF vs. PR: histograms of slopes from least-square-error line fits. Estimated probability density and cumulative probability distributions are also shown.

Figure 3.6  PFF vs. PR: all data points from 1000 firefly optimizations of randomized benchmark functions.

Figure 3.7  PFF vs. PoD: histograms of slopes from least-square-error line fits.

Figure 3.8  PFF vs. PoD: all data points from 1000 firefly optimizations of randomized benchmark functions.

Figure 4.1  Examples of pressure-driven and direct shear modes for MRFDs.

Figure 4.2  Three examples of MRFD types.

Figure 4.3  An absorbent matrix MRFD.

Figure 4.4  The elastomeric baffle MRFD design.

Figure 4.5  Some practical details of the elastomeric baffle MRFD design.

Figure 4.6  Parameterized spool and housing geometry used for design optimization.

Figure 4.7  Lumped parameter magnetic circuit model. Magneto motive force drives magnetic flux through a series of n reluctances.

Figure 4.8  Detailed parametric geometry for a magnetic circuit model corresponding to Figure 4.6. Evaluation of one-quarter of the cross-section is sufficient due to symmetry of the design.

Figure 4.9  Lumped parameter electric circuit model. Voltage source drives current through a series of resistance and inductance.

Figure 4.10  Shear parametric dependencies of MRF.

Figure 4.11  Magnetic parametric dependencies of MRF and low-carbon steel.

Figure 4.12  The magnetic circuit model results in a transcendental set of equations.

Figure 4.13  A benchmark 2D cost function and the corresponding CSD plot.

Figure 4.14  Solid model images of the prototype MRF damper.

Figure 4.15  A photograph of the prototype MRFD.

Figure 4.16  Experimental setup.
Figure 4.17 Results summary for the modified firefly portion of Algorithm 2.1. ........................................ 86

Figure 4.18 Distances (Euclidean norms) of fireflies from the lowest cost result, for the last migration...87

Figure 4.19 Cost and geometry of the five cluster champions from Figure 4.18 (blue) and their conjugate gradient optimal counterparts (red). .................................................................................................. 89

Figure 4.20 FEA result for the best design from Table 4.5 showing $|B|$ contour lines in the ferromagnetic spool and housing. ................................................................................................................................ 92

Figure 4.21  a. Cross-section of the spool and housing showing locations sampled for FEA results in Table 4.6.  b. FEA predicted $|B|$ in the MRF, between piston and cylinder. ............................................... 93

Figure 4.22  Cyclic joystick motion with current commands from 0 to 0.5 A in 10 steps: a. force vs. time and b. force vs. displacement. ............................................................................................................. 94

Figure 4.23  A step in current command: a. force vs. time and b. displacement vs. time. ........................... 95
Figure A. 1 Electromechanical energy conversion is modeled with a lossless magnetic field coupling electrical and mechanical subsystems.................................................................108

Figure A. 2 Magnetic field energy is assumed a function of states $\lambda$ and $\gamma$..............................................110

Figure A. 3 Lumped parameter electric circuit model............................................................................................111

Figure A. 4 A research assistant records data to characterize B vs. H for a ferrous material. .....................116

Figure A. 5 Schematic for measurement of sample B-H curve...........................................................................116

Figure A. 6 Material sample dimensions (mm)....................................................................................................117

Figure A. 7 A research assistant actuates the MRFD to measure device performance.................................118

Figure A. 8 Six seconds of typical MRFD force and displacement data..............................................................119

Figure A. 9 Assembled MRFD parts..................................................................................................................120

Figure A. 10 The ferrous cylinder built to the optimal geometry specified in Table 4.4.............................121

Figure A. 11 The optimal ferrous spool with geometry specified in Table 4.4..............................................122

Figure A. 12 Two of these aluminum shafts are used to suspend the spool....................................................123

Figure A. 13 These non-ferrous caps are attached to either end of the ferrous housing..............................124

Figure A. 14 A spacer for the rolling elastomeric baffle...................................................................................125

Figure A. 15 A “swabable” check valve used for injecting/removing MRF by syringe..............................126

Figure A. 16 Stainless steel adapter compatible with the check valve shown in Figure A. 15................126

Figure A. 17 This spherical bearing secured the MRFD to the base block as shown in Figure A. 9...........127

Figure A. 18 A force sensor was threaded between a rod end and the shaft shown in Figure A. 12........128

Figure A. 19 This signal conditioner was used to convert sensor force measurements for the ECM. .....129
LIST OF ALGORITHMS

Procedure 2.1  Tuning FA parameters for a broad class of optimization problems......................... 22

Algorithm 2.1  Enhanced FA pseudo code. ............................................................................................ 11

Algorithm 2.2  Random additional motion applied to fireflies in step 5.2.2 of Algorithm 2.1............ 12

Algorithm 2.3  The histogram approach used in step 5.4.1 of Algorithm 2.1 to identify cluster champions. ............................................................................................................................................. 18

Algorithm 2.4  A function for quantifying the extent to which a set of design vectors $\mathbf{x}$ have converged near the seeds $c$ of the function $f$ ................................................................................................................................................. 23

Algorithm 4.1  Optimization pseudo code. .......................................................................................... 79
Chapter 1

Introduction

1.1 Background and Motivation

The commercial application of microcontrollers and servo-controlled actuators has led to the popularity of digital drive-by-wire interfaces for machine control – where operators manipulate input devices (levers, pedals, steering wheels, etc.) that are electronically sensed, not mechanically coupled with actuators. Figure 1.1 shows a modern construction machine and its operator station, with drive-by-wire interface. This type of interface has many benefits (such as the accommodation of complex logic, modularity, design flexibility, minimal noise/weight/maintenance, etc.), but another result is that operator feel (haptic feedback) is not enhanced by the interface itself. In fact, considerable attention must be paid in designing drive-by-wire interfaces that enhance, rather than negatively impact, machine controllability.
Caterpillar’s small track-type tractor (shown in Figure 1.1) became fully drive-by-wire in the early 2000s. However, digital drive-by-wire (or fly-by-wire) control was originally developed for aerospace applications in the 1950s-1970s. The first piloted system demanding fully fly-by-wire control (as opposed to mechanical operator control, tele-operator control, or completely automatic computer control, which were all used in the same timeframe) was NASA’s lunar module, built in the 1960s. The vehicle was designed to be piloted by a computer guidance system, but with the ability for the human pilot to intervene (incidentally, this feature was crucial to the success of the Apollo moon landings – even preventing an aborted mission on the first lunar landing [1]). The lander provided electronic, rather than mechanical, operator override. NASA proceeded in the 1970s to develop fully fly-by-wire aircraft control [1] and many such aircraft systems have since been developed.

Early fly-by-wire systems were also first to encounter the extra development which must be undertaken to successfully design human interfaces for piloted machines. A precursor to fully fly-by-wire systems was the yaw-control of the CF-105s, an experimental Canadian military
aircraft, flown in 1958. Due to the extreme performance demanded of this airplane, computer augmented yaw control was necessary to provide stable operation. The pilot interface was at first made without an attempt to emulate traditional mechanically controlled flight systems. However, to gain pilot acceptance, it later had to be modified with springs to emulate the feel of cable control [1]. A similar experience was had for NASA’s fully fly-by-wire system, as documented by Tomayko, “This system had no feedback to the pilot, so a set of springs, bobweights, etc., was arranged to give artificial “feel” similar to that in a cable-only control system… In the early days of fly-by-wire, engineers thought that such an artificial feel system would be unnecessary, probably reasoning that the electronic feedback would be sufficient for control.” [1].

Iteration of this sort is to be expected with the transition from mechanical to electronic control interfaces. Haptic feedback is inherent in many mechanical interfaces, so designers updating to digital control may be unaware of it, fail to fully appreciate the role it serves, or may even view it as a problem to be solved (for example, if force feedback causes operator fatigue). With drive-by-wire digital control, haptic feedback must be intentionally created by the system designer and it is likely to be lacking if the designer has not explicitly considered it.

Another problem encountered by NASA in development of fly-by-wire is unwanted feedback. Biodynamic feedthrough or operator induced oscillation (OIO) [2] is machine vibration where the operator unwillingly participates in unstable closed-loop command feedback. This happens when the machine accelerates the operator in a mode and at a frequency that he or she cannot avoid passing motion through the input device. It is a common nuisance for many machines, but can have devastating consequences in aircraft control where sustained or escalating oscillation and can lead to loss of control. OIO may be mitigated by reducing the efficiency of the feedback path e.g. by better isolating the operator from the machine or the machine control from the operator. This remains an active research topic (e.g. [3], [4], [5]).

Possibilities for improving haptic feedback and addressing OIO include passive mechanical enhancement through improved damping and spring forces; and active solutions such as force feedback through motors and other electronically controlled actuators. This research seeks to
develop advanced design optimization methods that can be applied to an electronically controllable magnetorheological fluid device (MRFD) suitable for these purposes.

Designing electromechanical devices and other similarly complex engineering systems can be challenging. Methods for predicting a design’s performance are often either computationally prohibitive or overly simplistic; and predicting a given design’s performance is usually a small part of the challenge. The designer must solve the inverse of this problem, which is to select the design that gives the best performance possible. This requires a method for predicting any feasible design’s performance (usually a parameterized, many-degree-of-freedom model), and a way of intelligently choosing from the, perhaps infinite, design set. Furthermore, the meaning of “best performance possible” may not be straightforward. Real world designs are almost always the result of compromises between multiple conflicting performance measures. The goal of this dissertation is to develop and demonstrate an optimal design tool set to address these challenges.

1.2 Research Objectives

The overall goal of this research is to develop and demonstrate methods for optimal engineering design, and ultimately to use these methods to design a novel MRFD. The specific objectives are to:

1. Develop a new firefly algorithm (FA) suitable for multimodal optimization.
2. Demonstrate the cost-sorted distance (CSD) method for design space exploration.
3. Tune the FA to reliably cluster “fireflies” at multiple objective function minima.
4. Demonstrate FA optimization and CSD visualization for benchmark problems and electromechanical system design.
5. Examine the usefulness of FA and CSD for design under uncertainty.
6. Introduce a novel MRFD concept and optimize it using the methods developed.
1.3 Organization

Chapter 2 introduces novel methods for optimal engineering design. A state of the art FA that incorporates recommendations from recent literature with further improvements described here is developed. A complimentary technique, the CSD method, for visualization and clustering is also introduced and demonstrated.

In Chapter 3, the extent to which FA and CSD are useful for addressing uncertainty in the design process is investigated. Various robustness measures are described and the hypothesis that the percentage of fireflies per cluster correlates linearly with robustness of the associated designs is tested.

Chapter 4 describes the concept of a novel elastomeric baffle MRFD and its optimal design using the methods developed in earlier chapters. Numerical and experimental validation of the design is documented. Concluding remarks and avenues for related future research are provided in Chapter 5.
Chapter 2
Methods for Streamlined Firefly Optimization and Interpretation: Application to Engineering Design

2.1 Introduction

Numerical optimization is often the only practical means for solving nonlinear engineering design problems with multiple degrees of freedom and conflicting objectives. Although sophisticated optimization methods for such applications have been developed, it can be difficult for a designer to use them to explore the design space. This chapter describes a methodology that can be used to easily identify and assess multiple attractive designs for these problems.

Design optimization can be facilitated by development of a scalar objective function $f(x)$ of the design variables $x = [x_1, x_2, ..., x_n]$ which quantifies design fitness. By convention, a design is optimal if it minimizes $f(x)$ and satisfies inequality and equality constraints, $g(x)$ and $h(x)$:

$$
\min_{x} f(x) \\
\text{subject to:}
$$

$$
g_i(x) \leq 0, \ i \in [1, 2, ..., n_g] \\
h_j(x) = 0, \ j \in [1, 2, ..., n_h]
$$

(2.1)

where $n_g$ is the number of inequality constraints and $n_h$ is the number of equality constraints. Multiple design objectives may be incorporated using a weighted sum, $f(x) = \sum_i K_i f_i(x)$,
where $K_i$ is a scalar weighting factor. The objective and constraint equations may be combined to form a penalty function:

$$p(x) = f(x) + K_g \sum_{i=1}^{n_g} \max(0, g_i(x)) + K_h \sum_{i=1}^{n_h} \max(0, |h_i(x)| - \text{TOL}) \quad (2.2)$$

where $K_g$ and $K_h$ are penalty factors used for inequality and equality constraint violations, respectively, and $\text{TOL}$ is the equality constraint convergence tolerance. Minimizing (2.2) is thus equivalent to solving (2.1).

Solutions to (2.1) or (2.2) may be approximated using a variety of optimization methods; gradient-based and population-based methods are two types commonly used. Gradient approaches \cite{6} (e.g. steepest descent, conjugate gradient, Newton-based methods) can be effective for smooth, differentiable functions, but the rate and stability of convergence may be inferior to other methods for the multimodal or discontinuous cost functions frequently associated with multi-degree-of-freedom design optimization. Population-based methods \cite{7} such as genetic algorithms, particle swarm optimization, and other nature inspired algorithms take advantage of large and diverse populations of solutions to overcome these limitations.

The firefly algorithm (FA), introduced by Yang \cite{8} in 2009, is a population-based optimization method inspired by firefly behavior, where design vectors ($x$’s, the fireflies) migrate toward better fit neighbors in the design space. FA can be tuned to favor local solutions, rather than more exhaustively seeking the global optimum, and thus can be used to identify promising design alternatives. The core concept of FA is that the cost function of a design vector $x_i$ can be represented as the light intensity of a firefly: a design vector $x_j$ with a more favorable cost function is represented by a firefly with a brighter light intensity (i.e. $f(x_j) < f(x_i)$). At each design iteration, fireflies with lower intensities migrate toward brighter neighbors. The distance traversed is exponentially dependent on their spatial separation:

$$x_i(k + 1) = x_i(k) + \alpha \text{rand}(0) + \beta e^{-\gamma \|r_{ij}\|^m} r_{ij} \quad (2.3)$$

where $x_i(k)$ is a firefly’s current location and $x_i(k + 1)$ is its next location; the vector between fireflies is $r_{ij} = x_j(k) - x_i(k)$; $\alpha, \beta, \text{and} \gamma$ are tuning parameters, possibly varying with migration.
number; and \( \text{rand}() \) generates a vector of randomly distributed numbers, enhancing design space exploration.

Yang [8] discussed FA parameter tuning, compared and contrasted it with particle swarm optimization (PSO) and genetic algorithms (GA), and used 10 benchmark problems to demonstrate that FA can be superior to these alternative methods. Yang showed that FA converged to the global minimum in 99% of his simulations, 11% more often than GA and 6% more often than PSO, while requiring only 19% of GA’s function evaluations and 40% of PSO’s.

Improved versions of FA have been applied to a wide array of problems in recent years. Of note, Lukasik and Zak [9] in 2009 incorporated random variation in proportion to firefly distance from search space boundaries, and Yang [10] in 2010 proposed generating randomized step sizes from the Lévy distribution [11] (based on an inverse power law), providing a more thorough search of the design space. These improvements are incorporated in the algorithm developed here. Fister, et al. [12] in 2013 surveyed 172 articles related to FA, and documented avenues for continued research and development. The authors summarized FA method contributions and classified them as classical, modified or hybrid types. They also documented and categorized FA application examples from literature. In conclusion, they highlighted the method’s simplicity, flexibility and versatility, but noted that ideal FA parameter tuning can itself be a difficult optimization problem; more studies are needed to improve parameter specification.

Although population-based methods such as FA can be employed to solve “real world” multi-dimensional design optimization problems, interpreting results and identifying good alternative designs (i.e. local optima) can be challenging. These methods produce large sets of design vectors, and while attractive alternative solutions may be included, only the lowest cost solution is obvious. Dimensionality reduction methods [13] [14] such as principal component analysis (PCA), correlation graphs, and Pareto charts may be useful for both visually and programmatically discovering trends in data and parameters that account for variations, but these methods are not suitable for finding minima. Multi-objective optimization
methods typically involve identifying a Pareto frontier [15] of non-dominated designs from a large population of results. The Pareto frontier is a useful programmatic and graphical analysis tool for problems where the designer needs to understand trade-offs between two or more competing cost functions, but while a potentially infinite set of optimal designs can be identified, little information may be gained regarding dominated minima. In cases where population-based methods are capable of grouping design vectors near attractive solutions, these groups may be identified using clustering methods [16], which are optimization algorithms themselves, but can be computationally prohibitive.

In recent literature, tools to assist the designer in interpreting optimization results and finding attractive alternate designs have begun to emerge. Mattson and Messac [17] proposed a methodology for identifying attractive designs from multi-objective optimization results and demonstrated interactive software for down-selecting from the design space. Stump, et al. [18] developed “visual steering commands that help decision makers form their preference while exploring the trade space (i.e., “shopping”) to focus in on regions/points of interest as their preference sharpens.” Daskilewicz and German [19] proposed a set of features of interactive decision-support tools to promote effective and practical engineering design, and implemented these in an open-source computer program designed to be part of a user-in-the-loop optimization process.

This chapter describes a computationally efficient FA that incorporates a new method for identifying local minima. Computational efficiency is enhanced because, in addition to including improvements from recent literature [9] [10], processing order has been improved. First, randomized motion is applied only as often as necessary (once per firefly, each migration), incorporating step sizes drawn from exponential distributions. This allows for significant random motion and reduces redundant motion relative to other implementations that do so during each interaction with every lower cost design. Second, fireflies are sorted by cost prior to migration so that firefly dominance is known a priori. This enables parallel processing during migration and reduces objective function evaluations relative to other implementations that require comparisons of firefly pairs during migration. A novel approach
for visually identifying local minima is introduced: a plot of the distances between cost-sorted fireflies (the “cost-sorted distance” or CSD plot). The CSD plot is shown to reveal clusters near local minima, making the best design associated with each cluster (the local champion) readily apparent to the designer. A methodology for tuning FA parameters is shown to achieve clustering of results at the objective function minima. Finally, the tuned algorithm is demonstrated using benchmark problems and a “real world” electromechanical case study.

2.2 Methods

This section details the FA enhancements for imparting random firefly motion, the CSD plot and its programmatic implementation, the parametric benchmark function used to tune FA, and the tuning method.

2.2.1 Firefly Algorithm Implementation

Algorithm 2.1 is pseudo-code for an enhanced FA incorporating improvements to the original algorithm [8] suggested in the literature [9] [10] [11] [7]. Further improvements, highlighted in red, include normalization of the design vectors, parameters that can change throughout migrations, and randomized motion performed only as often as necessary, incorporating step sizes drawn from exponential distributions.
Algorithm 2.1 Enhanced FA pseudo code. $\alpha, \beta,$ and $\gamma$ are the FA tuning parameters that may vary with migration. Improvements relative to the literature are highlighted in red.

Parameters $N, M, \alpha, \beta,$ and $\gamma$ affect convergence to local minima of cost function $f$. The number of cost function evaluations is proportional to the number of fireflies $N$ and number of migrations $M$. Parameters $\alpha, \beta,$ and $\gamma$ can be fixed (in which case they are scalars) or be specified functions of the migrations $M$ (e.g. $\alpha$ can be linearly interpolated between $\alpha_1$ on the first migration and $\alpha_2$ on the last migration, in which case they are vectors). The probability of significant random motion is determined by $\alpha$; if it is large (e.g. $\alpha > 20$) the probability is small.
that fireflies move appreciably (see Figure 2.2). Attraction to brighter fireflies is dictated by parameters $\beta$ and $\gamma$. Large $\gamma$ (e.g. $\gamma > 20$) results in insignificant attraction to distant fireflies; $\beta \in [0,1]$ is a scaling factor (typically unity). Normalization of the design vectors can be used to enhance generalization at the expense of computational efficiency.

There are notable differences between Algorithm 2.1 and those previously documented in the literature. First, firefly locations and costs are evaluated prior to migration; they are sorted by cost and their relative distances are computed. Fireflies are not compared during migration. The number of cost function evaluations is therefore reduced by 50% (compared to evaluating the cost and location of each pair of fireflies within loop 5.2). This also enables efficient vector operations and parallel computing of migrations. Second, random movement is applied to each firefly outside of loop 5.2.1, after the movement due to attraction to all brighter fireflies. Third, some of the worst performing fireflies after migration may be replaced with variations of those identified as cluster champions. The benefit is a more thorough search near potential local minima at the expense of reduced exploration.

Algorithm 2.2 is pseudo-code to generate random motion in step 5.2.2 of Algorithm 2.1. It includes randomly distributed step sizes, between lower and upper limits, based on an exponential distribution.

```
r and(a, x):
6. Given $x$ an n-d vector, and $\alpha$ a positive real number:
7. define $\rho \leftarrow \text{runif}(n, 0, 1): n \times 1$ uniformly distributed random numbers, [0,1]
8. FOR $k = 1, 2, \ldots, n$
8.1. IF ($\rho_k < 0.5$)
8.1.1. define $r_k \leftarrow -|x_{l_k} - x_k| \exp(\alpha, rnd(1))$
8.2. ELSE
8.2.1. define $r_k \leftarrow |x_{l_k} - x_k| \exp(\alpha, rnd(1))$
9. Return the sum, $r + x$
```

Algorithm 2.2 Random additional motion applied to fireflies in step 5.2.2 of Algorithm 2.1.
In Algorithm 2.2, $\text{rnd}(1)$ is a uniformly distributed random number $\in [0,1]$ and

$$f_{\text{exp}}(\alpha, t) = e^{-\alpha t} - e^{-\alpha}$$

(2.4)

is $\leq 1$ in the range of interest, and intersects the abscissa at 1 (Figure 2.1).

![Figure 2.1 Equation (2.4) for five different values of $\alpha$. In Algorithm 2.2, $x \in [0,1]$ is a random number and $f_{\text{exp}}(\alpha, x)$ is multiplied by the distance to a design vector boundary.](image)

Figure 2.2 shows two histograms resulting from (2.4) given 10000 uniformly distributed random selections of $x \in [0,1]$ for two different $\alpha$. The larger ($\alpha = 20$) clearly generates output near the origin much more frequently. Thus, in Algorithm 2.1, while there is always the possibility of firefly movement anywhere between $x_L$ and $x_U$, fireflies are less likely to move far from their nominal positions as $\alpha$ grows larger.
2.2.2 The Cost-Sorted Distance Method

Steps 5.4 and 6 of Algorithm 2.1 are accomplished by sorting, according to cost, the distances between each pair of fireflies (\(|d|\), the Euclidean norm of the relative position vector). This cost-sorted distance (CSD) approach can be efficiently incorporated into FA, since the costs and distances are already required computations. For problems with multiple local minima, fireflies typically converge in clusters to these minima, with similar designs being separated by relatively small distances in the design space. For this reason, their cost-sorted distances appear as blocks or strips of similar color in a 3D scatter plot (e.g. Figure 2.3.b). Promising design alternatives can be selected from the lowest-cost members (champions) of each cluster.
Figure 2.3  a. A typical 2D cost function contour plot with fireflies (white data points) in their final migration positions – fireflies have clustered near three local minima; b. CSD plot: distances between normalized fireflies, arranged in order of cost. Clusters of fireflies appear as blocks of uniform color; their relative performance corresponds to distance from the origin.

An advantage of the CSD plot is that it provides a visualization tool for down-selecting from a large population of results. Figure 2.3, for example, shows for a two-degree-of-freedom system (where the fireflies can also be overlaid on a cost function contour plot) how the method can be used to find promising results. Figure 2.3.a shows the cost function along with 50 fireflies (white data points) in their final migration positions, near local minima. The corresponding CSD plot is shown in Figure 2.3.b. Note that 21 of the fireflies have converged near the lowest cost design, point A, and appear as purple data points in the bottom CSD row; 13 fireflies have converged near point B and are green in the bottom CSD row; 14 fireflies have converged near point C and are yellow in the bottom CSD row; two fireflies labeled D have not converged near local minima – they are blue in the bottom row of Figure 2.3.b. Fireflies that are close to one another generate purple CSD data points, for the “rainbow” color scaling in Figure 2.3.b, and fireflies far from one another are red. A banded (or plaid) CSD
plot indicates the likely presence of local minima. This ability to clearly display clusters of results and their champions is a significant benefit for designers dealing with high order problems (3D and higher), where cost functions can be difficult or impossible to visualize graphically.

2.2.3 Identifying CSD Cluster Champions

The CSD method is automated in step 5.4 of Algorithm 2.1, where local champions are extracted from clusters of fireflies. If desired, variations of these champions may replace the population’s lowest cost members after a subset of firefly migrations – this could accelerate convergence toward identified minima at the expense of reducing the scope of the search. For the examples discussed later in this chapter, replacement of the highest-cost fireflies is enabled for migrations ≥ \( M/3 \), and the replacement fireflies are randomly distributed within 1% of the distance between the champion’s elements and those of the lower or upper bounds.

To better illustrate how the CSD method can be used to identify promising design alternatives, projections of a typical plot are presented in Figure 2.4. Figure 2.4.a shows the typical vertical projection; Figure 2.4.b shows a rotated view with black lines connecting the first row of data; Figure 2.4.c shows only this first row of data. Figure 2.4.c data are distances from the lowest cost design to each of the other designs – apparent discontinuities easily visible in this plot indicate that FA has likely found local minima.
Clusters of results, indicated by patches of uniform color in Figure 2.4.a and stratified layers in the other plots, can be programmatically identified using the histogram approach of Algorithm 2.3. First, the population is sorted by distance into bins, and the lowest cost member of the first bin is recorded. Rows and columns of data with distances in the first bin are then removed from the data and the process is repeated until the desired number of bins have been processed or the fireflies have been depleted. Red boxes in Figure 2.4.c mark the champions of identified clusters for this data. It is notable that a histogram of only the first row of distance data would likely have missed the last champion in this dataset since it and the third are similar distances from the first champion.
Algorithm 2.3 The histogram approach used in step 5.4.1 of Algorithm 2.1 to identify cluster champions.

1. Given a set of cost-sorted fireflies $x$, and a maximum number of histogram bins $h$:

2. Compute the CSD matrix: $CSD_{i,j} \leftarrow |d|_{i,j}$ for all combinations $i, j$ of fireflies.

3. Compute the maximum distance between fireflies $dh \leftarrow \max(CSD)/h$.

4. The lowest cost firefly is always the first champion: $x^*_1 \leftarrow x_1$.

5. WHILE bins recorded is less than $h$ and there are still unprocessed fireflies,

   5.1. Extract the first column of the CSD matrix: $ds \leftarrow CSD^{(1)}$.

   5.2. FOR $k = 1, 2, ..., rows(ds)$, remove fireflies in the current bin:

      5.2.1. IF $ds_k < dh$, eliminate $x_k$.

   5.3. Record the champion of the next bin: $x^*_{next} \leftarrow x_1$.

   5.4. Compute the CSD matrix: $CSD_{i,j} \leftarrow |d|_{i,j}$ for all remaining fireflies $i, j$.

6. Return the champions: $x^*$.

2.2.4 Configurable Benchmark Function

Numerous benchmark functions for optimization algorithm evaluation and development have been documented [7], [11]. A special type is required for assessing an algorithm’s capability to find alternative solutions to the global minimum. A parametric, multi-degree-of-freedom, multimodal benchmark function (2.5) is introduced for this purpose:

$$ f(m, n, a, b, c, d, x) = \left(1 + d \sum_{i=1}^{n} (x_i - 0.5)^2 \right) \prod_{j=1}^{m} \left(1 - a_j e^{-b_j |x - e^{(j)^2}|} \right). $$

(2.5)

This is an n-dimensional (nD) spherical basis function multiplied by m nD exponential functions for generating local minima. Vectors $a$ and $b$ are mx1 factors for specifying depth and breadth, respectively, of local minima, $e$ is an nxm array of minimum locations (seeds), and $d$ is a scalar that affects the shape of the spherical basis function.
Some notable properties of (2.5) include: the basis function has a minimum value of 1.0 at $x_i = 0.5$ for $i = [1, 2, \ldots, n]$, corresponding to the center of the design space (without loss of generality, since design variables are to be normalized $[0, 1]$); locations of local minima, neglecting cross-effects, are specified by $e$ where $c_{ij} \in [0, 1]$ for $j = [1, 2, \ldots, m]$; function values at those locations, neglecting cross-effects, are specified by $1 - a_j$ where $a_j \in [0, 1]$; for $d > 0$, the lowest possible minimum value at any point is 0; breadth of the local minima is dictated by $b$, where larger values make them more narrowly focused (in the cases explored, $b_j \in [10, 250]$). The parametric nature of (2.5) enables programmatic generation of nD cost functions with specified (randomly if desired) shape and seed locations.

Figure 2.5 shows 1D, 2D, and 3D plots of (2.5) for different randomly-generated parameters. Seed locations for these plots are shown in red. In like manner, cost functions can be generated for problems of arbitrary dimension with approximately known minima. This configurability can be used to explore the tuning robustness of an optimization algorithm in a measured way: the successful algorithm should routinely locate one or more of these minima.
Figure 2.5  Examples of a. 1D, b. 2D, and c. 3D cost functions from (2.5) with randomly generated parameters. The number of seeds is $2n$ for each of these, where $n$ is the degrees of freedom. Seeds, shown in red, were randomly generated within prescribed subsections of the design space to prevent overlapping minima. Cost is indicated by density of the points in the 3D case; darker is lower.

In Figure 2.5, parameters are generated from uniform random distributions as follows: $a = \text{runif}(2n, 0.5, 1)$, $b = \text{runif}(2n, 25, 250)$, $c^{(j)} = \text{runif}(n, 0, 1)$ for $j \in [1, 2..2n]$, $d = 1$, where $n$ is the dimension, and $\text{runif}(n, 0, 1)$ generates an nD vector of uniformly distributed random data, [0,1].
Seeds, shown in red, were randomly generated within prescribed subsections of the design space to prevent substantially overlapping minima.

### 2.2.5 FA Parameter Tuning

While the FA method is capable of identifying local minima, tuning its parameters to reliably do so can be challenging. A configurable benchmark function may be used as part of an optimization process (Procedure 2.1) to tune FA parameters for prototypical optimization problems. The prototypical problem is represented by (2.5) with parameters \( m, n, a, b, c, \) and \( d \) chosen randomly from a predetermined range as in Figure 2.5. Optimization using Algorithm 2.1 with FA parameter set \( N, M, \alpha, \beta, \) and \( \gamma \) is performed and its result assessed. Results for different FA parameter sets are compared and the most reliably effective tuning set is chosen.
Procedure 2.1  Tuning FA parameters for a broad class of optimization problems.  Statistical methods are employed because the FA and the prototypical problem with randomized parameters (2.5) are stochastic.  The resulting tuning is desirable in the sense that it minimizes the cost function described in Algorithm 2.4.

1. Develop a cost function $\text{cost}_x$ to assess the fitness of given firefly population for indicating multiple minima of typical parametric benchmark functions.  i.e. ideally, each firefly should be near a local minimum; multiple local minima should have been found; etc.  This cost will be associated with a given FA tuning parameter set.
2. Choose a representative FA tuning parameter set – e.g. the designer’s best guess at good tuning.
3. Determine the number of evaluations required to achieve repeatable results (results are stochastic and must be averaged):
   3.1. FOR $N = 10, 25, 50, 100, 250, 500, 750, 1000$ (for example)
      3.1.1. FOR $i = 1, 2, \ldots, i_{\text{last}}$ (e.g. $i_{\text{last}} = 25$)
         3.1.1.1. Compute the mean $\text{cost}_x$ from $N$ optimizations of randomly generated benchmark problems using the representative FA tuning parameter set:
            \[ \text{mean}_i \leftarrow \text{mean} (\text{cost}_x) \]
      3.1.2. Compute the average and standard deviation of $\text{mean}_i$.
            \[ \text{mean}_N \leftarrow \frac{\sum_i \text{mean}_i}{i_{\text{last}}} = \text{mean} (\text{mean}_i) \]
            \[ \sigma_N \leftarrow \text{std} \text{dev}(\text{mean}_i) \]
4. Analyze $\text{mean}_N$ and $\sigma_N$.  Choose $N^*$ as small as possible but so that $\text{mean}_N$ is repeatable for a given FA parameter tuning set.  Consider for comparison, $\text{mean}_N$ associated with a random (initial, prior to optimization) firefly population.
5. Select a range for each FA tuning parameter and perform a full factorial search for the best tuning set – that which achieves minimum $\text{mean}_N(N^*)$.
   5.1. For each FA tuning set:
      5.1.1. Perform $N^*$ optimizations of randomly generated benchmark problems.
      5.1.2. Record $\text{mean}_N(N^*)$.
   5.2. Select the FA tuning parameter set with minimum $\text{mean}_N(N^*)$.

Procedure 2.1  Tuning FA parameters for a broad class of optimization problems.  Statistical methods are employed because the FA and the prototypical problem with randomized parameters (2.5) are stochastic.  The resulting tuning is desirable in the sense that it minimizes the cost function described in Algorithm 2.4.

Step 1 of Procedure 2.1 requires assessment of FA tuning.  Algorithm 2.4 is designed for this; $\text{cost}_x(f, x, c)$ where $f$ is an arbitrary instance of (2.5), its parameters $a$, $b$, and $c$ randomly
generated each time it is called, and $x$ the set of final firefly locations. Tuning parameters that minimize $cost_x$ provide the highest likelihood that fireflies will migrate to multiple minima for problems of the type (2.5), and provide a basis for choosing initial tuning for generic problems of similar complexity. It should be emphasized that FA and (2.5) with randomly generated parameters are stochastic; therefore, $cost_x$ for a given tuning parameter set is averaged over multiple runs to minimize the effects of variation (steps 1 through 4 of Procedure 2.1 involve determining a sufficient number of runs). A full factorial optimization (as specified in Procedure 2.1, step 5) or other robust method may be employed in finding tuning parameters that minimize Algorithm 2.4 most reliably.

<table>
<thead>
<tr>
<th>$cost_x(f, x, c)$:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Given $N$ normalized, cost-sorted $nD$ design vectors $x$ and a scalar cost function $f$ with $m$ cost-sorted $nD$ seed locations $c$, assess the extent to which design vectors have converged upon $f$’s minima:</td>
</tr>
</tbody>
</table>
| 2. FOR $i = 1, 2, ..., N$
| 2.1.1. $dx_{ci} \leftarrow$ distance from $x_i$ to the nearest min location |
| 3. FOR $j = 1, 2, ..., m$
| 3.1.1. $dc_{cj} \leftarrow$ distance from $c_j$ to the nearest design vector |
| 4. $dcx \leftarrow$ subset of the best few elements of $dcx$, if desired |
| 5. Return: $\text{mean}(dcx) + \text{mean}(dcx) + f(x_0) - f(c_0)$ |

Algorithm 2.4 A function for quantifying the extent to which a set of design vectors $x$ have converged near the seeds $c$ of the function $f$. Returned is a sum of the average distance from each design vector to its closest seed location, the average distance from each of a subset of minima to its nearest design vector, and the difference between the cost function of the best design vector and the global minimum. Firefly tuning is sought which minimizes this cost function.

The FA (Algorithm 2.1) was tuned using Procedure 2.1. The cost associated with each tuning parameter set was averaged from multiple optimizations of randomized 4D benchmark problems, these being a proxy for typical optimization problems. Four FA parameters $[\alpha_1 \alpha_2 \gamma_1 \gamma_2]$ of Algorithm 2.1 were tuned (others were prescribed: $N = 50, M = 50$, and $\beta =$
1) by selecting those from a full factorial search that minimize Algorithm 2.4 (i.e. $cost_x(f,...)$ where $f$ is the 4D benchmark problem (2.5) with parameters randomly generated each time it is called). Nine values of each tunable FA parameter were considered in the search, from 10 to 50, for a total of $9^4 = 6561$ FA parameter sets. The parameter set with the lowest $\text{mean}(cost_x)$ is used for subsequent optimizations.

The stochastic nature of FA and $f$ results in variation in $cost_x$ for a given FA parameter set; averaging was performed to reduce the effects of this variation. Figure 2.6 shows the mean and standard deviation of average $cost_x$ versus the number of averages taken for a given FA parameter set. The number of averages selected for the full factorial FA tuning runs was 500, this being judged a reasonable trade-off between computation time and repeatability.
Figure 2.6  Averages and standard deviations from 25 batches of cost, as functions of the number of optimizations in each batch for fixed FA tuning parameters: \([\alpha_1 \ \alpha_2 \ \gamma_1 \ \gamma_2] = [10 \ 40 \ 25 \ 25] \).  

a. The means, indicated by circles, with one standard-deviation bands extending above and below.  

b. The standard deviations of batches of the averages as a function of the number of FA evaluations in each batch.  

With 500 FA optimizations (circled in red), the average cost for these FA parameters is 0.57 and the standard deviation of the averages is 0.03.

Performing 500 sequential FA optimizations required 3 minutes on a desktop computer with a 2.7 GHz quad-core processor, translating to 14 days of sequential computing for the full factorial search. Table 2.1 shows the 10 lowest resulting cost FA parameter sets. Averages of cost, from 500 evaluations each of the 6561 FA parameter sets ranged from 0.469 to 1.434; in contrast, the average from 25 batches of 500 evaluations of initial firefly populations (a worst case scenario) is 7.262 and the standard deviation is 0.077. The parameter set resulting in lowest cost, \([\alpha_1 \ \alpha_2 \ \gamma_1 \ \gamma_2] = [10 \ 35 \ 15 \ 20] \) is employed in the remaining optimizations.
Table 2.1 The 10 lowest cost results from a full-factorial FA parameter search. For each of the 6561 parameter sets, an average cost was computed from 500 FA optimizations. Other FA parameters were prescribed: \( N = 50,\ M = 50,\) and \( \beta = 1.\) The lowest cost set \( [\alpha_1\ \alpha_2\ \gamma_1\ \gamma_2] = [10\ 35\ 15\ 20] \) is employed for results of the following sections.

<table>
<thead>
<tr>
<th>set</th>
<th>( \alpha_0 )</th>
<th>( \alpha_1 )</th>
<th>( \gamma_0 )</th>
<th>( \gamma_1 )</th>
<th>( \text{cost}_x )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>35</td>
<td>15</td>
<td>20</td>
<td>0.469</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>35</td>
<td>15</td>
<td>25</td>
<td>0.490</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>40</td>
<td>15</td>
<td>20</td>
<td>0.491</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>40</td>
<td>35</td>
<td>20</td>
<td>0.494</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>25</td>
<td>35</td>
<td>20</td>
<td>0.498</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>35</td>
<td>20</td>
<td>15</td>
<td>0.508</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>25</td>
<td>25</td>
<td>15</td>
<td>0.509</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>25</td>
<td>35</td>
<td>15</td>
<td>0.510</td>
</tr>
<tr>
<td>9</td>
<td>15</td>
<td>35</td>
<td>30</td>
<td>25</td>
<td>0.510</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
<td>25</td>
<td>25</td>
<td>15</td>
<td>0.511</td>
</tr>
</tbody>
</table>

**Benchmark Optimization: 1D, 2D, 3D, and 4D**

Optimization of randomized benchmark functions (2.5) was conducted for 1D, 2D, 3D, and 4D problems. In all of these cases, FA parameters from the full factorial search resulting in lowest \( \text{cost}_x \) were employed: \( N = 50,\ M = 50,\) \( \alpha = [10\ 35]^T\sqrt{4/n},\) \( \beta = [1\ 1]^T,\) \( \gamma = [15\ 20]^T\sqrt{4/n}.\) Note, \( \alpha \) and \( \gamma \) are multiplied by \( \sqrt{4/n} \) to scale the FA parameters (which were tuned using 4D benchmark problems) to the nD case. Computing each of these results required approximately 0.25 seconds on a desktop computer with a 2.7 GHz quad-core processor.

Figure 2.7 shows a typical 1D result. The cost function was generated using (2.5) with: \( m = 2,\ n = 1,\ a = [0.75\ 0.52]^T,\ b = [95.5\ 15.6]^T,\ c = [0.86\ 0.17],\ d = 1.\) Seeds of the cost function are located at \( c,\) with respective costs \( [0.28\ 0.53]^T.\) The optimization champions are \( [0.86\ 0.19],\) with respective costs \( [0.28\ 0.53]^T.\)
Figure 2.7 1D cost function (2.5) with initial and final firefly positions and corresponding CSD plots.
Fireflies are red data points in the cost curves.

The initial and final CSD plots in Figure 2.7 provide useful information about their respective firefly populations. Figure 2.7.c, corresponding to the initial population, indicates a fairly scattered population. There is variation in color among data point near one another (colors can be seen to blend from one to another) and they are interrupted in terms of cost (i.e. blue colored data points are scattered from left to right, interrupted by other colors). Note that the 1st and 9th fireflies can already be identified as the two most interesting design possibilities (the two lowest cost, obviously different solutions). Figure 2.7.d, corresponding to the final firefly
population, clearly shows a bimodal result. The 1st and 33rd fireflies are obvious champions. Note that the CSD plot does not indicate the relative cost of the two solutions, just that two have been found and one is better than the other.

Figure 2.8 shows a typical 2D result. The cost function was generated using (2.5) with parameters: $m = 4$, $n = 2$, $a = [0.89 \ 0.68 \ 0.64 \ 0.58]^T$, $b = [55.7 \ 34.7 \ 17.9 \ 48.6]^T$, $c = [0.19 \ 0.82 \ 0.78 \ 0.30]^T$, $d = 1$. Seeds of the cost function are located at $c$, with respective costs $[0.12 \ 0.40 \ 0.41 \ 0.47]^T$. The optimization champions are $[0.19 \ 0.81 \ 0.77]^T$, with respective costs $[0.12 \ 0.40 \ 0.41]^T$. 
Figure 2.8  2D cost function (2.5) with initial and final firefly positions and corresponding CSD plots. Fireflies are white data points in the contour plots.

The initial and final CSD plots in Figure 2.8 provide similarly useful information about their respective firefly populations (the comments regarding Figure 2.7 also apply). Clearly, Figure 2.8 shows that three distinct solutions have resulted; there are three clusters of fireflies (indicated by the purple data points) that are not close to the other designs (indicated by the orange, red, and yellow colors of other data points). The three champions to consider are fireflies 1, 17, and 32, the lowest cost member of each cluster.
Figure 2.9 shows a typical 3D result. The cost function was generated using (2.5) with: \( m = 6, n = 3, \ a = [0.87 \ 0.80 \ 0.57 \ 0.63 \ 0.74 \ 0.75]^T, \ b = [70.0 \ 52.8 \ 66.5 \ 59.0 \ 88.0 \ 95.8]^T, \ c = [0.23 \ 0.15 \ 0.85 \ 0.87 \ 0.28] \]
\[0.30 \ 0.32 \ 0.18 \ 0.69 \ 0.71], \ d = 1. \ Seeds \ of \ the \ cost \ function \ are \ located \ at \ c, \ with \ respective \ costs \ [0.14 \ 0.24 \ 0.32 \ 0.35 \ 0.48 \ 0.52]^T. \ The \ optimization \ champions \ are \ [0.23 \ 0.15 \ 0.15 \ 0.30 \ 0.67 \ 0.32 \ 0.86 \ 0.28] \]
\[0.30 \ 0.67 \ 0.32 \ 0.69 \ 0.71], \ with \ respective \ costs \ [0.14 \ 0.24 \ 0.32 \ 0.48 \ 0.52]^T.

Figure 2.9 3D cost function (2.5) with initial and final migration firefly positions and corresponding CSD plots. Fireflies are scatter plot black circles; grey circles are their projections on the axis planes.
Figure 2.9.c shows a less ordered initial population compared to the previous two figures because the design space is larger relative to the number of fireflies. Figure 2.9.d clearly indicates 5 distinct clusters within the final firefly population (blocks of dark blue data points). Comparing this to Figure 2.9.c, the found clusters are near 5 of the 6 cost function seeds.

Figure 2.10 shows typical 4D results. The cost function was generated using (2.5) with parameters: $m = 8$, $n = 4$, $a = [0.76 ~ 1.00 ~ 0.57 ~ 0.77 ~ 0.96 ~ 0.60 ~ 0.85 ~ 0.55]^T$, $b = [92.4 ~ 44.9 ~ 63.8 ~ 94.2 ~ 27.1 ~ 39.9 ~ 10.8 ~ 65.1]^T$, $c = [0.17 ~ 0.82 ~ 0.63 ~ 0.66 ~ 0.19 ~ 0.31 ~ 0.87 ~ 0.86]^T$, $d = 1$. Seeds of the cost function are located at $c$, with respective costs $[0.00 ~ 0.05 ~ 0.17 ~ 0.24 ~ 0.30 ~ 0.52 ~ 0.52 ~ 0.63]^T$. The optimization champions are $[0.17 ~ 0.81 ~ 0.63 ~ 0.48 ~ 0.63]$, $[0.79 ~ 0.16 ~ 0.77 ~ 0.79 ~ 0.78]$, $[0.35 ~ 0.31 ~ 0.65 ~ 0.63 ~ 0.48]$, with respective costs $[0.01 ~ 0.05 ~ 0.17 ~ 0.38 ~ 0.42]^T$.

Figure 2.10 4D cost function (2.5) with initial and final firefly positions and corresponding CSD plots.
The 4D cost function and firefly locations are not easily plotted in Figure 2.10 as they were for
the 1D through 3D problems. The initial and final CSD plots, though, are shown and provide
the same useful information about their respective populations. Three clusters of fireflies are
apparent in Figure 2.10.b, and their champions are near the three lowest cost seeds.

Table 2.2 shows a statistical summary of results from application of Algorithm 2.1 for 1000
each of 1D, 2D, 3D, and 4D randomized benchmark optimizations. Both of two minima were
found (i.e. a champion was within 1% of a seed or had a lower objective function value than
the nearest seed) by the optimizer in 85% of 1D runs – one of two minima was found in 15%
of runs. Two or more of four minima were found in 100% of 2D runs. The mode of minima
found was 4 and the average was 3.65. Two or more of 6 minima were found in 93% of 3D
runs. The mode of minima found was 4 and the average was 3.97. Two or more of 8 minima
were found in 89% of 4D runs. The mode of minima found was 4 and the average was 3.57.
At least one minimum was found by the optimizer in all runs.

Table 2.2 Summary of 1D-4D benchmark optimization results. Optimization was completed for 1000 each
of 1D, 2D, 3D, and 4D randomized benchmark problems. The first measure is the percentage of runs
where multiple minima were found. The number of minima found (mode and mean) for those runs is also
shown. Mode and mean are not shown for the 1D problem because in all cases where multiple minima
were detected, two were detected (there were always exactly two).

<table>
<thead>
<tr>
<th>measure</th>
<th>1D</th>
<th>2D</th>
<th>3D</th>
<th>4D</th>
</tr>
</thead>
<tbody>
<tr>
<td>multiple minima found</td>
<td>85.1%</td>
<td>100%</td>
<td>93.2%</td>
<td>89.0%</td>
</tr>
<tr>
<td>mode (minima found)</td>
<td>-</td>
<td>4/4</td>
<td>4/6</td>
<td>4/8</td>
</tr>
<tr>
<td>mean (minima found)</td>
<td>-</td>
<td>3.65/4</td>
<td>3.97/6</td>
<td>3.57/8</td>
</tr>
</tbody>
</table>
2.3 Example: Electromechanical Design Optimization

2.3.1 Active Magnetic Bearing and Controller

Typically in the design of electromechanical devices, the hardware and control algorithms are designed separately. A more satisfactory solution may be realized by parameterizing and optimizing these tasks simultaneously. The electromechanical system shown in Figure 2.11 is an active magnetic bearing (AMB), where voltage is manipulated to produce electromagnetic forces that maintain a desirable separation (levitation) between the shaft and electromagnet poles. Shaft displacement is sensed and fed back to a control algorithm, in this case a proportional + integral + derivative (PID) controller with gains \((k_p, k_i\) and \(k_d\), respectively) that must be optimized for stability and performance. The coil length \(L_c\), outside diameter \(D_c\), and wire gage \(g_c\) must also be specified. The 6D design vector is therefore \(x = [g_c \ D_c \ L_c \ k_p \ k_i \ k_d]^T\). Limits are assumed for the maximum current \(i_{\text{max}} = 2 A\), maximum voltage \(v_{\text{max}} = 12 V\), coil length \(5 \text{ mm} \leq L_c \leq 20 \text{ mm}\), coil outside diameter \(10 \text{ mm} < D_c \leq 30 \text{ mm}\), wire gauge \(26 \leq g_c \leq 32\), and maximum magnetic flux density \(B_{\text{max}} = 1 T\). It is desirable to maintain a reference gap \(y_d(t) = 2 mm\), in the presence of disturbance forces, with minimal electrical power.
Figure 2.11 Schematic of a longitudinal AMB, where the separation between stator and mover is maintained by a PID controlled coil voltage. The shaft and electromagnet core are made of electrical steel with relative permeability $\mu_r = 1000$ and mass density $\rho = 7.874 \text{ g/m}^3$. The shaft diameter is $D_s = 1 \text{ cm}$, the shaft length is $L_s = 10 \text{ cm}$, and the ferrous portion of the magnetic flux path, labeled $\phi$, also has length $L_s$. One-dimensional motion is assumed.

Assessing performance requires a dynamic model of the system, parametric in terms of the design variables. Such a model, a coupled set of nonlinear ordinary differential equations (ODEs) derived from physical laws, is fully developed in the appendix (A.25):

$$\frac{d}{dt}\begin{bmatrix} \xi \\ i \end{bmatrix} = \begin{bmatrix} \frac{\xi}{m_s} + \frac{F_d(t)}{m_s} - \frac{\mu_0 A N_c^2 i^2}{L_s (\mu_r + 2y)} \\ \frac{L_s}{\mu_r + 2y} + 2y \frac{\xi}{\mu_0 A N_c^2 (v - IR)} + \frac{2}{L_s} i \xi \end{bmatrix},$$  \hspace{1cm} (2.6)

where $y$ is the mover displacement, $\xi = \frac{dy}{dt}$ is its derivative, $i$ is coil current, $g$ is the gravitational acceleration ($y$-direction), $F_d(t)$ is a disturbance force applied to the mover, $m_s$ is the mass of the mover, $\mu_0$ is the magnetic permeability of the gap, $A$ is the uniform cross-sectional area of the magnetic flux path, $N_c$ is the number of coil turns, $L_s$ is the length of steel flux path, $\mu_r$ is the steel’s relative permeability, $v$ is the (controller applied) coil voltage, and $R$ is the coil
resistance. The many parameters of (2.6) are reduced to a set of three through the geometry of Figure 2.11 and relationships detailed in the appendix.

The PID control voltage is

\[
v(t) = -k_p \cdot e(t) - k_i \cdot \int e(t) \, dt - k_d \cdot \dot{e}(t)
\]

(2.7)

where \(e(t) = y_d(t) - y(t)\) is the displacement error and \(y_d(t)\) is the desired displacement. A disturbance force such as (2.8) may be used to evaluate the system’s robustness:

\[
F_d(t) = F_1 \Phi(t - T_1) + F_2 \Phi(t - T_2) \cdot (1 - \sin(\omega \cdot (t - T_2)))
\]

(2.8)

where \(F_1 = 5 \, m_s \, g\) and \(F_2 = -F_1/2\) are disturbance amplitudes, \(\Phi()\) is the Heaviside step function, \(T_1 = 1/3 \, s\), \(T_2 = 2/3 \, s\), and \(\omega = 10 \, Hz\) is a disturbance frequency. While the choice of disturbance force profile is somewhat arbitrary, the designer must ensure a physical limitation that net force remains positive; electromechanical forces of this type (reluctance forces) are exclusively attractive.

Stability and performance of this AMB system can be quantified using a penalty function

\[
p(x) = f(x) + K_g \sum_{i=1}^{rows(g)} \max(0, g_i(x))
\]

(2.9)

where

\[
f(x) = \int_0^{t_{end}} K_1 e(t)^2 + K_2 v \, i \, dt + K_3 \max(|\dot{e}(t)|),
\]

(2.10)

\[
g(x) = [\max(B) - B_{max}],
\]

(2.11)

and the simulation time is \(t \in [0, t_{end}]\), \(t_{end} = 1 \, s\), \(K_1 = 1000\), \(K_2 = 1/25\), and \(K_3 = 1/10\) are scale factors, \(\max(B)\) is the maximum magnetic flux density, and \(K_g = 1000\) is the inequality violation penalty factor.
2.3.2 FA Optimization of AMB+PID Design Parameters

The 6D engineering design (AMB with PID control) cost function (2.9) was optimized using Algorithm 2.1 with the FA tuning resulting from Procedure 2.1 \( (N = 50, M = 50, \alpha = [10 \ 35]^T \sqrt{\frac{4}{n}}, \beta = [1 \ 1]^T, \gamma = [15 \ 20]^T \sqrt{\frac{4}{n}}, n = 6 ) \). The system of equations (2.6) was simulated using an Adams-Bashforth ODE solver with convergence tolerance \( TOL = 0.001 \). Computing this result required approximately 18 minutes for a computer with a 2.7 GHz quad-core processor.

Figure 2.12.a and b show initial and final CSD plots for the AMB design example. Figure 2.12.c shows the design champions identified using Algorithm 2.3.

![Initial CSD plot](image1)
![Final CSD plot](image2)
![Optimization result: champions](image3)

Figure 2.12 AMB design example with initial and final CSD plots and designs resulting from firefly optimization. Table 2.3 provides detail regarding the champions and related performance measures.
While the resulting CSD plot (Figure 2.12.c) is not as simple as those associated with benchmark problems (this is not an atypical plot for larger problems), there is still evidence of firefly clustering and there are some clear candidates for further investigation. The first 10 fireflies (the lowest cost grouping, dark blue in the bottom left corner of the plot) are relatively near one another. The second cluster begins with firefly 11 and ends with firefly 23; a third cluster comprising fireflies 18, 19, and 21 are intermingled with these. The fourth cluster begins with firefly 24 and ends with firefly 39. Fireflies 26 and 41 are the fifth cluster.

Table 2.3 lists the design parameters and other pertinent results for each identified AMB design champion.

<table>
<thead>
<tr>
<th>value</th>
<th>symbol</th>
<th>units</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>wire gauge</td>
<td>$g_c$</td>
<td>-</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>28</td>
</tr>
<tr>
<td>coil diameter</td>
<td>$D_c$</td>
<td>mm</td>
<td>12.9</td>
<td>13.0</td>
<td>13.0</td>
<td>11.3</td>
<td>10.8</td>
</tr>
<tr>
<td>coil length</td>
<td>$L_c$</td>
<td>mm</td>
<td>16.4</td>
<td>16.4</td>
<td>16.3</td>
<td>18.6</td>
<td>12.9</td>
</tr>
<tr>
<td>proportional gain</td>
<td>$k_p$</td>
<td>V/m</td>
<td>79824</td>
<td>79415</td>
<td>78904</td>
<td>49058</td>
<td>89356</td>
</tr>
<tr>
<td>integral gain</td>
<td>$k_i$</td>
<td>V/(m s)</td>
<td>720211</td>
<td>372362</td>
<td>246117</td>
<td>703236</td>
<td>729545</td>
</tr>
<tr>
<td>derivative gain</td>
<td>$k_d$</td>
<td>V s/m</td>
<td>883</td>
<td>896</td>
<td>855</td>
<td>627</td>
<td>674</td>
</tr>
<tr>
<td>resistance</td>
<td>$R$</td>
<td>Ω</td>
<td>4.39</td>
<td>4.43</td>
<td>4.44</td>
<td>3.61</td>
<td>5.61</td>
</tr>
<tr>
<td>coil loops</td>
<td>$N_c$</td>
<td>-</td>
<td>597</td>
<td>600</td>
<td>600</td>
<td>539</td>
<td>544</td>
</tr>
<tr>
<td>max. acceleration</td>
<td>$\max(</td>
<td>\dot{y}(t)</td>
<td>)$</td>
<td>m/s²</td>
<td>79.8</td>
<td>78.2</td>
<td>80.1</td>
</tr>
<tr>
<td>average power</td>
<td>$\int (v(t)/t_{end})$</td>
<td>W</td>
<td>4.60</td>
<td>4.62</td>
<td>4.65</td>
<td>4.65</td>
<td>7.08</td>
</tr>
<tr>
<td>max. flux density</td>
<td>$\max(B)$</td>
<td>T</td>
<td>0.32</td>
<td>0.32</td>
<td>0.32</td>
<td>0.33</td>
<td>0.30</td>
</tr>
<tr>
<td>cost</td>
<td>$p(x)$</td>
<td>-</td>
<td>6118</td>
<td>6203</td>
<td>6314</td>
<td>6343</td>
<td>6638</td>
</tr>
</tbody>
</table>

The first three champions (Designs 1-3) are physically similar to one another, differing significantly only with respect to integral gain. Design 4 has the same wire gauge as the first three, but otherwise different geometry and different controller gains. Design 5 reveals significant differences in wire gauge, geometry and controller gains from the others. All of
the designs achieve maximum magnetic flux density well within the required limit. The average power consumption is similar between the first 4 designs, and is somewhat higher for design 5.

Figure 2.13 shows the transient responses associated with these designs. It should be noted that the initial displacement error is -1mm, an impulse is applied at \( t = T_1 \), and a sinusoidal disturbance is applied at \( t = T_2 \). All five designs exhibit stable AMB control; their responses appear similar in Figure 2.13.a (the full 2mm scale), and are similar enough that transient response does not seem a significant differentiator in the design choice. Small differences in response are apparent in Figure 2.13.b (the 0.2mm scale). Design 2 has the smallest initial overshoot; Design 4 has the largest. Design 1 has the smallest response to the step in disturbance force at \( T_1 \); Design 4 has the fastest recovery from it; Design 3 has the slowest. Compared to the others, Designs 4 and 5 have similarly higher amplitude responses to the oscillatory disturbance initiated at \( T_2 \).

![Figure 2.13 Transient simulation results](image)

**Figure 2.13** Transient simulation results (error: \( e = y_d - y \) (mm) vs. time: \( t \) (s) for the AMB FA optimization champions. The plots show identical data, but plot a is scaled to show the full scale and plot b is magnified to show differences between the design responses.
2.4 Discussion/Conclusions

The improved FA (Algorithm 2.1) has been demonstrated capable of accurately finding multiple solutions (i.e. design alternatives) for 1D through 4D benchmark problems and of finding multiple satisfactory designs for a practical engineering problem. Procedure 2.1, a method to find good FA tuning using Algorithm 2.4 and the randomized 4D benchmark problem has also been demonstrated. The resulting tuning was shown to work well for all the problems explored.

FA tuning resulting from application of Procedure 2.1 worked well for finding multiple accurate solutions of 1D, 2D, 3D, and 4D benchmark problems even though it was derived using the 4D problem. Individual examples of optimization results were given for 1D-4D benchmark problems. Both minima were found (meaning a champion was within 1% of a seed or had lower cost) for the 1D problem; the best 3 of 4 for the 2D problem; the best 3 and 5 out of 6 for the 3D problem; and the best 3 of 8 for the 4D problem. A summary of results was presented in Table 2.2 for optimization of 1000 each of 1D, 2D, 3D, and 4D randomized benchmark problems. In these optimizations, two minima were found for 85% of 1D problems, two or more were found for 100% of 2D problems, 93% of 3D problems, and 89% of 4D problems. At least one minimum was always found.

Many of the methods developed (those summarized in Procedure 2.1) provide a basis for systematically finding optimal FA parameters for typical problems – a need identified in the literature [12]. This is not meant to imply that it is difficult to find reasonably good tuning for Algorithm 2.1. Accurate results were produced using the optimal FA tuning, but many other tuning sets would likely give similarly accurate results. The range noted of \( cost_x \) averages (0.47 to 1.43 for the full factorial search) is not large considering that there are 6561 tuning sets and a random firefly population achieves \( cost_x = 7.26 \) on average, with standard deviation 0.08. Also, the ten best tuning sets ranged in cost from 0.47 to 0.51 and their standard deviations are likely near 0.03 (that shown in Figure 2.6 for another 500-average tuning set), therefore, it cannot be concluded that one of these is clearly superior.
The CSD method for identifying champions of clustered design populations was introduced. It is naturally compatible with FA and was incorporated in Algorithm 2.1. Interpretation of a CSD plot was demonstrated for a 2D problem by comparison to a contour plot showing the firefly locations overlaid with the cost function. This visualization method is more valuable for higher dimension problems where visualization by other means is difficult, as was demonstrated for 3D, 4D and 6D problems. The useful character of the CSD plot (the blocked or plaid appearance with champions near local minima) is highly dependent upon the ability of an optimization algorithm to produce clusters near local minima of the cost function. FA was shown to be capable of producing clustered populations and Algorithm 2.1 was successfully optimized using Algorithm 2.4 to do so.

Finally, thorough development of a 6D electromechanical design problem was presented. It was optimized using the methods developed, with the same FA tuning that was used for optimizing the benchmark problems. A CSD plot of the optimized population was presented and five diverse designs were extracted using Algorithm 2.3. All of the designs were feasible, and provided similarly good control of the AMB in response to the disturbance forces applied. The optimization and visualization methods presented have proved useful in designing this electromechanical system.
Chapter 3
Robustness Considerations
Associated with Firefly and the Cost-Sorted Distance Method

3.1 Introduction

A modified firefly algorithm (FA) was demonstrated in Chapter 2 for optimizing nonlinear multimodal systems with multiple degrees of freedom. The algorithm was tuned to produce results clustered near multiple minima of the objective function. The cost-sorted distance (CSD) method, a tool for exploring the design space, was shown to clearly indicate clusters of designs and their relative performance so that the most attractive design alternatives may be easily found.

Another critical aspect of optimal design, one that was not discussed in Chapter 2, involves the consideration of uncertainty. There are many potential sources of uncertainty in the design and implementation of engineered products. Uncertainty pertinent to engineering design has been classified according to four types [20], [21]: (A) changing environment and operating conditions (a.k.a. Type I); (B) production tolerances and actuator imprecision (a.k.a. Type II); (C) uncertainties in the system output; and (D) feasibility (design constraint) uncertainties. A very common example of Type II uncertainty involves the precision with which a dimension critical to the design can be achieved. Ideal real-world engineering designs are not simply deterministically optimal, but also insensitive to uncertainty. Specialized optimization
methods are suitable for achieving this characteristic. The purpose of this chapter is to examine whether the tools developed in Chapter 2 are useful in this regard.

Robust design optimization (RDO), i.e. methods for development of designs optimally insensitive to uncertainty [20], has been summarized by several authors in recent years. Notably, Beyer et al. [20] in 2007 provided a comprehensive survey of the field including classification of uncertainty types and appropriate methodologies for each of them. Extensive treatment was given to evolutionary algorithms in the context of robustness. Park et al. [21] in 2006 published an overview of robust design including definitions of key terms and concepts with thorough discussion of the Taguchi method of robust design. Jin et al. [22] in 2005 published a concise summary of evolutionary optimization methods for addressing uncertainty in the fitness function and design variables. Lelièvre et al. [23] in 2016 categorized methods by suitability for addressing objective function and/or constraint uncertainties and demonstrated development of appropriate objective functions.

Typically, RDO involves optimization of a function \( F(\mu_x, \mu_z) \) that is the robust counterpart of the deterministic objective function \( f(x,z) \). The designer seeks to specify optimal expected values \( \mu_x \) of possibly uncertain design variables \( x \) with standard deviations \( \sigma_x \), considering possibly uncertain design parameters \( z \), with expected values \( \mu_z \) and standard deviations \( \sigma_z \). Design parameters are quantities imposed on the design or provided by the system’s environment and the extent of their uncertainty must be tolerated. Multiple design objectives may be incorporated in \( F \) using a weighted sum approach (e.g. (2.10)), or Pareto optimization techniques may be applied [24]. Robust counterparts of the deterministic constraint functions may also be incorporated by weighted sum with the objective function to form a scalar penalty function, e.g. (2.9). Uncertain constraints and uncertain design parameters are often handled in a “worst case” sense (e.g. \( \min_{\mu_x} \max_{\mu_x} (F(\mu_x, \mu_z)) \) [20] [21] that can be effectively implemented using, for example, an iterative optimization technique [25] [26]. Ultimately, with methods that depend on the system and how uncertainty presents itself, a scalar objective function can be generated. In this chapter, without loss of generality, the objective function and its robust counterpart will be treated as scalar functions of random variables, \( x \).
Robust objective functions $F$ differ from their deterministic counterparts by incorporation of robustness measures. Beyer et al. described five types of robustness measures for design optimization. These are *robust regularization*, which involves finding the “worst case” objective function value associated with a range of design space; *expectancy measures* such as mean and standard deviation of the objective function for each design; *probabilistic threshold* measures of robustness, i.e. the probability that the objective function will achieve some result for a given design; *statistical feasibility* with regard to constraint satisfaction; and *possibilistic uncertainties* that can incorporate subjective assessment [20]. Two types are considered in this work. They are expectancy measures (mean and standard deviation - the most widely encountered robustness measures [26]), and a probabilistic threshold measure called probability of dominance [27] (PoD).

RDO can be computationally prohibitive compared to deterministic optimization because robustness measures must be estimated for all evaluated designs. Monte Carlo analysis is often the most direct and reliable method for estimating a given design’s robustness [20]. This requires computing $f$ for a large number of neighboring designs and is likely to increase the number of function evaluations by orders of magnitude. Alternative approaches may be available depending on the problem type; these include estimates based on Taylor series expansion (robustness measures from analytically or numerically estimated local function expansions), and meta-model approaches (including Taguchi robust design, neural networks, kriging, etc.) where robustness measures are estimated from a surrogate function that is fit using a strategic few local design points. Meta model approaches are particularly useful when objective function evaluation is very difficult (if a physical experiment is required, for example). In this work (although deterministic measures derived by Taylor series expansion are demonstrated) the standard basis of comparison is Monte Carlo analysis

As noted, RDO can be accomplished using any capable optimization method by incorporation of a robust objective function. Some evolutionary algorithms (EAs), however, due to their inherent stochasticity and large populations can generate robust designs by assessing the objective function directly [20].
The hypothesis tested in this chapter is that the percentage of fireflies (PFF) in clusters generated by Algorithm 2.1 (without modification) reliably indicates the relative robustness of associated designs. The procedure is to empirically assess the degree of linear correlation between PFF and traditional robustness measures for optimizations of randomized parametric benchmark problems (2.5).

3.2 Methods for Quantifying Robustness

The robustness measures considered here are expectancy measures (mean and standard deviation), and probability of dominance, i.e. PoD. Evaluation of these measures is considered in this section.

3.2.1 A Deterministic Approach

Symbolic objective functions facilitate deterministic expectancy measures based on Taylor series expansion. The Taylor series of \( f(x) \) through second order terms is

\[
f(x + \Delta x) = f(x) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} \Delta x_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 f}{\partial x_i \partial x_j} \Delta x_i \Delta x_j + \cdots. \tag{3.1}
\]

It is important to include terms through at least second order to accurately estimate expectancy measures for functions with stationary points (e.g. minima). Assuming \( x \) uncorrelated, zero-mean, normally distributed, random variables, the mean of \( f \) (i.e. \( \mu_f \)) may be estimated [20]

\[
\mu_f \equiv \mu_f(\mu_x) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 f}{\partial x_i \partial x_j} \bigg|_{\mu_x} \sigma_{x_i} \sigma_{x_j}, \tag{3.2}
\]

and the standard deviation of \( f \) (i.e. \( \sigma_f \)) may be estimated [28]
\[ \sigma_f \cong \left[ \sum_{i=1}^{n} \left( \frac{\partial f}{\partial x_i} \right)^2 \sigma_{x_i}^2 + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left( \frac{\partial^2 f}{\partial x_i \partial x_j} \right) \sigma_{x_i} \sigma_{x_j} \right]^{\frac{1}{2}}. \] (3.3)

This can be demonstrated with the simplest multimodal version of (2.4) (i.e. \( m = 2, n = 1 \)):

\[ f(x) = (1 + d \ (x - 0.5)^2) \sum_{j=1}^{2} \left( 1 - a_j \ e^{-b_j \ (x - c_j)^2} \right)^2. \] (3.4)

Applying (3.2) to (3.4),

\[ \mu_f(\mu_x, \sigma_x) \]

\[ \cong (1 + d \ (\mu_x - 0.5)^2) \left( 1 - a_1 \ e^{-b_1 (\mu_x - c_1)^2} \right) \left( 1 - a_2 \ e^{-b_2 (\mu_x - c_2)^2} \right) \]

\[ + \frac{\sigma_x^2}{2} \left[ 2d \ \left( a_1 \ e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( a_2 \ e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \right. \]

\[ - 2 \ a_1 b_1 \ e^{-b_1 (\mu_x - c_1)^2} \left( a_2 \ e^{-b_2 (\mu_x - c_2)^2} - 1 \right) (d \ (\mu_x - 0.5)^2 + 1) \]

\[ - 2 \ a_2 b_2 \ e^{-b_2 (\mu_x - c_2)^2} \left( a_1 \ e^{-b_1 (\mu_x - c_1)^2} - 1 \right) (d \ (\mu_x - 0.5)^2 + 1) \]

\[ + a_1 b_1^2 \ e^{-b_1 (\mu_x - c_1)^2} \left( 2 \mu_x - 2c_1 \right)^2 \left( a_2 \ e^{-b_2 (\mu_x - c_2)^2} - 1 \right) (d \ (\mu_x - 0.5)^2 + 1) \]

\[ + a_2 b_2^2 \ e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_2 \right)^2 \left( a_1 \ e^{-b_1 (\mu_x - c_1)^2} - 1 \right) (d \ (\mu_x - 0.5)^2 + 1) \]

\[ - 2 \ a_1 b_1 d \ e^{-b_1 (\mu_x - c_1)^2} \left( 2 \mu_x - 2c_1 \right) (2 \mu_x - 1) \]

\[ - 2 \ a_2 b_2 e^{-b_2 (\mu_x - c_2)^2} (2 \mu_x - 2c_2)(2 \mu_x - 1) \left( a_1 \ e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \]

\[ + 2a_1 b_1 a_2 b_2 e^{-b_1 (\mu_x - c_1)^2} e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_1 \right) \left( 2 \mu_x - 2c_2 \right) \left( d \ (\mu_x - 0.5)^2 + 1 \right) \]. \] (3.5)

Applying (3.3) to (3.4),
\[\sigma_f(\mu_x, \sigma_x)\]

\[\equiv \left[\sigma_x^2 \left[ d \left( 2 \mu_x - 1 \right) \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( a_2 e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \right] - 2a_1 b_1 e^{-b_1 (\mu_x - c_1)^2} \left( 2 \mu_x - 2c_1 \right) \left( a_2 e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \right.\]

\[\left. - a_2 b_2 e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_2 \right) \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \right] \left( \sigma_x^2 \right)^2 + \frac{\sigma_x^4}{2} \left[ 2d \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( a_2 e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \right.\]

\[\left. - 2a_1 b_1 e^{-b_1 (\mu_x - c_1)^2} \left( a_2 e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \right] - 2a_2 b_2 e^{-b_2 (\mu_x - c_2)^2} \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \]

\[+ a_1 b_1^2 e^{-b_1 (\mu_x - c_1)^2} \left( 2 \mu_x - 2c_1 \right)^2 \left( a_2 e^{-b_2 (\mu_x - c_2)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \]

\[+ a_2 b_2^2 e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_2 \right)^2 \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \]

\[\left. - 2a_1 b_1 d e^{-b_1 (\mu_x - c_1)^2} \left( 2 \mu_x - 2c_1 \right) \left( 2 \mu_x - 1 \right) \right] - 2a_2 b_2 d e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_2 \right) \left( 2 \mu_x - 1 \right) \left( a_1 e^{-b_1 (\mu_x - c_1)^2} - 1 \right) \right]\]

\[+ 2a_1 b_1 a_2 b_2 e^{-b_1 (\mu_x - c_1)^2} e^{-b_2 (\mu_x - c_2)^2} \left( 2 \mu_x - 2c_1 \right) \left( 2 \mu_x - 2c_2 \right) \left( d (\mu_x - 0.5)^2 + 1 \right) \right]^2.\]

If \(a = [0.2 \ 0.5]^T, b = [100 \ 500]^T, c = [0.15 \ 0.75], d = 1\), then (3.4) is

\[f(x) = (1 + (x - 0.5)^2)\left( 1 - 0.2 e^{-100 (x-0.15)^2} \right)\left( 1 - 0.5 e^{-500 (x-0.75)^2} \right).\]

Equation (3.7) is plotted in Figure 3.1. Its lowest minima are \(f_0 = 0.8946\) at \(x_0 = 0.1624\) and \(f_1 = 0.5312\) at \(x_1 = 0.7495\). The corresponding objective function \(\mu_f\) and \(\sigma_f\) from (3.2) and (3.3), respectively (for normally distributed \(x\) about \(\mu_x = x_0\) and \(\mu_x = x_1\) with \(\sigma_x = 0.01\)) are \(\mu_{f_0} = 0.89680, \sigma_{f_0} = 0.00308, \mu_{f_1} = 0.55780, \) and \(\sigma_{f_1} = 0.03760\).
Figure 3.1  An example of the simplest possible multimodal objective function.

Clearly, (3.2) and (3.3) provide a way to directly evaluate these robustness measures, but they lead to large symbolic expressions for even relatively simple multimodal objective functions. Also, as dimensions and the number of minima increase, the complication apparent in these solutions increases exponentially. Partial derivatives can be numerically approximated to reduce the symbolic burden, but this reduces the approximation fidelity and can result in significant numerical expense as problem size grows. For large problems other methods are warranted.

3.2.2 Monte Carlo Approach

Monte Carlo methods involve statistical analyses via multiple computer simulations with randomly perturbed inputs to approximate solutions to otherwise intractable problems. In contrast to Taylor series based methods, Monte Carlo methods are easy to implement with modern computers and do not require evaluation of an objective function’s derivatives or make
assumptions regarding input distributions. They do, however, require many cost function evaluations.

Computing $\mu_f$ and $\sigma_f$ involves simply taking the average and standard deviation of a number $N$ of computer simulations:

$$\mu_f \cong \frac{1}{N} \sum_{i=1}^{N} f(x + \Delta x_i),$$

(3.8)

and

$$\sigma_f \cong \left[ \frac{1}{N} \sum_{i=1}^{N} f(x + \Delta x_i) - \mu_f \right]^\frac{1}{2},$$

(3.9)

where $\Delta x$ are samples from set of random numbers.

The Monte Carlo method was employed using (3.8) and (3.9) to estimate $\mu_f$ and $\sigma_f$ of (3.7) about its two lowest minima. Figure 3.2 depicts the experiments as uncertainty is propagated from the input variables to the output. Ten-thousand computer simulations (with $\Delta x$ normally distributed about $\mu_{x_0}$ and $\mu_{x_1}$ with $\sigma_x = 0.01$) result in $\mu_{f_0} = 0.89682$, $\sigma_{f_0} = 0.00316$, $\mu_{f_1} = 0.55573$, and $\sigma_{f_1} = 0.03245$. 
a. normally distributed inputs propagate to outputs  

b. histograms of the output distributions

Figure 3.2 Uncertainty propagation for each of two minima of a typical 1D benchmark function. The function is evaluated at 10000 randomly selected points (normally distributed about each minimum), a. The input distributions are identical with $\sigma_x = 0.01$. Normalized histograms of output distributions are shown on the vertical axis. b. The output histograms are shown in closer detail.

The precision of Monte Carlo methods can be increased (until round-off due to floating point number representation becomes a limitation) by increasing the number $N$ of simulations. This is another advantage over Taylor series methods which require additional series terms to improve precision. The main advantage, though, is that it is negligibly more difficult to analyze any objective function using Monte Carlo than it is to analyze this simplest one.

### 3.2.3 Probability of Dominance

PoD is a probabilistic measure for comparing relative dominance of a number of proposed design points, considering uncertainty. It can be evaluated using Monte Carlo simulations at identical randomly distributed perturbations from each of the proposed designs. The objective function is evaluated for each of $N$ random perturbations from each of the proposed designs. PoD for a given design is the number of times its perturbed counterparts dominate those for the other designs’ identically perturbed counterparts, divided by $N$ [27]:
where $\Delta x \in \mathcal{N}(0, \sigma_x)$, $\sigma_x$ is a vector of standard deviations corresponding to the rows of $xs$ the set of proposed designs, $x_i \in xs$, and

$$P_{D_i}(x_{s}, \sigma_{x}) = \frac{1}{N} \sum_{j=1}^{N} P_{D}(x_{i} | x_{s}, x_{i} + \Delta x^{(j)}) ,$$

(3.10)

where $\Delta x \in \mathcal{N}(0, \sigma_x)$, $\sigma_x$ is a vector of standard deviations corresponding to the rows of $xs$ the set of proposed designs, $x_i \in xs$, and

$$P_{D}(x_{i} | x_{s}, x_{i} + \Delta x^{(j)}) = \begin{cases} 1 & \text{if } f(x_{i} + \Delta x^{(j)}) = \min_{k=1, \text{cols}(xs)} f(x_{k} + \Delta x^{(j)}) \\ 0 & \text{otherwise} \end{cases} .$$

(3.11)

The standard deviation $\sigma_x$ of $\Delta x$ distribution is critical for PoD. If it is very small then the design with lowest $f(x)$ will clearly dominate other designs. For example, in Figure 3.2 where $\sigma_x = 0.01$, no perturbed designs associated with $x_0$ dominate any associated with $x_1$ (none of the black data points have $f$ lower than any of the red data points); so $PoD(x_0, \sigma_x = 0.01) = 0$ and $PoD(x_1, \sigma_x = 0.01) = 1$. As $\sigma_x$ grows in magnitude, some perturbations of $x_0$ are superior to those of $x_1$. In Figure 3.3 where $\sigma_x = 0.025$, the two output distributions overlap somewhat (in contrast to Figure 3.2 where they do not); here $PoD(x_0, \sigma_x = 0.025) = 0.02230$ and $PoD(x_1, \sigma_x = 0.025) = 0.97770$.

![Figure 3.3](image)

**Figure 3.3** Output distributions overlap somewhat for $\sigma_x = 0.025$ (compare to Figure 3.2). The function (3.7) is evaluated at 10000 randomly selected points (normally distributed about each minimum), a. The input distributions are identical with $\sigma_x = 0.025$. Normalized histograms of output distributions are shown on the vertical axis. b. The output histograms are shown in closer detail.
Table 3.1 shows how PoD changes with \( \sigma_x \) for the lowest two minima of (3.7). \( PoD_0 \) increases from 0 to 16% as \( \sigma_x \) increases from 0.01 to 0.05. Here, the minimum at \( x_1 \) is dominant for all \( \sigma_x \); often however, the largest PoD does not belong to the deterministic minimum.

<table>
<thead>
<tr>
<th>( \sigma_x )</th>
<th>( PoD_0 )</th>
<th>( PoD_1 )</th>
<th>( \mu_{f_0} )</th>
<th>( \sigma_{f_0} )</th>
<th>( \mu_{f_1} )</th>
<th>( \sigma_{f_1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>0.0000</td>
<td>1.0000</td>
<td>0.8968</td>
<td>0.0032</td>
<td>0.5557</td>
<td>0.0325</td>
</tr>
<tr>
<td>0.015</td>
<td>0.0002</td>
<td>0.9998</td>
<td>0.8993</td>
<td>0.0065</td>
<td>0.5827</td>
<td>0.0638</td>
</tr>
<tr>
<td>0.020</td>
<td>0.0049</td>
<td>0.9951</td>
<td>0.9027</td>
<td>0.0109</td>
<td>0.6149</td>
<td>0.0952</td>
</tr>
<tr>
<td>0.025</td>
<td>0.0223</td>
<td>0.9777</td>
<td>0.9072</td>
<td>0.0168</td>
<td>0.6472</td>
<td>0.1219</td>
</tr>
<tr>
<td>0.030</td>
<td>0.0437</td>
<td>0.9563</td>
<td>0.9118</td>
<td>0.0226</td>
<td>0.6774</td>
<td>0.1444</td>
</tr>
<tr>
<td>0.035</td>
<td>0.0694</td>
<td>0.9306</td>
<td>0.9175</td>
<td>0.0293</td>
<td>0.7079</td>
<td>0.1601</td>
</tr>
<tr>
<td>0.040</td>
<td>0.1041</td>
<td>0.8959</td>
<td>0.9230</td>
<td>0.0350</td>
<td>0.7353</td>
<td>0.1733</td>
</tr>
<tr>
<td>0.045</td>
<td>0.1306</td>
<td>0.8694</td>
<td>0.9286</td>
<td>0.0414</td>
<td>0.7599</td>
<td>0.1831</td>
</tr>
<tr>
<td>0.050</td>
<td>0.1565</td>
<td>0.8435</td>
<td>0.9362</td>
<td>0.0498</td>
<td>0.7838</td>
<td>0.1903</td>
</tr>
</tbody>
</table>

PoD can be used to select from among identified minima, e.g. different points on a Pareto frontier, or FA design champions.

### 3.2.4 Correlation Between Robustness Measures and the Fireflies Clustered

The hypothesis that the percentage of fireflies (PFF) in clusters generated by Algorithm 2.1 reliably indicates the relative robustness of associated designs can be empirically tested using the methods of sections 3.2.2 and 3.2.3 and the benchmark function (2.5). The first task in finding PFF for a design population is to determine the number of fireflies per cluster. This can be done for a generic objective with a modification to Algorithm 2.3, but is more directly accomplished for the benchmark function by recording the closest minima (or seed) of (2.5) to
each firefly. PFF is the number of fireflies in each cluster $f f s_i$ divided by the total number of fireflies $f f s$:

$$PFF_i = \frac{f f s_i}{f f s}.$$  \hspace{1cm} (3.12)

This is a normalized quantity i.e. $PFF_i \in [0,1]$ and $\sum_i PFF_i = 1$. In order to test linear correlation between PFF and comparative robustness measures, the robustness measures should also be normalized. Conveniently PoD is already such a quantity, but the expectancy measures are not. A new measure, percent robustness (PR), derived from $\sigma_f$ will be employed:

$$PR_i = \frac{1}{\sum_{i=1}^{n \text{ champions}} \frac{1}{\sigma_f_i}}.$$  \hspace{1cm} (3.13)

The procedure is to statistically assess the degree of linear correlation between PoD and PFF and between PR and PFF for multiple optimizations of randomized parametric benchmark problems. Linear correlation is assessed by measuring the slope of a least square error (LSE) line fit through the data points corresponding to champions of each optimization. Also, Pearson’s correlation coefficient (R) is computed for a LSE line fit through all data points. The degree of reliability with which positive slope may be expected is statistically measured (i.e. the probability that more fireflies cluster about more robust minima).

Figure 3.4.c shows best-fit lines (black: PFF vs. PR, and red: PFF vs. PoD) for typical 2D benchmark optimization results. Data points are labeled 1 through 4 (in all plots of Figure 3.4) corresponding to the champions, in order of minimum cost. Figure 3.4.a is a contour plot of the cost function with data points for final firefly locations. Figure 3.4.b is the associated CSD plot. Figure 3.4.d shows $\mu_f$ vs. $\sigma_f$ from (3.8) and (3.9), respectively, with $N = 1000$ for each firefly throughout all migrations. Black data points are for the final migration fireflies and grey data points are for all previous migrations. Red lines in this plot correspond to the deterministic minima.
Figure 3.4  Typical 2D optimization results. In each plot, champions are labeled 1 through 4 from lowest to highest objective function value.  

a. Contour plot of the cost function with fireflies in their final migration locations;  
b. final migration CSD plot;  
c. data points and least-square error linear fits: PFF vs. PR (black) and PFF vs. PoD($\sigma_x = 0.05$) (red) for final migration champions;  
d. Monte Carlo $\mu_f$ vs $\sigma_f$ for each firefly throughout all migrations; black circles correspond to final migration locations, grey circles are for all previous migrations. Horizontal red lines are cost function minima.
3.3 Results

One thousand random variations of the parametric benchmark function (2.5) were optimized using FA for each of 1D, 2D, 3D, and 4D problems. Champions from each multimodal result were analyzed according to the methods of section 3.2.4. Histograms of slopes from the LSE line fits of PFF vs. PR (e.g. Figure 3.4.c) are shown in Figure 3.5, along with best-fit Gaussian distribution curves (red) and associated cumulative distributions (black, dashed). Red, dashed lines indicate the mean and the mean-plus-one-standard deviation of the slopes. Data for these measures are shown in Table 3.2. Notably, the cumulative probability distributions indicate that the PFF vs. PR trend-line has positive slope in: 79% of 1D optimizations; 96% of 2D optimizations; 98% of 3D optimizations; and 94% of 4D optimizations.
Figure 3.5  PFF vs. PR: histograms of slopes from least-square-error line fits. Estimated probability density and cumulative probability distributions are also shown. One-thousand each of 1D, 2D, 3D, and 4D randomized benchmark problems were solved using firefly.

Figure 3.6 shows all PFF vs. PR data points for all optimizations, with associated LSE lines. Notably as shown in Table 3.2, $R$ is: 0.55 for 1D data; 0.73 for 2D data; 0.74 for 3D data; and 0.62 for 4D data.
Figure 3.6 PFF vs. PR: all data points from 1000 firefly optimizations of randomized benchmark functions.

Table 3.2 linear correlation measures for $PFF$ vs $PR$

<table>
<thead>
<tr>
<th>Measure</th>
<th>1D</th>
<th>2D</th>
<th>3D</th>
<th>4D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R$</td>
<td>0.55</td>
<td>0.73</td>
<td>0.74</td>
<td>0.62</td>
</tr>
<tr>
<td>positive slope</td>
<td>79%</td>
<td>96%</td>
<td>98%</td>
<td>94%</td>
</tr>
<tr>
<td>$\mu$(slopes)</td>
<td>0.54</td>
<td>0.55</td>
<td>0.68</td>
<td>0.76</td>
</tr>
<tr>
<td>$\sigma$(slopes)</td>
<td>0.66</td>
<td>0.30</td>
<td>0.34</td>
<td>0.49</td>
</tr>
</tbody>
</table>

Histograms of slopes from the LSE line fits of PFF vs. PoD (e.g. Figure 3.4.c) are shown in Figure 3.7, along with best-fit Gaussian distribution curves (red) and associated cumulative distributions (black, dashed). Red, dashed lines indicate the mean and the mean-plus-one-
standard deviation of the slopes. Data for these measures are shown in Table 3.3. Notably, the cumulative probability distributions indicate that the PFF vs. PR trend-line has positive slope in: 88% of 1D optimizations; 94% of 2D optimizations; 92% of 3D optimizations; and 84% of 4D optimizations.

1D

2D

3D

4D

Figure 3.7 PFF vs. PoD: histograms of slopes from least-square-error line fits. Corresponding estimated probability density and cumulative probability distributions are also shown. One-thousand each of 1D, 2D, 3D, and 4D randomized benchmark problems were solved using firefly. Note, \( PoD(\sigma_x = 0.05) \).
Figure 3.8 shows all PFF vs. PR data points for all optimizations, with associated LSE lines. Notably as shown in Table 3.3, R is: 0.79 for 1D data; 0.72 for 2D data; 0.74 for 3D data; and 0.63 for 4D data.

Figure 3.8  PFF vs. PoD: all data points from 1000 firefly optimizations of randomized benchmark functions. Note, $PoD(\sigma_z = 0.05)$. 
Table 3.3  linear correlation measures for PFF vs. PoD.  Note, $PoD(\sigma_x = 0.05)$:

<table>
<thead>
<tr>
<th>Measure</th>
<th>1D</th>
<th>2D</th>
<th>3D</th>
<th>4D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R$</td>
<td>0.79</td>
<td>0.72</td>
<td>0.74</td>
<td>0.63</td>
</tr>
<tr>
<td>positive slope</td>
<td>88%</td>
<td>94%</td>
<td>92%</td>
<td>84%</td>
</tr>
<tr>
<td>$\mu$(slopes)</td>
<td>0.47</td>
<td>0.40</td>
<td>0.54</td>
<td>0.67</td>
</tr>
<tr>
<td>$\sigma$(slopes)</td>
<td>0.40</td>
<td>0.26</td>
<td>0.38</td>
<td>0.68</td>
</tr>
</tbody>
</table>

3.4 Discussion/Conclusions

The hypothesis tested in this chapter, that the percentage of fireflies (PFF) in clusters generated by Algorithm 2.1 (without modification) reliably indicates the relative robustness of associated designs, is to some extent supported by the data, depending on what is meant by “reliably”. Data indicates that positive correlation exists between PFF and each of the robustness measures considered, PR and PoD, i.e. if more fireflies are in a cluster then the objective is probably more robust to uncertainty in the associated design. The histograms in Figure 3.5 and Figure 3.7 seem well approximated by Gaussian distributions and indicate, with approximately 90% probability, LSE lines fit through the data have positive slope. However, the relationship is not very strong. This is indicated by the R values in Table 3.2 and Table 3.3, or the standard deviation of slopes, considering their means. This is also intuitively evident in the plots of Figure 3.5, Figure 3.6, Figure 3.7, and Figure 3.8.

FA and CSD with minor modification may have the potential to be more strongly correlated with robustness measures. Notably, the FA tuning is optimal according to Algorithm 2.4; but this does not include any consideration of PFF and robustness measures. Inclusion of such a term in this cost function might result in FA tuning that gives stronger linear correlation between PFF and the robustness measures.
Chapter 4

Design Optimization: Elastomeric Baffle MRFD

4.1 Introduction

Magnetorheological fluid (MR fluid or MRF) is a suspension of micron-sized paramagnetic particles in a base fluid (typically water, hydrocarbon, or glycol) wherein resistance to shear is dramatically affected by an imposed magnetic field [29]. Early implementations of devices made to exploit the properties of MRF were described in Rabinow’s patents from 1951 [30] and 1954 [31]. These fluids have been refined and increasingly employed in applications where controllable shear leads to improved performance, such as automobile shock absorbers [32], seat suspensions [33], and seismic dampers [34], and haptic feedback devices for drive-by-wire vehicles [35] and surgical robots [36].

MR fluid devices (MRFDs) can be classified according to the mode by which shear forces are produced. The most common classifications are pressure driven, and direct shear [29] as illustrated in Figure 4.1. In pressure driven mode, the MRF is pushed through a gap or orifice where variable magnetic fields are generated by manipulating current in electrical coils. Pressure driven devices require enclosures, seals, and surfaces suitable to contain the pressurized fluid; these features increase device complexity and cost. In direct shear mode, the MRF fills a gap between parallel or concentric surfaces that move relative to one another. Control of the magnetic flux within the gap results in variable resistance to motion of the opposing surfaces. Other modes of operation have also been documented, including squeeze...
and magnetic gradient pinch [38]. Mixed mode devices employ more than one mode simultaneously.

Figure 4.1 Examples of pressure-driven and direct shear modes for MRFDs. a. In pressure-driven mode, MRF passes through an orifice, where an applied magnetic field varies the resistance to flow. b. Direct shear mode involves relative motion between parallel or concentric surfaces, where an applied magnetic field varies the resistance to flow. Arrows in the fluid regions indicate local velocity gradients.

Devices can also be classified according to their motion – most notably, rotary and translational devices (Figure 4.2). Rotary devices usually employ direct shear mode [39] while translational devices typically operate in pressure driven mode [40].
Figure 4.2 Three examples of MRFD types: a. direct shear, rotational; b. pressure driven, translational; and c. mixed mode, translational. The mixed mode device exhibits both pressure driven and direct shear modes since the piston moves relative to the cylinder and the (nearly incompressible) MRF is pushed from one side of the piston to the other to allow movement. Dark grey regions indicate ferromagnetic materials. Motion is indicated by dashed arrows. Dashed red lines indicate applied magnetic field.

Many factors influence MRFD performance and cost. Regardless of their mode of operation, important design considerations include fluid and material selection, physical dimensions of parts, and electromagnetic design. To efficiently use resources and satisfy design constraints,
it is important to accurately predict device performance and cost for a given set of design parameters; this is the essence of engineering design. Various methodologies have been proposed in the literature for designing MRFDs – these can be categorized as fundamental, suboptimal, or optimal design techniques. Fundamental design refers here to the practical application of basic equations and relationships that govern the behavior of MRFs.

Jolly, Bender, and Carlson [29] summarized fundamental MRFD design theory and presented three figures of merit for comparing fluids. They provided data for some commercially available fluids, and gave examples of practical devices. Yang, Spencer, Carlson, and Sain [34] presented a methodology for designing mixed-mode axial dampers by considering quasi-static controllable force, dynamic range, and minimum active fluid volume. Also presented were dynamic performance considerations and practical advice regarding control. Grunwald and Olabi [41] demonstrated basic design principles and derived useful methods of comparison for certain parameters of a pressure-driven MR valve. Rosenfeld and Werely [42] provided a case study of a simplified MR valve design. Two non-dimensional parameters (plug thickness and damping coefficient) were compared as functions of power density for various valve designs, demonstrating a methodology for specification.

Suboptimal design methods employ experimental data to enhance the fundamental MRFD design. These methods can be especially useful if physical testing and prototyping resources are abundant or if simulation resources are limited. Statistical methods can be used to guide experimentation and to extrapolate results to yield design improvements. Parlak, Engin, and Sahin [43] used Taguchi experimental design methods with FEA (finite element analysis) to specify four design parameters of an MR damper with improved maximum force and dynamic range. Ten different variants of the device were built and tested. Erol and Gurocak [44] demonstrated an interactive design process including Taguchi methods and FEA to improve the design of an MR brake (a direct shear rotary device). Three levels of 12 parameters were considered. Braking torque was improved and size of the device reduced, compared to a prototype design.
Optimal design methods employ numerical simulation and optimization algorithms. Park, Falcao da Luz, and Suleman [45] parametrically optimized the design of an MR brake. They used FEA and fundamental design equations to optimize geometrical parameters and compare multiple configurations (one or two disks and different MRF types). Nguyen and Choi [46] used PSO (particle swarm optimization) to minimize mass of an MR brake, subject to various constraints, using both magnetic circuit modeling and FEA modeling of the parameterized hardware. Eight design parameters were considered. Optimal geometry was found to be similar by either method, but the FEA optimization took 160 times longer to conduct. Assadsangabi, Daneshmand, Vahdati, Eghtesad, and Bazargan-Lari [47] used GA (genetic algorithm) optimization with FEA to minimize a cost function (a weighted sum of mass and braking torque terms) with eight design variables. Gudmundsson, Jonsdottir, and Thorsteinsson [48] demonstrated the application of PSO with FEA to optimize the design of a rotary MR brake for a prosthetic knee. Nine geometrical design variables and three cost functions were considered (maximum braking torque, minimum off-state braking, and mass).

Optimal design can result in maximum performance, but manufacturing costs can be prohibitive for MRFDs. Carlson, in 1999 [49], cited cost as “a barrier to widespread commercial acceptance in many areas” for MRFDs. The most significant cost factors being “seals, rod surface finish, precision mechanical tolerances, electromagnet assembly including the coil(s), poles and flux conduit, and the volume of MR fluid.” He described a novel, moderate-force type of device that eliminates many of these factors by the nature of its design. It operates in direct shear mode with MRF held between magnetic poles by the wicking action of an absorbent matrix, such as open-celled foam (Figure 4.3). A dense base fluid, such as grease, is typically used to prevent settling. Since the fluid is not pressurized by action of the device’s motion, cost can be reduced through elimination of seals, expensive rod finishes, and precision tolerances. Minimal MR fluid is required because, for the example of a piston-cylinder device shown, filling only the matrix between piston and cylinder and coating the cylinder wall is necessary, as opposed to filling the entire cylinder. The absorbent matrix, however, introduces its own limitations; because it must resist strain along with the fluid, it is likely to wear over time. Also, it occupies volume within the magnetically active part of the
device, which would otherwise contain ferromagnetic material, thus increasing magnetic path reluctance.

Figure 4.3 An absorbent matrix MRFD. Grey areas are ferromagnetic material. Electricity flows in a coil wrapped around the center of the piston. Dashed red lines indicate applied magnetic field.

This chapter introduces a novel type of MRFD with many of the functional benefits of Carlson’s absorbent matrix design, the *elastomeric baffle design* (Figure 4.4). Rolling elastomeric baffles contain MRF between a magnetically conductive piston and cylinder, eliminating the need for absorbent matrix materials and enabling the use of lower viscosity MR fluids. These baffles are contained between cylindrical guides, attached to the piston and the outer housing. They roll axially as the piston translates relative to the housing. Like Carlson’s absorbent matrix design, this device avoids the necessity of high pressure dynamic seals and requires a relatively small amount of MRF.
Design optimization using lumped-parameter models of the electrical, magnetic, and mechanical subsystems is detailed in the following sections. This approach is shown to be computationally efficient and accurate for design optimization. Validation of the resulting design is demonstrated by comparing predicted magnetic characteristics to those obtained using FEA and experimental results from a prototype.

4.2 Methods

The elastomeric baffle MRFD shown in Figure 4.5 provides controllable resistance to axial motion of the center shaft, relative to the housing. The central part of the shaft (the spool) and adjacent housing are made of ferromagnetic materials (typically solid iron or electrical steel). Other parts are made of nonferrous materials (aluminum or plastic). An electrical coil is wrapped around the spool and its leads extend through the hollow shaft. The gap between the spool and the housing is filled with MRF, which is contained by a pair of elastomeric baffles that are each attached to the housing on one end and to the spool on the other end. The shear resistance of the MRF, and therefore the force required to move the shaft relative to the housing, is controlled by adjusting the coil’s electrical current.
Figure 4.5 Some practical details of the elastomeric baffle MRFD design: ferromagnetic parts (the spool and central housing are low carbon steel) are separated only by MRF; other parts are not necessarily ferromagnetic materials; the shaft has a passage for the coil wire.

Figure 4.6 details the parameterized spool and housing geometry used for design optimization. Parameters included in the optimization are \( N_c \), the number of coil loops, \( r = [r_0, r_1, \ldots, r_6] \) and \( \zeta = [\zeta_0, \zeta_1, \zeta_2] \), vectors of radial and axial dimensions, respectively, \( V_s \), the applied coil voltage, \( g_c \), the coil wire gauge, and \( R_c \), the electrical resistance of the coil.

Figure 4.6 Parameterized spool and housing geometry used for design optimization.
4.2.1 Magnetic Circuit Model

The shear resistance of MRF is known to depend on the magnetic flux density within it. Accurately predicting magnetic fields and fluxes is therefore paramount to design optimization. Analytical approaches are based on Ampere’s law and the conservation of magnetic flux density, two of Maxwell’s equations. For simple geometries, low frequency excitation, and unsaturated magnetic fields these equations can be combined to yield Hopkinson’s law,

\[ F = R \phi \]  

where \( F = N_i \) is magneto motive force, \( i \) is the coil electrical current, \( \phi \) is the magnetic flux, \( R = \frac{\ell}{\mu A} \) is the magnetic reluctance, \( \ell \) is the mean flux path length, \( \mu \) is the magnetic permeability, and \( A \) is the cross-sectional area of the magnetic flux path. Hopkinson’s law (A.14) is the basis of magnetic circuit modeling, and is most accurate in the range where magnetic flux density increases proportionally to magnetic field intensity - the linear range [50]. The convenience of this method is that the reluctance of each part in the magnetic circuit (\( R_1, R_2, \) etc.) can be computed separately, and a composite model constructed (Figure 4.7).

![Figure 4.7 Lumped parameter magnetic circuit model. Magneto motive force drives magnetic flux through a series of \( n \) reluctances.](image)

68
Programmatic application of (A.14) to the spool and housing components of Figure 4.6 is accomplished using the subdivided geometry of Figure 4.8. Three distinct cases, each with different flux paths, are deemed sufficient to accurately model the circuit while not unnecessarily constraining the design.

![Figure 4.8](image)

Figure 4.8 Detailed parametric geometry for a magnetic circuit model corresponding to Figure 4.6. Evaluation of one-quarter of the cross-section is sufficient due to symmetry of the design.

For each case, the magnetic flux path is broken into \( k \) discrete reluctances (highlighted in red in Figure 4.8.a-c),

\[
\mathcal{R}_k = \frac{\ell_k}{\mu_k A_k}
\]  

(4.2)

The strategy for computing representative areas \( A_k \) is generally to average the area of the conic section at the beginning and end of each segment. Table 4.1 summarizes these parametric expressions for each segment in Figure 4.8.b. More accurate, and more computationally expensive, calculations are warranted for segments 5 and 6, especially when the radial distance between \( r_a \) and \( r_b \) becomes large:

\[
\int_a^b \frac{dr}{A(r)} = \frac{r_b - r_a}{A_{ab}}
\]  

(4.3)

where \( A_{ab} \) is the area used to approximate the reluctance of a ring between \( r_a \) and \( r_b \). The Appendix includes similar expressions to those in Table 4.1 for Figure 4.8.a and c. Representative lengths \( \ell_k \) are taken between the midpoints of the areas.
Table 4.1  Parameters of (4.2), corresponding to Figure 4.8.b.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$\ell_k$</th>
<th>$A_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\frac{\zeta_0}{2}$</td>
<td>$\pi \left( r_1^2 - r_0^2 \right)$</td>
</tr>
<tr>
<td>1</td>
<td>$\frac{\zeta_2 - \zeta_1}{2}$</td>
<td>$\frac{\pi}{2} \left( r_1^2 - r_0^2 + (r_1 + r_0) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_0)^2} \right)$</td>
</tr>
<tr>
<td>2</td>
<td>$\frac{r_1 - r_3}{2}$</td>
<td>$\frac{\pi}{2} \left( (r_1 + r_0) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_0)^2} + (r_1 + r_3) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_3)^2} \right)$</td>
</tr>
<tr>
<td>3</td>
<td>$\frac{\zeta_2 - \zeta_1}{2}$</td>
<td>$\frac{\pi}{2} \left( (r_1 + r_3) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_3)^2} + (r_1 + r_3) \sqrt{\zeta_2^2 + (r_1 - r_3)^2} \right)$</td>
</tr>
<tr>
<td>4</td>
<td>$\frac{r_1 - r_3}{2}$</td>
<td>$\frac{\pi}{2} \left( (r_1 + r_3) \sqrt{\zeta_2^2 + (r_1 - r_3)^2} + 2 r_1 \zeta_2 \right)$</td>
</tr>
<tr>
<td>5</td>
<td>$r_4 - r_1$</td>
<td>$2 \pi \left( r_4 - r_1 \right) \frac{\zeta_2}{\ln \left( \frac{t_4}{r_1} \right)}$</td>
</tr>
<tr>
<td>6</td>
<td>$r_5 - r_4$</td>
<td>$2 \pi \left( r_5 - r_4 \right) \frac{\zeta_2}{\ln \left( \frac{t_5}{r_4} \right)}$</td>
</tr>
<tr>
<td>7</td>
<td>$\frac{\zeta_0 + \zeta_2}{2}$</td>
<td>$\pi \left( r_6^2 - r_5^2 \right)$</td>
</tr>
</tbody>
</table>

Each magnetic circuit has reluctances arranged in series, thus the net path reluctance is their sum. Magnetic flux for a given set of design parameters is

$$\phi = \frac{N_c i}{\sum_k R_k} \quad (4.4)$$

The magnetic flux density, $B$, in any portion of the circuit can be computed by dividing (A.15) by the appropriate cross-sectional area:

$$B_k = \frac{\phi}{A_k} \quad (4.5)$$
4.2.2 Electrical Circuit Model

A lumped-parameter electrical model (Figure 4.9), analogous to the magnetic circuit of Figure 4.7, is also used in the MRFD design optimization. In this model, $r_c$ represents coil resistance, $L_c$ is coil inductance, and $V_s$ is the source voltage.

![Figure 4.9 Lumped parameter electric circuit model. Voltage source drives current through a series of resistance and inductance.](image)

The electrical state equation can be obtained using Kirchhoff’s voltage law. Summing component voltages around the circuit of Figure 4.9, assuming $R = \sum k R_k$ is time-invariant, and rearranging terms yields

$$\frac{di}{dt} + \frac{R R_c}{N_c^2} i = \frac{R}{N_c^2} V_s \quad (4.6)$$

The circuit’s transient response can be characterized by this first-order differential equation’s time constant:

$$\tau = \frac{N_c^2}{R R_c} \quad (4.7)$$

Also from (4.6), the steady-state current is $i_{ss} = \frac{V_s}{R_c}$, and the associated electrical power is

$$W_e = i_{ss}^2 R_c \quad (4.8)$$

Parametric expressions relating coil geometry and resistance to the spool and housing dimensions are also considered. Magnet wire diameter as a function of gauge [51], $g_c$ is
\[
d_{w} = 92^{36-\theta_{c}} 0.005 \text{ in} (4.9)
\]

Therefore, for a given \( r \) and \( \zeta \), the number of coils that fit within the spool is approximately

\[
N_{e} = f_{c} \frac{\zeta_{o} r_{2} - r_{1}}{d_{w}} (4.10)
\]

where \( f_{c} \) is an empirical filling factor for the coil; \( f_{c} = 0.75 \) was used for this design.

The length of conductor necessary to construct the coil is:

\[
\ell_{w} = 2\pi \frac{r_{1} + r_{2}}{2} N_{e} (4.11)
\]

Coil resistance is therefore

\[
R_{c} = \rho_{e} \frac{\ell_{w}}{\pi \left( \frac{d_{w}}{2} \right)^{2}} + R_{0} (4.12)
\]

where \( \rho_{e} \) is the electrical resistivity of the conductor, and \( R_{0} \) represents the added resistance of the coil leads and circuit wiring.

**4.2.3 Mechanical System Properties**

Mechanical aspects of the MRFD design include the spool mass and the force required to actuate the device. The spool mass is

\[
m = m_{c} + m_{s} (4.13)
\]

where

\[
m_{c} = \frac{\pi}{4} d_{w}^{2} \ell_{w} \rho_{m,w}
\]

is a parametric expression for mass of the coil, \( \rho_{m,w} \) is the coil’s mass density,

\[
m_{s} = \pi \left( (\zeta_{o} + 2\zeta_{2}) (r_{a}^{2} - r_{0}^{2}) - 2\zeta_{1} (r_{3}^{2} - r_{0}^{2}) - \zeta_{o} (r_{4}^{2} - r_{1}^{2}) \right) \rho_{m,s}
\]

is the mass of the ferromagnetic part of the spool, and \( \rho_{m,s} \) is the its mass density.
The piston actuation force is primarily due to shear resistance in the MRF, which can be modeled as a Bingham fluid [41]. The Bingham fluid model assumes two additive components of shear force; one is a function of viscosity and the other of magnetic flux density:

\[ F = F_\eta + F_\beta \]

The viscous force component \( F_\eta \) is shear rate multiplied by shear area and dynamic viscosity. A parametric expression for this, given the design geometry, is

\[ F_\eta = 2 \pi r_3 \frac{\nu}{r_5 - r_4} \eta \left( \frac{\nu}{r_5 - r_4} \right) + \pi 2r_2 \frac{\nu}{r_5 - r_2} \eta \left( \frac{\nu}{r_5 - r_2} \right) \tag{4.14} \]

where \( \nu \) is spool velocity relative to the housing, \( \eta \) is dynamic viscosity, and shear rate is \( \nu \) divided by the radial distance across the fluid. Idealized incompressible Newtonian flow conditions are assumed.

The magnetic force component \( F_\beta \) is approximately the yield stress (a function of the magnetic flux density) multiplied by the magnetically active spool area:

\[ F_\beta = 2 \pi r_4 \frac{\nu}{r_5 - r_4} \tau_y(B) \tag{4.15} \]

where \( \tau_y(B) \) is the field-dependent yield stress. Empirical relationships for \( \eta \) and \( \tau_y \) are shown in Figure 4.10.a and b, respectively.

### 4.2.4 Material Properties

Several material properties and dependencies must be considered in the design process: MRF dynamic viscosity versus shear rate, MRF yield strength versus magnetic flux density, coil electrical resistivity, coil mass density, spool mass density, MRF magnetic permeability, and spool magnetic permeability. The nonlinear shear parametric dependencies of MRF are shown in Figure 4.10.
Figure 4.10  Shear parametric dependencies of MRF: a. Dynamic viscosity vs. shear rate, with negligible magnetic flux density; b. yield stress vs. magnetic flux density with negligible shear rate. Adapted from [29] MRX-336AG, a 36% iron by volume, silicone-based MRF.

Copper magnet wire is used for the coil. Typical mass density and resistivity values for copper are \( \rho_m = 8.96 \, \text{g/cm}^3 \) and \( \rho_e = 1.68 \times 10^{-8} \, \Omega \cdot \text{m} \), respectively. The spool and housing are made from low-carbon steel; a mass density of \( \rho_{m,s} = 7.87 \, \text{g/cm}^3 \) is assumed.

The magnetic dependencies of MRF and low-carbon steel are shown Figure 4.11. Relative permeability vs. magnetic flux density are shown in Figure 4.11.a. Magnetic flux density versus field intensity are shown in Figure 4.11.b.
4.2.5 Design Optimization

Equations (4.2) through (4.15) are expressed in terms of the design parameters of Figure 4.6 \( (r, \zeta, V_s, g_c, \text{and } R_c) \). MRFD performance can be quantified for a specific parameter set using the expressions for time constant \( \tau \), electrical power \( W_e \), spool mass \( m \), viscous force \( F_\eta \), and magnetic force \( F_B \) ((4.7), (4.8), (4.13), (4.14), and (4.15), respectively). An optimal design can be obtained by minimizing the cost function \( f(x) \), a weighted sum of these terms, while also satisfying inequality and equality constraints, \( g(x) \) and \( h(x) \):

\[
\min_x f(x) \\
\text{subject to:} \\
g_i(x) \leq 0 \text{ for each } i \in \{1 \ldots n_g\} \\
h_j(x) = 0 \text{ for each } j \in \{1 \ldots n_h\}
\]  

(4.16)

Figure 4.11 Magnetic parametric dependencies of MRF and low-carbon steel: a. relative permeability \( \mu_r \) vs. magnetic flux density \( B \); b. Magnetic flux density \( B \) vs. field intensity \( H \). Low carbon steel parameters determined experimentally. MRF parameters adapted from [29]: MRX-336AG, a 36% iron by volume, silicone-based fluid.
where $x$ is the set of optimization variables, $n_g$ is the number of inequality constraints, and $n_h$ is the number of equality constraints.

The optimization variables $x = [x_0, x_1, \ldots, x_{14}]$ given in Table 4.2 include magnetic flux $\phi$, which is needed to solve the transcendental equations of Figure 4.12.

Table 4.2  Optimization variables are a partially modified version of the design parameters with the addition of magnetic flux density ($\phi$). Units for each variable are shown along with lower and upper bounds used in the design. These limits indicate the designer’s preference for overall size, material limitations, and production capability.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$x_k$</th>
<th>units</th>
<th>$x_L$</th>
<th>$x_U$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$r_0$</td>
<td>mm</td>
<td>1.5</td>
<td>5.0</td>
</tr>
<tr>
<td>1</td>
<td>$r_1$</td>
<td>mm</td>
<td>15.0</td>
<td>20.0</td>
</tr>
<tr>
<td>2</td>
<td>$r_2 - r_1$</td>
<td>mm</td>
<td>0.1</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>$r_3$</td>
<td>mm</td>
<td>7.0</td>
<td>9.0</td>
</tr>
<tr>
<td>4</td>
<td>$r_4$</td>
<td>mm</td>
<td>17.5</td>
<td>20.5</td>
</tr>
<tr>
<td>5</td>
<td>$r_5 - r_4$</td>
<td>mm</td>
<td>0.5</td>
<td>2.5</td>
</tr>
<tr>
<td>6</td>
<td>$r_6$</td>
<td>mm</td>
<td>21.0</td>
<td>30.0</td>
</tr>
<tr>
<td>7</td>
<td>$r_7$</td>
<td>mm</td>
<td>17.5</td>
<td>19.5</td>
</tr>
<tr>
<td>8</td>
<td>$\zeta_0$</td>
<td>mm</td>
<td>5.0</td>
<td>35.0</td>
</tr>
<tr>
<td>9</td>
<td>$\zeta_1$</td>
<td>mm</td>
<td>5.0</td>
<td>15.0</td>
</tr>
<tr>
<td>10</td>
<td>$\zeta_2$</td>
<td>mm</td>
<td>5.0</td>
<td>25.0</td>
</tr>
<tr>
<td>11</td>
<td>$g_c$</td>
<td>-</td>
<td>25.1</td>
<td>36.9</td>
</tr>
<tr>
<td>12</td>
<td>$V_s$</td>
<td>V</td>
<td>6.0</td>
<td>24.0</td>
</tr>
<tr>
<td>13</td>
<td>$R_0$</td>
<td>$\Omega$</td>
<td>1.0</td>
<td>50.0</td>
</tr>
<tr>
<td>14</td>
<td>$\phi$</td>
<td>mWb</td>
<td>0.1</td>
<td>2.0</td>
</tr>
</tbody>
</table>

This magnetic circuit does not have a closed-form solution due to the nonlinear dependencies highlighted in Figure 4.11.a; the relative permeabilities of the MRF and electrical steel cannot be considered constants without unnecessarily limiting the range of flux density. Designating $\phi$ as an optimization variable and (A.15) as an equality constraint provides an efficient means of obtaining the numerical solution while also performing the optimization.
The magnetic circuit model results in a transcendental set of equations. Arrows indicate the circular linkage between these equations: magnetic permeabilities are functions of flux densities; reluctance is a function of permeabilities; magnetic flux is a function of reluctance; and flux densities are a function of magnetic flux. This circular dependency is prevented if permeabilities can be considered constant.

The cost function, including weights, used in the design optimization is:

\[
f(x) = 
\ldots F_B(x) \cdot (-50/lbf) + 
\ldots F_\eta(x)|_{v_d} \cdot (2.0/lbf) + 
\ldots \tau(x) \cdot (0.1/msec) + 
\ldots m(x) \cdot (2.0/lbm) + 
\ldots W_e(x) \cdot (0.5/W)
\]

where viscous force \( F_\eta(x) \) is assessed at the design velocity \( v_d = 0.2m/s \), a typical value from testing. The inequality constraints for this system are

\[
g(x) = \left\{ \frac{x_0 - x_1}{x_0 - x_3}, \frac{x_4 + x_5 - x_6 + 5}{x_2 + x_1 - x_4}, \frac{x_9 - x_8/2 - x_10}{\max(|B(x)_k|)/T - 0.65} \right\}
\]

The first five terms of \( g(x) \) are geometrical limitations. The last term reflects a preference to avoid highly saturated magnetic fields. The single equality constraint is
\[ h(x) = x_{14} - \phi(x) \]  

(4.19)

where \(\phi(x)\) is from (A.15).

The cost function and constraints can be combined using a penalty function

\[ p(x) = f(x) + K_g \sum_{i=0}^{n_g} \max(0, g_i(x)) + K_h \sum_{i=0}^{n_h} \max(0, |h_i(x)| - TOL) \]  

(4.20)

where \(K_g = 1000\) and \(K_h = 1000\) are penalty factors used for inequality and equality constraint violation, respectively, and \(TOL = 0.00001\) is the convergence tolerance used for equality constraints.

The optimization problem defined by (A.14) to (4.19) is nonlinear and discontinuous; specifying initial optimization variables that result in convergent solutions for many optimization methods is non-trivial. Gradient-based optimization methods such as the conjugate gradient method [52] can efficiently drive to minimum solutions, but frequently present difficulty converging initially or when not in the vicinity of a locally minimum solution. For this reason, a variant of the firefly algorithm [7] is employed to find candidate solutions and a conjugate gradient method is used to efficiently drive from the candidate solutions to local minima. Pseudo code for this composite method is given in Algorithm 2.1.

The firefly algorithm is an evolutionary optimization method modeled upon firefly behavior, where a population of optimization vectors \((x)'s, \text{referred to as fireflies}\) migrate toward better fit neighbors in the design space. It can be tuned to favor local solutions, rather than more exhaustively seeking the best global solution; therefore, it can be used to identify promising local minima for examination.
Algorithm 4.1  Optimization pseudo code.  Assignment is indicated by “←”.  Items within loops are indented.  α, β, and γ are the firefly algorithm tuning parameters and can be functions of migration number.

One of the benefits of Algorithm 2.1 is that it facilitates design exploration.  Steps 4.d and 5 of the algorithm are accomplished using cost-sorted distance (CSD) method described by Elliott, et al. [53], where the distances between the fireflies and their relative costs are used to identify potentially good design alternatives.  The normalized fireflies are ranked according to cost and the distance (the Euclidean norm of the vector) between each pair of them is computed.  Similar designs are separated in the design space by relatively small distance so that local minima, when they occur, may be found in clusters.  The alternative designs to be considered are the lowest cost members of each cluster.
A major advantage of this method is in giving the designer a graphical tool for down-selecting from a large population of results, for problems with many degrees of freedom, where direct visualization is not practical. Figure 2.3, however, shows for a two-degree-of-freedom system (where the cost function can also be shown as a contour plot) how the method can be used to find promising results. The contour plot of Figure 2.3.a shows the cost function along with 50 fireflies in their final migration positions, which are obviously near local minima. Figure 2.3.b shows a CSD plot where colors correspond to the distance between the normalized fireflies. Clusters can be easily seen as bands of similar color in this plot. Thirty-two of the fireflies have converged near point A, and are dark blue in the bottom row of Figure 2.3.b. Two data points have converged near point B and are teal-colored in the bottom row; 8 data points have converged near point C and are also teal-colored in the bottom row; 6 data points have converged near point D and are green in the bottom row; and the right-most two data points have not converged as near a minimum as the other points – these two points can be seen near point D in Figure 2.3.a. Points that are close to one another can be distinguished by dark blue data points in Figure 2.3.b and local minima can be identified by a banded or plaid appearance in the CSD plot. This plot is symmetric since the distance is the same between the same pair of points (the distance from firefly x to firefly y is the same as the distance from firefly y to firefly x), so it is actually redundant to show both the lower and upper diagonal portions. A comparison of these two figures demonstrates the value of the CSD method. It is easy to see clusters of results where local minimums are likely, and to find the champion (the lowest cost member) of each. This is of great value when the cost function is not otherwise easy to visualize and can be done for problems with any number of degrees of freedom.
Figure 4.13  A benchmark 2D cost function and the corresponding CSD plot.  a. A contour plot of an example 2-D cost function with fireflies in their final migration positions - clusters of fireflies have resulted near local minimums; b. distances between normalized fireflies, arranged in order of cost – clusters of fireflies can also be seen in this plot as bands of color, and their

The CSD method is automated in step 4.d of Algorithm 2.1, where champions are detected by finding the lowest cost members of populated histogram bins. These champions replace the population’s lowest cost members after each firefly migration. It can be used graphically by the designer, in step 5 to investigate the population before proceeding with conjugate gradient optimization.

4.2.6 FEA Validation

Though computationally expensive compared to magnetic circuit analysis, and therefore preferably not included in the optimal design process, FEA is ideal for validating the optimal design prior to prototype construction. Validation is performed by comparing field predictions made using FEA to those from magnetic circuit analysis.
Predicting magnetic flux density throughout the simple geometry of Figure 4.6 is an axisymmetric, steady-state problem, and is within the capability of many FEA programs. FEMM [54] is one such program; it has been used with the settings shown in Table 4.3 to validate the optimal MRFD design resulting from Algorithm 2.1. Performing the analysis requires specification of part geometry as well as the geometry of a bounded region around the parts; material definitions for the different enclosed regions; and specification of the electrical circuit properties. An infinite boundary can be approximated using the asymptotic boundary condition. Measured B-H curves (Figure 4.11.b) should be specified where possible. Properties for other materials can be selected from the software’s database.

Table 4.3  FEMM settings for solving the magnetic field problem using FEA

<table>
<thead>
<tr>
<th>Setting</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem Type</td>
<td>Axisymmetric</td>
</tr>
<tr>
<td>Frequency (Hz)</td>
<td>0</td>
</tr>
<tr>
<td>Solver Precision</td>
<td>1e-008</td>
</tr>
<tr>
<td>Min Angle</td>
<td>30</td>
</tr>
<tr>
<td>AC Solver</td>
<td>Successive Approximation</td>
</tr>
</tbody>
</table>

Results can be examined after the mesh generation and analysis steps are run. Useful results include a contour plot of $|\mathbf{B}|$ throughout the spool and housing regions and throughout the MRF, and specifically at points corresponding to those predicted using magnetic circuit analysis. Centroids of sections shown in Figure 4.8 are selected for comparison to magnetic circuit analysis.

### 4.2.7 Prototype Development

Figure 4.14 shows a solid model prototype of the MRFD. The spool and central housing are made of electrical steel. Cylindrical parts adjacent to the spool and central housing are
machined from translucent acrylic. Threaded shafts machined from aluminum are screwed into the spool. Latex elastomeric baffles or sleeves are held with O-rings between the spool and cylindrical guide, and between the inner and outer housing pieces, on both sides. The volume between the two baffles is filled with MRF. Two ports, with medical check valves, are used to evacuate air and inject the fluid by syringe. The housing is attached to a base block through a spherical bearing (Figure 4.15).

Figure 4.14 Solid model images of the prototype MRF damper. A spherical bearing suspends the MRFD within the base block and the shaft moves axially with the cylinder.

A photograph of the prototype MRFD is shown in Figure 4.15.
Figure 4.15  A photograph of the prototype MRFD. The base block and center shafts are machined from aluminum. The housing is a composite of three pieces: the center piece is machined from low-carbon steel; and the outer pieces are machined from translucent acrylic. The force sensor generates a voltage between 0.5 and 4.5 volts, proportional to the axial force (between -50 and 50 lbf, respectively).

4.2.8 Experimental Validation

A prototype MRFD can be tested using the experimental setup of Figure 4.16. Spool motion is induced by manipulation of the joystick. One shaft of the MRFD is attached through an axial force sensor and rod end (with spherical bearing) to the joystick. Axial displacement of the spool is computed from joystick angle which is sensed by a potentiometer in the joystick base. Force applied to the MRFD is measured using the axial force sensor. Various other sensors are used to indicate the desired current level, and the ECM’s 2A driver controls current. Sensor signals are measured by the ECM and can be relayed to a PC using controller area network (CAN) serial communication.
Figure 4.16 Experimental setup: a Caterpillar A5M2 electronic control module (ECM) is used to measure operator inputs (current enabled, current level, current on/off), sensor signals (force and angular displacement), and to supply controlled current to the MRFD coil. Data is recorded to a computer using controller area network (CAN) communication.

4.3 Results

A population of 100 design vectors (fireflies) were processed through 21 migrations, requiring 4300 cost function evaluations, to produce the results of Figure 4.17. Shown for a subset of the migrations are the migration number, maximum distance between any two fireflies $|d_{\text{max}}|$, the mean distance between all fireflies $|d_{\text{mean}}|$, the lowest cost $f(x_0)$, and a scatter plot of $|d|$ between each pair of members of the normalized and sorted population (the CSD plot). Normalization is done by mapping each element of each firefly between 0 and 1 depending on its position between corresponding elements of $x_L$ and $x_U$. Therefore, the maximum possible distance is $|d|_{\text{max_possible}} = \sqrt{15} \approx 3.87$, where 15 is the length of the design vector. In each plot, the population is arranged in order of $f(x)$, with the lowest cost design at the origin. Abscissa and ordinate axes are identical: firefly population number, from 1 to 100. The color of each data point corresponds to the distance between the fireflies of those coordinates. Colors
cannot be compared from one plot to another because each plot’s color scale is normalized to show local contrast. The usefulness of the plots is that a designer can observe at a glance whether results are near one another, and whether promising local minimums (clusters) are likely exhibited. Clusters appear as bands of similar color. Contrasting bands that intersect the axes near the origin indicate low cost results that are relatively dissimilar to the population’s lowest cost result.

Figure 4.17 Results summary for the modified firefly portion of Algorithm 2.1. An initial population of 100 fireflies is randomly derived within the limits and then processed through 21 migrations. Color is auto-scaled for each plot so that patterns are visible in all migrations. Cost of the best individual as well as the mean and max distance (Euclidean norm) between all of the fireflies are shown for each migration.
Producing the results of Figure 4.17 required 60 seconds of computation using a Mathcad script on a computer with one 2.7 GHz quad-core processor. Throughout this progression, \(|d|_{\text{max}}\) was reduced from 0.280 to 0.094; \(|d|_{\text{mean}}\) was reduced from 0.156 to 0.029; and \(f(x_0)\) was reduced from 2197 to -449. The plot for migration 21 exhibits a banded appearance that is indicative of alternative low cost designs.

Figure 4.18 shows \(|d|\) from the normalized lowest cost result, \(\overline{x}_0\), to each of the other normalized results for the last migration of Figure 4.18 – this is the bottom-most row of the CSD plot. The fireflies are arranged in order of cost (lowest cost at the origin, highest cost at 100).

![Figure 4.18 Distances (Euclidean norms) of fireflies from the lowest cost result, for the last migration. Results corresponding to the boxed data points are used as initial design vectors for a gradient based optimization. These data points are also the bottom-most row from the contour plot for migration 21 in Figure 4.17.](image)

Clusters of results are identified using a histogram approach, the population is sorted into bins according to distance and the lowest cost member of each bin extracted. Red boxes in Figure
4.18 mark the lowest cost fireflies (champions) of identified clusters. These correspond to bands of contrasting color in plots of Figure 4.17. Champions from Figure 4.18 are used as initial design vectors for a subsequent conjugate gradient optimization step.

Figure 4.19 summarizes the results of this final step. The first column gives the cost of each of the champions from Figure 4.18, and of the vector resulting from a conjugate gradient optimization using that champion as an initial value. Firefly results are indicated with a subscript $ff$, and the conjugate gradient results are indicated with a subscript $cg$. Geometry of these results are shown in the second column of Figure 4.19 here blue dashed lines are the firefly champions and solid red lines are the result of continued conjugate gradient optimization. Conjugate gradient optimization was done using an inbuilt method of Mathcad’s $Minimize()$ function with convergence and constraint tolerances each set to 0.00001. This method would not converge to a solution with these settings given the average of $x_L$ and $x_U$ as a starting point (the firefly optimization proved a useful first step in providing candidate solutions).
Figure 4.19 Cost and geometry of the five cluster champions from Figure 4.18 (blue) and their conjugate gradient optimal counterparts (red). Cost of each result is shown in the first column with respective subscripts and colors. Dashed blue lined geometry are firefly results that are used as initial values, and solid red lined geometry are associated conjugate gradient optimization results.
Table 4.4 shows the 5 local minima resulting from Algorithm 2.1. The lower and upper limits for each variable are included again for reference.

Table 4.4 The optimization variables, their units, firefly lower and upper limits, and optimal results after conjugate gradient optimization for each of the cluster champions of Figure 4.18. These correspond to the results with solid red lines in Figure 4.19.

<table>
<thead>
<tr>
<th>( k )</th>
<th>( x_k )</th>
<th>units</th>
<th>( x_L )</th>
<th>( x_U )</th>
<th>( x_0 )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( r_0 )</td>
<td>mm</td>
<td>1.5</td>
<td>5.0</td>
<td>1.72</td>
<td>3.48</td>
<td>3.42</td>
<td>3.53</td>
<td>3.55</td>
</tr>
<tr>
<td>1</td>
<td>( r_1 )</td>
<td>mm</td>
<td>15.0</td>
<td>20.0</td>
<td>18.95</td>
<td>19.34</td>
<td>19.24</td>
<td>18.96</td>
<td>19.10</td>
</tr>
<tr>
<td>2</td>
<td>( r_2 - r_1 )</td>
<td>mm</td>
<td>0.1</td>
<td>2.0</td>
<td>1.40</td>
<td>0.90</td>
<td>0.42</td>
<td>1.14</td>
<td>0.94</td>
</tr>
<tr>
<td>3</td>
<td>( r_3 )</td>
<td>mm</td>
<td>7.0</td>
<td>9.0</td>
<td>7.00</td>
<td>7.95</td>
<td>8.14</td>
<td>8.04</td>
<td>8.16</td>
</tr>
<tr>
<td>4</td>
<td>( r_4 )</td>
<td>mm</td>
<td>17.5</td>
<td>20.5</td>
<td>20.35</td>
<td>20.25</td>
<td>19.65</td>
<td>20.09</td>
<td>20.04</td>
</tr>
<tr>
<td>5</td>
<td>( r_5 - r_4 )</td>
<td>mm</td>
<td>0.5</td>
<td>2.5</td>
<td>0.50</td>
<td>0.53</td>
<td>0.50</td>
<td>0.85</td>
<td>0.91</td>
</tr>
<tr>
<td>6</td>
<td>( r_6 )</td>
<td>mm</td>
<td>21.0</td>
<td>28.0</td>
<td>28.00</td>
<td>28.00</td>
<td>28.00</td>
<td>28.00</td>
<td>28.00</td>
</tr>
<tr>
<td>7</td>
<td>( r_7 )</td>
<td>mm</td>
<td>17.5</td>
<td>19.5</td>
<td>17.50</td>
<td>18.43</td>
<td>18.46</td>
<td>18.41</td>
<td>18.44</td>
</tr>
<tr>
<td>8</td>
<td>( \zeta_0 )</td>
<td>mm</td>
<td>5.0</td>
<td>35.0</td>
<td>25.12</td>
<td>21.29</td>
<td>8.50</td>
<td>18.36</td>
<td>19.92</td>
</tr>
<tr>
<td>9</td>
<td>( \zeta_1 )</td>
<td>mm</td>
<td>5.0</td>
<td>15.0</td>
<td>7.99</td>
<td>10.14</td>
<td>7.11</td>
<td>9.83</td>
<td>9.64</td>
</tr>
<tr>
<td>10</td>
<td>( \zeta_2 )</td>
<td>mm</td>
<td>5.0</td>
<td>25.0</td>
<td>8.49</td>
<td>10.14</td>
<td>13.54</td>
<td>11.37</td>
<td>12.24</td>
</tr>
<tr>
<td>11</td>
<td>( g_c )</td>
<td>-</td>
<td>25.1</td>
<td>36.9</td>
<td>30.18</td>
<td>31.83</td>
<td>36.60</td>
<td>31.14</td>
<td>31.02</td>
</tr>
<tr>
<td>12</td>
<td>( V_s )</td>
<td>V</td>
<td>6.0</td>
<td>24.0</td>
<td>16.66</td>
<td>21.14</td>
<td>22.24</td>
<td>18.24</td>
<td>18.35</td>
</tr>
<tr>
<td>13</td>
<td>( R_0 )</td>
<td>( \Omega )</td>
<td>1.0</td>
<td>50.0</td>
<td>16.93</td>
<td>22.90</td>
<td>18.78</td>
<td>25.25</td>
<td>27.85</td>
</tr>
<tr>
<td>14</td>
<td>( \phi )</td>
<td>mWb</td>
<td>0.1</td>
<td>2.0</td>
<td>0.68</td>
<td>0.69</td>
<td>0.67</td>
<td>0.63</td>
<td>0.62</td>
</tr>
</tbody>
</table>

Table 4.5 shows notable results for the designs examined, including components of \( f(x) \). Terms associated with the coil, shown but not directly included as costs or constraints in the optimization are design current, current density, resistance, turns, and wire length. There are practical limits for some of these, so they are monitored in the results, and can be constrained if necessary (but this is not preferable because unnecessary additional terms slow optimization convergence).
Table 4.5 Important results for the designs of Figure 4.19.

<table>
<thead>
<tr>
<th>quantity</th>
<th>units</th>
<th>expression</th>
<th>$x_0$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>max force</td>
<td>lbf</td>
<td>$F_B$</td>
<td>18.80</td>
<td>18.01</td>
<td>15.63</td>
<td>15.03</td>
<td>14.05</td>
</tr>
<tr>
<td>min force</td>
<td>lbf</td>
<td>$F_{\eta}</td>
<td>_{\epsilon_d}$</td>
<td>1.41</td>
<td>1.49</td>
<td>1.45</td>
<td>1.42</td>
</tr>
<tr>
<td>time constant</td>
<td>msec</td>
<td>$\tau$</td>
<td>33.56</td>
<td>22.92</td>
<td>8.53</td>
<td>26.80</td>
<td>26.10</td>
</tr>
<tr>
<td>mass</td>
<td>lbm</td>
<td>$m$</td>
<td>0.88</td>
<td>0.83</td>
<td>0.68</td>
<td>0.81</td>
<td>0.87</td>
</tr>
<tr>
<td>electric power</td>
<td>W</td>
<td>$W_e$</td>
<td>4.10</td>
<td>4.88</td>
<td>6.38</td>
<td>3.29</td>
<td>2.98</td>
</tr>
<tr>
<td>cost</td>
<td>-</td>
<td>$f(x)$</td>
<td>-930</td>
<td>-891</td>
<td>-773</td>
<td>-743</td>
<td>-694</td>
</tr>
<tr>
<td>design current</td>
<td>$A$</td>
<td>$i_{ss}$</td>
<td>0.49</td>
<td>0.46</td>
<td>0.49</td>
<td>0.36</td>
<td>0.37</td>
</tr>
<tr>
<td>max flux density</td>
<td>$T$</td>
<td>$\max_{k}(</td>
<td>B_k</td>
<td>)$</td>
<td>0.65</td>
<td>0.65</td>
<td>0.60</td>
</tr>
<tr>
<td>current density</td>
<td>$A/mm^2$</td>
<td>$i_{ss}/A_w$</td>
<td>9.78</td>
<td>14.35</td>
<td>38.53</td>
<td>11.39</td>
<td>11.46</td>
</tr>
<tr>
<td>coil resistance</td>
<td>$\Omega$</td>
<td>$R_c - R_0$</td>
<td>16.53</td>
<td>23.11</td>
<td>26.79</td>
<td>24.75</td>
<td>22.15</td>
</tr>
<tr>
<td>coil turns</td>
<td>-</td>
<td>$N_c$</td>
<td>405.84</td>
<td>354.22</td>
<td>165.36</td>
<td>384.56</td>
<td>343.40</td>
</tr>
<tr>
<td>wire length</td>
<td>ft</td>
<td>$\ell_w$</td>
<td>164.37</td>
<td>144.53</td>
<td>66.28</td>
<td>154.78</td>
<td>138.55</td>
</tr>
<tr>
<td>wire gauge</td>
<td>-</td>
<td>$g_e$</td>
<td>30</td>
<td>32</td>
<td>36</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>gap distance</td>
<td>mm</td>
<td>$x_5$</td>
<td>0.50</td>
<td>0.53</td>
<td>0.50</td>
<td>0.85</td>
<td>0.91</td>
</tr>
</tbody>
</table>

The lowest cost result, $x_0$, from Table 4.5, is the basis for the design built and tested. FEA was performed on this design to validate results predicted by magnetic circuit analysis. Figure 4.20 shows a contour plot of $|B|$ for design $x_0$, produced using FEMM. Roughly 20 seconds was required to compute this result. Notably, (assuming this run is typical) replacing magnetic circuit analysis with FEA for the 4300 firefly cost function evaluations of Figure 4.17 with the same computer and settings, would have required 1 day of computation, rather than 1 minute.
Figure 4.20  FEA result for the best design from Table 4.5 showing $|B|$ contour lines in the ferromagnetic spool and housing. The 405-turn 30 AWG coil, filling the gap within the spool, is driven with 0.5 A.

Table 4.6 shows a comparison between magnetic circuit analysis and FEA results for the optimal design. In the two columns are $|B|$ predicted by each method, at locations corresponding to the segments of Figure 4.8. FEA sample locations are the segment centroids (shown in Figure 4.21.a). All results are within 0.06 T, or 10% difference. It is notable that the magnetic flux density from optimization is at the limit of 0.65 T for segment 2 and is not as high for the other segments (it was not optimal to push all segments to this limit).
Table 4.6 Predicted $|B|$ for the optimal design. Magnetic circuit model results are shown in the first column, FEA results in the second, and percent difference in the third, for each segment corresponding to Figure 4.8.a. The centroid of each segment was used for FEA sample points; these locations are shown in Figure 4.21.a.

<table>
<thead>
<tr>
<th>$k$</th>
<th>circuit predicted (T)</th>
<th>FEA predicted (T)</th>
<th>% difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.62</td>
<td>0.64</td>
<td>3.2</td>
</tr>
<tr>
<td>1</td>
<td>0.62</td>
<td>0.63</td>
<td>1.6</td>
</tr>
<tr>
<td>2</td>
<td>0.65</td>
<td>0.71</td>
<td>9.2</td>
</tr>
<tr>
<td>3</td>
<td>0.62</td>
<td>0.59</td>
<td>-4.8</td>
</tr>
<tr>
<td>4</td>
<td>0.55</td>
<td>0.55</td>
<td>0.0</td>
</tr>
<tr>
<td>5</td>
<td>0.54</td>
<td>0.51</td>
<td>-5.6</td>
</tr>
<tr>
<td>6</td>
<td>0.50</td>
<td>0.46</td>
<td>-8.0</td>
</tr>
<tr>
<td>7</td>
<td>0.61</td>
<td>0.62</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Figure 4.21.b shows $|B|$ from FEA along the gap between the piston and housing. The location of this measurement is indicated by the red line in Figure 4.21.a.

Figure 4.21 a. Cross-section of the spool and housing showing locations sampled for FEA results in Table 4.6. b. FEA predicted $|B|$ in the MRF, between piston and cylinder.
The hardware shown in Figure 4.15 was tested using the setup of Figure 4.16. Displacement and force data are shown in Figure 4.22 for a range of current commands. The joystick was cycled manually between its limits at approximately 1 Hz for about 200 sec. This means the average velocity of the piston was approximately 0.2 m/sec. Coil current was increased in 0.05 A increments from 0 to 0.5 A and applied intermittently. Figure 4.22.a shows the time trace of force measured and the same data is plotted relative to displacement in Figure 4.22.b. Color of data points in these traces corresponds to the current command as indicated by the color bar. The force measured at 0.5 A has a root-mean-square (rms) value of 19.6 lbf with rms velocity, 0.21 m/sec. The force predicted at 0.5 A and 0.2 m/sec is 20.2 lbf ($F_b + F_n$).

Figure 4.22  Cyclic joystick motion with current commands from 0 to 0.5 A in 10 steps: a. force vs. time and  b. force vs. displacement.  The MRFD was exercised manually at roughly one cycle per second throughout the test. Figure A. 8 shows a 6 second segment of this data.

Force and displacement time traces are shown in Figure 4.23 for a step change in current command: 0 to 0.5 A. Force increased to approximately 66% of its maximum level within 100 msec. A similar result can be seen for force reduction when current switches back to 0 A. The predicted time constant is 34 msec. Velocity was not consistently maintained in the data of Figure 4.23.
The MRFD was exercised manually. Color in the force plot corresponds to current command. Blue is zero, red is 0.5 A current command.

### 4.4 Discussion/Conclusions

An optimal MRFD has been designed, built, and tested using the methods of section 4.2. Results are presented in section 4.3 for each of the stages of design and validation. Table 4.4 shows the lower and upper limits and resulting designs.

Magnetic circuit analysis was used as part of a lumped parameter modeling approach to optimize the design. The lumped parameter and FEA models differed by not more than 0.06 T, or 9.2% in predicting $|\mathbf{B}|$, as shown in Table 4.6. This was deemed acceptable agreement to justify using the lumped parameter model for optimization. The benefit was a roughly 1400 fold improvement over FEA in processing time.

In addition to providing validation for the lumped parameter model, FEA was helpful for evaluation of the MRFD design. Results for the optimal design produced fairly uniform
gradient contours throughout ferromagnetic material regions of Figure 4.20, indicating that there are not dominant areas of magnetic reluctance in the design.

The CSD method was used as part of an automated firefly optimization procedure to optimize the MRFD. The firefly champions were used as initial designs for a final conjugate gradient optimization step. Figure 4.18 shows that the champions in this case happen to be also the lowest cost set of results. This is atypical - often, as in Figure 2.3, there is a cluster of lowest cost results following $x_0$. This may indicate, for example, that $x_0$ is not as robust as $x_1$ for the MRFD design.

The cost function is heavily weighted for driving to high max force and design $x_0$ is best in this regard, but design $x_1$ has 96% of its max force with 66% of its time constant. It is not obvious that one of these is a better choice; slightly different cost function weighting would result in reversal of their ranking. This is one example of the rationale for developing a method to explore local minima rather than identifying only the lowest cost result. Also, it is clear from Table 4.5 that variables $x_5$ and $x_6$ might be specified rather than optimized since for all five of the best designs they are near the same constraint limits. Removing these from the optimization variables would reduce the degrees of freedom and number of constraints and might improve convergence for future optimization.

The data Figure 4.22 and Figure 4.23 show good agreement with analytical force prediction, but are not conclusive for comparing transient capability. The rms force value at 0.5 A was within 0.6 lbf (3 %) of the predicted value. The measured forces of Figure 4.22 are generally larger in magnitude for negative displacement than for positive and are higher for motion from positive to negative displacement than for negative to positive displacement. These variations may be affected by non-constant actuation velocity, imperfect alignment of the assembled parts, or bore diameter inconsistency. It is not possible to accurately compare predicted and measured time constants using the data of Figure 4.23 because the piston velocity was not consistently maintained during the transition. At face value, the time constant is roughly 3 times the value predicted. Automated piston actuation might enable a better conclusion regarding the transient prediction.
Chapter 5

Conclusions

The overall goal of this research has been to develop and demonstrate methods for optimal engineering design, and ultimately to use those methods to design a novel magnetorheological fluid device (MRFD) suitable for haptic feedback. The following research objectives have been addressed:

1. An improved population-based firefly algorithm (FA) for optimization was developed and demonstrated capable of solving multimodal engineering design problems. Chapter 2 described improvements of this new FA over versions described previously in the literature.

2. The cost-sorted distance (CSD) method for visualizing and exploring the design space was introduced in Chapter 2. The designer can use the CSD method to explore the results of optimization and to identify the most attractive design alternatives found.

3. A methodology was presented for tuning the FA to reliably generate solutions with clustered populations at multiple minima for multimodal problems.

4. FA and CSD capability was demonstrated in Chapter 2 using 1D-4D benchmark problems and a realistic 6D electromechanical system design.

5. Chapter 3 considered whether FA and CSD, as implemented in Chapter 2, provide indications of design robustness to uncertainty. The percentage of fireflies (PFF) clustered at multiple minima was shown to be correlated with robustness measures. The correlation was demonstrable, but not strong (i.e. LSE lines fit through the data have approximately 90% probability of positive slope, but correlation coefficients for those fits were only 0.7, on average).
6. A novel MRFD concept was described and contextualized in Chapter 4. Parameterized models were developed and optimized using FA and CSD. The optimal design was validated through simulation and testing.

5.1 Future Research Opportunities

The FA and CSD methods documented in this dissertation have proved useful for designing optimal real-world electromechanical systems. There are many possible avenues for continued research in this area. One opportunity is to quantitatively compare these methods to other tools with similar capability. In Chapter 3, FA and CSD were shown to indicate, to some extent, robustness to uncertainty rather than just deterministic fitness; however, the methods were not explicitly developed to accomplish this. A stronger correlation between PFF and robustness measures might result from purposefully including robustness indication as a goal in the algorithm design.

Even with the measurement uncertainty caused by inconsistent velocity control during testing, the observed MRFD transient response seems somewhat slower than predicted. This may well have been caused by the current driver used in the experiment. Additional tests are needed to determine if the transient response is accurately predicted by the model; if not, then this is a deficiency that must be addressed in future research. Interesting avenues of continued research regarding elastomeric baffle MRFDs include: finding a range of optimal designs with different force capability; development of a robust version of the device, considering production limitations and other sources of uncertainty; determining if this device type has a niche of superiority to other moderate force, translational devices.
REFERENCES


Diameters and Cross-Sectional Areas of AWG Sizes of Solid Round Wires Used as


[53] C. M. Elliott, G. Buckner and S. Ferguson, "A Modified Firefly Implementation and


[57] B. Mettler, T. Kanade and M. Tischler, "System Identification of a Model-Scale


Appendices
Appendix A: Active Magnetic Bearing Model

Following is the detailed development of the AMB (Figure 2.11) transient equations. Considering only translation in the $y$ direction and applying Newton’s second law of motion,

$$ F(t) = m_s \ddot{y}(t) \quad (A.1) $$

where $m_s = A \rho L_s$ is the shaft mass, $A = \frac{\pi}{4}D_s^2$ is the shaft area, and $\rho$ is the shaft density. Three vertical forces act on the shaft:

$$ F(t) = m_s g + F_B(t) + F_d(t) \quad (A.2) $$

where $g$ is the gravitational acceleration, $F_B(t)$ is the electromagnetic force, and $F_d(t)$ is a disturbance force.

An expression for the magnetic force may be derived using the energy method [55] where the magnetic field is assumed to be lossless as indicated in Figure A.1.

![Figure A.1 Electromechanical energy conversion is modeled with a lossless magnetic field coupling electrical and mechanical subsystems.](image)

A power balance on the magnetic field may be written

$$ \frac{dE}{dt} = P_e + P_m \quad (A.3) $$

where $E$ is energy stored, $P_e = i \nu_L$ is the electrical power input, $i$ is the coil current, $\nu_L = \frac{d\lambda}{dt}$ is the magnetic field induced voltage, $\lambda$ is the flux linkage, and $P_m = F_B \frac{dy}{dt}$ is the mechanical power input. Equation (A.3) can be rewritten
\[
\frac{dE}{dt} = i \frac{d\lambda}{dt} + F_B \frac{dy}{dt} . \quad (A.4)
\]

Eliminating \(dt\) from both sides of (A.4) yields
\[
dE = i \, d\lambda + F_B \, dy . \quad (A.5)
\]

An alternate expression for \(\frac{de}{dt}\) results from assuming that \(E = E(\lambda, y)\) is a state equation and taking its total derivative:
\[
\frac{dE}{dt} = \frac{\partial E}{\partial \lambda} \frac{d\lambda}{dt} + \frac{\partial E}{\partial y} \frac{dy}{dt} . \quad (A.6)
\]

A comparison of (A.4) and (A.6) shows that
\[
\frac{\partial E}{\partial \lambda} \bigg|_y = i \quad (A.7)
\]

and
\[
\frac{\partial E}{\partial y} \bigg|_\lambda = F_B . \quad (A.8)
\]

The field is assumed a conservative function of states \(\lambda\) and \(y\) only; therefore, integration to find \(E\) may be carried out along the most convenient path from \((0, 0)\) to \((\lambda, y)\). Referring to Figure A.2, integrating (A.5) along the bold, dashed path yields an expression for \(E\):
\[
E = \int i \, d\lambda . \quad (A.9)
\]

If the device’s operation is restricted to the range where current is proportional to flux linkage, \(i = \frac{\lambda}{L(y)}\) where \(L(y)\) is the inductance, then
\[
E = \frac{\lambda^2}{2 \, L(y)} = \frac{i^2}{2} \, L(y) . \quad (A.10)
\]
Figure A. 2 Magnetic field energy is assumed a function of states $\lambda$ and $y$.

Substituting (A.10) into (A.8),

$$ F_B = \frac{i^2}{2} \frac{d}{dy} L(y). $$

(A.11)

The inductance can also be expressed in terms of magnetic flux,

$$ L(y) = \frac{N_c}{l} \phi(y) $$

(A.12)

where $N_c$ is the number of coil loops and $\phi$ is the magnetic flux. Substituting this into (A.11) yields a useful relationship for magnetic force:

$$ F_B = \frac{N_c}{2} \frac{d\phi}{dy}. $$

(A.13)

The magnetic flux $\phi$ in (A.13) can be estimated using Hopkinson’s law (the basis of magnetic circuit analysis):

$$ F = R \phi $$

(A.14)

where $F = N_c i$ is magneto-motive force, $R = \frac{l_v}{\mu A} + \frac{2y}{\mu_0 A}$ is the magnetic reluctance, $L_v$ is the length of steel flux path, $2y$ is the length of the air gaps, $\mu_0$ is the magnetic permeability of the
gap, \( \mu = \mu_r \mu_0 \) is the steel’s magnetic permeability, \( \mu_r \) is its relative permeability, and \( A \) is the cross-sectional area of the magnetic flux path (uniform for this design).

Substituting and solving (A.14) for magnetic flux yields:

\[
\phi = \frac{\mu_0 A N_c i}{\frac{L_s}{\mu_r} + 2 y} \quad (A.15)
\]

Considering (A.15), the electromagnetic force (A.13) can be expressed in terms of parameters and states:

\[
F_B = -\frac{\mu_0 A N_c^2 i^2}{\left(\frac{L_s}{\mu_r} + 2 y\right)^2} \quad (A.16)
\]

A differential equation describing the shaft motion may be written, based on (A.1), (A.2), (A.16):

\[
y(t) = \ddot{y} + \frac{F_d(t)}{m_s} - \frac{\mu_0 A N_c^2 i^2}{m_s \left(\frac{L_s}{\mu_r} + 2 y\right)^2} \quad (A.17)
\]

An electrical subsystem dynamic equation may be developed using Kirchoff’s voltage law. The lumped-parameter coil model is shown in Figure A. 3. The sum of voltages around this loop is

\[
v = v_R + v_L \quad (A.18)
\]
where \( v \) is the control voltage, \( v_r = iR \) is the voltage due to coil resistance, and \( v_L = \frac{dI}{dt} = \frac{d}{dt} (iL) = \frac{d}{dt} (N_c \phi) \) is the voltage due to the coil inductance. Substituting (A.15) and simplifying the total derivative of \( \lambda(i,y) = N_c \phi(i,y) \), yields

\[
v_L = \frac{\mu_0 A N_c^2}{\mu_r + 2y} \frac{di}{dt} - \frac{2 \mu_0 A N_c^2}{\left(\frac{L_s}{\mu_r} + 2y\right)^2} i \frac{dy}{dt} . \tag{A.19}
\]

Substituting (A.19) into (A.18) and rearranging yields:

\[
\frac{di}{dt} = \frac{\frac{L_s}{\mu_r} + 2y}{\mu_0 A N_c^2} (v - iR) + \frac{2}{\frac{L_s}{\mu_r} + 2y} i \frac{dy}{dt} \tag{A.20}
\]

which is the differential equation governing the electrical circuit dynamics.

It remains only to develop parametric expressions for \( N_c \) and \( R \). The number of coil turns is approximately

\[
N_c = f_c \frac{L_c}{D_w} \frac{D_c - D_s}{2} \tag{A.21}
\]

where \( f_c \) is an empirical filling factor for the coil (\( f_c = 0.75 \) is realistic for a hand-wound coil), and the magnet wire diameter [51] is

\[
D_w = 92^{-36-g_c} 0.005 \text{ in}, \tag{A.22}
\]

where \( g_c \) is the coil gauge. The length of conductor necessary to construct the coil is approximately

\[
\ell_w = 2\pi \frac{D_c + D_s}{4} N_c . \tag{A.23}
\]

Coil resistance is therefore

\[
R = \rho_c \frac{\ell_w}{\frac{\pi}{4} D_w^2} \tag{A.24}
\]

where \( \rho_c = 1.68 \times 10^{-8} \) \( \Omega \) m is the electrical resistivity of the magnet wire.
Nonlinear differential equations (A.17) and A.20) comprise the dynamic system and may be numerically integrated to approximate the AMB’s transient behavior. Integration by numerical solver (e.g. Adams-Bashforth [56]) is facilitated using the first-order system of equations:

\[
\frac{d}{dt} \begin{pmatrix} \xi \\ \xi \end{pmatrix} = \begin{pmatrix} \xi \\ \frac{\xi}{m_s} - \frac{\mu_0 A N_c}{m_s} \frac{N_c^2 i^2}{(m_s \left( \frac{L_s}{\mu_r} + 2 \gamma \right))^2} \end{pmatrix}, \tag{A.25}
\]

where \( \xi = \frac{dy}{dt} \).
Appendix B: Parametric Length and Area Expressions

Lengths and areas used to compute reluctance of the magnetic circuit, for cases a and c of Figure 4.8 are shown in Table A.1 and Table A.2, respectively.

Table A.1  Parameters of (4.2), corresponding to Figure 4.8.a.

<table>
<thead>
<tr>
<th>( k )</th>
<th>( \ell_k )</th>
<th>( A_k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( \frac{1}{2} \left( \frac{\zeta_0}{2} + \zeta_2 - \zeta_1 \right)^2 + \left( \frac{r_3 - r_0}{2} \right)^2 )</td>
<td>( \pi (r_1^2 - r_0^2) )</td>
</tr>
<tr>
<td>1</td>
<td>( \zeta_1 - \zeta_2 )</td>
<td>( \pi (r_1^2 - r_3^2) )</td>
</tr>
<tr>
<td>2</td>
<td>( \frac{\zeta_3}{2} )</td>
<td>( \frac{\pi}{2} \left( r_1^2 - r_3^2 + (r_1 + r_3) \sqrt{\zeta_2^2 + (r_1 - r_3)^2} \right) )</td>
</tr>
<tr>
<td>3</td>
<td>( \frac{r_1 - r_3}{2} )</td>
<td>( \frac{\pi}{2} \left( (r_1 + r_3) \sqrt{\zeta_2^2 + (r_1 - r_3)^2 + 2 r_1 \zeta_2} \right) )</td>
</tr>
<tr>
<td>4</td>
<td>( \frac{r_4 - r_1}{2} )</td>
<td>( 2 \frac{\pi}{\ln \left( \frac{r_4}{r_1} \right)} (r_4 - r_1) \zeta_2 )</td>
</tr>
<tr>
<td>5</td>
<td>( r_5 - r_4 )</td>
<td>( 2 \frac{\pi}{\ln \left( \frac{r_5}{r_4} \right)} (r_5 - r_4) \zeta_2 )</td>
</tr>
<tr>
<td>6</td>
<td>( \frac{\zeta_0 + \zeta_2}{2} )</td>
<td>( \pi (r_6^2 - r_5^2) )</td>
</tr>
</tbody>
</table>
Table A.2 Parameters of (4.2), corresponding to Figure 4.8.c.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$\ell_k$</th>
<th>$A_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\frac{\zeta_0}{2}$</td>
<td>$\pi \left( r_1^2 - r_0^2 \right)$</td>
</tr>
<tr>
<td>1</td>
<td>$\frac{\zeta_2 - \zeta_1}{2}$</td>
<td>$\frac{\pi}{2} \left( r_1^2 - r_0^2 + (r_1 + r_0) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_0)^2} \right)$</td>
</tr>
<tr>
<td>2</td>
<td>$\frac{r_1 - r_0}{2}$</td>
<td>$\frac{\pi}{2} \left( (r_1 + r_0) \sqrt{(\zeta_2 - \zeta_1)^2 + (r_1 - r_0)^2} + 2 r_1 (\zeta_2 - \zeta_1) \right)$</td>
</tr>
<tr>
<td>3</td>
<td>$r_3 - r_1$</td>
<td>$2 \pi \left( r_3 - r_1 \right) \left( \zeta_2 - \zeta_1 \right) / \ln \left( \frac{r_3}{r_1} \right)$</td>
</tr>
<tr>
<td>4</td>
<td>$r_4 - r_3$</td>
<td>$2 \pi \left( r_4 - r_3 \right) \zeta_2 / \ln \left( \frac{r_4}{r_3} \right)$</td>
</tr>
<tr>
<td>5</td>
<td>$r_5 - r_4$</td>
<td>$2 \pi \left( r_5 - r_4 \right) \zeta_2 / \ln \left( \frac{r_5}{r_4} \right)$</td>
</tr>
<tr>
<td>6</td>
<td>$\frac{\zeta_0 + \zeta_2}{2}$</td>
<td>$\pi \left( r_6^2 - r_5^2 \right)$</td>
</tr>
</tbody>
</table>
Appendix C: Experiments to Measure B vs H

Figure A. 4 shows a research assistant recording data to characterize the pertinent magnetic characteristics of a ferrous material sample.

Figure A. 4  A research assistant records data to characterize B vs. H for a ferrous material.

The procedure was to measure magnetic flux density (using the 912 Gaussmeter) in the sample gap (Figure A. 5) for a series of DC current commands (generated using the Agilent E3634A). Coils were 1000 turns of 28 gauge magnet wire.

Figure A. 5  Schematic for measurement of sample B-H curve.
Hopkinson’s law can be used to estimate a material’s magnetic permeability $\mu_{fe}$ as follows. Substituting $F = N_c i$, $R = \frac{L_{fe}}{\mu_{fe} A} + \frac{L_{gap}}{\mu_0 A}$, and $\phi = BA$ into (A.14), assuming zero flux leakage, and simplifying yields an expression for the sample’s magnetic permeability:

$$\mu_{fe} = \frac{L_{fe}}{N_c i} \frac{L_{gap}}{B}$$

(A.26)

where $L_{gap} = 2mm$, $L_{fe} = 194.35\ mm$, $N_c = 1000$, $\mu_0 = 4 \pi 10^{-7} N/A^2$, and $i$ and $B$ are measured. The magnetic permeability vs. magnetic flux curve of Figure 4.11.a was measured in this manner. The B-H curve (Figure 4.11.b) results from the same data assuming $H \equiv \frac{B}{\mu_{fe}}$.

The sample geometry is shown in Figure A. 6.

![Figure A. 6 Material sample dimensions (mm).]
Appendix D: MRFD Performance Experiments

Figure A. 7 shows a research assistant actuating the MRFD to measure performance characteristics of the device.

The MRFD was manually displaced using the joystick arrangement shown in Figure 4.16. Data was streamed at 100Hz from the control module to a PC and post processed using a Matlab script to produce the data shown in Figure 4.22 and Figure 4.23. Six seconds of transient data at 0.5A coil current is shown in Figure A. 8.
Figure A. 8  Six seconds of typical MRFD force and displacement data.  Displacement (the first row) is estimated from the joystick rotary position.  Velocity (the third plot) is estimated from displacement. This is part of the data shown in Figure 4.22.  Coil current is 0.5 A.
Appendix E: MRFD Prototype Parts

Figure A. 9 Assembled MRFD parts.
Figure A. 10 The ferrous cylinder built to the optimal geometry specified in Table 4.4.
Figure A. 11  The optimal ferrous spool with geometry specified in Table 4.4.
Figure A. 12  Two of these aluminum shafts are used to suspend the spool (Figure A. 11) within the ferrous housing (Figure A. 10). One end of the shaft screws into the spool and the other end slides within the non-ferrous cap. A nut secures the elastomeric baffle spacer (Figure A. 14) between this shaft and the spool.
Figure A. 13  These non-ferrous caps are attached to either end of the ferrous housing (Figure A. 10). One end of each elastomeric baffle is secured between these and o-rings seated in ferrous housing.
Figure A. 14 A spacer for the rolling elastomeric baffle. These are attached by nuts between the shafts of Figure A. 12 and the spool (Figure A. 11). The elastomeric baffle rolls on these. Also, the elastomeric baffles are secured to the spool between these and o-rings seated in the spool.
Figure A. 15 A “swabable” check valve used for injecting/removing MRF by syringe.

Figure A. 16 Stainless steel adapter compatible with the check valve shown in Figure A. 15. Two of these are attached to opposite sides of the cylinder of Figure A. 10. They are used for injecting/removing MRF from the device.
Figure A. 17 This spherical bearing secured the MRFD to the base block as shown in Figure A. 9.
Figure A. 18  A force sensor was threaded between a rod end and the shaft shown in Figure A. 12.
Figure A. 19 This signal conditioner was used to convert sensor force measurements for the ECM.