ABSTRACT

Title of Dissertation: ROBUST OPTIMIZATION AND SENSITIVITY ANALYSIS WITH MULTI-OBJECTIVE GENETIC ALGORITHMS: SINGLE- AND MULTI-DISCIPLINARY APPLICATIONS

Mian Li, Doctor of Philosophy, 2007

Dissertation directed by: Shapour Azarm, Professor
Department of Mechanical Engineering

Uncertainty is inevitable in engineering design optimization and can significantly degrade the performance of an optimized design solution and/or even change feasibility by making a feasible solution infeasible. The problem with uncertainty can be exacerbated in multi-disciplinary optimization whereby the models for several disciplines are coupled and the propagation of uncertainty has to be accounted for within and across disciplines. It is important to determine which ranges of parameter uncertainty are most important or how to best allocate investments to partially or fully reduce uncertainty under a limited budget. To address these issues, this dissertation concentrates on a new robust optimization approach and a new sensitivity analysis approach for multi-objective and multi-disciplinary design optimization problems that have parameters with interval uncertainty.

The dissertation presents models and approaches under four research thrusts. In the first thrust, an approach is presented to obtain robustly optimal solutions which are as best as possible, in a multi-objective sense, and at the same time their sensitivity of objective and/or constraint functions is within an acceptable range. In the second thrust, the robust optimization approach in the first thrust is extended to design optimization

problems which are decomposed into multiple subproblems, each with multiple objectives and constraints. In the third thrust, a new approach for multi-objective sensitivity analysis and uncertainty reduction is presented. And in the final research thrust, a metamodel embedded Multi-Objective Genetic Algorithm (MOGA) for solution of design optimization problems is presented.

Numerous numerical and engineering examples are used to explore and demonstrate the applicability and performance of the robust optimization, sensitivity analysis and MOGA techniques developed in this dissertation. It is shown that the obtained robust optimal solutions for the test examples are conservative compared to their corresponding optimal solutions in the deterministic case. For the sensitivity analysis, it is demonstrated that the proposed method identifies parameters whose uncertainty reduction or elimination produces the largest payoffs for any given *investment*. Finally, it is shown that the new MOGA requires a significantly fewer number of simulation calls, when used to solve multi-objective design optimization problems, compared to previously developed MOGA methods while obtaining comparable solutions.

ROBUST OPTIMIZATION AND SENSITIVITY ANALYSIS

WITH MULTI-OBJECTIVE GENETIC ALGORITHMS:

SINGLE- AND MULTI-DISCIPLINARY APPLICATIONS

By
Mian Li

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2007

Advisory Committee:

Professor	Shapour Azarm, Advisor/Chair
Associate Professor	Steven A. Gabriel
Associate Professor	Satyandra K. Gupta
Associate Professor	P. K. Kannan (Dean's representative)
Professor	Ali Mosleh

To My Dear Wife, Sophia

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Dr. Shapour Azarm, for his gracious encouragement, guidance and support throughout my graduate study at University of Maryland, College Park. He provided me a great opportunity on the study, excellent academia training, and marvelous communication practice.

I also would like to thank the committee members, Dr. Steven Gabriel, Dr. S. K. Gupta, Dr. P. K. Kannan, and Dr. Ali Mosleh for agreeing to serve on my committee and their priceless time, effort, comments, and suggestions for improving my thesis.

I would like to acknowledge the partial support from the U.S. Office of Naval Research for this research work.

The software, ATSV, developed by the Applied Research Laboratory (ARL) of Penn State University is used in the dissertation for plotting correlation figures in Chapter 5.

Mr. Paul Higgins is gratefully acknowledged for providing gracious assistance with editing of my dissertation.

I want to thank my colleagues and friends, past and present, from the Design Decision Support Laboratory: Dr. Subroto Gunawan, Dr. Kumar Maddulapalli, Dr. Babak Besharati, Dr. Ali Farhang-Mehr, and Mr. Vikrant Aute, who have provided much valuable inputs and suggestions. Special thank goes to Dr. Genzi Li and LT. Nathan Williams for their gracious comments and generous help during my study. I also want to thank my parents for their continuous support during my study. Finally, my grateful thanks are given to my wife, Sophia, whose love and understanding always help me carry on during difficult times in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

ACVR	Acceptable Constraint Variation Region
AOVR	Acceptable Objective Variation Region
CK-MOGA	Circled Kriging assisted Multi-Objective Genetic Algorithm
CO	Collaborative Optimization
CSR	Constraint Sensitivity Region
GA	Genetic Algorithm
genc	the generation counter
K-MOGA	Kriging assisted Multi-Objective Genetic Algorithm
MaxNumGeneration	Maximum number of the generations in the MOGA
MEP	Multi-objective Entropy Performance
MCO	Multi-objective Collaborative Optimization
McRO	Multi-objective collaborative Robust Optimization
MDO	Multi-disciplinary Design Optimization
MOGA	Multi-Objective Genetic Algorithm
MORO	Multi-Objective Robust Optimization
MOOP	Multi-Objective Optimization Problem
NumGeneration	Number of generation when stop criteria are satisfied
NumSimCall	Number of simulation calls used in optimizations
OSR	Objective Sensitivity Region
popsize	the population size in the MOGA
rep_{genc}	the number of individuals replaced as new offspring in the *genc*-th generation
RTR	Retained Tolerance Region
SA	Sensitivity Analysis
SS*i*	Subsystem *i*
WCCSR	Worst Case Constraint Sensitivity Region
WCOSR	Worst Case Objective Sensitivity Region

$\boldsymbol{f}$	design objectives
$\tilde{\boldsymbol{f}}$	design objective values when parameters vary
$\boldsymbol{g}$	inequality design constraints
J	number of inequality constraints
K	number of design parameters
M	number of design objectives
N	number of design variables
N_1	nominal design used for the verification of robustness
$\boldsymbol{p}$	design parameters
$\boldsymbol{p}_{sh}$	design parameters at the system level
$\boldsymbol{p}_0$	nominal (or most likely) value of design parameters

$\boldsymbol{p}_i$	local parameters in subsystem i
$\tilde{\boldsymbol{p}}$	possible parameter value over the tolerance region
R_1	robust design used for the verification of robustness
R_2	robust design used for the verification of robustness
R_f	radius of WCOSR
R_{g}	radius of WCCSR
R_u	radius of uncertainty from the approximation model
$\boldsymbol{t}$	target variables
$\boldsymbol{t}_{ij}$	corresponding target variables for coupling variables $\boldsymbol{y}_{ij}$
$\boldsymbol{t}_{ij,0}$	nominal value of target variables $\boldsymbol{t}_{ij}$
$\boldsymbol{x}$	design variables
$\boldsymbol{x}_0$	particular design trail
$\boldsymbol{y}_{ij}$	coupling variables, output from subsystem i and inputs to subsystem j
$\Delta\boldsymbol{f}$	variations in objectives
$\Delta\boldsymbol{f}_0$	maximum acceptable variations in the objectives
$\Delta f_{0,m}$	maximum acceptable variation in the m-th objective
$\Delta\boldsymbol{f}_{i,0}$	maximum acceptable variations of objectives in subsystem i
$\Delta f_{i,0,m}$	maximum acceptable variation of m-th objectives in subsystem i
$\Delta\boldsymbol{g}$	variations in constraints
$\Delta\boldsymbol{g}_0$	maximum acceptable increment in the constraints
$\Delta\boldsymbol{p}$	known range of parameter variations
$\Delta\boldsymbol{p}_0^-$	lower bound of known parameter variation
$\Delta\boldsymbol{p}_0^+$	upper bound of known parameter variation
η_0	desired level of robustness specified by designer
η_c	collaborative robustness index
η_f	multi-objective robustness index
η_g	feasibility robustness index
η_{perf}	performance robustness index

CHAPTER 1: INTRODUCTION

This dissertation presents a new robust optimization method and a new sensitivity analysis method for single- and multi-disciplinary optimization with uncertainty. Robust optimization and sensitivity analysis is usually used when an optimization model has parameters with uncontrollable variations due to uncertainty or noise. In this dissertation uncertainty in design variables or parameters is represented by intervals; probability distributions are not required.

This dissertation first presents a new robust optimization method for single-disciplinary multi-objective optimization problems, based on a forward mapping from the parameter space to the objective and/or constraint space. A worst-case distance calculated based on this forward mapping is used as a metric for the robustness. After that, the single-disciplinary robust optimization method is extended to handle the uncertainty and its propagation in multi-objective Multi-disciplinary Design Optimization (MDO) problems. In addition, a novel global Sensitivity Analysis (SA) and uncertainty reduction method is developed based on this worst-case distance measure for uncertain parameters whose uncertainty is reducible. Finally, the distance measure of uncertainty is further used as a metric for error prediction in a metamodeling assisted Multi-Objective Genetic Algorithm (MOGA) to significantly reduce the number of simulation calls during the optimization. Numerous numerical and engineering examples are used to demonstrate the merits and applicability of the proposed approaches.

1.1 MOTIVATION AND OBJECTIVE

Many engineering design optimization problems have multiple nonlinear objectives and constraints, mixed continuous-discrete design variables, and more critically,

parameters with uncontrollable variations. These problems and corresponding solution methods form the research area called "robust optimization." For optimization problems in this area, the objective functions of and/or the feasibility of an optimal design can be significantly degraded or changed due to uncertainty. There are essentially two different sources for uncertainty: 1) noisy input data (or noisy input factors), this includes noisy parameters in the problem, and 2) noisy control factors, these refer to design variables in the optimization problem whose optimizer-specified solution cannot be achieved exactly (or deterministically) in practice, e.g., geometrical dimensions cannot be made to their exact size due to manufacturing errors. The intent in robust multi-objective optimization is to obtain optimal design solutions which are as "best" as possible and at the same time variation in their objective and/or constraint functions due to noisy factors is still within an acceptable range. Many methods and approaches have been proposed in the literature to obtain robust design solutions. Depending upon whether the variations are considered in the objective or constraint functions, a robust optimization approach can be classified into two types [Parkinson et al., 1993]: "objective robustness" or "feasibility robustness", respectively. For objective robustness, the goal is to seek a design solution whose value of the optimal objective functions remains relatively the same regardless of the variability. Similarly, for feasibility robustness, the goal is to obtain a design solution which is feasible regardless of the variability. Since feasibility of a solution is often referred to as reliability, feasibility robust optimization is also called reliability optimization.

MDO is another important aspect of optimizing design of "complex" engineering systems [Sobieski-Sobieszczanksi, 1988], [Sobieszczanski-Sobieski and Balling, 1996],

[Sobieszczanski-Sobieski and Haftka, 1997]. By a complex system we mean a system design optimization problem that is decomposed into multiple interacting subsystems or disciplines. Examples of MDO problems are abundant and can be typically found in aircraft, spacecraft, automobiles, ships, and other engineering design applications. As a simple example, the bevel gear and the universal motor are two interconnected components of an angle grinder. They are considered as the two disciplines of the angle grinder (see Section 3.6.4 for the detail). However, even though there are many applications of MDO, the effective handling of uncertainties in MDO problems is still rare. The problem with handling variability is exacerbated in MDO whereby several disciplines and multi-objective optimization problems are coupled and as a result the complexity of design optimization problems increases. The situation becomes worse particularly because uncertainties may exist not only in each discipline but also in couplings across disciplines and hence methods for handling uncertainty within and across disciplines have become quite important [Du and Chen, 2000(a) and 2002], [Gu et al., 2000], [Gu and Renaud, 2002], [Gu et al., 2006]. Couplings in this dissertation refer to the variables (or outputs) generated by one discipline and used (as inputs) by other disciplines. For example, in the design of the angle grinder (see Section 3.6.4), the motor output such as power is used as an input by the bevel gear discipline. However, even though there are many reported applications for robust MDO approaches, those MDO methods are essentially for single-objective robust optimization problems that have continuous objective/constraint functions or when input probability distributions are known, which is not valid for complex system design during the conceptual design stage, due to inadequate information or insufficient samplings. The literature is still short in

handling uncertainty for fully coupled multiobjective decentralized MDO problems with interdisciplinary uncertainty propagation.

Sensitivity Analysis (SA) is a natural next step of robust optimization and has been investigated since it has been gaining more and more interest, especially for the applications where it is critical to identify the (reducible) uncertain parameters whose reduction or elimination of their interval will produce the largest payoffs in the performance. For instance, after performing a robust optimization and obtaining a set of optimal and robust solutions, a deeper analysis of the effects of individual parameters should be investigated to determine if any opportunities exist for further reduction in uncertainty given a variety of possible investment levels.

In the broadest sense there are two motivations for taking uncertainty into account in engineering design [Saltelli et al., 1999(a)]. Approaches whose goal is to find the range and frequency of possible model outcomes as a result of all model input uncertainty are broadly filed as Uncertainty Analysis. In contrast, those methods that seek to connect the uncertainty in model inputs to model outputs are classified as Sensitivity Analysis (SA) [Iman and Helton, 1988]. Sensitivity analysis can further be classified as either local or global in nature. Local sensitivity analysis methods examine the change in model outputs with respect to small variations in model inputs. These methods have the obvious drawback of being valid only for small regions of uncertainty. Global Sensitivity Analysis (GSA) takes into account the entire range of model inputs to determine the affect on overall model uncertainty. In GSA, the affect of parameters on model uncertainty are generally compared to overall model uncertainty in order to quantify parameter importance or sensitivity. Moreover, it is more common in the literature to

convert the multi-objective problem to a single objective one and then perform sensitivity analysis for a single solution point or design [Saltelli et al., 1999(b)]. On the whole, the approaches to multi-objective optimization problems have focused on sensitivity for a single solution based on a weighted objective or have been local in nature. Therefore, there is impetus to extend GSA to multi-objective design problems with respect to multiple designs.

To help solve the previously mentioned problems and obtain global optimal solutions, effective approaches to handle multi-objective optimization must also be addressed. Generally speaking, there are two classes of design optimization methods that can be used for engineering design optimization (see, e.g., [Papalambros and Wilde, 2000], [Arora, 2004], [Belegundu and Chandrupatla, 1999]). These two classes are gradient- and non-gradient-based methods. Gradient-based methods (see, e.g., [Bazaraa et al., 1993]) require derivative information for the optimization functions (i.e., objective and constraint functions) and usually have an implicit assumption that these functions are "smooth" and that design variables are continuous. In general, gradient-based methods can only guarantee obtaining a local optimal design solution unless the functions used in the problem have special properties such as convexity over convex feasible regions. Unfortunately, derivative information usually is not available because of uncertainty in engineering problems themselves during the conceptual design stage or due to the complexity of simulations used to evaluate designs. The smoothness assumption can be relaxed in non-gradient-based techniques. One popular and general class of non-gradient-based techniques for design optimization is Genetic Algorithms (GAs) [Holland, 1975], [Goldberg, 1989]. GAs were developed by John Holland [Holland, 1975] and are based

on evolutionary concepts. These algorithms can handle non-smooth functions and mixed continuous-discrete design variables, a situation common in engineering design problems. Generally, GAs can obtain or converge to a global (or near to) optimal design solution. Moreover, they can be easily extended to handle multiple design objectives, i.e., Multi-Objective Genetic Algorithms or MOGAs [Deb, 2001], [Coello Coello et al., 2002], [Narayanan and Azarm, 1999], [Kurapati et al., 2002]. One important advantage of MOGAs is that Pareto optimal solutions can be obtained by a single run of the GA. The main shortcoming of MOGAs is that they require a large number of simulation calls. A simulation call here means that the performance (objectives and/or constraints) of a design is calculated by a simulation. (Here, a simulation model refers to a set of functions or a computer program.) Researchers have been quite active in developing models and methods that improve the efficiency of GAs and MOGAs in terms of the number of simulation calls. Therefore, it will be extremely useful to develop a MOGA approach with significantly less computational effort, in terms of simulation calls, compared to conventional MOGAs.

The overall objective of this dissertation is to present a roust optimization and sensitivity analysis method for multi-objective and feasibility robust optimization to single- and multi-disciplinary design optimization problems. Also, an efficient metamodel assisted MOGA will be presented as part of the approach for these problems.

1.2 RESEARCH THRUSTS

To achieve the overall objective, we pursued four research thrusts for the research in this dissertation described as follows: (1) Multi-Objective and Feasibility Robust Optimization in Single-disciplinary Design Optimization (Chapter 3); (2) Performance

and Collaborative Robust in Multi-disciplinary Design Optimization (Chapter 4); (3) Interval Uncertainty Reduction and Multi-Objective Sensitivity Analysis in Single-disciplinary Design Optimization (Chapter 5); and (4) Metamodel Assisted Multi-Objective Genetic Algorithm (Chapter 6). The detailed motivation and objective of each research thrust are described in the following subsections.

1.2.1 Research Thrust 1: Multi-Objective and Feasibility Robust Optimization in Single-disciplinary Design Optimization

The first research thrust is concerned with variations in the objective and/or constraint functions of an optimal design due to uncertainty in the design variables or input parameters in a single-disciplinary design optimization, or called "all-at-once" formulation of MDO. A design is called "multi-objectively robust" if its variation in objective functions still remains within an acceptable range when parameters vary. A design is called "feasibly robust" if it is always feasible even if there are parameter variations. We extend the approach to "performance robust optimization," invoking objective and feasibility robustness together. This so-called "performance robustness" of an optimal design is especially important because its objective functions can degrade significantly or its feasibility can change (the feasible design may no longer be feasible) due to the variations in parameters.

The objective of Research Thrust 1 is to develop a method to obtain robust solutions which are as best as possible, in a multi-objective sense, and at the same time their variation in objective and/or constraint functions, due to uncontrollable parameter uncertainty is within an acceptable range.

A portion of this research thrust was presented in Li et al., [2006].

1.2.2 Research Thrust 2: Performance and Collaborative Robustness in Multi-disciplinary Design Optimization

The second research thrust extends the performance robustness for the MDO problems. MDO is concerned with methods for optimizing design of a system governed by multiple coupled disciplines. The existing robust optimization methods for this type of problems generally focus on continuous variations (i.e., continuous distributions) where distributions are presumed generally for single-objective optimization problems in each subsystem. Although the approaches proposed previously, i.e., [McAllister and Simpson, 2003] and [Kalsi et al., 2001], can handle robust design problems with multiple objective functions, they can only apply to robust MDO problems with no coupling or one-way coupling MDO problems (e.g., only upstream coupling parameters, from the follower to leader discipline, had uncontrollable variations). The literature is particularly limited in handling uncertainty for fully coupled multiobjective multilevel MDO problems with interdisciplinary uncertainty propagation.

The objective of Research Thrust 2 is to develop an approach that can find robust solutions for multi-objective MDO problems in which mixed continuous-discrete variation happens not only for parameters within disciplines but also across disciplines.

A portion of this research thrust was presented in Li and Azarm [2007].

1.2.3 Research Thrust 3: Interval Uncertainty Reduction and Multi-Objective Sensitivity Analysis in Single-disciplinary Design Optimization

The third research thrust is concerned with the determination of the effect of input uncertainty on the overall system's multiple responses using a GSA method. GSA methods can be classified as sampling based (Monte Carlo), analytical, or as interval

analysis. Sampling methods are the most prolific with variance being the principle measure of uncertainty. The greatest drawbacks of these methods are computational cost, the availability of probability distributions and treatment of tail probabilities. Moreover, iteratively "leave-one-out" strategy (fix one parameter at its mean value per time) used in analytical methods may not reflect a real situation in engineering design optimization in which some types of uncertainty cannot be eliminated entirely, such as manufacturing tolerance. Some degree or "grayscale" uncertainty reduction for one uncertainty or a combination of several uncertainties should become more attractive for improving the sensitivity of designs.

The objective of Research Thrust 3 is to develop a novel Multi-objective Sensitivity Analysis for multi-output (*multi-objective*) *problems and provide designers and program managers in multi-disciplinary design an environment that allocates investments for parameters whose uncertainty is reducible and should be reduced to achieve the acceptable variations in the model outputs.*

A portion of this research thrust was presented in Li et al., [2007c].

1.2.4 Research Thrust 4: Metamodel Assisted Multi-Objective Genetic Algorithm

Although GAs and MOGAs have been widely used in engineering design optimization, the important challenge still faced by researchers in using these methods is their high computational cost due to the population-based nature of these methods. In particular, a number of techniques incorporating metamodeling with GA based methods have been reported in the literature [Jin, 2005]. A metamodel means a simplified approximation of the original simulation model. Some of these methods use metamodeling in the GA's fitness estimation [Jin, 2005]. Others incorporate

metamodeling in the GA reproduction schemes to guide the search in the design space [Shan and Wang, 2005]. While the fitness estimation methods have been reported to reduce the computational cost significantly [Jin, 2005], these methods have the risk of generating false optimal solutions because of uncertainty (error) in the predicted objective and constraint value from the metamodels.

The objective of Research Thrust 4 is to develop an objective criterion to measure the uncertainty in the prediction of responses from the metamodels so that the risk of generating false optima can be reduced. The goal is to develop a MOGA that can converge to the Pareto front using significantly fewer number of simulation calls compared to a conventional MOGA.

A portion of this research thrust was presented in Li et al., [2007b].

1.3 ASSUMPTIONS

In developing our robust optimization and sensitivity analysis methods, we make the following assumptions:

- We assume that there exists a trade-off between objective functions of a design in the system or each subsystem. If such a trade-off does not exist, then it is not necessary to conduct multi-objective optimization.
- We assume that the range of parameter uncertainty is known as an interval (or several discrete intervals) *a priori*. Interval uncertainty is not required to be continuous.
- We presume an acceptable variation range for each objective function in the robust optimization.

- We assume, without considering uncertainty, the calculation of objective and constraint functions for a design is deterministic. That is, the same designs always have the identical outputs from the simulation model in the deterministic case.
- We assume that the parameter uncertainty considered in Chapter 5 is reducible.

There are also several properties about the simulation and optimization problems that should be noted in this dissertation, including:

- Simulations used in optimization problems are considered as "black boxes" that will provide the identical responses (outputs) when the same inputs are supplied.
- Design variables and/or parameters in optimization problems can be mixed continuous-discrete.

1.4 ORGANIZATION OF DISSERTATION

The rest of the dissertation is organized as follows. Chapter 2 gives the definitions of concepts and terminologies used throughout the dissertation. The proposed multi-objective and feasibility robust optimization approach for single-disciplinary design optimization is described in Chapter 3 (Research Thrust 1), and we extend it to multi-objective robust MDO problems in Chapter 4 (Research Thrust 2). In Chapter 5, we develop a new method for multi-objective sensitivity analysis and uncertainty reduction for single-disciplinary design optimization (Research Thrust 3). Chapter 6 presents an online metamodel assisted MOGA approach based on an objective criterion for the uncertainty in the predicted responses from metamodels. To demonstrate the applications of our methods, numerous numerical and engineering examples are given in Chapters 3 through 6. Chapter 7 concludes the dissertation with some remarks as well as a discussion on the contributions of the dissertation and potential future research directions.

Figure 1.1 shows the organization and flow of information in this dissertation.

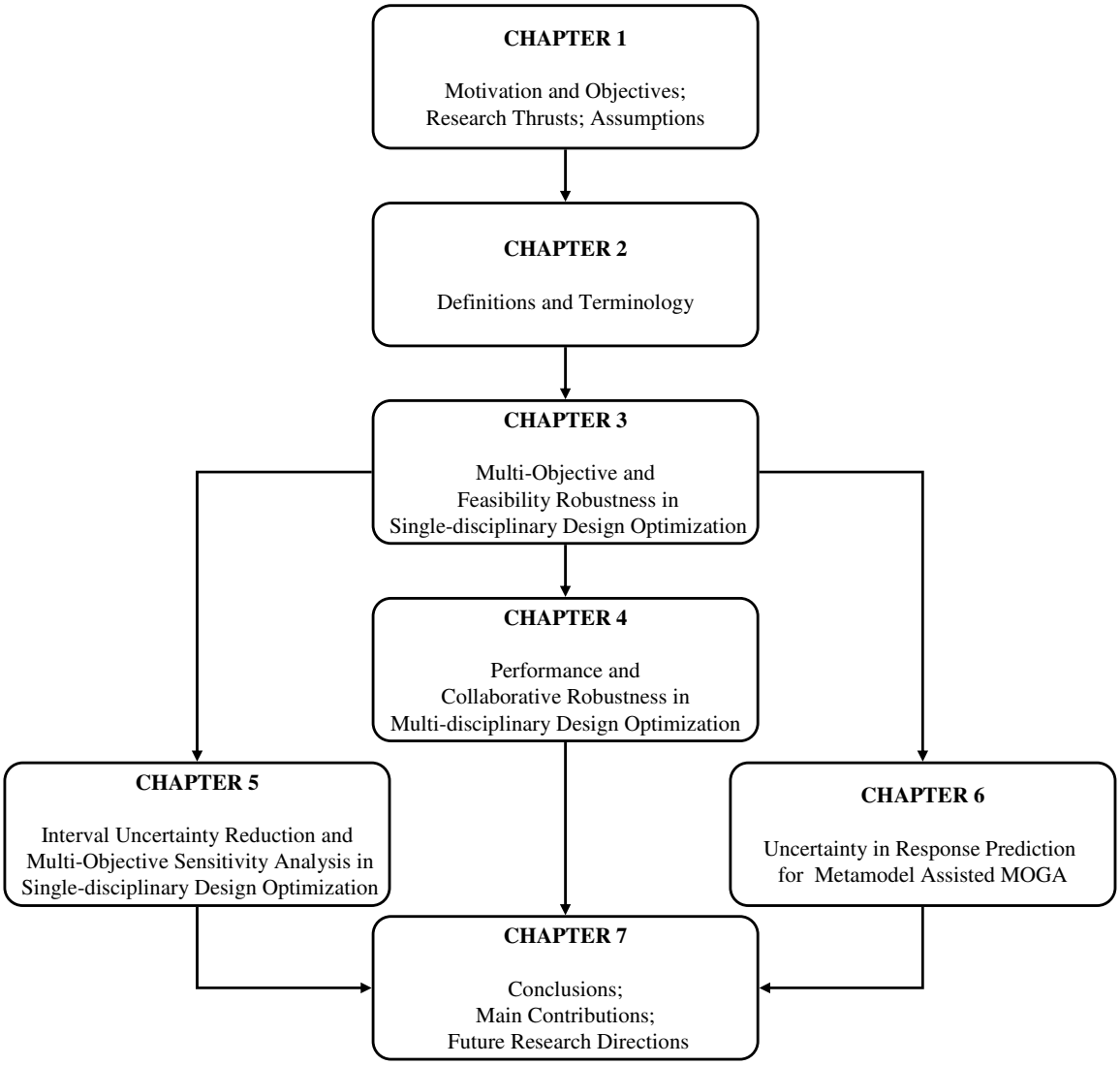


Figure 1.1 Organization of dissertation

CHAPTER 2: DEFINITIONS AND TERMINOLOGY

2.1 INTRODUCTION

In this chapter, we provide several definitions and terminologies that will be used in the dissertation. First in Section 2.2, we describe a typical formulation of multi-objective optimization problems with interval uncertainty. After that, an MDO formulation is introduced in Section 2.3. We briefly discuss several definitions used in the sensitivity analysis in Section 2.4. Following in Section 2.5 is a brief description of Multi-Objective Genetic Algorithm (MOGA) that is used as an optimizer in this dissertation. Finally in Section 2.6, we briefly discuss a typical metamodeling approach, called kriging, which is used in the metamodel assisted MOGA approach described in Chapter 6.

2.2 MULTI-OBJECTIVE OPTIMIZAITON WITH UNCERTAINTY

In this section, we set the basic optimization problem and explain several definitions and terminologies used in this dissertation. A general formulation of multi-objective optimization problems with uncertain parameters is given in Eq. (2.1).

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}) \qquad m = 1, \ldots, M \\ \text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}) \le 0 \quad l = 1, \ldots, L \\ & \boldsymbol{x}^{lower} \le \boldsymbol{x} \le \boldsymbol{x}^{upper} \end{aligned} \tag{2.1}$$

where $\boldsymbol{x} = (x_1,\ldots,x_N)$ is the N-element design variable vector; $\boldsymbol{x}^{lower}$ and $\boldsymbol{x}^{upper}$ are the lower and upper bounds of $\boldsymbol{x}$, respectively. The optimization is performed by changing the $\boldsymbol{x}$ components. $\boldsymbol{p} = (\boldsymbol{p}_1,\ldots,\boldsymbol{p}_K)$ is the vector of parameters, fixed for a particular optimization run but can have uncertainty. In this dissertation, uncertainty in $\boldsymbol{p}$ is represented by known intervals. The vector $\boldsymbol{x}$ or $\boldsymbol{p}$ or both might have continuous and discrete components. These sets may be binary restrictions, integer constraints, or just

specifications that indicate only a discrete set of choices available. Note that any design variable that has uncontrollable variation is included in both $\boldsymbol{x}$ and $\boldsymbol{p}$. As an example, a continuous noisy range for the design variables $\boldsymbol{x}$ may refer to an "implementation noise", relating to the inability of a manufacturer to achieve exact levels of design variables due to errors in a manufacturing setting. A discrete range for the parameters $\boldsymbol{p}$ refers, for example, to different applications or use scenarios. For instance, a consumer durable product like a grinder power tool could be used in different applications such as concrete or wood or metal and such applications or use conditions can vary and are not under the control of a product designer. $\{f_1,\ldots,f_M\}$ are the objective functions and are commonly represented as a vector $\boldsymbol{f} = (f_1,\ldots, f_M)$. The functions $g_1,\ldots,g_L$ are the constraints; a design that does not violate any of the constraints is called "feasible." Since the convexity of a feasible region is not assumed in this dissertation, an equality constraint can be transformed to two corresponding inequality constraints. Thus we presume that all the constraints can be represented as inequality functions.

Parameter space ($\boldsymbol{p}$-space): A K-dimensional space in which the coordinate axes are the parameter values.

Objective space ($\boldsymbol{f}$-space): An M-dimensional space in which the coordinate axes are the objective values.

Feasibility space ($\boldsymbol{g}$-space): An L-dimensional space in which the coordinate axes are the constraint values.

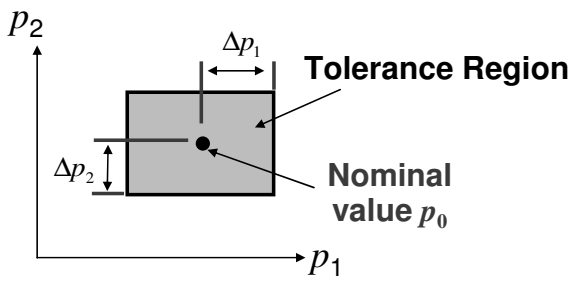


Figure 2.1 Tolerance region for design x_0 in parameter space

Tolerance Region: We consider problems where the uncertain parameters have a nominal (or most likely) value $\boldsymbol{p}_0 = (p_{0,1},\ldots,p_{0,K})$ and a known interval (or range). We describe the objective variation in terms of the parameter variations. The tolerance region is defined as a hyper-rectangle in $\boldsymbol{p}$-space, formed by $\tilde{\boldsymbol{p}}$, all possible parameter values (positive and negative directions of the nominal value). The parameters' tolerance region is defined as $\left\{\tilde{\boldsymbol{p}} \mid \boldsymbol{p}_0 - \Delta\boldsymbol{p}^- \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p}^+\right\}$, in which $\Delta\boldsymbol{p}^-$ and $\Delta\boldsymbol{p}^+$ are the lower and upper bounds of the known parameter variation range, respectively. Note that this region need not be symmetric about the origin, and it is not necessary to be constant for all designs. However, for simplicity, in this dissertation this region is considered symmetric about the nominal value, but not necessarily constant for all designs. Given the symmetry of the tolerance region, $\Delta\boldsymbol{p}^- = \Delta\boldsymbol{p}^+ = \Delta\boldsymbol{p}$ is used to define the tolerance region as shown in Figure 2.1. However, the tolerance region is not necessarily continuous. In that case, a discrete set of available choices of $\tilde{\boldsymbol{p}}$ should be indicated.

Pareto set and Pareto frontier: Since there are trade-offs among the M objective functions, the optimization problem Eq. (2.1) generally has more than one optimal solution. Those solutions are optimal in the Pareto sense, and the set of them is called the Pareto set: no design in the set is better, in all objectives, than any other design in the set.

The Pareto solutions lie on a boundary in objective function space, called the Pareto frontier. For further definitions and reviews of multi-objective optimization concepts and methods, see [Miettinen, 1999] and [Deb, 2001].

Dominance status: In the context of multi-objective optimization, a feasible design point is said to be "non-dominated" if no other feasible point under consideration (e.g., points in one generation in MOGA) is better than that point with respect to all objectives. The set of all non-dominated points under consideration forms a non-dominated set. The remaining points under consideration form a "dominated" set. The "dominance status" of a point determines whether a point is dominated or non-dominated. A two-objective example of the dominance status is shown in Figure 2.2.

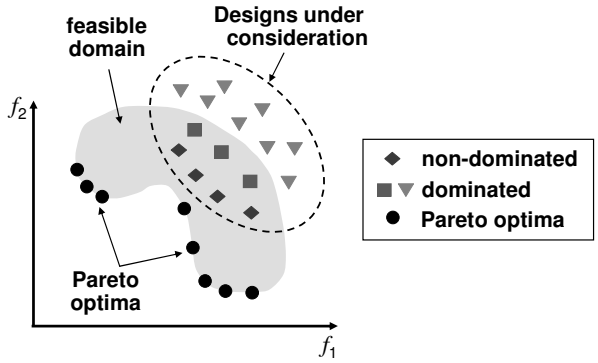


Figure 2.2 Dominance status in a two-objective case

At the convergence of an optimization procedure (such as MOGA), the set of non-dominated points eventually evolves to form the Pareto set (or an estimate of it).

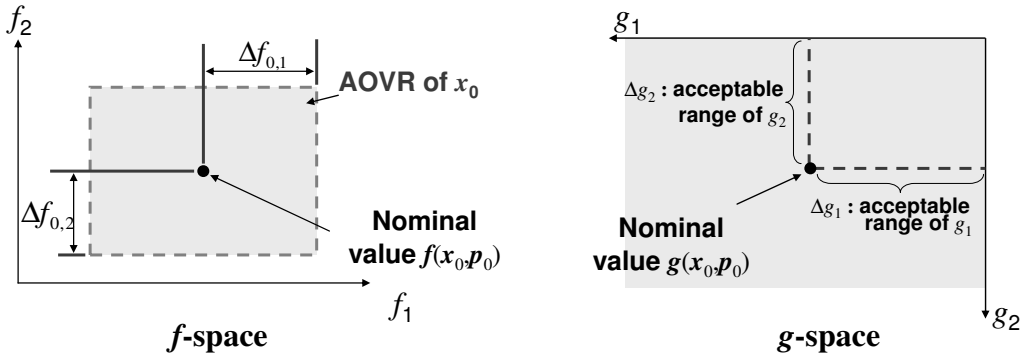
Nominal Pareto set: This refers to the Pareto set of a multi-objective optimization problem with nominal parameters. That is, nominal Pareto solutions are the solutions of the optimization problem in Eq. (2.1) when $\boldsymbol{p} = \boldsymbol{p}_0$.

Robust Pareto set: This is a set whose elements are both robust and Pareto optimal. This set refers to a set of designs that is Pareto optimal, with the additional properties that,

with $\boldsymbol{p}$ varying over a tolerance region, the values of the objective functions for each design remain within an acceptable range, and the designs remain feasible.

For a particular design $\boldsymbol{x}_0 = (x_{0,1},\ldots,x_{0,N})$ the nominal values of the objective functions are $\boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{p}_0) = (f_1(\boldsymbol{x}_0, \boldsymbol{p}_0),\ldots,f_M(\boldsymbol{x}_0, \boldsymbol{p}_0))$, and the nominal values of the constraint functions are $\boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{p}_0) = (g_1(\boldsymbol{x}_0, \boldsymbol{p}_0),\ldots, g_L(\boldsymbol{x}_0, \boldsymbol{p}_0))$. We will speak of objective function variations and constraint function variations of $\boldsymbol{x}_0$ caused by parameter variations $\tilde{\boldsymbol{p}}$: $\Delta\boldsymbol{f}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) = \boldsymbol{f}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{p}_0)$ and $\Delta\boldsymbol{g}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) = \boldsymbol{g}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - g(\boldsymbol{x}_0, \boldsymbol{p}_0)$.

We can also define three variation spaces. $\Delta\boldsymbol{p}$-space is the K-dimensional parameter variation space, whose axes are parallel to the axes of $\boldsymbol{p}$-space and whose origin is at $\boldsymbol{p}_0$ in $\boldsymbol{p}$-space. Similarly, we define $\Delta\boldsymbol{f}$-space, the M-dimensional objective variation space whose origin in $\boldsymbol{f}$-space is at $\boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{p}_0)$; and $\Delta\boldsymbol{g}$-space, the L-dimensional feasibility variation space whose origin is at $\boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{p}_0)$ in $\boldsymbol{g}$-space.



(a) (b)

Figure 2.3 (a) An AOVR in f-space and (b) an ACVR in g-space for design x_0

<u>*Acceptable Objective Variation Region*</u>: The Decision Maker (DM) specifies the maximum acceptable objective variation $\Delta\boldsymbol{f}_0 = (\Delta f_{0,1},\ldots, \Delta f_{0,M})$, which determines the Acceptable Objective Variation Region (AOVR) in $\Delta\boldsymbol{f}$-space: the hyper-rectangle centered at the origin, with dimensions $\pm\Delta f_{0,m}$, $m = 1,\ldots,M$, as shown in Figure 2.3(a).

Here we use the subscript "0" to represent the acceptable $\Delta \boldsymbol{f}$ range, different from the variation $\Delta \boldsymbol{f}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}})$. For simplicity, in the dissertation we assume that AOVR is a connected symmetric hyper-rectangle.

Acceptable Constraint Variation Region: If design $\boldsymbol{x}_0$ is feasible for the nominal parameter values, it will remain feasible with parameter variation $\tilde{\boldsymbol{p}}$ as long as the constraint function $\boldsymbol{g}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \leq 0$. (Any design that is infeasible even for the nominal values of $\boldsymbol{p}_0$ cannot be robust, and is not considered.) Hence, each nominally feasible design automatically gives an Acceptable Constraint Variation Region (ACVR): the hyper-rectangle in $\Delta \boldsymbol{g}$-space for which $0 \leq \Delta g_{0,l} \leq \left| g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) \right|$, l=1,…,L, based on the design $\boldsymbol{x}_0$ and nominal value of parameters $\boldsymbol{p}_0$, as shown in Figure 2.3(b). In other words, ACVR is determined by the values of the design's constraint functions, $g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)$, l=1,…, L; different nominally feasible designs have different ACVRs.

Distance metrics: For any two points $\boldsymbol{x}$ and $\boldsymbol{y}$ in N-dimensional space ($N \geq 2$), the distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ can be defined in three commonly used distance metrics, $\|\cdot\|_q$, q =1, 2, or ∞, as shown in Eq. (2.2):

$$\left\| \boldsymbol{x} - \boldsymbol{y} \right\|_q = \begin{cases} \sum_{n=1}^{N} \left| x_n - y_n \right|, \; q = 1 \\ \left(\sum_{n=1}^{N} \left| x_n - y_n \right|^2 \right)^{\frac{1}{2}}, \; q = 2 \\ \max(\left| x_n - y_n \right|, n = 1,..., N), \quad q = \infty \end{cases} \tag{2.2}$$

The illustration of three distance metrics is shown in Figure 2.4, as in a two-dimensional case.

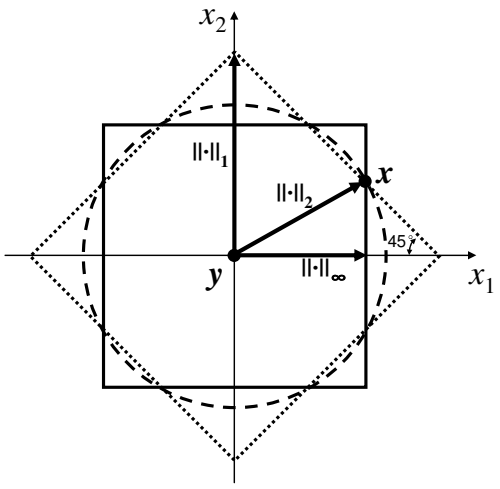


Figure 2.4 Distance metrics in a two-dimensional case

2.3 MULTI-OBJECTIVE MDO

MDO refers to a class of optimization methods that are used to solve a system optimization problem that consists of multiple coupled subsystems. For simplicity and without loss of generality, we consider a fully coupled two-subsystem optimization problem representing two coupled disciplines i, i=1, 2, as shown in Figure 2.5. The optimization formulation for the two disciplines, as shown in Eq. (2.3), includes a vector of interdisciplinary coupling variables $\boldsymbol{y}$, while the shared variables and shared uncertain parameters are $\boldsymbol{x}_{sh}$ and $\boldsymbol{p}_{sh}$, respectively:

$$\begin{aligned} &\min_{\boldsymbol{x}_{sh},\,\boldsymbol{x}_i} && \boldsymbol{f}_i(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_i, \boldsymbol{y}_{ji}) \\ &\text{s.t.} && \\ & && \boldsymbol{g}_i(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_i, \boldsymbol{y}_{ji}) \le 0 \\ &\text{where} && \boldsymbol{y}_{ij} = \boldsymbol{Y}_i(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_i, \boldsymbol{y}_{ji}) \quad i = 1,2 \end{aligned} \tag{2.3}$$

The vector $\boldsymbol{y}_{ij}$, called coupling variables, represents a coupled variable vector: Outputs from discipline i and inputs to discipline j. For instance, consider the two-discipline example shown in Figure 2.5, $\boldsymbol{y}_{12}$ represents the outputs from discipline 1 and the inputs to disciplines 2 while $\boldsymbol{y}_{21}$ represents the outputs from disciplines 2 and the inputs to

discipline 1. Shared variables refer to those variables that will be used in more than one discipline. Please see Section 3.6.5 and Section 4.3.3 for a detailed example of MDO.

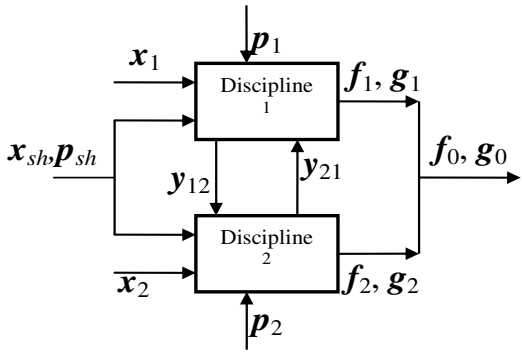


Figure 2.5 A fully coupled two-discipline system

The vectors $\boldsymbol{x}_i$, $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$ are the vectors of "local" (discipline's) design variables, objective and constraint functions, respectively. Here local means within one discipline, without sharing with other disciplines. The vector $\boldsymbol{Y}_i$ in Eq. (2.3) represents the functions that are used to calculate the coupling variables $\boldsymbol{y}_{ij}$. Parameters $\boldsymbol{p}_i$ represent the local parameters that have interval uncertainty. The entire system objective and constraint functions, represented by vectors $\boldsymbol{f}_0$ and $\boldsymbol{g}_0$, can be assumed as functions of local objective and/or constraint functions, as shown in Figure 2.5.

2.4 SENSITIVITY ANALYSIS

In this section, we discuss several definitions used in Chapter 5 for Multi-Objective Sensitivity Analysis.

Inner product: We define an inner (or array) product operation for two vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, with the same number of elements, as: $\boldsymbol{\alpha}\cdot\boldsymbol{\beta} = (\alpha_1\beta_1, \alpha_2\beta_2,\ldots,\alpha_K\beta_K)$. This inner multiplication results in a new vector that has the same number of elements as $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and whose k-th element is the product of the k-th element of $\boldsymbol{\alpha}$ and the k-th element of $\boldsymbol{\beta}$.

Grayscale: Similar to the membership function in fuzzy logic [Zadeh, 1965], the original meaning of grayscale in image processing is the different shades of gray in an image [Vincent, 1993], not just black-and-white. In this dissertation, the grayscale of a value β is defined as a continuous varied level of β, from zero to itself. In other words, given a scalar α between 0 and 1, the grayscale of any value β is defined as the product of α and β, as $\alpha\beta$.

Parameter Uncertainty Retention Index (PURI): In this dissertation, the vector $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_K)$ is called the Parameter Uncertainty Reduction Index (*PURI*), with $0 \le \alpha_k \le 1$, for $k = 1,\ldots, K$, corresponding to each parameter. That is, each element of $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$, $\alpha_k\Delta p_k$, is a grayscale of a parameter variation Δp_k, for $k = 1,\ldots, K$.

Retained Tolerance Region (RTR): *RTR* is a retained tolerance region in $\boldsymbol{p}$-space defined as a grayscale of the original tolerance region $\Delta\boldsymbol{p}$, represented by the inner product of the *PURI* and the original tolerance region: $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p} = (\alpha_1\Delta p_1, \alpha_2\Delta p_2,\ldots, \alpha_K\Delta p_K)$. Essentially, *RTR* can represent any symmetric hyper-rectangle within the original tolerance region. When $\boldsymbol{\alpha} = \mathbf{1}$, *RTR* is the original tolerance region; and when $\boldsymbol{\alpha} = \mathbf{0}$, *RTR* is reduced to the nominal point of parameter, $\boldsymbol{p}_0$. For instance, in a two-dimensional case shown in Figure 2.6, $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$ can represent any rectangle within the original tolerance region.

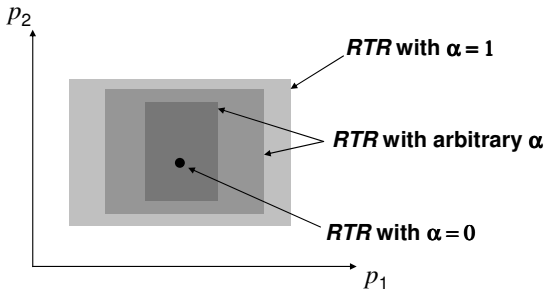


Figure 2.6 Retained tolerance regions

2.5 MULTI-OBJECTIVE GENETIC ALGORITHMS (MOGA)

The conventional MOGA used in this dissertation is based on NSGA [Deb, 2001] combined with an elitism strategy. Elitism strategies refer to the technologies used in the evolutionary algorithms that can keep all or part of the non-dominated individuals in the population as long as these elite individuals are not dominated by others. As shown in Figure 2.7, the conventional MOGA begins with coding all design variables (continuous or discrete) into binary chromosomes. It randomly (or based on some sampling strategies) generates an initial population (i.e., a family of starting points) of individuals (or design points) whose objective/constraint function values are calculated by simulations. Based on a fitness evaluation, the current population is divided into two sets: non-dominated and dominated. The next population is composed of two parts (see the "Next population" in Figure 2.7): elite and offspring points. The elite points are non-dominated points that are directly inherited from the previous generation. Offspring points are generated by GA operations, such as crossover and mutation. Then the algorithm evolves to the next generation (i.e., the algorithm goes into the next iteration) until the MOGA stops when all stopping criteria are satisfied. Then all non-dominated points in the last generation are considered as optimal Pareto solutions or good estimate of them. The stopping criteria and quality measures used in the dissertation are described in detail in Section 6.2.4. By comparing Pareto solutions obtained from MOGAs to the known Pareto frontiers for test examples, the convergence and diversity of obtained Pareto frontiers has been verified in Section 6.3 for the test problems.

In this dissertation, a strategy similar to NSGA-II [Deb, 2001] has been used to ensure that the number of non-dominated points is not more than a pre-specified

percentage (e.g., 60%) of the population. The remaining points are offspring design points that are produced by GA operations. Such a strategy ensures that a pre-specified percentage (e.g., 40%) of individuals in the population is generated by genetic operations so that the algorithm can keep searching for new optima. For offspring design points, we use a probability of 0.95 for crossover and a probability of 0.05 for mutation. The choice of these percentage and probability values is suggested by Deb [2001] or by MATLAB GADS User's Manual. For constraints, a previously reported constraint handling approach [Narayanan and Azarm, 1999], [Kurapati et al., 2002] has been used. Essentially, the constraint handling method is based on a penalty function which takes into account both the amount and the number of violated constraints. Moreover, using this method, the feasible solutions always have better fitness value than infeasible ones.

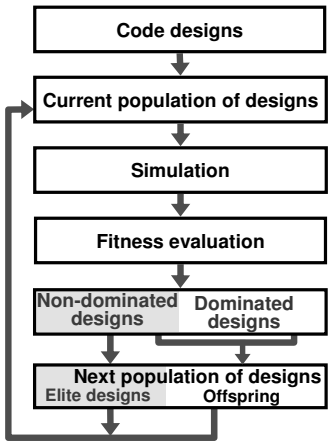


Figure 2.7 Flowchart of a conventional MOGA in one generation

In this dissertation, we define and use the following terms.

genc: The generation counter, which is an integer from 1 to the *MaxNumGeneration*.

MaxNumGeneration: The maximum number of the generations in the MOGA. In general, *MaxNumGeneration* is not larger than 100.

NumGeneration: The number of generations used in the MOGA when stop criteria are satisfied. The *NumGeneration* is less than the *MaxNumGeneration*. Typically for the examples used in the dissertation, The *NumGeneration* is about 30~50.

NumSimCall: The total number of simulation calls used in the MOGA. Based on our setting, the *NumSimCall* is calculated as:

$$NumSimCall = \sum_{genc=1}^{NumGeneration} rep_{genc}, \text{ and}$$

$$40\% \, popsize \times NumGeneration \le NumSimCall \le popsize \times MaxNumGeneration .$$

Typically in this dissertation, the total *NumSimCall* is less than 2500 in one run of MOGA.

popsize: The population size in the MOGA. It is generally dependent on the number of design variables. For the examples used in this dissertation, the *popsize* is no more than 100.

rep_{genc}: The number of new offspring individuals in the *genc*-th generation. According to our elitism strategy, rep_{genc} is at least 40% of the *popsize*. Basically, rep_{genc} represents the maximum number of simulation calls used in the *genc*-th generation to evaluate the individuals without metamodels.

In this dissertation, an individual in the MOGA represents a design or a design point in the design optimization problems. Hereafter, an individual or a point always refers to a design.

In the conventional MOGA, the response (i.e., objective and constraint functions) values of points in the initial population are calculated by a simulation model. Our conventional MOGA is different from NSGA-II [Deb, 2001] with respect to the elitism strategy. NSGA-II requires more computational effort since it combines the offspring

population with the parent population and then non-dominated sorting is used to classify the entire population. In the MOGA used here, only non-dominated points (or part of them) are migrated to the next generation.

2.6 KRIGING METAMODELING

For completeness, a brief description of the kriging metamodeling which is used in Chapter 6 is given in this section [Simpson et al., 2001], [Simpson et al., 2004]. As mentioned before, a design point is usually evaluated by a simulation, which is referred to as "observed by simulation." However, due to the intensive computational cost of the simulation, a reduced form model is usually used as a replacement of the simulation to predict or estimate the response values of the design in less time. As a popular metamodeling approach, kriging [Simpson et al., 2001], [Sacks et al., 1989] has been widely used in recent years for metamodeling of computationally expensive deterministic simulations [Koch et al., 2002], [Martin and Simpson, 2006], [Sasena et al., 2005].

Belonging to the family of ordinary linear least squares estimation algorithms, kriging predicts the response of unobserved points (i.e., those whose response has not been obtained by the simulation) based on all of the observed points (i.e., those whose response has already been obtained), as shown in Figure 2.8. The kriging method used in this dissertation is also called ordinary kriging in the literature [Simpson et al., 2001], [Sacks et al., 1989] and it is often used for predicting a simulation's response values at discrete input locations (or design points) which is the situation for the proposed approach. In addition to the property that kriging computes the best linear unbiased estimate of the response, the reason we have used the kriging approach is that the

uncertainty (i.e., error) in an estimated response can be easily obtained as a byproduct with the kriging metamodel. However, the kriging metamodeling needs to perform matrix inversions for predicting the response, which can increase the computational time when the dimension of the problem is high [Jin, 2005].

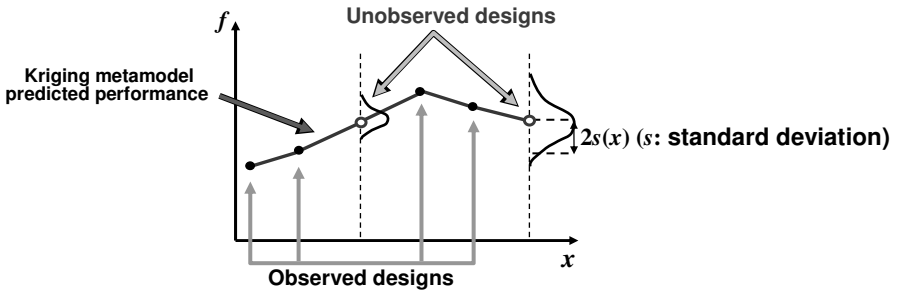


Figure 2.8 Kriging metamodeling for response prediction

In kriging, a one-dimensional response value from a simulation is globally estimated by a known polynomial and a random departure from the polynomial:

$$u(\boldsymbol{x}) = v(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{2.4}$$

where $u(\boldsymbol{x})$ is the unknown response of interest, $v(\boldsymbol{x})$ is a known polynomial, and $Z(\boldsymbol{x})$ is assumed as a realization of a Gaussian random process with a mean of zero, variance of σ^2, and a non-zero covariance between any two distinct observed points [Simpson et al., 2001], [Sacks et al., 1989]. The $v(\boldsymbol{x})$ term provides a "global" approximation of the design space; the $Z(\boldsymbol{x})$ term creates a "localized" deviation so that the kriging metamodel interpolates with respect to n_o observed points. The covariance matrix of $Z(\boldsymbol{x})$ is given by

$$\text{cov}[Z(\boldsymbol{x}^i), Z(\boldsymbol{x}^j)] = \sigma^2 \mathbf{R}; \quad \mathbf{R} = [\text{R}(\boldsymbol{x}^i, \boldsymbol{x}^j)] \tag{2.5}$$

where $\mathbf{R}$ is a $n_o \times n_o$ symmetric correlation matrix with ones along diagonal, and $\text{R}(\boldsymbol{x}^i, \boldsymbol{x}^j)$, which is the correlation function between any two observed points $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$ for off-diagonal elements. The correlation function $\text{R}(\boldsymbol{x}^i, \boldsymbol{x}^j)$ used in this dissertation is

$$\mathrm{R}(\boldsymbol{x}^i, \boldsymbol{x}^j) = \exp[-\sum_{n=1}^{N} \theta_n \left| x_n^{\,i} - x_n^{\,j} \right|^2] \tag{2.6}$$

where θ_n is an unknown correlation parameter. The quantities $x_n^{\,i}$ and $x_n^{\,j}$ are the n-th components of the observed points $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$, respectively. The estimate, $\hat{u}(\boldsymbol{x}_0)$, of the response $u(\boldsymbol{x}_0)$ at an unobserved point $\boldsymbol{x}_0$ is given by

$$\hat{u} = \hat{\beta} + \mathbf{r}^T(\boldsymbol{x}_0)\mathbf{R}^{-1}(\boldsymbol{u} - \boldsymbol{v}\hat{\beta}) \tag{2.7}$$

where $\boldsymbol{u}$ is a column vector of length n_o which contains the values of the response at each observed point, and $\boldsymbol{v}$ is a column vector with n_o components which is filled with ones when $v(\boldsymbol{x})$ is constant. In Eq. (2.7),

$$\begin{aligned} &\hat{\beta} = (\boldsymbol{v}^T\mathbf{R}^{-1}\boldsymbol{v})^{-1}\boldsymbol{v}^T\mathbf{R}^{-1}\boldsymbol{u} \\ &\mathbf{r}^T(\boldsymbol{x}_0) = [\mathrm{R}(\boldsymbol{x}_0,\boldsymbol{x}^1),\ \mathrm{R}(\boldsymbol{x}_0,\boldsymbol{x}^2),\ldots,\mathrm{R}(\boldsymbol{x}_0,\boldsymbol{x}^{n_o})] \end{aligned} \tag{2.8}$$

The estimate of the variance $\hat{\sigma}^2$ for Eq. (2.5) is given by

$$\hat{\sigma}^2 = (\boldsymbol{u} - \boldsymbol{v}\hat{\beta})^T\mathbf{R}^{-1}(\boldsymbol{u} - \boldsymbol{v}\hat{\beta}) / n_o \tag{2.9}$$

The Mean Squared Error (MSE) s^2 for an unobserved point $\boldsymbol{x}_0$ using this kriging metamodel predictor is given by

$$s^2(\boldsymbol{x}_0) = \hat{\sigma}^2(1 - \mathbf{r}^T\mathbf{R}^{-1}\mathbf{r} + \frac{(1 - \boldsymbol{v}^T\mathbf{R}^{-1}\mathbf{r})^2}{\boldsymbol{v}^T\mathbf{R}^{-1}\boldsymbol{v}}) \tag{2.10}$$

Statistically, the Root Mean Squared Error (RMSE) or the standard deviation $s(\boldsymbol{x}_0)$ represents the predicted deviation of the kriging metamodel from the actual response as shown in Figure 2.8. In this dissertation, it is assumed that the predicted deviation from the kriging metamodel has a conditional normal distribution with a mean that is equal to the prediction and variance equal to the kriging variance. This normally distributed standard deviation s will be used in the dissertation to decide a prediction interval.

However, even if this normal distribution assumption does not hold, it is possible to find a transformation that makes the random process approximately normal [Albada and Robinson, 2007].

The maximum likelihood estimate of θ_n in Eq. (2.6) can be obtained by maximizing the following expression:

$$\max_{\theta > 0} \frac{-[n_o \ln(\hat{\sigma}^2) + \ln(\mathbf{R})]}{2} \tag{2.11}$$

Some new schemes used to update kriging metamodel parameters θ_n have been reported in the literature, e.g., [Gano et al., 2006], [Martin, 2007].

The next chapter will present a new method for single-disciplinary robust optimization.

CHAPTER 3: MULTI-OBJECTIVE AND FEASIBILITY ROBUSTNESS IN SINGLE-DISCIPLINARY DESIGN OPTIMIZATION

3.1 INTRODUCTION

Engineering design optimization problems can have parameters with interval uncertainty. Such an uncertainty can degrade the performance of optimized objective functions and/or change the feasibility of the optimal solutions significantly. A robust optimal design is a feasible design alternative that is optimal according to its objectives and whose variation in its objective or feasibility (or both) is still within the acceptable range when parameters vary.

The goal of optimization under uncertainty is to obtain a solution with an optimal expected value of the objective function under some chance constraints, such as the two-stage model and recourse methods in stochastic programming [Birge and Louveaux, 1997], [Ruszczynski and Shapiro, 2003]. However, in engineering design, with the introduction of the robust approach by Taguchi [Taguchi, 1978], there was a paradigm shift in design optimization under uncertainty. Instead of optimizing the expected value, Taguchi argued that the goal should also include minimizing the sensitivity of the solution with respect to variations, i.e., to obtain a robust optimal solution. Later researchers developed numerous stochastic and deterministic approaches for robust optimization. The stochastic approaches use probability information of the variable parameters, usually mean and variance, to optimize the expected value and minimize the sensitivity of the solutions (see, e.g., [Parkinson et al., 1993], [Yu and Ishii, 1998] and [Jung and Lee, 2002] for objective robust optimization; see [Chen et al., 1999], [Tu et al.,

1999], [Du and Chen, 2000(b)], [Choi et al., 2001], [Youn et al., 2003], [Ray, 2002] and [Gunawan and Papalambros, 2007] for feasibility robust optimization). Currently, robust optimization methods based on possibility theory [Mourelatos and Zhou, 2006] or using "imprecise probabilities" [Aughenbaugh and Paredis, 2006] have also been addressed in the literature. The main shortcoming of stochastic approaches is that probability distributions must be known or presumed (the difficulty on discontinuous distributions can be overcome by using scenario trees [Birge and Louveaux, 1997]). However, this requirement cannot be satisfied during the early stage of design due to insufficient samplings for calculating the probability distribution or inadequate information about the problem itself.

Many of the deterministic approaches obtain robust optimal design solutions using gradient information of parameters (see, e.g., [Balling et al., 1986], [Sundaresan et al., 1992], [Zhu and Ting, 2001], [Lee and Park, 2001], [Su and Renaud, 1997], [Messac and Yahaya, 2002]). "Deterministic" here implies that no stochastic information (i.e., probability distribution) for uncontrollable parameter variations is presumed. The main shortcoming of deterministic methods is that their objective or constraint functions must be differentiable with respect to the variables with uncontrollable variations. Some of the methods also assume that the objective or constraint functions can be treated as linear with respect to the parameter variations, which might not hold for large variations. Others use a "minimax regret criterion" over discrete scenarios [Kouvelis and Yu, 1997], [Kasperskia and Zieli´nskib, 2006], which could result in an overly conservative solution. There has been an attempt to extend deterministic methods to optimization problems with multiple objectives and with variations beyond a linear range [Gunawan and Azarm,

2004, 2005(a), 2005(b)], [Li et al., 2005]. However, Gunawan's method, where the backward mapping from objectives to parameters was established, is only applicable when the multiple objectives/constraints are continuous with respect to uncertain parameters. The detail comparison with Gunawan's method is described in Appendix and in Li et al., [2006]. Hence effective optimization methods that can handle mixed continuous-discrete design variables/parameters in robust optimization problems, where the simulation is a "black box," are of great interest. For such black box simulations, the Karush-Kuhn-Tucker (KKT) conditions or convexity assumptions can not be established [Floudas, 1995]. In summary, the study of deterministic robust formulations that are applicable to multi-objective optimization problems that have mixed continuous-discrete parameters with variability beyond a linear range, as will be proposed in this research thrust, is still an active research topic.

In this section, we present a new deterministic, non-gradient based approach for objective robust and feasibility robust optimization in multi-objective design problems with interval uncertainty in parameters. We extend the approach to "performance robust optimization," requiring objective and feasibility robustness together. "Deterministic" here implies that no stochastic information (i.e., probability distribution) for uncontrollable parameter variations is presumed. That is, we assume that we know a range of variation for parameters.

In order to assess the robustness of design alternatives, the problem considered in our robust optimization approach consists of a bi-level optimization, a special case of a Mathematical Program with Equilibrium Constraints (MPEC) [Luo et al., 1996]. These bi-level problems are notoriously hard to solve in that they are nonconvex and the entire

feasible region may not even be known in closed form. However, unlike MPECs discussed in Luo et al., [1996] and for which most studies have concentrated on, the robust optimization approach under consideration allows for mixtures of both continuous and discrete design variables. Thus, the problems we consider are of the hardest type due to non-convexities from the discrete nature, the black box functions that are potentially present, as well as the two levels that need to be considered. MOGA or GA has been used to overcome the difficulties.

The organization of the rest of this chapter is as follows. First we present a new objective robustness measure in Section 3.2. Section 3.3 presents a new deterministic Multi-Objective Robust Optimization (MORO) approach using the new measure. After that we develop a constraint robustness measure, similar to the objective robustness measure, and use it for Feasibility Robust Optimization in Section 3.4. In Section 3.5 we develop "performance robust optimization," invoking objective robustness and feasibility robustness together. Five numerical and engineering examples and the corresponding results are given in Section 3.6. Finally, in Section 3.7 the main observations for the proposed performance robust optimization approach are summarized.

The basic idea in the robust optimization approach described in this chapter was also presented in Li et al., [2006]. However, a new distance measure for robustness indices and five new numerical and engineering test examples are added in this chapter.

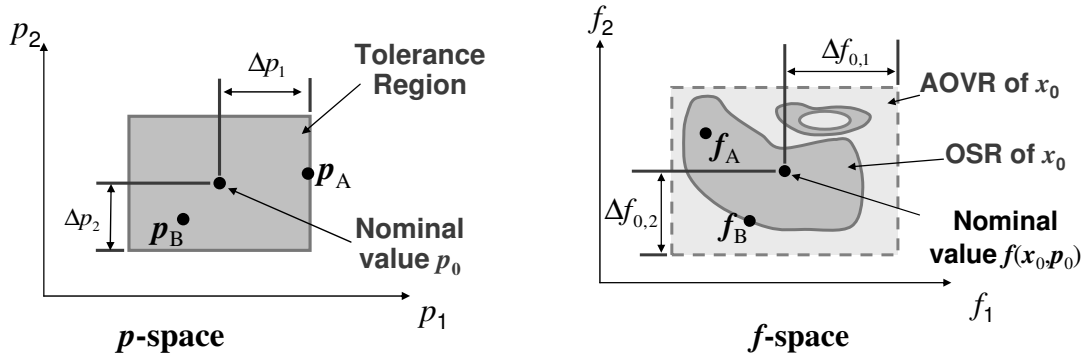
3.2 OBJECTIVE ROBUSTNESS MEASURE

In this section, we present a new objective robustness measure based on the mapping of the $\Delta \boldsymbol{p}$-space tolerance region into an objective sensitivity region in $\Delta \boldsymbol{f}$-space.

3.2.1 Objective Sensitivity Region

The effect of parameter variations on the objective values of a design $\boldsymbol{x}_0$ can be represented by a mapping from $\boldsymbol{x}_0$'s tolerance region in $\boldsymbol{p}$-space, i.e., $\{\tilde{\boldsymbol{p}} \mid \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p}\}$, to a corresponding region in $\Delta\boldsymbol{f}$-space. The latter region is called the Objective Sensitivity Region (OSR). The function in Eq. (3.1) defines this mapping from $\boldsymbol{p}$-space to $\boldsymbol{f}$-space.

$$\Delta f_m = f_m(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - f_m(\boldsymbol{x}_0, \boldsymbol{p}_0);\ m = 1,\ldots,M;\ \boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p} \tag{3.1}$$



(a) (b)

Figure 3.1 Forward mapping from (a) tolerance region in $\boldsymbol{p}$-space to (b) the corresponding OSR in $\boldsymbol{f}$-space for design $\boldsymbol{x}_0$

Figure 3.1 illustrates the mapping for a case with two parameters and two objective functions. Figure 3.1(a) shows the tolerance region (shaded rectangle) for a design alternative. Figure 3.1(b) shows the corresponding OSR (shaded area), which is obtained by the mapping of all the possible points $\tilde{\boldsymbol{p}}$ in the tolerance region to the $\boldsymbol{f}$-space. The mapping might be a "Many-to-One" mapping, with more than one point in $\boldsymbol{p}$-space mapping to one point in the OSR or might be a non-linear mapping. As shown in Figure 3.1, the point $\boldsymbol{p}_A$ is on the boundary of the tolerance region; however, after the mapping,

the corresponding point $\boldsymbol{f}_A$ in $\boldsymbol{f}$-space may not be on the boundary of the OSR. Further, the OSR might be disconnected or have holes as shown in Figure 3.1(b). Note that since the OSR is not necessarily symmetric (in fact we do not know the exact shape of the OSR upfront) and it is obtained as a result of a mapping from the tolerance region, the tolerance region also can be asymmetric or discontinuous.

The design $\boldsymbol{x}_0$ is objectively robust if its AOVR encloses its corresponding OSR, as illustrated in Figure 3.1. That is, no $\tilde{\boldsymbol{p}}$ in $\boldsymbol{x}_0$'s tolerance region will cause objective value to be outside the DM's acceptable region. If the AOVR does not enclose the OSR, then $\boldsymbol{x}_0$ is not an objectively robust design; that is, some value(s) of $\tilde{\boldsymbol{p}}$ in the tolerance region will cause at least one of $\boldsymbol{x}_0$'s objectives to be outside the DM's acceptable region. However, to allow the DM to trade robustness for optimality, we seek a quantitative measure of robustness rather than a binary indicator.

3.2.2 Objective Robustness Index

As mentioned before, the enclosure of the OSR in the AOVR is the criterion for objective robustness. However, determining the enclosure and calculating the quantitative measure of robustness can be intractable. To overcome this, we use the ratio of a worst case estimate of the "size" of design alternative $\boldsymbol{x}_0$'s OSR to the size of $\boldsymbol{x}_0$'s AOVR as a measure of $\boldsymbol{x}_0$'s robustness, i.e., the objective robustness index.

To calculate the objective robustness index, we need to obtain (i) an estimate of the size of the OSR, and (ii) a "direction" along the OSR which the combined objective functions are most sensitive (i.e., a point on the boundary of OSR which is furthest from the origin) as a result of variation in parameters. We wish to use the vector distance from the origin in the $\Delta\boldsymbol{f}$-space to the point with the maximum distance metric in the OSR as a

measure of the size and direction of OSR. However, obtaining this "vector distance" in the $\Delta \boldsymbol{f}$-space can be problematic because the objective values may have incommensurable units and scales (e.g., tens of dollars for one objective, thousands of millimeters for another). We therefore normalize each axis in the $\Delta \boldsymbol{f}$-space by the DM's acceptable variation on that axis, using Eq. (3.2).

$$\overline{\Delta f_m} = \frac{f_m(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - f_m(\boldsymbol{x}_0, \boldsymbol{p}_0)}{\Delta f_{0,m}};\ m = 1,\ldots,M \quad \boldsymbol{p}_0 - \Delta \boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta \boldsymbol{p} \tag{3.2}$$

For simplicity, our notation in the remainder of this dissertation does not include the bar over $\Delta \boldsymbol{f}$; the normalization is understood.

In the normalized $\Delta \boldsymbol{f}$-space the AOVR becomes a hyper-cube. We define R_I as the radius of AOVR in $\|.\|_\infty$. Since the AOVR is a hyper-cube, according to Eq. (2.2), $R_I = 1$ as shown in Figure 3.2. R_I is used as the measure of the size of the AOVR.

Our worst-case estimate of the OSR in the normalized $\Delta \boldsymbol{f}$-space is R_f, which is defined as the $\|.\|_\infty$ distance from the origin in $\Delta \boldsymbol{f}$-space to the point with the maximum $\|.\|_\infty$ in the OSR, as shown in Figure 3.2. The hyper-cube with radius R_f is called the Worst Case Objective Sensitivity Region, WCOSR. This WCOSR is a hyper-cube in the normalized space so that its radius R_f determines its size.

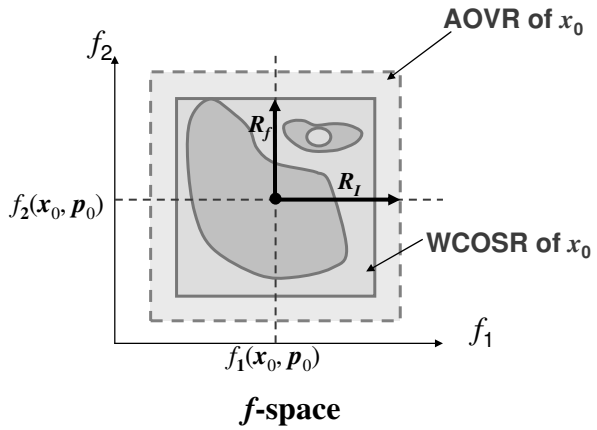


Figure 3.2 Worst-case estimate of the OSR in normalized *f*-space

The normalized AOVR encloses the WCOSR if R_f does not exceed R_I, as illustrated in Figure 3.2. Next, we define an objective robustness index: $\eta_f = \frac{R_f}{R_I}$. Because $R_I = 1$, this becomes $\eta_f = R_f$. A design is totally objectively robust if: $\eta_f \leq 1$. Notice here we use the $\|.\|_\infty$ distance instead of the $\|.\|_2$ distance metric as in [Li et al., 2006], so that failure shown in Figure 3.3 can be avoided. In Figure 3.3, the trial point $\boldsymbol{x}_0$ is robust but the robustness index mistakenly identifies it non-robust; because using the $\|.\|_2$ metric, R_f is larger than R_I (in $\|.\|_2$ distance, R_I is still one).

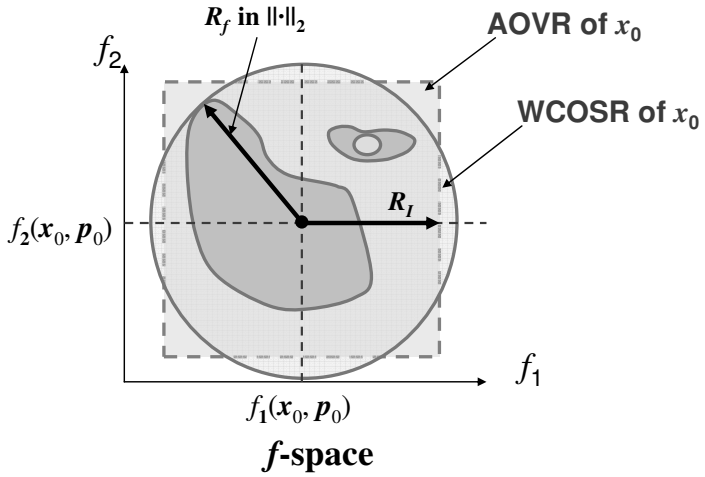


Figure 3.3 An example where $\|.\|_2$ metric fails for the objective robustness index

3.2.3 Calculating the Objective Robustness Index

Directly calculating the entire OSR for every design alternative considered when solving an optimization problem will generally be burdensome because the tolerance region must be covered finely enough to account for any possible non-convexity or discontinuity in $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p})$ (in an even worse case, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p})$ may be a black box). Fortunately, R_f can be calculated using an optimization method, as shown in Eq. (3.3), where we have included the normalization.

$$
\begin{aligned}
& R_f(\boldsymbol{x}_0) = \max_{\tilde{\boldsymbol{p}}} \left[\left\| \Delta \boldsymbol{f}(\tilde{\boldsymbol{p}}) \right\|_\infty \right] \\
& \text{where} \quad \Delta f_m(\tilde{\boldsymbol{p}}) = \frac{f_m(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - f_m(\boldsymbol{x}_0, \boldsymbol{p}_0)}{\Delta f_{0,m}}; \; m = 1, \ldots, M \\
& \qquad \boldsymbol{p}_0 - \Delta \boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta \boldsymbol{p}
\end{aligned}
\tag{3.3}
$$

In this optimization problem, the variables are the K elements of the parameter variation vector $\tilde{\boldsymbol{p}}$. The quantity Δf_m is the m-th objective function variation due to $\tilde{\boldsymbol{p}}$. In the normalized $\Delta \boldsymbol{f}$-space, this optimization problem finds the maximum $\|.\|_\infty$ distance from the origin to the furthest point on the OSR, thus the radius of the WCOSR. A robust design will not be considered as a non-robust design by using the $\|.\|_\infty$ distance metric in Eq. (3.3). Suppose a design, $\boldsymbol{x}_0$, is actually robust but considered as a non-robust one as obtained by Eq. (3.3). This means that the R_f value for $\boldsymbol{x}_0$ must be larger than 1 by the $\|.\|_\infty$ metric (see Figure 3.2). Then there must be at least one $\tilde{\boldsymbol{p}}$ point in the tolerance region that leads to at least one Δf_m to be larger than $\Delta f_{0,m}$, according to our robust criterion in Eq. (3.3). By the definition of objective robustness in this approach, design $\boldsymbol{x}_0$ is not robust, which is a contradiction to the initial assumption (i.e., design $\boldsymbol{x}_0$ is robust).

In the next section we show how we use η_f (that is, R_f from Eq. (3.3)) in the optimization problem Eq. (2.1) to obtain the robust alternatives.

3.3 MULTI-OBJECTIVE ROBUST OPTIMIZATION USING ROBUSTNESS INDEX

In this section, we present our new approach for robust Multi-Objective Robust Optimization (MORO) problems using η_f, the objective robustness index introduced in the previous section.

3.3.1 MORO Problem

The goal of the MORO problem is to identify design alternatives that simultaneously have optimal objective values and satisfy the objective robustness requirement: variations in the parameters will not cause the objective values to vary beyond the AOVR. Our approach for achieving robust solutions is to constrain the designs' objective robustness index to be at most a threshold value $\eta_{0,f}$ that is selected by the DM. This formulation of the MORO problem is shown in Eq. (3.4).

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}_0) && m = 1, \ldots, M \\ \text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}_0) \le 0 && l = 1, \ldots, L \\ & \eta_f - \eta_{0,f} \le 0 \\ & \boldsymbol{x}^{lower} \le \boldsymbol{x} \le \boldsymbol{x}^{upper} \end{aligned} \tag{3.4}$$

Here, η_f is calculated from Eq. (3.3), where it is called R_f. Setting $\eta_{0,f} = 1$ will ensure that the designs are robust. The DM may choose $\eta_{0,f} > 1$ to gain in the designs' nominal objective values, risking that some instances of the designs might have objective values outside the acceptable region. (The DM might choose $\eta_{0,f} < 1$ to get designs that are more than fully robust, perhaps reflecting some uncertainty in the accuracy of the tolerance region or in the calculation of objective functions. However, the "extra" robustness would, in general, degrade the nominal optimal objective function values.)

We assume here that robust solutions exist with a presumed AOVR. If they do not and if it is permissible, the AVOR is iteratively enlarged until such solutions do exist. Note that knowing upfront whether robust solutions exist for a presumed AOVR is difficult, if not impossible. This is because we have not assumed any mathematical form for the objective functions (e.g., they can be discontinuous with respect to uncontrollable parameters). A possible approach to knowing existence of a solution a-priori is for the DM to start with applying the sensitivity analysis approach that will be presented in Chapter 5 of this dissertation and determine an initial AOVR based on the results for the sensitivity analysis. If robust solutions can not be obtained, then the DM can enlarge the AOVR. This enlargement of the AVOR should be continued iteratively until robust solutions can be obtained. Note that here the AOVR is assumed to be continuous and symmetric with respect to nominal objective values for each objective function. However, our approach can be easily extended to an asymmetric, discontinuous AOVR or even to a non-constant AVOR for different designs, depending upon the nominal objective values.

3.3.2 Outer-Inner Optimization Structure

Similar to, but more difficult than MPECs [Luo et al., 1996] (due to non-convexities and black box functions), the formulation in Eq. (3.4) has two optimization problems: an outer problem to minimize the M objective functions f_m with respect to variables $\boldsymbol{x}$, subject to the L constraints g_l and the robustness constraint; and an inner problem to find η_f, for each design in the optimization process, by maximizing the radius R_f with respect to parameters $\tilde{\boldsymbol{p}}$ over the tolerance region.

Figure 3.4 shows this outer-inner structure; the outer problem is the upper problem in the figure, and the inner problem is the lower problem. The outer problem generates a candidate design alternative $\boldsymbol{x}_0$, which is feasible for all constraints other than objective robustness. The design variable values of $\boldsymbol{x}_0$ are sent to the inner problem, where the value of η_f (that is, R_f) is calculated for $\boldsymbol{x}_0$. The value of η_f is sent back to the outer problem, where it is used to evaluate the robustness constraint function, and to determine if $\boldsymbol{x}_0$ is feasible.

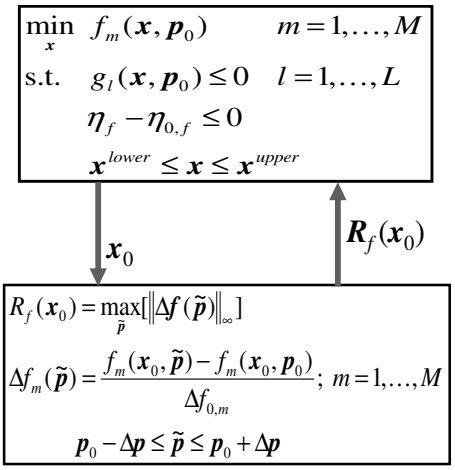


Figure 3.4 The outer-inner structure of MORO problems

Since the objective functions in the outer and inner problems might be black box functions with respect to $\boldsymbol{x}$ and $\tilde{\boldsymbol{p}}$, respectively, and because the OSR might be disconnected, we use a Genetic Algorithm (GA) to get a global optimal solution for the inner problem Eq. (3.3), and a MOGA (see Section 2.5) to get a Pareto optimal set for the outer problem Eq. (3.4). Note that the purpose of our outer-inner optimization is not to obtain the global optimal for (design alternative) $\boldsymbol{x}$ and (design parameter) $\tilde{\boldsymbol{p}}$ simultaneously. Rather, the purpose is to obtain, for a given design

alternative under consideration in the outer problem, the global optimal for robustness index (with respect to parameter $\tilde{\boldsymbol{p}}$) in the inner problem.

Concerning the computational cost of the approach, if *Nsc* is the number of simulation calls for the solutions of the outer problem and *Msc* is the number of simulation calls for the solution of the inner problem then the total number of simulation calls for our robust optimization approach will be in the order of *Nsc*×*Msc*.

3.4 FEASIBILITY ROBUST OPTIMIZATION USING CONSTRAINT ROBUSTNESS INDEX

Our approach, above, for objective robust optimization presumes that the parameter variations do not affect the designs' feasibility. However, this might not hold in general; the parameter variation might cause violation of some constraints. In this section we present a new deterministic method for Feasibility Robust Optimization (FRO). The goal of FRO is to minimize the objective function $\boldsymbol{f}$ with respect to $\boldsymbol{x}$, and simultaneously to ensure that the constraints $g_l \le 0$, $l=1,\ldots,L$, hold when the parameters vary. (Note that in FRO we are not concerned with variability of the objective function.) The approach is similar to our approach for MORO (Sections 3.2 and 3.3), employing a feasibility robustness index based on a sensitivity region.

3.4.1 Constraint Sensitivity Measure

For each feasible design alternative $\boldsymbol{x}_0$ ($g_l(\boldsymbol{x}_0,\boldsymbol{p}_0)\le 0, l=1,\ldots,L$), the tolerance region in $\Delta\boldsymbol{p}$-space maps into a sensitivity region in $\Delta\boldsymbol{g}$-space. We call this region in $\Delta\boldsymbol{g}$-space, the original constraint sensitivity region of $\boldsymbol{x}_0$, Figure 3.5 shows a two-constraint case. For this region, it is important to determine when $g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \ge g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)$ which corresponds to

$\boldsymbol{x}_0$ potentially becoming infeasible when $\tilde{\boldsymbol{p}}$ changes. (When $g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \le g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) \le 0$ the variation $\tilde{\boldsymbol{p}}$ will not change the feasibility of $\boldsymbol{x}_0$.)

Figure 3.5 shows the Constraint Sensitivity Region (CSR), which is essentially an estimate of the original constraint sensitivity region of $\boldsymbol{x}_0$. As with the OSR, we normalize the CSR using $\boldsymbol{x}_0$'s ACVR (see Section 2.2). The function in Eq. (3.5) defines the CSR.

$$\Delta g_l = \begin{cases} \dfrac{g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)}{\left| g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) \right|}, \text{if } g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \ge g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) & l = 1,\ldots,L \\ 0 \qquad \text{otherwise} \end{cases} \qquad (3.5)$$
$$\boldsymbol{p}_0 - \Delta\boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta\boldsymbol{p}$$

Note that the formulation in Eq. (3.5) causes the CSR to lie entirely in the first hyper-quadrant of the $\Delta\boldsymbol{g}$-space (i.e., the region for which $\Delta g_l \ge 0$, l=1,…,L). In normalized $\Delta\boldsymbol{g}$-space the ACVR is a hyper-cube in the first hyper-quadrant, with its lower vertex at nominal point $\boldsymbol{g}(\boldsymbol{x}_0,\boldsymbol{p}_0)$. Figure 3.5 also illustrates the CSR and ACVR in a normalized $\boldsymbol{g}$-space.

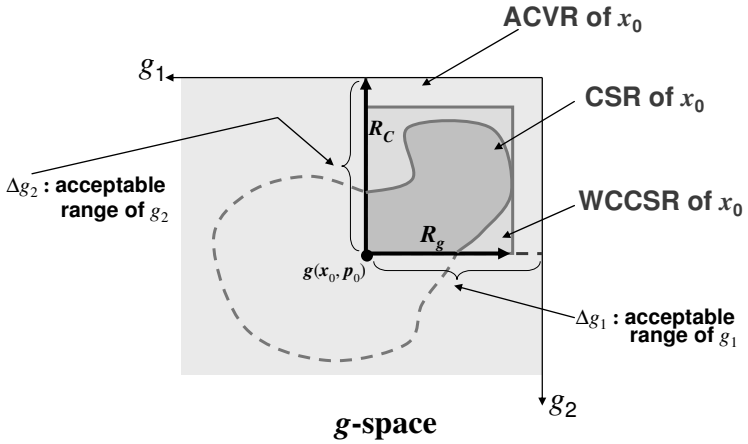


Figure 3.5 ACVR, CSR, and WCCSR in normalized g-space for design x_0

The criterion for feasibility robustness is to determine whether the ACVR of design $\boldsymbol{x}_0$ encloses the CSR. Applying the same rationale as we used for creating the WCOSR,

we use a worst-case estimate of the CSR, which is called Worst-Case Constraint Sensitivity Region (WCCSR). As illustrated in Figure 3.5, the WCCSR is the part of a hyper-cube in the first hyper-quadrant of the $\Delta \boldsymbol{g}$-space. The hyper-cube's radius, called R_g, is the $\|.\|_\infty$ distance from the origin to the furthest point of the CSR; this radius determines the size of the WCCSR. For the size of the ACVR we take the radius, R_C in Figure 3.5, of the ACVR from the origin to the ACVR boundary. We define the feasibility robustness index $\eta_g = \frac{R_g}{R_C}$ to correspond to our definition of objective robustness index (Section 3.2). A design is fully feasibility robust if its $\eta_g \leq 1$. Since $R_C =$ 1 in $\|.\|_\infty$ distance, normalized in $\Delta \boldsymbol{g}$-space, we have $\eta_g = R_g$.

3.4.2 Formulation of Feasibility Robust Optimization

The optimization problem for calculating the R_g for design x_0 is shown in Eq. (3.6).

$$
\begin{aligned}
& R_g(\boldsymbol{x}_0) = \max_{\tilde{\boldsymbol{p}}} [\left\| \Delta \boldsymbol{g}(\tilde{\boldsymbol{p}}) \right\|_\infty] \\
& \Delta g_l = \begin{cases} \dfrac{g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)}{\left| g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) \right|}, \text{if } g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \geq g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) \\ 0 \qquad\qquad \text{otherwise} \end{cases} \qquad l = 1, \ldots, L \\
& \boldsymbol{p}_0 - \Delta \boldsymbol{p} \leq \tilde{\boldsymbol{p}} \leq \boldsymbol{p}_0 + \Delta \boldsymbol{p}
\end{aligned}
\tag{3.6}
$$

Similar to our MORO approach, we constrain the designs' feasibility robustness index to be at least a threshold value $\eta_{0,g}$ selected by the DM.

Eq. (3.7) presents the feasibility robust optimization problem:

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}_0) && m = 1, \ldots, M \\
\text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}_0) \leq 0 && l = 1, \ldots, L \\
& \eta_g - \eta_{0,g} \leq 0 \\
& \boldsymbol{x}^{lower} \leq \boldsymbol{x} \leq \boldsymbol{x}^{upper}
\end{aligned}
\tag{3.7}
$$

As with our approach for MORO (recall Section 3.3 and Figure 3.4), we form an outer-inner optimization structure to solve the problem. The outer problem Eq. (3.7) here corresponds to the outer problem Eq. (3.4). The inner problem Eq. (3.6) here calculates η_g (that is, R_g), corresponding to calculation of η_f in Eq. (3.3). Since the feasibility robust optimization problem, too, may have non-differentiable or discontinuous functions or a disconnected CSR, we again use a MOGA for the outer problem and a GA for the inner problem.

Note that the computation algorithm for Eq. (3.6) must address the divide-by-zero that could occur if $g_l(\boldsymbol{x}_0,\boldsymbol{p}_0) = 0$. Our technique is to set Δg_j to a value much larger than 1 if $g_l(\boldsymbol{x}_0,\boldsymbol{p}_0) = 0$ and $g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) > 0$, which makes η_g much larger than 1, artificially forcing design $\boldsymbol{x}_0$ to violate the feasibility robustness constraint in Eq. (3.7).

3.5 PERFORMANCE ROBUST OPTIMIZATION

Following the separate discussions of objective robust optimization in Section 3.3 and of feasibility robust optimization in Section 3.4, an immediate development is to combine them for what we term "performance robust optimization". The goal for performance robust optimization is to obtain design alternatives meeting three criteria: they have Pareto minimized objective values; their objective values are within the prescribed bounds as the parameters vary over the tolerance region; and they remain feasible as the parameters vary over the tolerance region. In this section we present a method for performance robust optimization.

A direct approach for performance robust optimization is to add both the objective robust constraint and the feasibility robust constraint to the original problem Eq. (2.1) to form Eq. (3.8).

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}_0) && m = 1, \ldots, M \\
\text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}_0) \le 0 && l = 1, \ldots, L \\
& \eta_f - \eta_{0,f} \le 0 \\
& \eta_g - \eta_{0,g} \le 0 \\
& \boldsymbol{x}^{lower} \le \boldsymbol{x} \le \boldsymbol{x}^{upper}
\end{aligned}
\tag{3.8}
$$

In this problem, η_f is calculated from Eq. (3.3) and η_g is calculated from Eq. (3.6). With this formulation, for every design alternative $\boldsymbol{x}_0$ there are two inner optimization problems as shown in Figure 3.6. The DM can specify the thresholds for objective robustness index η_f and for feasibility robustness index η_g independently.

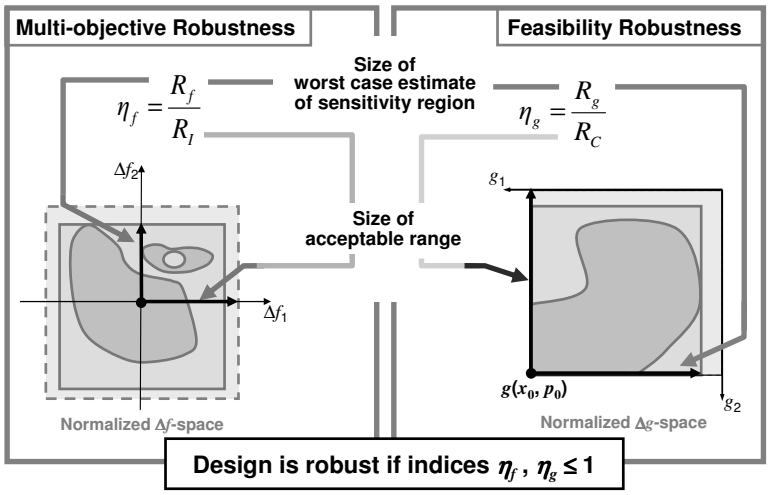


Figure 3.6 Performance robust optimization

In order to save the computational cost for two inner problems, we can combine the objective robustness index and feasibility robustness index in a single performance robustness index $\eta_{perf} = \max(\eta_f, \eta_g)$, requiring that neither exceed the threshold value. Accordingly, we need only one common threshold value η_0 for the performance robustness index. Thus we obtain the modified formulation in Eq. (3.9).

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}_0) && m = 1, \ldots, M \\ \text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}_0) \le 0 && l = 1, \ldots, L \\ & \eta_{perf} - \eta_0 \le 0 \\ & \boldsymbol{x}^{lower} \le \boldsymbol{x} \le \boldsymbol{x}^{upper} \end{aligned} \tag{3.9}$$

We calculate η_{perf} using one single-objective optimization problem as shown in Eq. (3.10).

$$\begin{aligned} & \eta_{perf}(\boldsymbol{x}_0) = \max\{\max_{\tilde{\boldsymbol{p}}}[\|\Delta \boldsymbol{f}(\tilde{\boldsymbol{p}})\|_\infty], \max_{\tilde{\boldsymbol{p}}}[\|\Delta \boldsymbol{g}(\tilde{\boldsymbol{p}})\|_\infty]\} \\ & \text{where} \Delta f_m(\tilde{\boldsymbol{p}}) = \frac{f_m(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - f_m(\boldsymbol{x}_0, \boldsymbol{p}_0)}{\Delta f_{0,m}};\ m = 1, \ldots, M \\ & \Delta g_l = \begin{cases} \dfrac{g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)}{|g_l(\boldsymbol{x}_0, \boldsymbol{p}_0)|}, \text{if } g_l(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) \ge g_l(\boldsymbol{x}_0, \boldsymbol{p}_0) & l = 1, \ldots, L \\ 0 \qquad \text{otherwise} \end{cases} \\ & \boldsymbol{p}_0 - \Delta \boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta \boldsymbol{p} \end{aligned} \tag{3.10}$$

It should be noted that the above mentioned formulation, for both multi-objective robust and feasibility robust optimization, is deterministic. It does not require any information about probability distributions of the parameter variations. Also, it does not assume linearity or continuity of the objective/constraint functions; hence, it is applicable even when the variations of parameters are large.

The proposed multi-objectively robust and feasibly robust formulations are conservative because directly calculating the sensitivity region is intractable in general. However, in the case when the mapping is linear and the original domain is polyhedral, it is well known that the sensitivity region will also be polyhedral. Both formulations estimate the sensitivity region by a worst case method, instead of calculating it directly. The benefit of the worst case method is that the robustness of obtained solutions is guaranteed and computational cost could be reduced compared with checking the

robustness along each dimension of the objective/constraint space. In this Eq. (3.10), the lower level problem is required to be solved globally which further exacerbates the computational cost.

In the next section, we use several numerical and engineering examples to demonstrate the applicability of this performance robust optimization approach. For simplicity, hereafter, we mean both multi-objective and feasibility robustness by robustness.

3.6 EXAMPLES AND RESULTS

In this section, we use five examples to demonstrate our method. The first one is a numerical and others are all taken from engineering applications, including the design of a vibrating platform, an angle grinder, a speed reducer, and a case study for a UUV-Payload. Several of these engineering examples are also used in later chapters and we have added parameters with interval uncertainty to form robust optimization problems. We also have discrete variables or parameters in these examples, to demonstrate the general applicability of the proposed approach. To save space, we only present the verification results for the first three examples.

3.6.1 Numerical Example

This bi-objective numerical example is developed as an extension from a single-objective MDO problem with two coupled disciplines [Gu and Renaud, 2002]. Here we use it as a single-disciplinary (or so-called "all-at-once" format) two-objective optimization problem given in Eq. (3.11). There are three design variables: $\boldsymbol{x} = [x_1, x_2, x_3]$, two objective functions: $\boldsymbol{f} = [f_1, f_2]$, and two constraint functions: $\boldsymbol{g} = [g_1, g_2]$.

$$
\begin{aligned}
\min_{\boldsymbol{x}} \quad & f_1 = {x_2}^2 + x_3 + y_1 + e^{-y_2} \\
\min_{\boldsymbol{x}} \quad & f_2 = x_1 + \sqrt{x_2} + ({y_2}^2 - {y_1}^3)/10^4 + 150 \\
\text{s.t.} \quad & g_1 = 8 - y_1 \le 0 \\
& g_2 = y_2 - 10 \le 0 \\
& -10 \le x_1 \le 10, \ 0 \le x_2, x_3 \le 10
\end{aligned}
\tag{3.11}
$$

The two inner variables y_1 and y_2 given in this example are:

$$
\begin{aligned}
y_1 &= \boldsymbol{Y}_1(\boldsymbol{x}, y_2) = {x_1}^2 + x_2 - 0.2\, y_2 \\
y_2 &= \boldsymbol{Y}_2(\boldsymbol{x}, y_1) = x_1 + x_3 + \sqrt{y_1}
\end{aligned}
\tag{3.12}
$$

For robust design optimization, we introduce an uncontrollable variation in design variable $\boldsymbol{x}$. The variations in Δx_2 and Δx_3 are known to be continuous and within 6% from the nominal value. Moreover, in this example, we assume that there are uncontrollable discrete variations for one of the variables. That is, Δx_1 is discretized to 12 possible values, ±1%, ±2%, ±3%, ±4%, ±5%, or ±6% from the nominal. The AOVR for each of bi-objective functions f_1 and f_2 is both ±10 units from their nominal. The threshold robustness index η_0 is set to be 1.

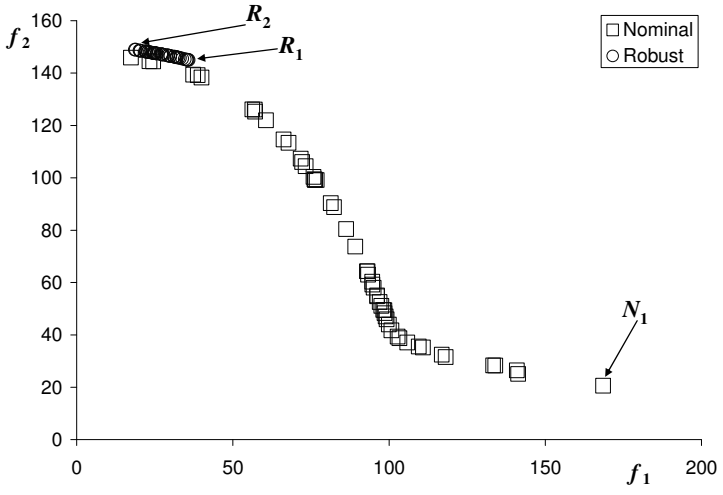
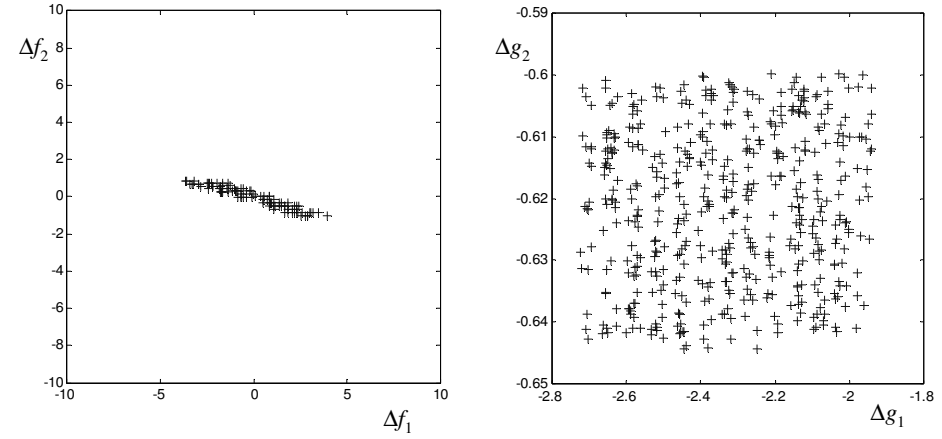


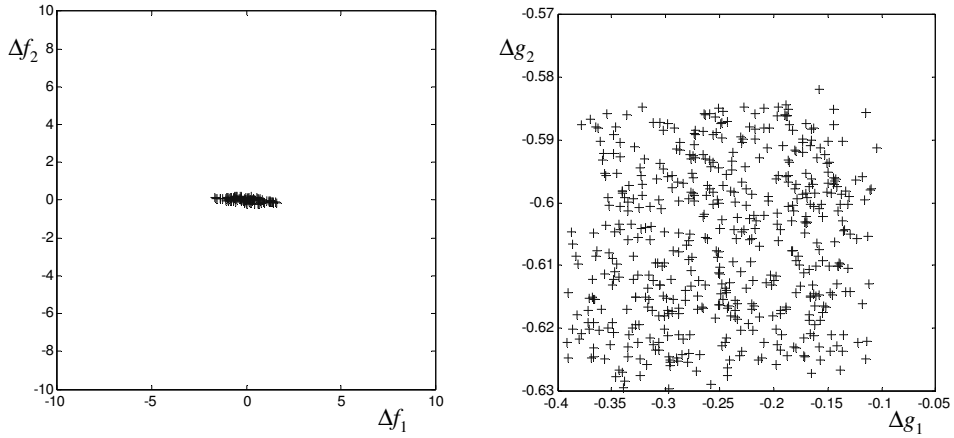
Figure 3.7 Nominal and robust Pareto solutions

We obtained the nominal Pareto solutions of Eq. (3.11) and the robust solutions for the above mentioned settings. Both of them are shown in Figure 3.7. As expected, robust Pareto solutions are slightly dominated by the nominal Pareto solutions. And in this problem, as shown in Figure 3.7, all robust solutions are clustered on the north-west side close to the nominal solutions. To verify the robustness of the robust designs obtained, a Monte-Carlo (MC) simulation (using uniform distributions) perturbing the variation over the tolerance region is applied. Two robust optimal designs R_1 and R_2, shown in Figure 3.7, are selected here to demonstrate the verification for the performance robustness. For both of these two points, we used 10,000 sample points for the tolerance region and verified that the objective and constraint functions remained within their acceptable range for these two robust designs, as shown in Figure 3.8 (a) and (b) (for visualization, only a subset of the MC points are shown), respectively. In this figure, small crosses represent the $\Delta \boldsymbol{f}$ and $\Delta \boldsymbol{g}$ values for each robust design. This verification indicates that obtained robust designs are both multi-objectively robust and feasibly robust. For these two robust designs, especially for design R_2, although their $\Delta \boldsymbol{f}$ values are much smaller than the acceptable range, their $\boldsymbol{g}$ values are close to the axes of the $\boldsymbol{g}$-space. This means that the variation in constraint values affects the performance robustness more significantly than the variation in objectives.

In addition, a typical nominal design N_1 obtained from the nominal Pareto set is verified as non-robust, because the variation in its objectives (about ±50 units in f_1 and ±20 units in f_2) are much larger than the acceptable range(±10 units for both f_1 and f_2), as shown in Figure 3.9.



(a)



(b)

Figure 3.8 Verification of performance robustness for design (a) R_1 and (b) R_2 as shown in Figure 3.7

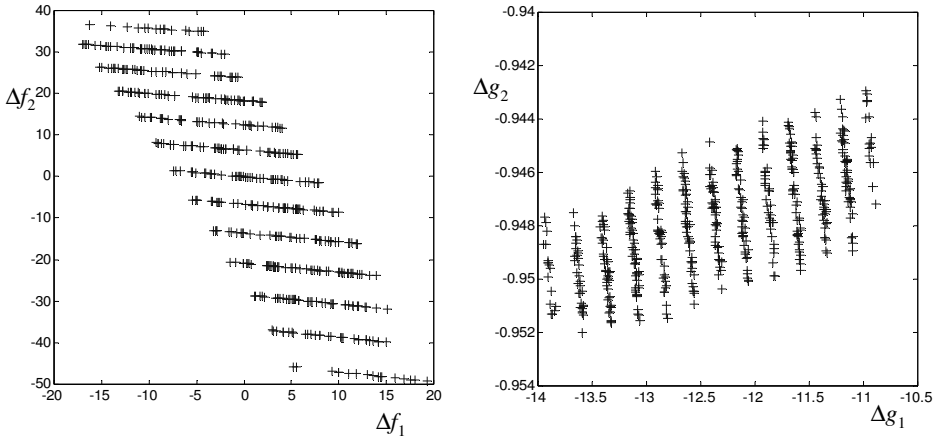


Figure 3.9 Verification of performance robustness for nominal design N_1 as shown in Figure 3.7

3.6.2 Vibrating Platform Design

The second example is to design a vibrating platform, which is modeled as a pinned-pinned sandwich beam with a vibrating motor on top [Narayanan and Azarm 1999], as shown in Figure 3.10. We formed a two-objective constrained optimization problem and use MOGA to obtain the nominal optimal solutions in the deterministic case.

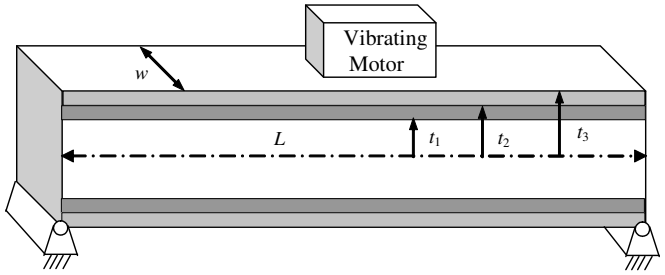


Figure 3.10 A pinned-pinned vibrating platform

The three layers of this platform (the inner layer, two middle layers sandwiching the inner layer, and two outer layers sandwiching the inner and middle layers) are made out of three different materials: type A, B, and C. The choice of materials for the layers must be mutually exclusive, i.e., no two layers can use the same material. The properties of each of the materials are shown in Table 3.1: ρ is the mass density, E is the modulus of elasticity, and c is the cost of the material per volume.

Table 3.1 Material properties

	A	B	C
ρ (kg/m^3)	100	2770	7780
E (GPa)	1.6	70	200
c (\$/m^3)	500	1500	800

Two objective functions of this optimization design are to maximize the natural frequency of a platform and to minimize its total material cost by controlling five sizing variables (continuous) and one combinatorial variable (discrete). The design variables are

the width of the platform (w), the length of the beam (L), and the thicknesses of the three layers (t_1, t_2, and t_3). The thicknesses of the middle and outer layers are represented as a difference between two sizing variables (e.g., thickness of the middle layer is equal to (t_2-t_1)). The combinatorial variable is the choice of materials for the layers (M). Since there are three possible material types, there are six possibilities for M (starting from the inner layer outward): {A,B,C}, {A,C,B}, {B,A,C}, {B,C,A}, {C,A,B}, and {C,B,A}. The platform design is subjected to five constraints: the maximum weight of the platform and the lower and upper limits on the thickness of the middle and outer layers.

The optimization formulation for this example is shown in Eq. (3.13).

$$
\begin{aligned}
\text{maximize} \quad & f_{\mathrm{n}} = \left(\frac{\pi}{2L^2}\right)\left(\frac{\mathrm{EI}}{\mu}\right)^{0.5} \\
\text{minimize} \quad & \text{cost} = 2wL\left[c_1 t_1 + c_2\left(t_2 - t_1\right) + c_3\left(t_3 - t_2\right)\right] \\
& \mathrm{EI} = \left(\frac{2b}{3}\right)\left[E_1 t_1^3 + E_2\left(t_2^3 - t_1^3\right) + E_3\left(t_3^3 - t_2^3\right)\right] \\
& \mu = 2w\left[\rho_1 t_1 + \rho_2\left(t_2 - t_1\right) + \rho_3\left(t_3 - t_2\right)\right] \\
\text{s. t.} \quad & g_1 \equiv \mu L\text{-}2800 \le 0 \\
& g_2 \equiv \left(t_1\text{-}t_2\right) \le 0 \\
& g_3 \equiv \left(t_2 - t_1\right) - 0.15 \le 0 \\
& g_4 \equiv \left(t_2 - t_3\right) \le 0 \\
& g_5 \equiv \left(t_3 - t_2\right) - 0.01 \le 0
\end{aligned}
\tag{3.13}
$$

In Eq. (3.13), the notations (ρ_1, ρ_2, ρ_3), (E_1, E_2, E_3), and (c_1, c_2, c_3) refer to the density, modulus of elasticity, and material cost for the inner, middle, and outer layer of the platform, respectively. The lower and upper bounds for the design variables are: $0.05 \le t_1 \le 0.5$, $0.2 \le t_2 \le 0.5$, $0.2 \le t_3 \le 0.6$, $0.35 \le w \le 0.5$, and $3 \le L \le 6$.

There are variations in the density and cost of "Material A" (ρ_{A} and c_{A}). For the sensitivity requirements, the parameter variations are known to be $[\Delta\rho_{\mathrm{A}}, \Delta c_{\mathrm{A}}] = [10$

kg/m^3, 25 \$/m^3]. The AOVR for the two objective functions f_1 and f_2 is defined by [$\Delta f_{0,1}$, $\Delta f_{0,2}$] = [\$5, 5Hz], presumed by the DM. Figure 3.11 shows the obtained robust Pareto solutions, compared to the nominal Pareto designs (shown as a min-min plot by taking the negative of the frequency). In this example, the objective values for nominal Pareto solutions are much better than those for robust Pareto solutions.

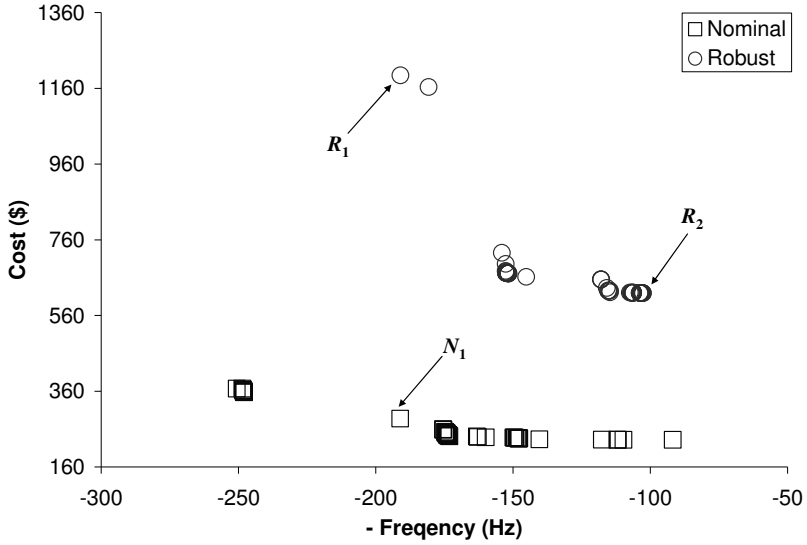
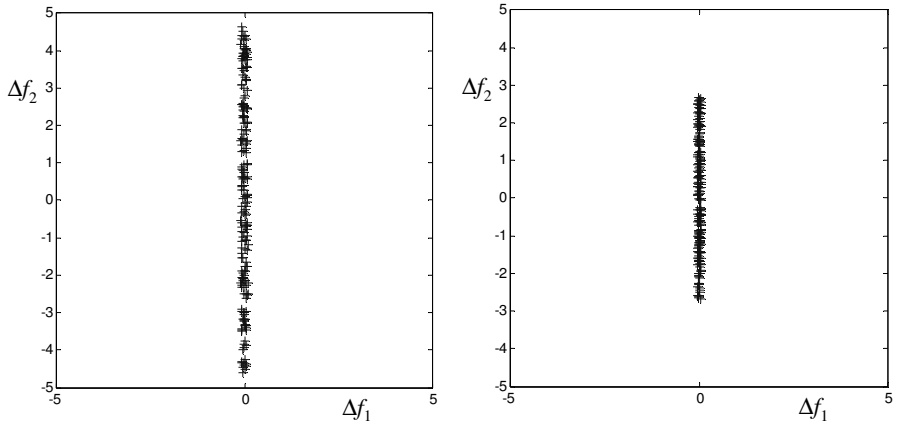


Figure 3.11 Nominal and robust Pareto solutions for vibrating platform design

To verify the robustness of the robust design obtained, the same Monte-Carlo simulation used in Section 3.6.1 is applied to two robust optimal designs R_1 and R_2, shown in Figure 3.11. For both of these two robust designs, their objective variations remained within their acceptable range [$\Delta f_{0,1}$, $\Delta f_{0,2}$] = [\$5, 5Hz], as shown in Figure 3.12(a) and (b), respectively. Notice in this figure we only show the verification for $\Delta \boldsymbol{f}$ values because the parameter variations [$\Delta \rho_A$, Δc_A] do not affect the constraint functions from g_2 to g_5 and g_1 is an inactive constraint (for all robust designs, the g_1 values are less than -300, which is far away from zero in $\boldsymbol{g}$-space). That is, the objective robustness

makes the main contribution for the performance robustness for this problem. This verification indicates that obtained robust designs are multi-objectively robust.



(a) (b)

Figure 3.12 Verification of objective robustness for design (a) R_1 and (b) R_2 as shown in Figure 3.11

In addition, a typical nominal design N_1 obtained from the nominal Pareto set is also verified as non-robust because the variation in its f_2 (about ±12Hz) is much larger than the acceptable range (±5Hz), as shown in Figure 3.13.

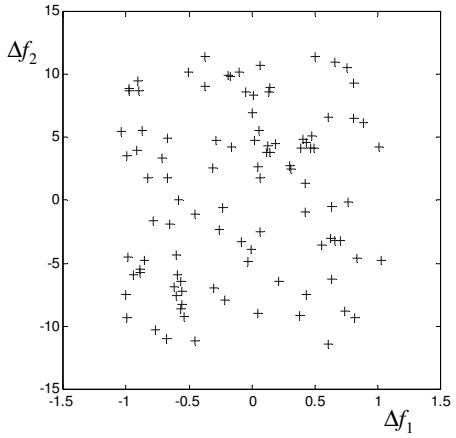


Figure 3.13 Verification of objective robustness for design N_1 as shown in Figure 3.11

3.6.3 Speed Reducer Design

The second engineering example is a well-known problem of designing a speed reducer [Kurapati et al., 2002]. Here, we modified the formulation to a two-objective optimization problem, as shown in Eq. (3.14).

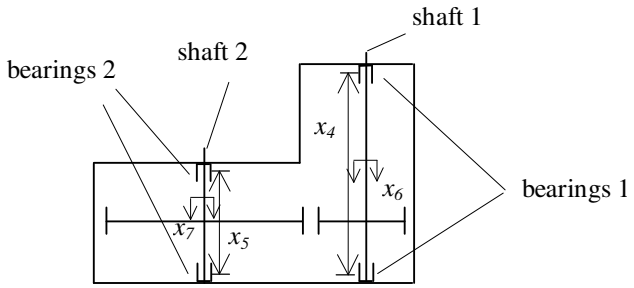


Figure 3.14 Design of a speed reducer

Figure 3.14 shows the configuration of the speed reducer. The objectives of the optimization problem are to minimize the total volume of the speed reducer as well as the stress in the first gear shaft. The problem has seven design variables: the gear face width (x_1), the teeth module (x_2), the number of teeth pinion (x_3), the distance between bearings on the first shaft (x_4) and on the second shaft (x_5), and the diameter of the first shaft (x_6) and second shaft (x_7). All design variables are continuous except for x_3 (the number of teeth), which must be an integer.

A lower and upper bound are imposed on each of the design variables. In addition, the design is subject to 11 inequality constraints. The constraints are: upper bound on the bending stress of the gear tooth (g_1), upper bound on the contact stress of the gear tooth (g_2), upper bound on the transverse deflection of the first shaft (g_3) and the second shaft (g_4), dimensional restrictions based on space and/or experience (g_5, g_6, and g_7), design requirements on the shaft based on experience (g_8 and g_9), and upper bound on the normal stress on the first shaft (g_{10}) and on the second shaft (g_{11}). The mathematical

formulation of the problem is given next. The units for all the variables are cm (except for x_3 – the integer variable). The unit for the first objective is cm^3 and for the second and the third objective is kPa.

$$\begin{aligned}
\min_{x} \quad f_1 &= 0.7854\ x_1 x_2^{\,2} \left(\frac{10\, x_3^{\,2}}{3} + 14.933\ x_3 - 43.0934\right) \\
&\quad - 1.508\ x_1 (x_6^{\,2} + x_7^{\,2}) + 7.477\ (x_6^{\,3} + x_7^{\,3}) \\
&\quad + 0.7854\ (x_4 x_6^{\,2} + x_5 x_7^{\,2}) \\
\min_{x} \quad f_2 &= \frac{\sqrt{\left(\frac{745\ x_4}{x_2 x_3}\right)^2 + 1.69 \times 10^7}}{0.1 x_6^3}
\end{aligned} \tag{3.14a}$$

$$\begin{aligned}
\text{s. t.} \quad & g_1 \equiv \frac{1}{x_1 x_2^2 x_3} - \frac{1}{27} \le 0 \quad ; \quad g_2 \equiv \frac{1}{x_1 x_2^2 x_3^2} - \frac{1}{397.5} \le 0 \\
& g_3 \equiv \frac{x_4^3}{x_2 x_3 x_6^4} - \frac{1}{1.93} \le 0 \quad ; \quad g_4 \equiv \frac{x_5^3}{x_2 x_3 x_7^4} - \frac{1}{1.93} \le 0 \\
& g_5 \equiv x_2 x_3 - 40 \le 0 \quad ; \quad g_6 \equiv \frac{x_1}{x_2} - 12 \le 0 \\
& g_7 \equiv 5 - \frac{x_1}{x_2} \le 0 \quad ; \quad g_8 \equiv 1.9 - x_4 + 1.5 x_6 \le 0 \\
& g_9 \equiv 1.9 - x_5 + 1.1 x_7 \le 0 \quad ; \quad g_{10} \equiv f_2 - 1800 \le 0 \\
& g_{11} \equiv f_3 - 1100 \le 0
\end{aligned} \tag{3.14b}$$

$$\text{where} \quad : \quad f_3 = \frac{\sqrt{\left(\frac{745\ x_5}{x_2 x_3}\right)^2 + 1.575\ x10^8}}{0.1 x_7^3}$$

$$\begin{aligned}
& 2.6 \le x_1 \le 3.6 \qquad 0.7 \le x_2 \le 0.8 \\
& 17 \le x_3 \le 28 \qquad 7.3 \le x_4 \le 8.3 \\
& 7.3 \le x_5 \le 8.3 \qquad 2.9 \le x_6 \le 3.9 \\
& 5.0 \le x_7 \le 5.5
\end{aligned}$$

Two design variables, the teeth module (x_2) and the first shaft diameter (x_6) have variation as $[\Delta x_2, \Delta x_6] = [0.01, 0.1]$. The acceptable $\boldsymbol{f}$ variation is given as $[\Delta f_{0,1}, \Delta f_{0,2}) = [100, 75]$. The obtained robust Pareto solutions, compared to the nominal Pareto solutions, are shown in Figure 3.15. We see in this figure that the robust Pareto set is a subset of the nominal Pareto set.

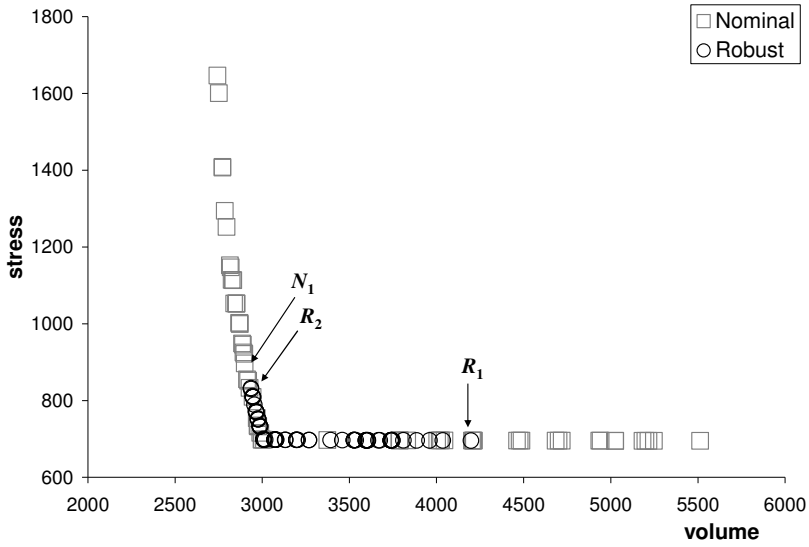


Figure 3.15 Nominal and robust Pareto solutions for speed reducer design

To verify the robustness of the designs obtained, a similar Monte-Carlo simulation is used to vary over the tolerance region $[\Delta x_2, \Delta x_6]$. Two robust optimal designs R_1 and R_2 have been verified as objectively robust, compared with the verification of a typical nominal optimal design N_1, as shown in Figure 3.16 and Figure 3.17, respectively. The feasibility robustness of robust Pareto solutions has also been verified.

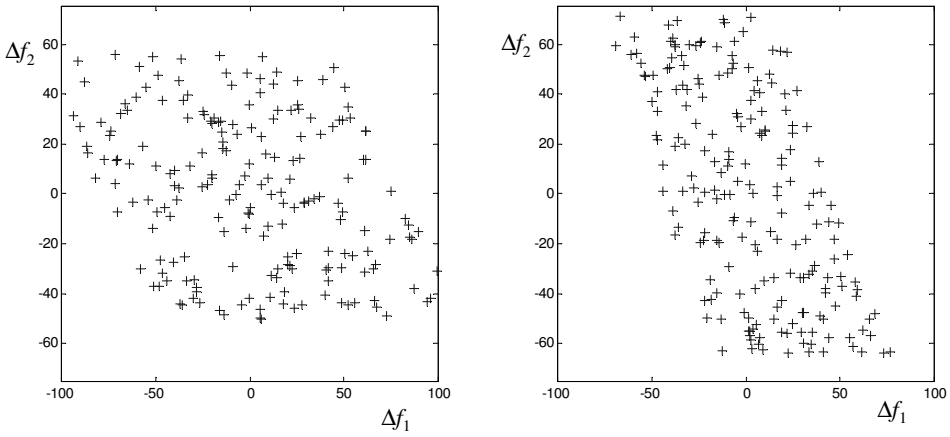


Figure 3.16 Verification of robustness for robust designs (a) R_1 and (b) R_2 as shown in Figure 3.15

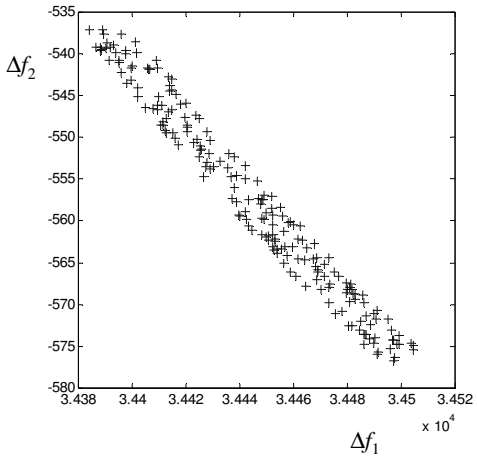


Figure 3.17 Verification of robustness for a nominal design N_1 as shown in Figure 3.15

3.6.4 Angle Grinder Design

This engineering example was also selected from the literature and is to optimize the design of an angle grinder tool [Williams et al., 2006]. Several existing and validated models exist for the major components of the angle grinder, such as the universal motor [Simpson, 1998] and the American Gear Manufacturers Association standard for bevel gears [Hurricks, 1994]. The two components of greatest interest (motor and bevel gear) of the grinder are shown in Figure 3.18. In this example we have nine design variables, two objective functions and thirteen constraint functions as described in Tables 3.2 - 3.6.

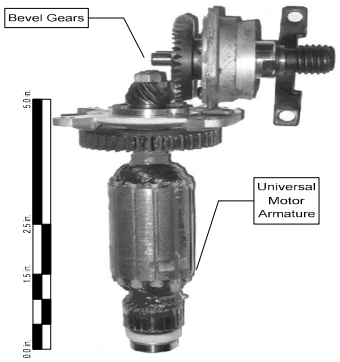


Figure 3.18 Engineering components for an angle grinder

Table 3.2 Grinder design variables

Pinion pitch diameter D_p (m)	$.009 \le D_p \le .03$
Current I (amps)	$6 \le I \le 12$
Gap thickness l_{gap} (m)	$.00005 \le l_{gap} \le .07$
Stack length L (m)	$.01 \le L \le .2$
Armature turns N_c (# of turns)	$20 \le N_c \le 300$
Stator turns N_s (# of turns)	$10 \le N_s \le 200$
Gear ratio r	$.2 \le r \le 4$
Stator outer radius R_o (m)	$.01 \le R_O \le .1$
Stator thickness t (m)	$.0001 \le t \le .1$

The nine design variables $\boldsymbol{x}$ with their lower and upper bounds make up the physical characteristics of the motor and bevel gear assembly (in Table 3.2). Two objective functions, the total mass of the grinder and its cost (Table 3.6) are calculated through a series of engineering computations in Table 3.3 and Table 3.4. The cost function is a presumed regression of two important attributes of the grinder. Table 3.3 and Table 3.4 develop in a sequential fashion to facilitate the readers understanding of computation dependencies. The physical constraints of this problem are shown in Table 3.5.

Table 3.3 Universal motor design computations

Armature diameter l_r (m)	l_r =2(R_o-t-l_{gap})
Armature section A_r (m^2)	$A_r = (\pi \cdot l_r^2)/4$
Wrap length l_{rw} (m)	$A_r = 2l_r + 2L$
ρ (ohm-m) 20 awg	*0.036 ohms-m*
Wire area A_w (m^2) 20 awg	*0.000504 m^2*
Arm. resistance R_a (ohms)	$R_a = (\rho N_c l_{rw})/A_w$
Stator resistance R_s (ohms)	$R_a = 2(\rho N_s l_{rw})/A_w$
Resistance losses P_{copper} (W)	$P_{copper} = I^2(R_s + R_a)$
Brush coefficient α (volts)	α =2
Brush losses P_{brush} (W)	$P_{brush} = \alpha \cdot I$
Voltage V (volts)	*V=120 v*
Power in P_{in} (W)	$P_{in} = I \cdot V$
Motor output P_{out} (W)	$P_{out} = P_{in} - P_{brush} - P_{copper}$
Density Steel ρ_s (kg/ m^3)	ρ_s = 8000(kg/ m^3)
ρ_{copper} (kg/ m^3)	ρ_{copper} = 8900 (kg/ m^3)
Stator mass M_s (kg)	$M_s = (\pi(R_o)^2 - \pi(R_o - t)^2) \cdot L \cdot \rho_s$
Armature mass M_a (kg)	$M_a = A_r \cdot L \cdot \rho_s$
Windings mass M_w (kg)	$M_w = l_{rw}(N_c + 2N_s)A_w \cdot \rho_{copper}$
Motor mass M_m (kg)	$M_m = M_s + M_a + M_w$
Motor constant K	$K = N_c/\pi$
Magnetomotive force $\Im$	$\Im = N_s I$
Mean stator path l_c (m)	$l_c = \pi(2R_o + t)/2$
Stator cross section A_s (m^2)	$A_s = L \cdot t$
Armature section A_a (m^2)	$A_a = L \cdot l_r$
Gap cross section A_g (m^2)	$A_g = L \cdot l_r$
Permeability of steel μ_{steel}	μ_{steel} =1000
Permeability, free space μ_o	$\mu_o = 4\pi \cdot 10^{-7}$
Stator reluctance $\Re_s$	$\Re_s = l_c/2(\mu_{steel} \cdot \mu_o \cdot A_s)$
Armature reluctance $\Re_r$	$\Re_r = l_r/(\mu_{steel} \cdot \mu_o \cdot A_a)$
Air gap reluctance $\Re_g$	$\Re_g = l_{gap}/(\mu_o \cdot A_g)$
Total reluctance $\Re_{tot}$	$\Re_{tot} = \Re_s + \Re_a + 2\Re_g$
Flux ϕ	$\phi = \Im/\Re_{tot}$
Torque T (N-m)	$T = K \cdot \phi / I$
Revolutions per minute N	$N = 9.549 \cdot P_{out}(kw)/T(N-m)$

Table 3.4 Bevel gear design computations

Pinion torque (load RPM) T_p (N-m)	$T_p = 9.459 \cdot P_{out} / 6500 rpm$
Gear torque (load RPM) T_g (N-m)	$T_g = T_p \cdot r$
Pressure angle φ_p	$\phi_p = 20°$
Cone distance C (m)	$C = D_p /(2 \cdot SIN(\phi_p))$
Face width b (m)	$b=.008\ m$
Gear pitch diameter D_g (m)	$D_g = D_p \cdot r$
Tooth loading intensity F_i (N)	$F_i = 2 \cdot T_p \cdot C /(D_p \cdot b(C-b))$
Elasticity factor (Carbon steel) Z_e	$Z_e = 189$
Zone factor Z_H	$Z_H = 4/(SIN(2 \cdot \phi_p)^2$
Pinion pitch angle θ_g	$\theta_p = ASIN(D_p / C)$
Shaft angle γ	$\gamma = 90°$
Gear pitch angle θ_g	$\theta_g = \gamma - \theta_p$
Pinion cone depth d_v (m)	$d_v = D_p \cdot SEC(\theta_p)$
Gear cone depth D_v (m)	$D_v = D_p \cdot SEC(\theta_g)$
Amplification (light/medium shock)	$K_a = 1.35$
Load distribution (precision gears)	$K_m = 1.2$
Geometry factor J	$J=.25$
Number of pinion teeth N_t	$N_t = 11$
Module (pinion) m	$m=D_p / N_t$
Pinion mass M_p (kg)	$M_p = (\pi \cdot D_p^2 \cdot b \cdot \rho_{steel}) / 4$
Gear mass M_g (kg)	$M_g = (\pi \cdot D_g^2 \cdot b \cdot \rho_{steel}) / 4$
Bevel gears mass M_{bg} (kg)	$M_{bg} = M_p + M_g$

Table 3.5 Grinder constraints $g(x)$

Flux density armature B_r (T)	$B_r = \phi / A_a \le 1.5T$
Flux density stator B_s (T)	$B_s = \phi /(2 \cdot A_s) \le 1.5T$
Flux density air gap B_g (T)	$B_g = \phi / A_g \le 1.5T$
Armature heat flux K_s (A/m)	$K_s = \dfrac{N_c \cdot I}{\pi \cdot l_r} \le 10000$
Stator heat flux K_s (A/m)	$K_s = \dfrac{N_s \cdot I}{\pi(l_r + t)} \le 10000$
Length to diameter ratio	$L / G \le 5$
Integer turns	$N_c, N_s = \text{int}$
Grinding wheel RPM N_{out}	$N_{out} = N / r \le 10000$
Bending stress σ_b (Pa)	$\sigma_b = (K_a K_m F_i) /(m \cdot J) \le 145 MPa$
Contact stress σ_f (Pa)	$\sigma_f = Z_H Z_e \sqrt{\dfrac{K_a K_m F_i (d_v + D_v)}{(d_v D_v)}} \le 720 MPa$
Armature tip velocity v_a	$v_a = \pi \cdot N \cdot l_r \le 3658\ (m/s)$

Table 3.6 Product attributes

Girth G (m)	$G=2(R_o+.004(m))$
Fixed mass M_f (kg)	$M_f = M_{cord} + M_{commutar} + ... = 1.58\ kg$
Total mass M_t (kg)	$M_t=M_{bg}+M_m+M_f$
Total *Cost* ($)	*Cost* = 3.61I+22.38pI/M_t-19.29, nominal p = 1

In this example, four design variables and parameters have uncontrollable uncertainty, represented by intervals. The standard alternating current voltage V (Table 3.3), stator outer radius R_o, stack length L, and the coefficient in the cost function p, $[\Delta V, \Delta R_o, \Delta L, \Delta p] = [5\text{v}, 3\%R_o, 3\%L, 0.02]$; $3\%R_o$ and $3\%L$ mean 3% of the nominal R_o and L values, respectively. The AOVR in this problem is: [ΔCost, ΔMass] = [4unit, 2kg]. The obtained nominal and robust Pareto designs are shown in Figure 3.19. For this example, it is clearly observed and as expected the robust Pareto solutions are dominated by the nominal solutions.

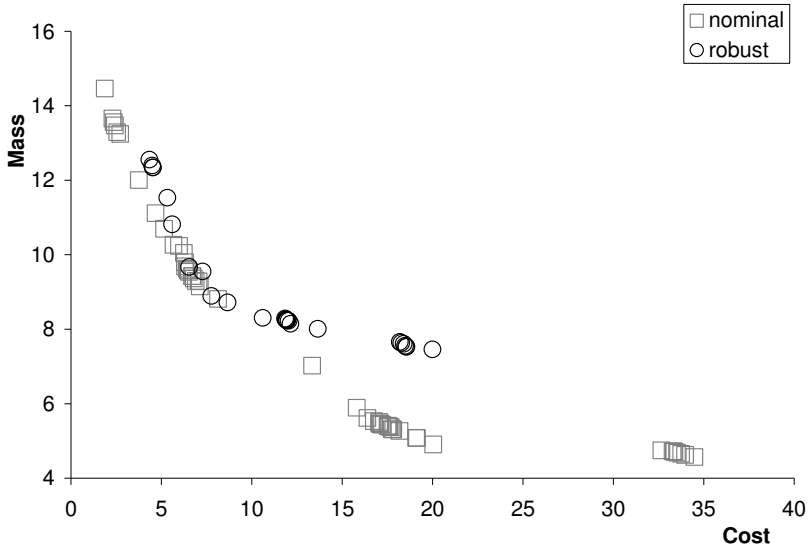


Figure 3.19 Nominal and robust Pareto solutions for grinder design

3.6.5 Unmanned Undersea Vehicle (UUV) with a Payload Design

As the final engineering example, we apply our performance robust optimization to the design of a UUV with payload. The original model for this example was developed previously in [Frits, 2004**],** [Burdic, 2003]. This design optimization consists of two interconnected disciplines: Payload and UUV. UUV discipline mainly includes the

design of a Propulsion part plus the guide and control component. Payload discipline focuses on the design of the payload itself.

Typically, the payload of a UUV must be effective in several different uses, called "scenarios." In this case study, we have two different scenarios: "Medium" and "Large". Effectiveness of the payload design in a scenario is measured by the probability of success of the payload delivery, $P_{S|UUV}$, in that scenario, given the probability of success of UUV, P_{UUV}. The design goal of the entire system is to simultaneously maximize the individual P_S's (the probability of success of the entire UUV) for two scenarios and minimize the total UUV weight (including Payload weight).

Two design variables, the Payload Length (PL), the Vehicle Diameter (VD) are used in both Payload and UUV disciplines. We call them shared design variables. These two variables are continuous and they are bounded by: $6.0 \leq VD \leq 12.75$ and $1.0(VD) \leq PL \leq 5.0(VD)$ (1~5 times of VD), respectively. There are four Payload design variables for Payload discipline itself: the Material of the Hull (HM), the Payload Type (PT), the first Inner material type (I1), and the second Inner material type (I2). All of the Payload variables are discrete: HM, PT, I1, and I2. The choices for HM, PT and I1 are [6061AL, 7075AL], [BULK, MULTI_MISS], and [I1_A, I1_B], respectively. For discrete variable I2, the options available are [I2_A, I2_B, I1_B], but I2 can be I1_B only if the variable I1 is I1_B also. In addition to the six design variables, there is a fixed continuous parameter, the ambient noise level (= 44 dB), on which the payload operates. However, unlike our other engineering examples, there is no closed-form formulation to map the design variables to the $P_{S|UUV}$'s. Rather, we are provided with a design analyzer, called Payload Sizing Model (a computer program) that maps the Payload design

variables to the payload size, weight and other intermediate attributes, based on which the P_{SlUUV}'s are calculated by a payload performance evaluator (called Payload Evaluator), as shown in Figure 3.20. The payload design is constrained by upper limits on the weight of the payload.

In addition to two shared design variables, we have four UUV design variables in UUV discipline: the Engine Type (ET) used in UUV, the HorsePower (HP) of the engine, the Run Distance (RD) of the UUV, and the Damper Thickness (DT) of the vehicle. The Engine Type (ET) is a discrete variable with two choices [OpenCyc, SCEPS]. Other three UUV variables are continuous and they are bounded by: $0.2 \le DT \le 2.0$, $10.0(VD) \le HP \le 20.0(VD)$ (10~20 time of VD value, but its unit is horse power), and $0.4(VD) \le RD \le 50.0(VD)$ (RD's unit is nautical mile). Similar to the Payload discipline, we are provided with a computer program (called UUV Sizing Model in Figure 3.20) to calculate the UUV's size, weight and other intermediate attributes and then these attributes are used to evaluate the probability of success for the UUV design, P_{UUV}. There exist couplings between two sizing models in this problem. For instance, the payload weight generated in Payload discipline is used in the UUV discipline as a variable; and G&C length, a variable generated in UUV discipline, is also used in Payload discipline. Chapter 4 will focus on how to handle the uncertainty propagated across disciplines through coupling variables. In this chapter, since we use all four computer programs all-at-once in only one performance analyzer, uncertainty across disciplines is masked within the performance analyzer as in a "black box."

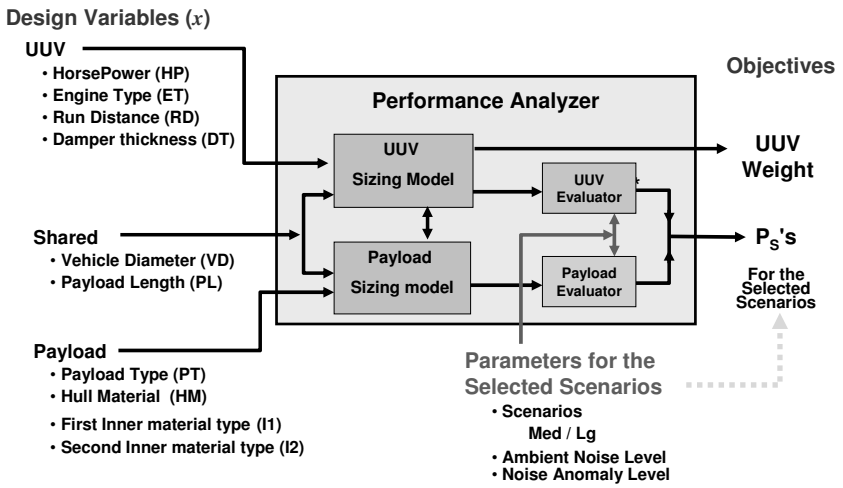


Figure 3.20 Performance analyzer for UUV-Payload design

In this example, we address a three-objective UUV-Payload design optimization. The two objectives are to maximize P_{SM} and P_{SL} for two different scenarios and to minimize the total UUV weight. The design variables are shown in Figure 3.20. In this example, three design variables that have uncontrollable uncertainty represented by intervals are $[\Delta RD, \Delta PL, \Delta VD] = [1\text{ nm}, 0.05\text{ inch}, 0.05\text{ inch}]$. The AOVR in this problem is: $[\Delta P_{SM}, \Delta P_{SL}, \Delta UUVWeight] = [0.1, 0.1, 40lb]$. The obtained, nominal and robust, Pareto designs are shown in Figure 3.21. A typical nominal design and a robust design that have the similar P_S's are also shown in Figure 3.21. In order to account for the robustness, the robust design requires more RD, which makes it larger and heavier (so that there is more energy in the propulsion part of the UUV) than the nominal counterpart. In this case, both of them have the similar payloads. The robustness of the robust Pareto designs is also verified by the Monte-Carlo simulations.

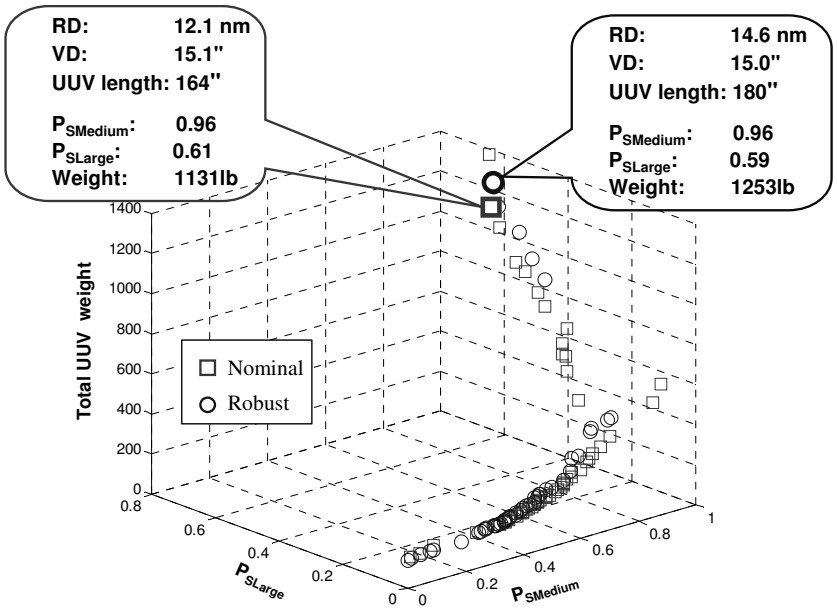


Figure 3.21 Nominal and robust Pareto UUV-Payload designs

3.7 SUMMARY

We have presented a new approach using robustness indices in robust design optimization for multi-objective problems in which uncontrollable variability in parameters causes variation in the objective functions and/or the constraint functions. The Decision Maker specifies the acceptable variation of the objective values. We presume that the range of the uncertain parameters is known. The approach can be used for objective robust optimization, for feasibility robust optimization, or for "performance robust optimization" (i.e., invoking both objective and feasibility robustness). The approach is deterministic, and so does not require probability distributions for the uncertain parameters. The approach is not gradient based, and so is applicable for cases with objective functions and constraint functions that are discontinuous (or merely non-differentiable) with respect to the parameter variations, and for cases where the variations are large, beyond the functions' linear range.

Our robustness measures, for objectives or for constraints, are based on mapping the parameter tolerance region into sensitivity regions in the objective space or the constraint space, respectively. For objectives or for constraints, we define a robustness index that is the ratio of the size of the corresponding sensitivity region to the size of the acceptable region of objective variation or constraint variation, respectively.

In the proposed approach, the sensitivity regions can be oddly shaped, or be disconnected or contain holes, so determining their "size" and most sensitive "direction" can be problematic. Accordingly, we define a worst-case estimate. Rather than calculating the sensitivity regions directly, we use an optimizer to solve for these worst-case estimates of the OSR and the CSR. The overall robust optimization problem thus becomes an outer-inner optimization problem. We use MOGA for the outer multi-objective optimization problem and GA for the two inner single-objective problems, one for objective robustness and the other for feasibility robustness. These two single-objective problem can also be combined into one problem to save the computational effort. The outer-inner structure can make this approach computation intensive. Further research to reduce the computational cost will be conducted as part of our future work, as discussed in Chapter 7.

We used five numerical and engineering examples to demonstrate the applicability of the proposed approach. For most of these examples, robust Pareto designs are dominated by (or are interior to) the nominal Pareto designs, as expected. Monte-Carlo simulations have been used to verify the robust Pareto designs. The variations of objective and constraint functions for those robust designs are still within the acceptable ranges.

The computer platform used in this dissertation is a Dell Optiplex GX620 (3.4GHz Pentium4 CPU with 2GB of RAM). For most of test examples in this section, the computation time for a nominal optimization using MOGA is usually about 10 minutes. For UUV-Payload example, it takes about 20 minutes to finish one nominal run. It usually takes less than 20 hours for each of the other test examples and no more than 36 hours for UUV-Payload to complete a single run of the robust optimization method. Note that the computational time for robust optimization is significantly longer than nominal optimization because of numerous evaluations in the inner problem.

In the next chapter, we will extend the robust optimization approach described in this chapter to a robust method for MDO problems.

CHAPTER 4: PERFORMANCE AND COLLABORATIVE ROBUSTNESS IN MULTI-DISCIPLINARY DESIGN OPTIMIZATION

4.1 INTRODUCTION

Engineering design optimization methods involving "complex" systems often fall under MDO [Sobieszczanski-Sobieski and Balling, 1996], [Sobieszczanski-Sobieski and Haftka, 1997]. The word complex refers to a system whose analysis involves multiple interacting subsystems or disciplines.

Typical MDO approaches can be classified into two categories: All-at-once and multilevel approaches, e.g., [Sobieszczanski-Sobieski and Balling, 1996]. In an all-at-once approach, every discipline works as an analyzer while all analyzers work together to compute objective and constraint function values for a centralized optimization problem (See Section 2.2). In contrast, in a multilevel approach, the overall analysis and optimization problem is decentralized into multiple interacting and disciplinary subproblems and the optimization is performed in each subproblem while they all work together in concert to obtain the solution to the MDO problem. The multilevel (or decentralized) MDO approaches are abundant and have been applied to examples in air- and space-crafts, automobiles, and other engineering design problems, e.g., [Sobieski-Sobieszczanksi, 1988], [Renaud and Gabriele, 1993], [Braun, 1996], [Seller et al., 1996], [Sobieszczanski-Sobieski et al., 1998], [Sobieski and Kroo 2000], [Kodiyalam and Sobieszczanski-Sobieski, 2000 and 2001], and [Kim, 2001], among others. Also, large-scale stochastic optimization problems can be decomposed by using Dantzig-Wolfe

method [Dantzig and Wolfe, 1961], Benders decomposition [Benders, 1962], or Lagrangian decomposition [Conejo et al., 2006] when the functions in each optimization subproblem are explicit in linear, nonlinear or mixed-integer forms.

In contrast to single-objective optimization, very few papers are reported in multiobjective MDO and they usually are based on the weighted sum or compromised DSP approaches, e.g., [Tapetta and Renaud, 1997], [Kalsi et al., 2001], and [McAllister et al., 2000]. More importantly, there often exist uncontrollable variations or uncertainties in parameters of an MDO problem. These uncertainties may exist not only in each discipline but also propagate across disciplines due to couplings and hence methods for handling uncertainty within and across disciplines have become quite important [Du and Chen, 2000(a) and 2002], [Gu et al., 2000], [Gu and Renaud, 2002], [Gu et al., 2006]. However, even though there are many reported applications for robust MDO approaches, those MDO methods are essentially developed for single-objective robust optimization problems that have continuous objective/constraint functions, e.g., [Koch et al., 1999], [Gu et al., 2000], [Gu and Renaud, 2002], [Gu et al., 2006] or when input probability distributions are known, e.g., [Mavris et al., 1999], [Chen and Lewis, 1999], [Sues et al., 2001], [McAllister and Simpson, 2003], [Du and Chen, 2002, 2004, and 2005], and [Liu et al., 2006]. Although the approaches proposed, i.e., [McAllister and Simpson, 2003] and [Kalsi et al., 2001], can handle robust design problems with multiple objective functions, they can only apply to robust MDO problems with no coupling or one-way coupling (e.g., only upstream linking parameters, from the follower to leader discipline, have uncontrollable variations). The literature is particularly short in handling uncertainty for fully coupled multiobjective multilevel MDO problems with interdisciplinary

uncertainty propagation or when simulations in each discipline are considered as a black box; that is where this approach is making its contributions. In this regard, we present an approach called Multiobjective Collaborative Robust Optimization (McRO) that can find robust solutions for multiobjective multilevel MDO problems in which uncontrollable variability happens not only in the parameters within disciplines but also in couplings across disciplines. The forward mapping and thus the robustness index in the all-at-once format developed in Chapter 3 for performance robust optimization will be extended and used to measure an additional collaborative robustness to address the uncertainty propagated across disciplines in this chapter.

Section 4.2 describes the details of the McRO approach. Three examples, the numerical example, Speed Reducer design and UUV-Payload design, are used to demonstrate the applicability of the proposed approach in Section 4.3. Concluding remarks are given in Section 4.4. Definitions and terminologies used in this section are given in Chapter 2.

The significant portion of this chapter, other than the UUV-Payload design example, was also presented in Li and Azarm, [2007].

4.2 MULTIOBJECTIVE COLLABORATIVE ROBUST OPTIMIZATION (McRO)

We present here a Multiobjective collaborative Robust Optimization (McRO) approach that can be used for the solution of multiobjective multilevel MDO problems that have mixed continuous-discrete parameters which have interval uncertainty, extended from the forward mapping and the robustness index developed in Chapter 3. For these MDO problems, since there are full couplings across different disciplines, the

variation in one discipline might affect the performance of other disciplines. By full couplings, we mean there is a two-way connection between any two disciplines. In this regard, not only do we account for variations in each discipline but also have to handle propagation of uncertainty across different disciplines.

In this section, we first present the formulation of McRO. After that, we describe the technique used for handling the propagation of uncertainty, based on the robustness index. Then we present the McRO approach and implementation.

4.2.1 McRO Formulation

Recall Figure 2.5 whereby the formulation of a coupled multiobjective MDO problem with two disciplines is given. This formulation can be converted for the case of two disciplines (or subsystems) to a bi-level Collaborative Optimization (CO) model [Braun, 1996] as shown in Figure 4.1. Essentially, the coupling variables $\boldsymbol{y}$ are decoupled and replaced by adding corresponding target variables $\boldsymbol{t}$ and an additional constraint in each subsystem optimization problem. This additional constraint ensures that the value of all coupling variables is matched to their corresponding target variables in all subsystems, also called "Interdisciplinary Consistency Constraint" (ICC). ICC requires that essentially each subsystem should eventually have the same value for coupling variables. As an example, one of the coupling variables in the UUV-Payload design example (recall Section 3.6.5) is the payload weight. This coupling variable is calculated as an output by the Payload discipline and used by the UUV discipline as an input. Thus when the multi-disciplinary analysis is decomposed, we need to generate a target variable for this payload weight, as proposed in the literature by the CO approach [Braun, 1996].

As shown in Figure 4.1, the subsystem optimization problem in the top is for "System 0" and the subsystem optimization problems in the lower level are for subsystem i or SSi, i =1, 2. (As mentioned for simplicity and without loss of generality, we consider a fully coupled two-subsystem optimization problem representing two coupled disciplines. This approach can be easily extended to design problems with more than two disciplines.) Each subsystem optimization problem in the lower level represents a discipline and has its own set of design variables, parameters, and corresponding objective and constraint functions, represented by $\boldsymbol{x}_i$, $\boldsymbol{p}_i$, $\boldsymbol{f}_i$, $\boldsymbol{g}_i$, i=1, 2, respectively. System 0 has its own shared design variables and shared parameters as well as the objectives and constraints.

From this point on, by using the term "System" we mean the upper level block in Figure 4.1 and by "subsystem" we mean the lower level blocks in Figure 4.1. For System 0 the optimization problem in Eq. (4.1) is formulated with $\boldsymbol{x}_{sh}$ and $\boldsymbol{p}_{sh}$ as the shared design variables and shared uncertain parameters, respectively. Also, the target variables $\boldsymbol{t} = [\boldsymbol{t}_{12}, \boldsymbol{t}_{21}]$ are for the system targets and correspond to the interdisciplinary coupling variables: $\boldsymbol{y} = [\boldsymbol{y}_{12}, \boldsymbol{y}_{21}]$. For System 0, we have the optimization problem as shown in Eq. (4.1).

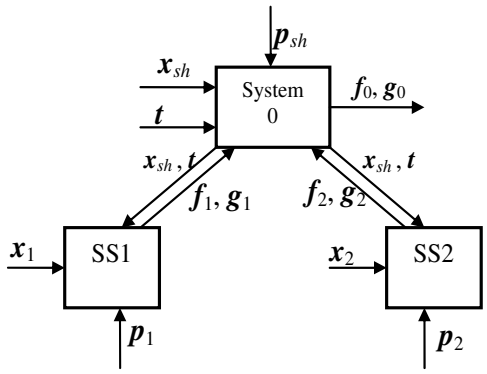


Figure 4.1 Collaborative optimization

$$\min_{X_0} \quad f_0(X_0, p_{sh}) \tag{4.1a}$$

s.t.

$$g_0(X_0, p_{sh}) \le 0 \tag{4.1b}$$

$$X_0 \equiv [x_{sh}, t] \tag{4.1c}$$

The objective functions in System 0 (Eq. (4.1a)) are minimized with respect to X_0, which includes both the shared design variable x_{sh} and target variables t. This formulation also considers the variability or interval uncertainty in p_{sh}.

Several previous formulations handle coupling variables in the CO models [Braun, 1996], [Alexandrov and Lewis, 2002], and [Aute and Azarm, 2006]. We present here two alternative formulations: formulation-1 and formulation-2. For both formulations, there is a target vector t_{ij}, corresponding to the vector y_{ij}, generated in System 0 and passed to SSi, i=1,2. Therefore, y_{ij}, the output variables from SSi to SSj, is a function of x_{sh}, x_i, p_i and t_{ji}. That is, the target value t_{ji}, instead of the coupling variables y_{ji}, is used for all the calculations in SSi as long as the value of all coupling variables are matched to their corresponding target values. The system design variables x_{sh} and target variables t are considered as fixed parameters in SSi.

In formulation-1, e.g., [Alexandrov and Lewis, 2002], [Sobieski and Kroo, 2000], [Aute and Azarm, 2006], the combined design variable vector in SSi, X_i in Eq. (4.2e), includes local design variables x_i and local target variables $t_{ji}^{\,i}$. The objectives in SSi are to minimize f_i, Eq. (4.2a), subject to local constraints g_i, Eq. (4.2b), and an additional Interdisciplinary Consistency Constraint (ICC), as in Eq. (4.2c). For instance in SS1, the ICC constraints are imposed by "$\|.\|_2$=0" type constraints to minimize the difference between y_{12} (output from SS1) and t_{12}, and also to minimize the difference between the

local variable t_{21}^{1} and the target t_{21}, while there are variations in p_i. In practice, the $\|.\|_2$ type constraints (Eq. (4.2c)) are satisfied within a small acceptable tolerance.

$$\min_{X_i} \quad f_i(x_{sh}, X_i, p_i, t_{ji}^{\,i}) \tag{4.2a}$$

s.t.

$$g_i(x_{sh}, X_i, p_i, t_{ji}^{\,i}) \le 0 \tag{4.2b}$$

$$\begin{aligned} \left\| y_{ij} - t_{ij} \right\|_2 &= 0 \\ \left\| t_{ji}^{\,i} - t_{ji} \right\|_2 &= 0 \end{aligned} \tag{4.2c}$$

$$y_{ij} = Y_i(x_{sh}, X_i, p_i, t_{ji}^{\,i}) \tag{4.2d}$$

$$X_i \equiv [x_i, t_{ji}^{\,i}], \quad i = 1,2 \tag{4.2e}$$

Using a more simplified alternative, formulation-2 [Haimes et al., 1990], it is not necessary to have a local target variables $t_{ji}^{\,i}$ in SSi; i.e., the system target variable t_{ji} will be used directly in SSi to calculate the coupling variables y_{ij} and local objective/constraint functions, as shown in Eq. (4.3):

$$\min_{x_i} \quad f_i(x_{sh}, x_i, p_i, t_{ji}) \tag{4.3a}$$

s.t.

$$g_i(x_{sh}, x_i, p_i, t_{ji}) \le 0 \tag{4.3b}$$

$$\left\| y_{ij} - t_{ij} \right\|_2 = 0 \tag{4.3c}$$

$$y_{ij} = Y_i(x_{sh}, x_i, p_i, t_{ji}), \quad i = 1,2 \tag{4.3d}$$

In formulation-2, x_i is the only vector of local design variables. ICC in Eq. (4.3c) forces the interdisciplinary coupling variables y_{ij} to match their corresponding system target values t_{ij}. In this way, the interdisciplinary consistency is met by a simpler formulation.

4.2.2 Interdisciplinary Propagation of Uncertainty

As shown in Figure 2.5, the propagation of uncertainty implies that outputs from one discipline are not only affected by the uncertainty from that discipline's parameters but also by the uncertainty from interdisciplinary coupling variables. In this section, we mainly focus on an approach that accounts for interdisciplinary propagation of uncertainty that is produced as a result of interdisciplinary coupling variables. The uncertainty that exists within the discipline can be handled by the performance robust optimization approach discussed in Chapter 3.

To handle the interdisciplinary propagation of uncertainty, two issues will have to be addressed. First, we need to decide how the uncertainty in each discipline should be represented. Second, we need to quantify this uncertainty. These two issues are discussed in the next two paragraphs.

In a CO framework, the coupling among the disciplines and satisfaction of the ICC are handled by matching the interdisciplinary coupling variables $\boldsymbol{y}$ to the corresponding target variables $\boldsymbol{t}$. As shown in Eqs. (4.2) and (4.3), when there is no uncertainty ICC forces all interdisciplinary $\boldsymbol{y}$ components to converge to a single value of their corresponding system target $\boldsymbol{t}$. However, with the uncertainty introduced in coupling variables due to uncontrollable parameters (coupling variables are another type of outputs of the subsystem's analyzer, in addition to the local objectives/constraints), ICC can not be satisfied. The variation in the interdisciplinary coupling variables $\boldsymbol{y}_{ij}$ leads to a range and not just a single value. Moreover, this variation in $\boldsymbol{y}_{ij}$ is propagated to subsystem SSj and may exacerbate the variation in $\boldsymbol{y}_{ji}$. Since in a CO framework the coupling variables $\boldsymbol{y}$ are decoupled and replaced by the target variables $\boldsymbol{t}$ (see Eqs. (4.2d) or (4.3d)), in the

same manner the variation from the coupling variables $\boldsymbol{y}$ should also be decoupled and represented by a variation in the target variables $\boldsymbol{t}$. In other words, to replace the variation in the coupling variables $\boldsymbol{y}$, a presumed variation range in the target variables $\boldsymbol{t}$ should be achieved. In both formulation-1, Eq. (4.2), and formulation-2, Eq. (4.3), the variation in the uncontrollable local parameters $\boldsymbol{p}_i$ leads to the variation in $\boldsymbol{y}_{ij}$ (see Eqs. (4.2d) or (4.3d)) even if we consider $\boldsymbol{t}_{ji}$ to be deterministic for the time being. Therefore, with the variation in $\boldsymbol{p}_i$, $\boldsymbol{y}_{ij}$ cannot match a single value of $\boldsymbol{t}_{ij}$ and moreover the variation in $\boldsymbol{y}_{ij}$ will propagate from SSi to SSj (see Figure 2.5). Similarly, the variation in $\boldsymbol{y}_{ji}$ will also propagate back to SSi and may aggravate the variation of $\boldsymbol{y}_{ij}$. Since $\boldsymbol{t}_{ji}$ is the system target of interdisciplinary variable $\boldsymbol{y}_{ji}$, it has to reflect the variation in $\boldsymbol{y}_{ji}$. That is, $\boldsymbol{t}_{ji}$ can not be a single deterministic value either: It should have a variation range. Therefore, the variation in both $\boldsymbol{p}_i$ and $\boldsymbol{t}_{ji}$ leads to the variation in $\boldsymbol{y}_{ij}$. For instance, in SS1, $\boldsymbol{y}_{12} = \boldsymbol{Y}_1(\boldsymbol{x}_{sh}, \boldsymbol{x}_1, \boldsymbol{p}_1, \boldsymbol{t}_{21})$. Similarly, in SS2, $\boldsymbol{y}_{21} = \boldsymbol{Y}_2(\boldsymbol{x}_{sh}, \boldsymbol{x}_2, \boldsymbol{p}_2, \boldsymbol{t}_{12})$. Suppose now that $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ have variations. As a result, both $\boldsymbol{y}_{12}$ and $\boldsymbol{y}_{21}$ will have variations due to variations in $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$, respectively. Moreover, $\boldsymbol{t}_{21}$ should also have variations due to $\boldsymbol{y}_{21}$, otherwise it cannot work as a replacement of $\boldsymbol{y}_{21}$ (see Eqs. (4.2c) and (4.3c)). Thus, the variations from both $\boldsymbol{p}_1$ and $\boldsymbol{t}_{21}$ lead to the variations in $\boldsymbol{y}_{12}$. In this way, the variations in the interdisciplinary coupling variables $\boldsymbol{y}$ can be represented by and transferred to the variations in the target variables $\boldsymbol{t}$.

The next issue to be addressed is how to quantify the variations in target variables $\boldsymbol{t}$. A presumed tolerance region is used to quantify the interval uncertainty in $\boldsymbol{t}$. Each component in the vector of target variables $\boldsymbol{t}$ is assumed to have a nominal value and a tolerance range around the nominal. The interdisciplinary coupling variables $\boldsymbol{y}_{ij}$ is considered as part of the output vector from SSi, together with $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$. Therefore, with

the presence of uncertainty in inputs to SSi (e.g., $\boldsymbol{t}_{ji}$ and $\boldsymbol{p}_i$), all the outputs from SSi: $\boldsymbol{y}_{ij}$, $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$ have variations. However, as long as the value of $\boldsymbol{y}_{ij}$ stays within tolerance region of $\boldsymbol{t}_{ij}$, it is acceptable for SSj to use $\boldsymbol{t}_{ij}$ as a vector of input parameters. That is, *the tolerance region of target variables* $\boldsymbol{t}_{ij}$ *is not only an input variation range to* SSj, *but is also an acceptable variation range for coupling variables* $\boldsymbol{y}_{ij}$ *in* SSi. The same applies to SSj: As long as the variation in $\boldsymbol{y}_{ji}$ stays within the tolerance region of $\boldsymbol{t}_{ji}$, SSi can use the target vector $\boldsymbol{t}_{ji}$ in Eqs. (4.2) or (4.3). The target variable $\boldsymbol{t}$ (with the corresponding nominal value and tolerance region) provides a cushion to absorb the variation in $\boldsymbol{y}$. In this way, the propagation of uncertainty in the coupling variables is replaced by target variables with a tolerance region. A design in McRO approach is defined by shared and design variables, uncertain parameters and target variables, and outputs from each subsystem. In this regard, a design is: 1) *performance-wise robust* if the variation in objective functions of a feasible design is within an acceptable range and this feasible design remains feasible when parameters and targets vary, and 2) *collaboratively robust* if the variation in coupling variables stays within an tolerance region of targets when parameters and targets vary.

4.2.3 McRO Approach

The all-at-once performance robust optimization formulation described in Chapter 3 is extended here to solve the McRO formulation in Eqs. (4.2) or (4.3). Here we assume that: i) the range of uncontrollable parameters $\boldsymbol{p}_{sh}$ and $\boldsymbol{p}_i$ are known; ii) the AOVR for each objective function in System 0 and all SSi are given; and iii) the tolerance region for target variables $\boldsymbol{t}$ is known for all subsystem disciplines. Based on these assumptions, a design is considered to be robustly optimal if it satisfies the following two conditions:

1) In each subsystem optimization problem SSi, the subsystem objective values $\boldsymbol{f}_i$ are optimized with their variation being within their AOVR and the subsystem constraint values $\boldsymbol{g}_i$ remain within their ACVRs. Additionally, the variation of the subsystem interdisciplinary outputs $\boldsymbol{y}_{ij}$ is within the tolerance region of the target variables $\boldsymbol{t}_{ij}$.
2) In System 0, the objective values are optimized with their variations being within the system AOVR. The system constraints also remain feasible, i.e., within their ACVR. However, as done in this dissertation, if we assume that the objective functions in System 0 are dependent on the objective values $\boldsymbol{f}_i$, i =1, 2, of subsystems; then the AOVR in System 0 is also dependent on and can be calculated from AOVRs of the subsystems. However, in general, for the proposed McRO approach this last requirement is not necessary to be imposed.

In the proposed formulations, the nominal values of system parameters $\boldsymbol{p}_{sh}$ and subsystem parameters $\boldsymbol{p}_i$ are $\boldsymbol{p}_{sh,0}$, and $\boldsymbol{p}_{i,0}$, respectively. $\boldsymbol{t}_0$ represents the nominal value of interdisciplinary target variables $\boldsymbol{t}$. In System 0, as given in Eq. (4.4a), the system objective functions are minimized with respect to shared design variables $\boldsymbol{x}_{sh}$ and the nominal value of interdisciplinary target variables $\boldsymbol{t}_0$, with all parameters $\boldsymbol{p} = [\boldsymbol{p}_{sh}, \boldsymbol{p}_1, \boldsymbol{p}_2]$ and the interdisciplinary target variable $\boldsymbol{t} = [\boldsymbol{t}_{12}, \boldsymbol{t}_{21}]$ fixed at their nominal values $\boldsymbol{p}_0$ and $\boldsymbol{t}_0$, respectively. The system constraints are given in Eq. (4.4b). The system robustness constraint, Eq. (4.4c) (recall Eq. (3.9) in Section 3.5), implies that the combined performance and collaborative robustness index: $\eta_{perf,c}$, to be less than or equal to 1 for a robust design.

$$\min_{\boldsymbol{X}_0} \quad \boldsymbol{f}_0(\boldsymbol{X}_0, \boldsymbol{p}_{sh,0}) \tag{4.4a}$$

s.t.

$$\boldsymbol{g}_0(\boldsymbol{X}_0, \boldsymbol{p}_{sh,0}) \le 0 \tag{4.4b}$$

$$\eta_{perf,c} - 1 \le 0 \tag{4.4c}$$

$$\boldsymbol{X}_0 \equiv [\boldsymbol{x}_{sh}, \boldsymbol{t}_0] \text{ and } \eta_{perf,c} = \max\{\eta_{perf}, \eta_c\} \tag{4.4d}$$

In this formulation, $\eta_{perf,c}$ is the maximum value between the performance robustness index η_{perf} and the collaborative robustness index η_c, which are calculated by Eq. (4.5) and Eq. (4.6) respectively, as follows:

$$\begin{aligned}\eta_{perf} = \max_{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}} \{ &\left\|\Delta \boldsymbol{f}_0(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}})\right\|_\infty, \left\|\Delta \boldsymbol{f}_1(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{21})\right\|_\infty, \\ &\left\|\Delta \boldsymbol{f}_2(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{12})\right\|_\infty, \left\|\Delta \boldsymbol{g}_0(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}})\right\|_\infty, \\ &\left\|\Delta \boldsymbol{g}_1(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{21})\right\|_\infty, \left\|\Delta \boldsymbol{g}_2(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{12})\right\|_\infty \}\end{aligned} \tag{4.5a}$$

$\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{t}}$ present the possible parameter and target variations over their tolerance regions, respectively. The variation of the each component in the objective and constraint vector in System 0 and SS*i* are defined by:

$$\Delta f_{0,m} = \frac{f_{0,m}(\boldsymbol{x}_{sh}, \tilde{\boldsymbol{p}}_{sh}, \tilde{\boldsymbol{t}}) - f_{0,m}(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh,0}, \boldsymbol{t}_0)}{\Delta f_{0,0,m}} \tag{4.5b}$$

$$\Delta g_{0,l} = \begin{cases} \dfrac{g_{0,l}(\boldsymbol{x}_{sh}, \tilde{\boldsymbol{p}}_{sh}, \tilde{\boldsymbol{t}}) - g_{0,l}(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh,0}, \boldsymbol{t}_0)}{\left|g_{0,l}(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh,0}, \boldsymbol{t}_0)\right|} \\ \qquad \text{if } g_{0,l}(\boldsymbol{x}_{sh}, \tilde{\boldsymbol{p}}_{sh}, \tilde{\boldsymbol{t}}) \ge g_{0,l}(\boldsymbol{x}_{sh}, \boldsymbol{p}_{sh,0}, \boldsymbol{t}_0) \\ 0 \qquad \text{otherwise} \end{cases} \tag{4.5c}$$

$$\Delta f_{i,m} = \frac{f_{i,m}(\boldsymbol{x}_i, \tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{ji}) - f_{i,m}(\boldsymbol{x}_i, \boldsymbol{p}_0, \boldsymbol{t}_{ji,0})}{\Delta f_{i,0,m}}, i = 1, 2 \tag{4.5d}$$

$$\Delta g_{i,l} = \begin{cases} \dfrac{g_{i,l}(\boldsymbol{x}_i, \tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{ji}) - g_{i,l}(\boldsymbol{x}_i, \boldsymbol{p}_0, \boldsymbol{t}_{ji,0})}{\left| g_{i,l}(\boldsymbol{x}_i, \boldsymbol{p}_0, \boldsymbol{t}_{ji,0}) \right|} & \\ & \text{if } g_{i,l}(\boldsymbol{x}_i, \tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{ji}) \geq g_{i,l}(\boldsymbol{x}_i, \boldsymbol{p}_0, \boldsymbol{t}_{ji,0}) \\ 0 & \text{otherwise} \end{cases} \quad i = 1, 2 \tag{4.5e}$$

$f_{0,m}$ and $f_{i,m}$ are the m-th objective function in System 0 and SSi, respectively. $\Delta f_{0,m}$ and $\Delta f_{i,m}$ are normalized with respect to $\Delta f_{0,0,m}$ and $\Delta f_{i,0,m}$, the AOVR for the m-th objective function in System 0 and SSi, respectively. The constraints $g_{0,l}$ and $g_{i,l}$ are the l-th constraint function in System 0 and SSi, respectively. Notice that in Eq. (4.5), the target variables $\boldsymbol{t}_{ji}$ are passed to SSi as uncontrollable parameters to calculate $\boldsymbol{y}_{ij}$, $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$. Therefore the variation in SSi comes from two sources: $\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{t}}_{ij}$.

The collaborative robustness index η_c is defined as the maximum $\|.\|_\infty$ distance from the coupling variable $\boldsymbol{y}_{ij}$ to the nominal target variable value $\boldsymbol{t}_{ij,0}$, as shown in Eq. (4.6). As long as this distance is within the tolerance region $\Delta \boldsymbol{t}_{ij,0}$, the variation within the coupling variables is acceptable.

$$\eta_c = \max_{[\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}]} \left\| C_i \right\|_\infty, \; i = 1, \; 2 \tag{4.6a}$$

where C_i is the difference between $\boldsymbol{y}_{ij}$ and the nominal value of target $\boldsymbol{t}_{ij}$ and normalized by $\Delta \boldsymbol{t}_{ij,0}$, as defined in (4.6b):

$$C_i = \frac{y_{ij}(\boldsymbol{x}, \tilde{\boldsymbol{p}}, \tilde{\boldsymbol{t}}_{ji}) - \boldsymbol{t}_{ij,0}}{\Delta \boldsymbol{t}_{ij,0}}, \; j \neq i \tag{4.6b}$$

By adding a robust constraint, $\eta_{perf,c} - 1 \leq 0$, (Eq. (4.4c)), into the System 0, the largest deviation from $\boldsymbol{y}$ to the nominal value of $\boldsymbol{t}$ should not exceed the tolerance region of target value $\boldsymbol{t}$ and the largest deviation from nominal of objective and constraints stays within an acceptable range. The design that satisfies this constraint is both performance and collaboratively robust. In this way, the uncertainty in the coupling variables $\boldsymbol{y}$ is

represented and quantified. This robust constraint is verified for any feasible system design $\boldsymbol{x}_{sh}$ together with its corresponding subsystem designs after obtaining all optimal subsystem designs. (The approach for calculating the robust constraint is discussed in the next section.) In this way, the two conditions mentioned at the beginning of this section are satisfied and the subsystem's solutions are guaranteed to be multiobjectively and feasibly robust.

In the subsystem level SSi, Eq. (4.7) is used to find the optimal solutions of the subsystem optimization problem in Eq. (4.3). (The same approach can be applied to Eq. (4.2) too.) The subsystem objective functions are minimized with respect to subsystem design variables $\boldsymbol{x}_i$. The shared design variables $\boldsymbol{x}_{sh}$ and the nominal value of interdisciplinary target variables $\boldsymbol{t}_0$ are fixed in Eq. (4.7). $\boldsymbol{y}_{ij,0}$, $\boldsymbol{t}_{ij,0}$ and $\boldsymbol{t}_{ji,0}$ are the nominal value for coupling and target variable $\boldsymbol{y}_{ij}$, $\boldsymbol{t}_{ij}$ and $\boldsymbol{t}_{ji}$, respectively.

$$\min_{\boldsymbol{x}_i} \quad \boldsymbol{f}_i(\boldsymbol{x}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_{i,0}, \boldsymbol{t}_{ji,0}) \tag{4.7a}$$

s.t.

$$\boldsymbol{g}_i(\boldsymbol{x}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_{i,0}, \boldsymbol{t}_{ji,0}) \le 0 \tag{4.7b}$$

$$\left\| \boldsymbol{y}_{ij,0} - \boldsymbol{t}_{ij,0} \right\|_2 = 0 \tag{4.7c}$$

$$\boldsymbol{y}_{ij,0} = \boldsymbol{Y}_i(\boldsymbol{x}_{sh}, \boldsymbol{x}_i, \boldsymbol{p}_{i,0}, \boldsymbol{t}_{ji,0}) \quad i = 1,2 \tag{4.7d}$$

An implementation of the McRO approach is discussed in the next section.

4.2.4 McRO Implementation

We use MOGA as the optimizer for system and subsystem subproblems and GA as the optimizer to calculate the robustness index as in Chapter 3. MOGA and GA are used because there might exist mixed continuous-discrete design variables and parameters in

these subproblems and we would like to find the global optimal solutions for the robustness index.

The steps in the implementation of the McRO approach are as follows:

1. At System 0, MOGA generates a set of design alternatives including a set of $\boldsymbol{x}_{sh}$ and $\boldsymbol{t}_0$ (nominal value of $\boldsymbol{t}$) values for individual designs in the initial generation.
2. For each individual in System 0, the values of $\boldsymbol{x}_{sh}$ and $\boldsymbol{t}_0$ are passed to each SSi as fixed parameters.
3. MOGA solves the SSi optimization problem, Eq. (4.7) with uncontrollable parameters $\boldsymbol{p}_{sh}$ and $\boldsymbol{p}_i$ fixed to their nominal value.
4. The objective functions $\boldsymbol{f}_i$ and constraint functions $\boldsymbol{g}_i$ and their corresponding local design variables $\boldsymbol{x}_i$ values are passed from SSi back to the System 0. At this point, since MOGA is a population-based approach, each SSi has a Pareto solution set corresponding to each $\boldsymbol{x}_{sh}$ and $\boldsymbol{t}_0$. As a result it is necessary to map multiple Pareto points from each SSi to an individual point at the system level. Here, we have adopted a simple strategy for selecting a solution from each subsystem Pareto set [Aute and Azarm, 2006]. In this strategy, the solution that has the best objective function value f_1 in each subsystem is chosen and its value is passed to the system level. However, other strategies could also be used.
5. MOGA solves System 0's optimization problem, Eq. (4.4), calculating the objective and constraint values. For each design $\boldsymbol{x}_{sh}$, the robustness index at the system level is calculated using a GA. The system level objective and constraint values are used to assign a fitness value to the population at the system level.

6. A new population of $\boldsymbol{x}_{sh}$ and $\boldsymbol{t}_0$ is generated based on the genetic operators of selection, crossover and mutation.

Steps 2 to 6 are repeated and the procedure stops when a pre-specified maximum number iterations is achieved or other stopping criteria are satisfied. The GA parameters used in the MOGA and GA are shown in Table 4.1.

Table 4.1 GA parameters used in McRO

GA Parameters	Values
popsize – system level	100
popsize – subsystem level	80
Crossover probability	0.9
Mutation probability	0.1
Elite number	40% of pop. size
Max. # of generations – system level	100
Max. # of generations – subsystem level	80
Number of bits (for each continuous design variable)	16
Number of bits (for integer design variable)	4

Next we demonstrate the applicability of the proposed approach using a numerical and two engineering examples.

4.3 EXAMPLES AND RESULTS

In this section, a numerical example and two engineering examples, described in Chapter 3, are used to demonstrate the applicability of the McRO approach. In this chapter, each of these three examples has two objective functions in each subsystem and has mixed continuous-discrete design variables and/or parameter variations. Moreover, two subproblems in each example are fully coupled, which are used to demonstrate the applicability of the proposed McRO in solving the uncertainty propagation.

4.3.1 Numerical Example

The bi-objective numerical example shown in Section 3.6.1 is developed here as a bi-level two-objective MDO problem with two fully coupled disciplines. The all-at-once formulation in Eq. (3.11) and Eq. (3.12) is converted into a system-level subproblem with two subsystem-level coupled subproblems. Between these two subsystems SSi, i=1 and 2, there are the two coupling variables $\boldsymbol{y} = [y_1, y_2]$. In each subproblem, there are two objective functions and one constraint. The collaborative optimization formulation for this problem is as follows.

In System 0, $\boldsymbol{x}_{sh}$ includes only one design variable, x_1; and the vector of target variables $\boldsymbol{t}$ includes two target variables t_1 and t_2, corresponding to the two coupling variable y_1 and y_2. Thus $\boldsymbol{x}_{sh} = [x_1]$, $\boldsymbol{t}_{12} = [t_1]$ and $\boldsymbol{t}_{21} = [t_2]$. The bi-objective optimization problem in System 0 is given in Eq. (4.8):

$$\begin{aligned} &\min_{\boldsymbol{X}_0} \quad f_{0,1} = f_{1,1} + f_{2,1} = y_1 + {x_2}^2 + x_3 + e^{-y_2} \\ &\min_{\boldsymbol{X}_0} \quad f_{0,2} = f_{1,2} + f_{2,2} = -{y_1}^3/10^4 + 150 \\ &\qquad\qquad + \sqrt{x_2 + {y_2}^2}/10^4 + x_1 \end{aligned} \tag{4.8}$$

where

$$\begin{aligned} &\boldsymbol{X}_0 \equiv [x_1, t_1, t_2] \\ &-10 \le x_1 \le 10 \\ &\quad 8 \le t_1 \le 120 \\ &-10 \le t_2 \le 10 \end{aligned}$$

SS1 has the local design variable: $\boldsymbol{x}_1 = [x_2]$, and the local copy of the target variable for t_2, i.e., t_2^1 . The formulation of the bi-objective optimization problem for SS1 is given in Eq. (4.9):

$$
\begin{aligned}
&\min_{\boldsymbol{X}_1} \quad f_{1,1} = x_2^{\,2} + y_1 \\
&\min_{\boldsymbol{X}_1} \quad f_{1,2} = \sqrt{x_2} - y_1^{\,3}/10^{\,4} + 150 \\
&\text{s.t.} \quad g_{1,1} = 8 - y_1 \le 0 \\
&\qquad \left\| y_1 - t_1 \right\| \le \varepsilon_1 \\
&\qquad \left\| t_2^1 - t_2 \right\| \le \varepsilon_2 \\
&\text{where} \\
&\qquad y_1 = x_1^{\,2} + x_2 - 0.2 t_2^1 \\
&\qquad \boldsymbol{X}_1 \equiv [x_2, t_2^1] \\
&\qquad 0 \le x_2 \le 10 \\
&\qquad -10 \le t_2^1 \le 11 \\
&\qquad \varepsilon_i = 10^{-3} t_i,\ i = 1,2
\end{aligned}
\tag{4.9}
$$

SS2 has the local design variable $\boldsymbol{x}_2 = [x_3]$ and the local copy of the target variable for t_1, i.e., t_1^2 . The formulation of the bi-objective optimization problem for SS2 is defined in Eq. (4.10):

$$
\begin{aligned}
&\min_{\boldsymbol{X}_2} \quad f_{2,1} = x_3 + e^{-y_2} \\
&\min_{\boldsymbol{X}_2} \quad f_{2,2} = x_1 + y_2^{\,2}/10^{\,4} \\
&\text{s.t.} \quad g_{2,1} = y_2 - 10 \le 0 \\
&\qquad \left\| y_2 - t_2 \right\| \le \varepsilon_2 \\
&\qquad \left\| t_1^2 - t_1 \right\| \le \varepsilon_1 \\
&\text{where} \\
&\qquad y_2 = x_1 + x_3 + \sqrt{t_1^2} \\
&\qquad \boldsymbol{X}_2 \equiv [x_3, t_1^2] \\
&\qquad 0 \le x_3 \le 10 \\
&\qquad 8 \le t_1^2 \le 120 \\
&\qquad \varepsilon_i = 10^{-3} t_i,\ i = 1,2
\end{aligned}
\tag{4.10}
$$

The coupling variable y_1 and y_2 are defined in Eq. (4.11):

$$
\begin{aligned}
y_1 &= \boldsymbol{Y}_1(\boldsymbol{x}_{sh}, \boldsymbol{x}_1, t_2^1) = x_1^{\,2} + x_2 - 0.2 t_2^1 \\
y_2 &= \boldsymbol{Y}_2(\boldsymbol{x}_{sh}, \boldsymbol{x}_2, t_1^2) = x_1 + x_3 + \sqrt{t_1^2}
\end{aligned}
\tag{4.11}
$$

For this example, we used the formulation-1, as in Eq. (4.2). For the nominal case (without uncertainty), the obtained Pareto solutions from the Multiobjective

Collaborative Optimization (MCO) approach are shown in Figure 4.2 ("Nominal: MCO"). To verify the Pareto optimal results obtained from the collaborative formulations, Eqs. (4.9) and (4.10), the optimal Pareto solutions for the original all-at-once problem, Eq. (3.11) in Chapter 3, are also shown in Figure 4.2.

Again, the variations in Δx_2 and Δx_3 are assumed to be within ±6% from nominal and Δx_1 is discretized to 12 possible values, ±1%, ±2%, ±3%, ±4%, ±5%, or ±6% from the nominal. The AOVR for each of the two components in the vector of bi-objective functions f_1 and f_2 in SS1 and SS2 are both ±5 units from their nominal. The tolerance region for the target variables Δt_1 and Δt_2 are also ±2 units from their nominal. Since the objective values in System 0 are only dependent on the local objectives from SS1 and SS2, the AOVR in System 0 is also only dependent on and, for this example, is the sum of AOVRs of subsystems. Note that there is no constraint in System 0, Eq. (4.8). As a result, for any design $\boldsymbol{x}_1$, if a corresponding local design: x_2 and x_3, is robust, then $\boldsymbol{x}_1$ is robust.

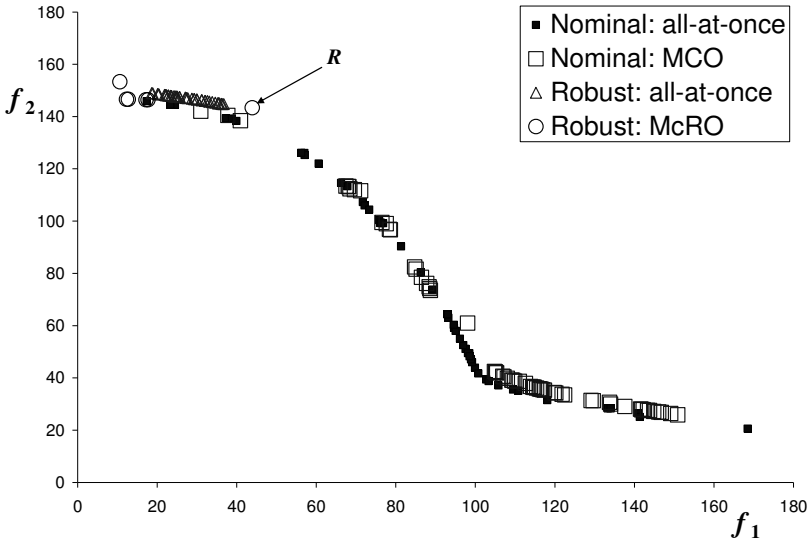


Figure 4.2 Obtained nominal and robust optimal solutions from McRO

The robust solutions for the above mentioned settings are shown in Figure 4.2. The robust solutions as in the all-at-once format in Chapter 3, with the same tolerance region and with each of $\Delta f_{0,1}$ and $\Delta f_{0,2}$ being 10 units from its nominal, have also been shown here. As shown, the nominal Pareto solutions from the MCO approach overlap with the nominal solutions from the all-at-once performance robust optimization approach. Moreover, while we need to also set an acceptable range for the coupling variables, the robust Pareto solutions from the McRO approach are comparable, though a little different from those obtained from the all-at-once approach. As a demonstration, the robustness of one of the robust optimal solutions shown in Figure 4.2, *R*, is verified by the Monte-Carlo simulation with 10,000 sample points (only a subset of them are shown in Figure 4.3 for visualization). The variation in objective functions in both SS1 and SS2 are shown in Figure 4.3.

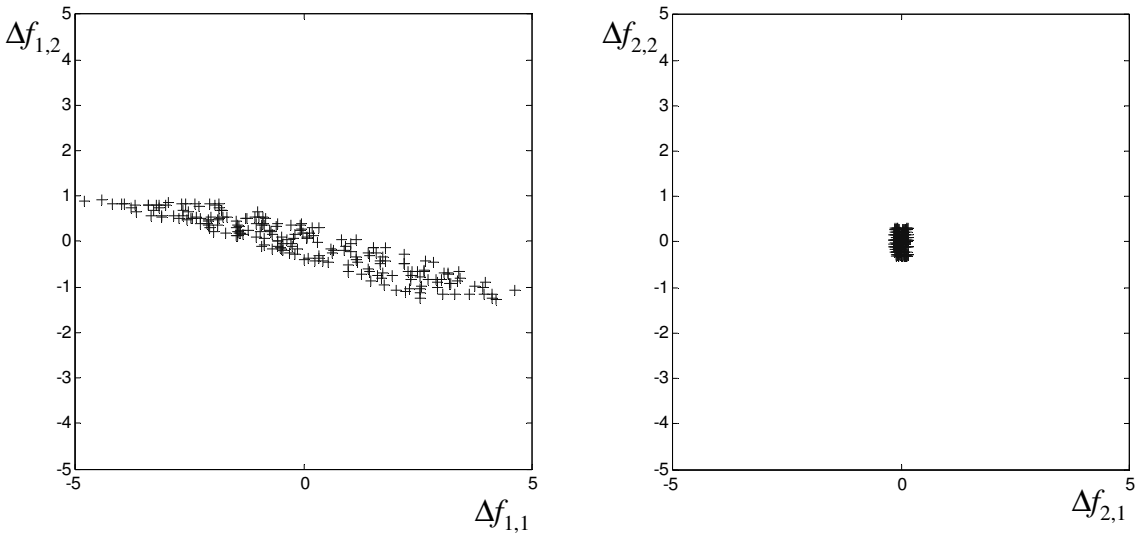


Figure 4.3 Verification of objective robustness for design *R* of Figure 4.2 in (a) SS1 and in (b) SS2

4.3.2 Speed Reducer Design

The second example is the design of a speed reducer described before in Section 3.6.3. Here, we modified the formulation to a two-objective optimization problem as shown in Eq. (4.12).

$$\begin{aligned}
\min_{x} \quad f_1 &= 0.7854\ x_1 x_2{}^2 \left(\frac{10\, x_3{}^2}{3} + 14.933\ x_3 - 43.0934\right) \\
&\quad - 1.508\ x_1 (x_6{}^2 + x_7{}^2) + 7.477\ (x_6{}^3 + x_7{}^3) \\
&\quad + 0.7854\ (x_4 x_6{}^2 + x_5 x_7{}^2) \\
\min_{x} \quad f_2 &= \max\ \{f_{12}, f_{22}\}
\end{aligned} \tag{4.12a}$$

$$\textit{where} \quad f_{12} = \frac{\sqrt{\left(\frac{745\ x_4}{x_2 x_3}\right)^2 + 1.69 \times 10^7}}{0.1 x_6^3} \quad \text{and} \quad f_{22} = \frac{\sqrt{\left(\frac{745\ x_5}{x_2 x_3}\right)^2 + 1.575\ x10^8}}{0.1 x_7^3}$$

$$\begin{aligned}
\text{subject to:} \quad & g_1 \equiv \frac{1}{x_1 x_2^2 x_3} - \frac{1}{27} \le 0 \quad ; \quad g_2 \equiv \frac{1}{x_1 x_2^2 x_3^2} - \frac{1}{397.5} \le 0 \\
& g_3 \equiv \frac{x_4^3}{x_2 x_3 x_6^4} - \frac{1}{1.93} \le 0 \quad ; \quad g_4 \equiv \frac{x_5^3}{x_2 x_3 x_7^4} - \frac{1}{1.93} \le 0 \\
& g_5 \equiv x_2 x_3 - 40 \le 0 \quad ; \quad g_6 \equiv \frac{x_1}{x_2} - 12 \le 0 \\
& g_7 \equiv 5 - \frac{x_1}{x_2} \le 0 \quad ; \quad g_8 \equiv 1.9 - x_4 + 1.5 x_6 \le 0 \\
& g_9 \equiv 1.9 - x_5 + 1.1 x_7 \le 0 \quad ; \quad g_{10} \equiv f_{12} - 1800 \le 0 \\
& g_{11} \equiv f_{22} - 1100 \le 0
\end{aligned} \tag{4.12b}$$

$$\begin{aligned}
&2.6 \le x_1 \le 3.6 \qquad 0.7 \le x_2 \le 0.8 \\
&17 \le x_3 \le 28 \qquad 7.3 \le x_4 \le 8.3 \\
&7.3 \le x_5 \le 8.3 \qquad 2.9 \le x_6 \le 3.9 \\
&5.0 \le x_7 \le 5.5
\end{aligned}$$

To decompose the problem, we use the formulation in Eq. (4.1) and Eq. (4.3). The entire system is decomposed into one system level subproblem: System 0, with two subsystem level subproblems: SS1 and SS2. These two subsystems are fully coupled by two coupling variables $y = [y_1, y_2]$, defined as in Eq. (4.13):

$$y_1 = 0.7854\ x_4 x_6^2$$
$$y_2 = 0.7854\ x_5 x_7^2 \tag{4.13}$$

In each discipline, there are two objective functions and several constraints. In System 0, $\boldsymbol{x}_{sh}$ includes three design variables $\boldsymbol{x}_{sh} = [x_1, x_2, x_3]$. The vector of target variables $\boldsymbol{t}$ includes two target variables t_1 and t_2, corresponding to two coupling variables y_1 and y_2, as shown in Eq. (4.14):

$$\begin{aligned}
&\min_{\boldsymbol{X}_0} \quad f_{0,1} = f_{1,1} + f_{2,1} \\
&\min_{\boldsymbol{X}_0} \quad f_{0,2} = \max\ \{f_{1,2}, f_{2,2}\} \\
&\text{s.t.} \\
&\qquad g_{0,1} = g_1 \equiv \frac{1}{x_1 x_2^2 x_3} - \frac{1}{27} \le 0 \\
&\qquad g_{0,2} = g_2 \equiv \frac{1}{x_1 x_2^2 x_3^2} - \frac{1}{397.5} \le 0 \\
&\qquad g_{0,3} = g_5 \equiv x_2 x_3 - 40 \le 0 \\
&\qquad g_{0,4} = g_6 \equiv \frac{x_1}{x_2} - 12 \le 0 \\
&\qquad g_{0,5} = g_7 \equiv 5 - \frac{x_1}{x_2} \le 0 \\
&\text{where} \\
&\qquad \boldsymbol{X}_0 \equiv [x_1, x_2, x_3, t_1, t_2] \\
&\qquad 2.6 \le x_1 \le 3.6 \quad 0.7 \le x_2 \le 0.8 \quad 17 \le x_3 \le 28 \\
&\qquad 49 \le t_1 \le 99 \quad 150 \le t_2 \le 180
\end{aligned} \tag{4.14}$$

The subsystem SS1 has the local design variable $\boldsymbol{x}_1 = [x_4, x_6]$. In this example, we do not have any local copy of the target variable for t_2. We used the formulation-2, as in Eq. (4.3): The target value t_2 is used directly in the SS1. The formulation of the bi-objective optimization problem for SS1 is defined in Eq. (4.15).

$$
\begin{aligned}
&\min_{\boldsymbol{X}_1} \quad f_{1,1} = 0.7854\ x_1 x_2{}^2 \left(\frac{10\, x_3{}^2}{3} + 14.933\ x_3 - 43.0934\right) \\
&\qquad\qquad - 1.508\ x_1 x_6{}^2 + 7.477\ x_6{}^3 + t_2 \\
&\min_{\boldsymbol{X}_1} \quad f_{1,2} = \frac{\sqrt{\left(\frac{745\ x_4}{x_2 x_3}\right)^2 + 1.69 \times 10^7}}{0.1 x_6^3} \\
&\text{s.t.} \\
&\qquad g_{1,1} = g_3 \equiv \frac{x_4^3}{x_2 x_3 x_6^4} - \frac{1}{1.93} \le 0 \\
&\qquad g_{1,2} = g_8 \equiv 1.9 - x_4 + 1.5 x_6 \le 0 \\
&\qquad g_{1,3} = g_{10} \equiv f_{1,2} - 1800 \le 0 \\
&\qquad \left\| y_1 - t_1 \right\| \le \varepsilon_1 \\
&\text{where} \\
&\qquad y_1 = 0.7854\ x_4 x_6{}^2 \\
&\qquad \boldsymbol{X}_1 \equiv [x_4, x_6] \\
&\qquad 7.3 \le x_4 \le 8.3 \qquad 2.9 \le x_6 \le 3.9 \\
&\qquad \varepsilon_1 = 10^{-3} t_1
\end{aligned}
\tag{4.15}
$$

The subsystem SS2 has the local design variable $\boldsymbol{x}_2 = [x_5, x_7]$. The formulation of the bi-objective optimization problem for SS2 is defined in Eq. (4.16):

$$
\begin{aligned}
&\min_{\boldsymbol{X}_2} \quad f_{2,1} = -1.508\ x_1 x_7{}^2 + 7.477\ x_7{}^3 + t_1 \\
&\min_{\boldsymbol{X}_2} \quad f_{2,2} = \frac{\sqrt{\left(\frac{745\ x_5}{x_2 x_3}\right)^2 + 1.575\ x10^8}}{0.1 x_7^3} \\
&\text{s.t.} \\
&\qquad g_{2,1} = g_4 \equiv \frac{x_5^3}{x_2 x_3 x_7^4} - \frac{1}{1.93} \le 0 \\
&\qquad g_{2,2} = g_9 \equiv 1.9 - x_5 + 1.1 x_7 \le 0 \\
&\qquad g_{2,3} = g_{11} \equiv f_{2,2} - 1100 \le 0 \\
&\qquad \left\| y_2 - t_2 \right\| \le \varepsilon_2 \\
&\text{where} \\
&\qquad y_1 = 0.7854\ x_5 x_7{}^2 \\
&\qquad \boldsymbol{X}_1 \equiv [x_5, x_7] \\
&\qquad 7.3 \le x_4 \le 8.3 \qquad 5.0 \le x_6 \le 5.5 \\
&\qquad \varepsilon_2 = 10^{-3} t_2
\end{aligned}
\tag{4.16}
$$

For the nominal case (i.e., without uncertainty), the obtained Pareto solutions are shown in Figure 4.4 by squares, shown as “Nominal: MCO”.

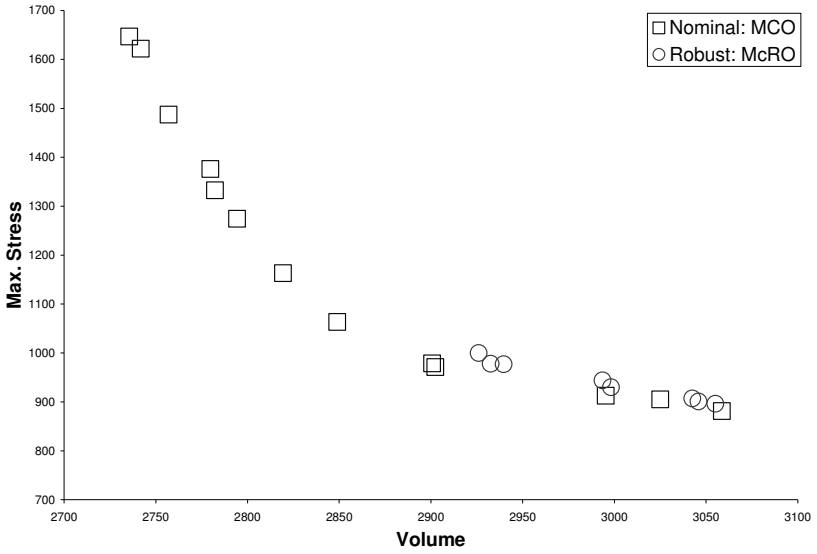


Figure 4.4 Nominal and robust designs using MCO and McRO, respectively

In robust optimization, the variation $[\Delta x_6, \Delta x_7]$ in the design variables $[x_6, x_7]$ is assumed to be between ±0.1. The AOVR for $[f_1\ f_2]$ in both SS1 and SS2 are: $[\Delta f_{i,0,1}, \Delta f_{i,0,2}]$=[60, 75]. The tolerance regions Δt_1 and Δt_2 of targets, are continuous and within ±5 and ±7 from nominal, respectively. The AOVR in $f_{0,1}$ of System 0 is the sum of local AOVRs from SSi, i=1, 2. The robustness index constraint as shown in Eq. (4.4c) is calculated by a GA. The obtained robust solutions are shown in Figure 4.4 by circles. All robust designs are interior to the nominal designs obtained from MCO. The total volume and maximum stress of robust designs are larger than the corresponding values of nominal designs, respectively.

4.3.3 UUV-Payload Design

This example is the same as the UUV-Payload example given in Section 3.6.5. Using the decentralized UUV-Payload performance analyzers shown in Figure 4.5, we developed the MDO framework for UUV-Payload design as shown in Figure 4.6.

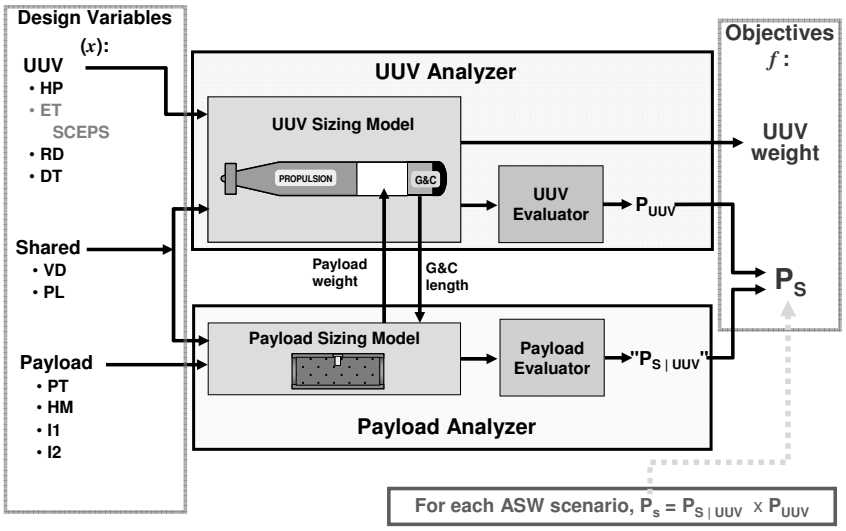


Figure 4.5 Decentralized UUV-Payload performance analyzers

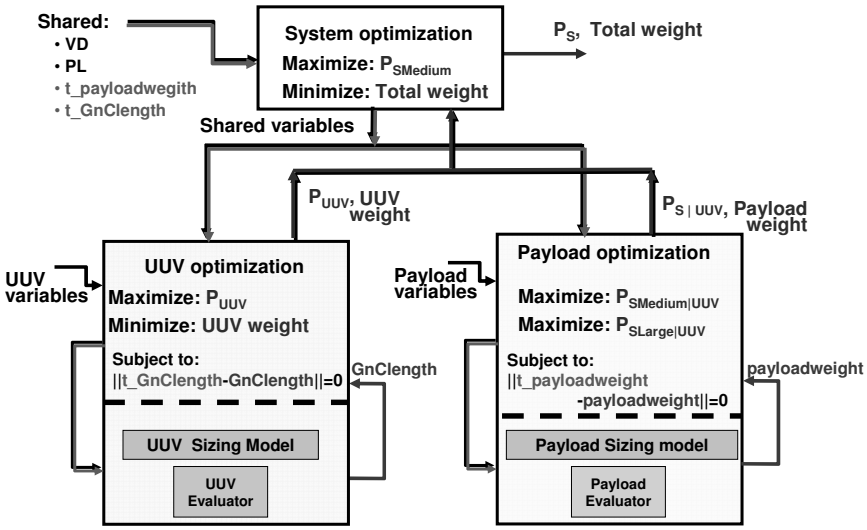


Figure 4.6 MDO framework in deterministic UUV-Payload design

As mentioned previously the framework uses a decentralized formulation: one MOGA controls the design variables in the UUV (without payload) discipline and another MOGA controls the Payload discipline to collaboratively develop UUV-Payload system design. Compared to the formulation in Section 3.6.5, we decentralize that all-at-once model into a two-level hierarchical model. In the decomposed model, we have a

system level subproblem and two fully coupled subsystem level subproblems. Each subproblem has its own associated inputs: local design variable, parameters and disciplinary outputs such as objectives and constraints. Note that in system 0, in addition to shared variables VD and PL, there are two target variables, each for one corresponding coupled variable. The target of the payload weight is called t_payloadweight as well as the target of the GnC length, t_GnClength. In order to maintain the consistency for all subsystems when the multidisciplinary analysis is decomposed, each subsystem should eventually have the same value of coupling variables by using the Interdisciplinary Consistency Constraints (ICC), as shown in Figure 4.6, with one for each coupled variable. The design variables, the optimization problem in each subsystem as well as in system subproblem of this example are also shown in Figure 4.6.

For UUV-Payload system design in this section, we try to maximize P_S in the "Medium" scenario and minimize total UUV weight, as a typical two-objective optimization problem at the system level, with P_S and total UUV weight having trade-offs. In order to achieve higher P_S, we generally need a heavier UUV with the payload.

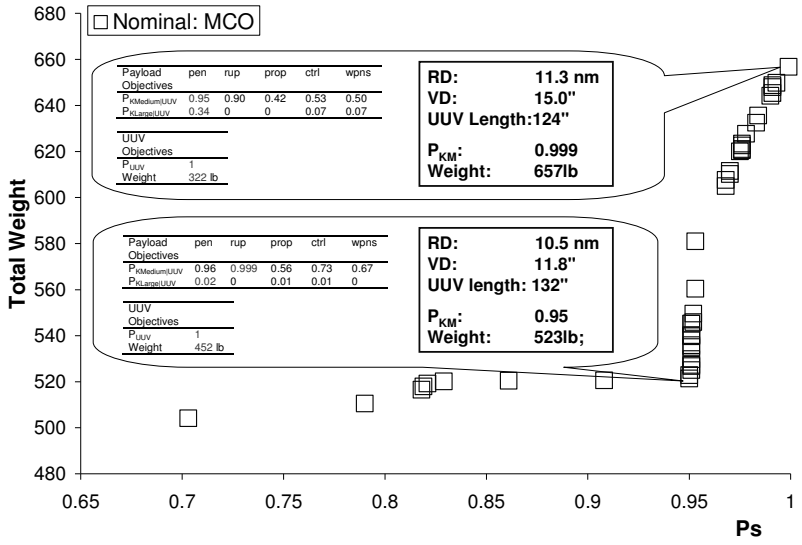


Figure 4.7 UUV-Payload nominal optimal designs in deterministic case

Two typical nominal UUV-Payload designs from the optima frontier, with their system level objective values are given in the Figure 4.7. Shown in the figure are also their subsystem objective function values in the Payload discipline and UUV discipline. For the Payload subsystem, we have five P_{SlUUV} values for each scenario. We pick the maximum one from each five (one for medium scenario and one for large scenario) as the two objective function values in the Payload discipline. In this case, the P_{SLarge} can not be better unless one sacrifices $P_{SMedium}$ since it is a Pareto solution in Payload subsystem. Autonomy of this subsystem is preserved. For UUV discipline, the two objectives are maximizing P_{UUV} and minimizing UUV weight.

Figure 4.8 shows the McRO model for UUV-Payload design. In addition to the steps in the deterministic model, we have an additional constraint in System 0 to determine the robustness, including both performance and collaborative robustness. In both deterministic and robust models, the ICC must be satisfied for all optimal designs. For the robust optimization in this example, we assume that two design variables have uncontrollable uncertainty represented by intervals, [ΔRD, ΔVD] = [1 nm, 0.05 inch]. The tolerance regions for the target variables, [Δt_payloadweigth, Δt_GnClength] are continuous within 20lb and 0.5 inch from nominal. The AOVR in SS1 is: [ΔP_{UUV}, ΔUUVWeight] = [0.1, 40lb] and the AOVR in SS2 is: [$\Delta P_{SMedium}$, ΔP_{SLarge}] = [0.1, 0.1].

In addition to the deterministic designs, we show the optimal robust designs using McRO approach in Figure 4.9. Robustness here implies that both performance and collaborative robustness are met. As expected and observed in the results, in order to account for the uncertainty from the inputs and across the disciplines, UUV weight (tons of lbs) must be enlarged to achieve a similar P_S. For each pair of nominal and robust

designs, robust ones have longer propulsion parts. That is, longer UUV bodies with heavier payloads are used in the robust case.

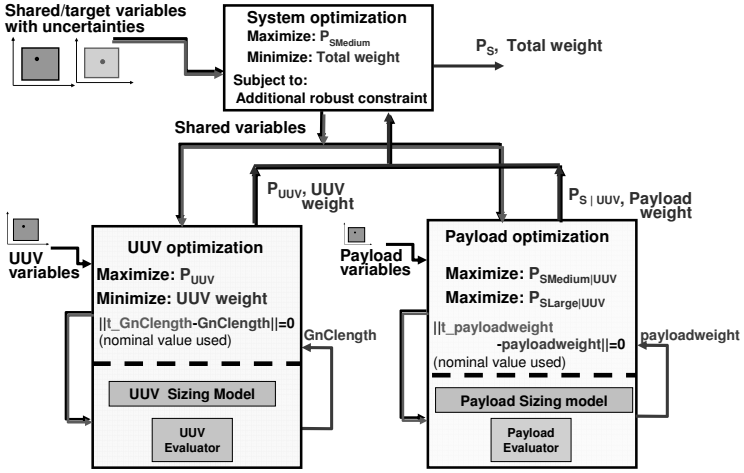


Figure 4.8 McRO approach for UUV-Payload design

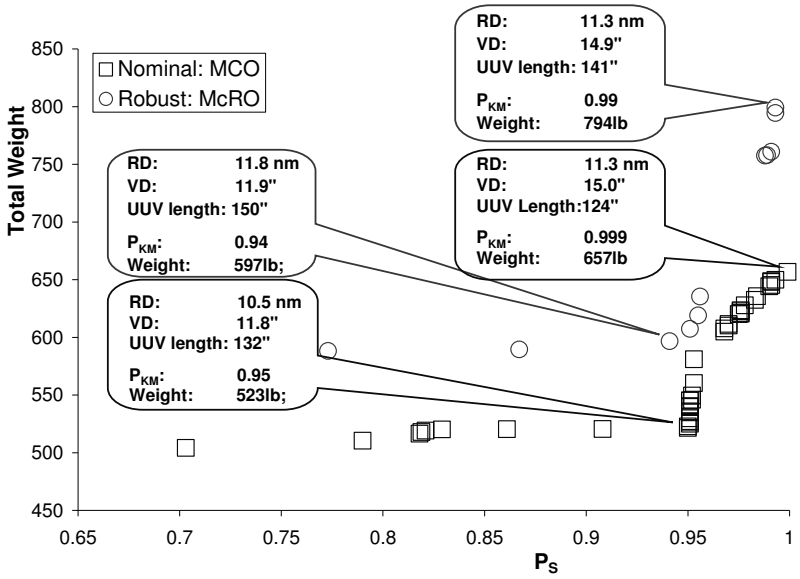


Figure 4.9 UUV-Payload optimal designs in nominal and robust case

4.4 SUMMARY

Based on the concepts of the forward mapping and the robustness index discussed in Chapter 3, we extend the performance robust optimization to handle collaborative robustness for MDO problems. Compared to the previous work, the McRO approach of

this chapter has several characteristics. First, McRO can find robust solutions for multiobjective multilevel coupled MDO problems in which uncontrollable variations exist not only in parameters within each subsystem but also across subsystems. Second, McRO can handle MCO problems that have multiple physical objectives in each of the subproblems. An important advantage of McRO is that it does not require presumed distributions for representing variations in uncontrollable parameters. Also, parameters and design variables can be of a mixed continuous-discrete type. However, McRO requires a tolerance region for target variables. It also requires an acceptable variation range for objective functions as in Chapter 3. This tolerance region of targets provides a cushion to absorb the variations in coupling variables.

McRO has been demonstrated with a numerical and two engineering examples, all of which have fully coupled subproblems. All of the examples have two objective functions in each subsystem and have mixed continuous-discrete design variables and parameter variations. From the results obtained, in the numerical example it was observed that the nominal solutions for the all-at-once and MCO methods are comparable. However, the robust solutions for the all-at-once and McRO methods appear to cover different portions of the robust Pareto frontier. The reason behind this is the setting for the acceptable region of coupling variables. In the speed reducer example, it was observed that both the McRO solutions and MCO solutions covered comparable solution space. However, in all examples, it was observed that: i) robust solutions are interior to or more conservative than the nominal ones, and ii) the number of robust solutions is fewer than the nominal ones. In UUV-Payload design example, to account for the uncertainty, all robust designs are interior to the nominal designs, with significant increased UUV weight.

Finally, McRO can be computationally expensive. We used the same computer platform (see Section 3.7) to test the McRO approach. It usually takes about 2 hours for the numerical and speed reducer example, and 3 hours for the UUV-Payload example to complete a nominal run since MOGA is used in each subsystem. However, it also takes about 100 hours for the UUV-Payload design and 36 hours for other examples to finish one run of robust optimization. Approximation methods can be used to address this aspect which will be considered as part of our future research directions as discussed in Chapter 7.

Next chapter will present a new sensitivity analysis approach for single-disciplinary optimization problems that have parameters with reducible interval uncertainty.

CHAPTER 5: INTERVAL UNCERTAINTY REDUCTION AND MULTI-OBJECTIVE SENSITIVITY ANALYSIS IN SINGLE-DISCIPLINARY DESIGN OPTIMIZATION

5.1 INTRODUCTION

In the broadest sense there are two motivations for taking into account uncertainty in engineering design. Uncertainty analysis approaches are developed to find the range and frequency of possible design or model outputs as a result of uncertainty in parameters or model inputs. In contrast, Sensitivity Analysis (SA) [Iman and Helton, 1988] seeks to connect the uncertainty in model inputs to model outputs. Sensitivity analysis can further be classified as either local or global in nature. Local sensitivity analysis methods such as Differential Analysis (DA) [Hamby, 1994], Most Probable Point (MPP) [Kern et al., 2003] and Response Surface Modeling (RSM) examine the uncertainty in model outputs with respect to small variations in model inputs [Frey and Patil, 2002]. These local SA methods have the limitation of being valid only for small regions of uncertainty. Global Sensitivity Analysis (GSA) takes into account the entire range of model inputs to determine the affect on overall model outputs. The methods proposed in this chapter fall into this global domain and thus will occupy the majority of our attention. In GSA, the portion of model output uncertainty attributed to a subset of uncertain input parameters is generally compared to the overall model output uncertainty considering all input parameters in order to quantify the parameter subset's importance or sensitivity.

Alternatively, SA methods can be classified as sampling based (Monte Carlo), analytical, or as interval analysis. Sampling methods are the most prolific with variance being the principle measure of uncertainty. They typically calculate total model output

uncertainty as a measure of comparison and then iteratively drop a parameter or "leave-one-out" (fixed at the mean) from the comparison uncertainty calculation [Liu et al., 2006]. Those parameters (when fixed) that create the greatest decrease in output uncertainty are considered the most important. Sampling methods with variance as a measure of uncertainty include (amongst others): pure Monte Carlo analysis [Helton, 1993], [Helton and Davis, 2003], Fourier Amplitude Sensitivity Test (FAST) [Saltelli et al., 1999a], Analyis of Variance (ANOVA) [Chen et al., 2005] and Sobol's Variance Decomposition or Index [Sobol 1993 and 2001], [Homma and Saltelli, 1996]. Less commonly, the relative entropy (difference between two probability distributions) has been considered as a measure of global uncertainty and implemented as a sampling technique though [Liu et al., 2006]. Although Bayesian based approaches have also been used in probabilistic sensitivity analysis approaches [Oakley and O'Hagan, 2004], the greatest drawbacks of these methods are computational cost, the availability of probability distributions and treatment of tail probabilities while the Monte Carlo based analyses are not always valid for SA approaches if prior distributions are not credible [Greenland, 2001]. As a major drawback of sampling methods, computational cost can be mitigated to some degree through the use of efficient sampling techniques such as Latin Hypercube Sampling [Helton and Davis, 2003]. The leave-one-out comparison implemented in these methods, however, may not reflect the real situation in engineering design or other domains where some types of uncertainty cannot be eliminated entirely, such as manufacturing tolerances or environmental conditions (e.g., temperature or alternating current voltage input). Rather some degree or "grayscale" version of the original uncertainty is considered to be a more appropriate characterization of uncertain

parameter importance. The reduction of uncertainty (assuming reducible) or a combination of several parameter uncertainties should become more attractive as a measure for improving the sensitivity of designed products than simply eliminating uncertainty.

Analytical methods such as differential analysis and Fast Probability Integration (FPI) [Wu, 1987] are less prolific than the sampling based methods. When these methods are grouped with the post-optimality information from the ubiquitous Simplex Method [Saltelli et al., 1999b] and the less common duality approach for nonlinear problems [Balbas et al., 1999] a sizable fraction of SA methods emerge. These methods suffer from the fact that they are primarily local in nature (e.g., DA, simplex, and duality) and partial derivatives can be difficult to compute (e.g., FPI, DA, Duality). It should be noted that the Simplex approach is not strictly local as it gives a range over which the optimal solution is still valid or can be easily predicted from current solution but does not provide sensitivity beyond that range as a new optimization solution is necessary once the slackness has been exceeded.

Difficulty in obtaining probability distributions for sparse data makes interval analysis attractive and applicable in early stages in engineering design for systems [Wu and Rao 2007], which assumes that a range (or interval) of uncertainty exists rather than a probability distribution for input parameters. Additionally, some feel that interval uncertainty better approximates uncertainty due to ignorance [Ferson and Ginzburg, 1996], [Ferson, et al., 2004] as very little is assumed about the uncertain parameter. A comprehensive review of typical SA methods mentioned previously can be found in [Saltelli et al., 2000], [Helton and Davis, 2003].

It is the author's belief that nearly all sensitivity analysis techniques can be extended to multiple outputs or objectives though it is far less common in the literature than single output [Barron and Schmidt, 1988]. Current multi-output design methods are generally local in nature. RSM-coefficients [Frey and Patil, 2002] and DA, for example, convert the multi-output problem to a single output problem. Avila et al., [2006] employ a local sensitivity method and examine dominated designs generated by a Multi-Objective Genetic Algorithm (MOGA) within an uncertain interval in design space. As an extension of differential analysis the Karush-Kuhn-Tucker's optimality conditions have also been evaluated as sensitivity measures for a Pareto set of solutions [Balbas et al., 2005], [Zhang, 2003]. These methods are limited by the fact the functions must be smooth, continuous in parameter variations and only applicable for small ranges of uncertainty. As a typical post optimality sensitivity analysis approach, it is more common to convert the multi-objective (or multi-output) problem to a single weighted output problem and then perform sensitivity analysis for a single solution point or design [Fiacco, 1983], [Barron and Schmidt, 1988]. FAST and the Sobol's Index have been used to evaluate the weighting scheme used to generate the single output solution [Saltelli et al., 1999b]. Additionally, a response surface model in conjunction with sampling has been employed to evaluate sensitivity for multi-attribute decision problem converted to a single objective problem [Bauer et al., 1999]. As a whole the approaches to multi-output problems have focused on sensitivity for a single solution on a weighted objective or have been local in nature.

There is impetus to conduct multi-objective GSA under the consideration of both a single design and multiple designs (i.e., a set of designs of interest which may or may not

be optimal). For instance, valuating all product designs (or a line of product designs) at once within the firm is a holistic approach and should produce fewer misaligned decisions.

In this chapter, we present a global sensitivity analysis approach for multi-output (multi-objective or -attribute) problems. The radius of the Worst-Case Objective Sensitivity Region (WCOSR) discussed in Chapter 3 has been proposed here to measure the variation on the multi-objectives for a single or a set of designs (i.e., designs from a Pareto frontier of a multi-objective design optimization problem) when input parameters have different level interval uncertainties. A similar metric, Shannon entropy [Cover and Thomas, 1991] has been used to verify the WCOSR metric. As well, a reducible tolerance region (i.e., adjustable uncertainty range) is presented and used to measure the variable ranges of uncertainty in input parameters. A two-objective optimization problem is formed to demonstrate the trade-off between reducing variation in the multiple outputs and the extra effort or investment used in reducing uncertainty in input parameters. The results of this two-objective optimization problem can provide the Decision Maker multiple choices for which uncertainties and whose uncertainty ranges to investigate or evaluate for investment. That is, it can be determined as to which sources of uncertainty should be eliminated or reduced from the model inputs in order to achieve the acceptable variations in the model outputs. Compared to typical SA methods, all obtained choices and the identification of importance can be used with respect to a family of designs under evaluation in the early design stages of a product cycle. Two engineering design examples with interval uncertainties are used to demonstrate the applicability of the proposed approach.

Details of the proposed SA approach with two metrics are presented in Section 5.2. Two engineering design problems and corresponding results are given in Section 5.3 to illustrate the applicability of the proposed approach. Concluding remarks are presented in Section 5.4.

This chapter was also presented in Li et al., [2007c].

5.2 UNCERTAINTY REDUCTION

After the set of (usually the Pareto set of) designs are known, we wish to minimize output uncertainty as a result of parameter variations due to uncertainty, as well as the investment in uncertainty reduction (assuming it is reducible). We first describe the measure for uncertainty reduction for the input parameters. Then, we describe the proposed SA approach starting with the first metric on the model outputs: the radius of the Objective Sensitivity Region (OSR), followed by another: Shannon entropy [Shannon, 1948], [Cover and Thomas, 1991]. We calculate these two metrics by varying uncertain parameters rather than by optimizing design variables. A two-objective optimization problem is formulated for the proposed SA approach.

5.2.1 Uncertainty Reduction in the Parameter Space

One of the reasons that SA is gaining interest is that designers and managers typically wish to know which uncertain parameter(s) should have their uncertainty reduced or totally eliminated to bring about significant variation reduction in model outputs. Extra but limited “investment” or “resource” (i.e., cost of new materials, conducting surveys, new methods, new manufacturing machines, etc.) must be allocated wisely and used efficiently to reduce the uncertainty in the most critical parameters so that the substantial

variation reduction in the multiple model outputs can be achieved. SA should provide this type of information to DMs.

Recall (in Section 2.4) that the Retained Tolerance Region (*RTR*) in $\boldsymbol{p}$-space is the grayscale of an original tolerance region $\Delta\boldsymbol{p}$, defined as the inner product of the *PURI* and the original tolerance region: $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p} = (\alpha_1\Delta p_1,\ \alpha_2\Delta p_2,\ldots,\ \alpha_K\Delta p_K)$. Essentially, *RTR* can represent any symmetric hyper-rectangle inside the original tolerance region, as shown in Figure 2.6.

For a particular design $\boldsymbol{x}_0 = (x_{0,1},\ldots,x_{0,N})$ the nominal values of the objective functions are $\boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{p}_0) = (f_1(\boldsymbol{x}_0, \boldsymbol{p}_0),\ldots,f_M(\boldsymbol{x}_0, \boldsymbol{p}_0))$, and the nominal values of the constraint functions are $\boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{p}_0) = (g_1(\boldsymbol{x}_0, \boldsymbol{p}_0),\ldots,\ g_L(\boldsymbol{x}_0, \boldsymbol{p}_0))$. We consider here objective function variations and constraint function variations of $\boldsymbol{x}_0$ caused by parameter variations $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$ within the *RTR*, as in Eq. (5.1):

$$\begin{aligned} \Delta\boldsymbol{f}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) &= \boldsymbol{f}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{p}_0) \\ \Delta\boldsymbol{g}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) &= \boldsymbol{g}(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - \boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{p}_0) \\ \boldsymbol{p}_0 - \boldsymbol{\alpha}\cdot\Delta\boldsymbol{p} \le \tilde{\boldsymbol{p}} &\le \boldsymbol{p}_0 + \boldsymbol{\alpha}\cdot\Delta\boldsymbol{p} \end{aligned} \tag{5.1}$$

Regardless of which type of uncertainty is under investigation we can map instances of parameter variations for the entire Pareto (or a set of) designs and consider the effect on all designs in a multi-objective sense. Figure 5.1 is an example of this mapping for a two-parameter two-objective function problem.

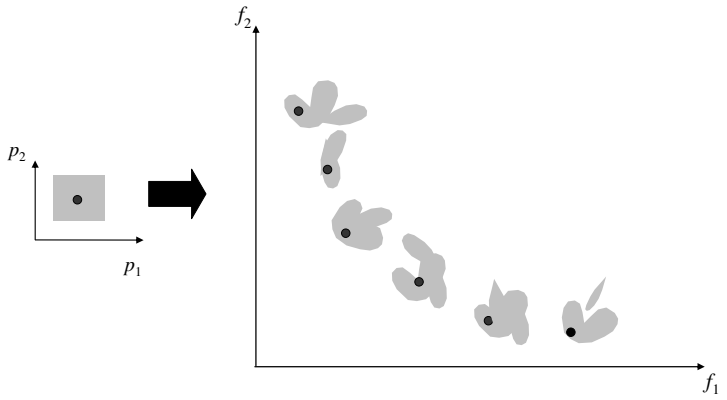


Figure 5.1 Variation in objective space due to uncertainty in parameters

Notice that in Figure 5.1 the parameter range on the left can map differently to each of the designs represented in objective space on the right. That is, the uncertainty can affect each design differently.

As mentioned in Section 2.4, the *PURI* vector $\boldsymbol{\alpha}$ is used to define *RTR* in the parameter space, thus it defines the retained amount of uncertainty in the input parameters. Accordingly, the quantities $\sum_{k=1}^{K} 2\alpha_k$ and $\prod_{k=1}^{K} 2\alpha_k$, respectively, represent the hyper-perimeter and hyper-volume of the uncertainty retained in the parameter space after the uncertainty is normalized by $\Delta\boldsymbol{p}$. That is, the quantities $\sum_{k=1}^{K} 2\alpha_k$ and $\prod_{k=1}^{K} 2\alpha_k$ represent unit-less uncertainty and could be used as a metric of the amount of uncertainty. However, in this dissertation, we would like to provide a metric to measure the *investment* put into reducing the uncertainty. That is, the DM wants to put as little resource as possible for a given *RTR*. So, after normalization the following two metrics *Perimeter Reduction Metric* (*PRM*) and *Volume Reduction Metric* (*VRM*) together are used as a relative measure of the investment as shown in Eq. (5.2):

$$PRM = \frac{K - \sum_{k=1}^{K} \alpha_k}{K} \text{ and } VRM = 1 - \prod_{k=1}^{K} \alpha_k \tag{5.2}$$

If $\alpha_k = 1$ for $k = 1,\ldots,K$, *PRM* and *VRM* go to zero, which means no extra resource needed to reduce the input uncertainty. On the contrary, if $\alpha_k = 0$ for $k = 1,\ldots,K$, then both *PRM* and *VRM* are equal to 1, meaning the maximum possible effort needed to eliminate all uncertainty in parameters. We define the weighted sum of these two metrics as the *Investment*, which correlates positively with the amount of investment or resource used for reducing uncertainty, as shown in Eq. (5.3):

$$Investment = \theta_1 PRM + \theta_2 VRM \tag{5.3}$$

The θ value can be selected and aligned according to the DM's preferences. In this dissertation, we use $\theta_1 = \theta_2 = 0.5$ which means that both the volume and perimeter metrics have equal weights according to the DM's preferences.

In the engineering design, the DM would like to use as little resource as possible to reduce the uncertainty in the input parameters, but also to reduce the variation in the objectives as much as possible. So next we will provide two metrics in objective function space (or multi-output space) to measure the variation in objectives for a multitude of candidate designs under investigation early in the product design stage.

5.2.2 Objective Sensitivity Region in the Objective Space

In this section, we will develop an uncertainty metric based on the concept of the OSR developed in Chapter 3 and describe this metric in terms of variation in the objective space. For simplicity, we consider the output space as the same as the multi-objective space of the optimization problem in Eq. (2.1).

The previous proposed formulation in Chapter 3, termed as a performance robust optimization approach, is based on the concept that the known tolerance region of parameters with uncertainty can be mapped to $\boldsymbol{f}$-space to form an Objective Sensitivity Region (OSR), as shown in Figure 5.2. In this dissertation, for any *RTR*, $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$, we could map it into the objective space to obtain the corresponding *Reduced-OSR*, *ROSR*. Based on that, R_f, considered as a radius of the *ROSR* (vector in Figure 5.2) for design $\boldsymbol{x}_n$, is the possible largest deviation of the objective values from the nominal. Basically, $R_f(\boldsymbol{\alpha})$ represents how far the deviation of multiple objectives can be from $\boldsymbol{x}_n$ nominal in the objective space due to a corresponding $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$. Given N_{np} nominal Pareto designs $\boldsymbol{x}_n$, n=1,…, N_{np}, the average R_f of the family of Pareto solutions for any $\boldsymbol{\alpha}$ can be calculated by an optimization problem as shown in Eq. (5.4); (N_{np} is the number of designs under consideration). In Eq. (5.4), for each design $\boldsymbol{x}_n$, its objective variation is normalized by its own nominal objective values. The obtained R_f for any $\boldsymbol{\alpha}$ is thus represented as a percentage of Pareto solutions' nominal objective values. Notice in this chapter we use $\|\cdot\|_2$ distance metric instead of using $\|\cdot\|_\infty$ because there is no requirement to compare the WCOSR (hyper-sphere in this chapter) with the normalized AOVR (a hyper-cube) as in Chapter 3.

$$R_f(\boldsymbol{\alpha}) = \max_{\tilde{\boldsymbol{p}}} \frac{[\sum_{n=1}^{N_{np}} \left\|\Delta \boldsymbol{f}(\tilde{\boldsymbol{p}})\right\|_2]}{N_{np}}$$

$$\text{where} \qquad \Delta \boldsymbol{f}(\tilde{\boldsymbol{p}}) = \frac{\boldsymbol{f}(\boldsymbol{x}_n, \tilde{\boldsymbol{p}}) - \boldsymbol{f}(\boldsymbol{x}_n, \boldsymbol{p}_0)}{\boldsymbol{f}(\boldsymbol{x}_n, \boldsymbol{p}_0)} \tag{5.4}$$

$$\boldsymbol{p}_0 - \boldsymbol{\alpha}\cdot\Delta\boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$$

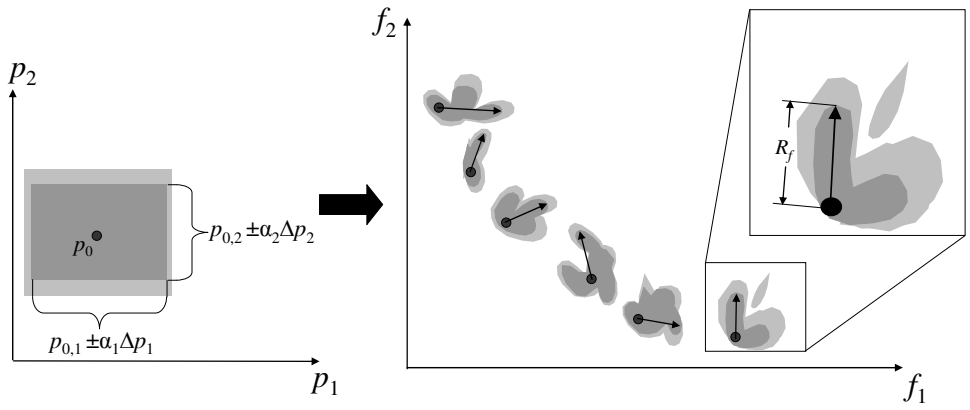


Figure 5.2 Mapping from a *RTR* to *ROSR*s with R_f

Figure 5.2 is an example of how parameter uncertainty is mapped to the *ROSR*s in the objective space. The effect of varying α and therefore the *RTR* to reduce the R_f and the *ROSR* is shown in Figure 5.3. It is also demonstrated in Figure 5.3 how two relatively similar volume sized *RTR*s can have very different R_f given that one parameter may be more important than the other. Clearly, given equal investment for combination A and B (Figure 5.3) the DM would prefer to invest in combination A as the R_f for the *ROSR* in A is smaller. This is the basis of comparison for candidate investment levels.

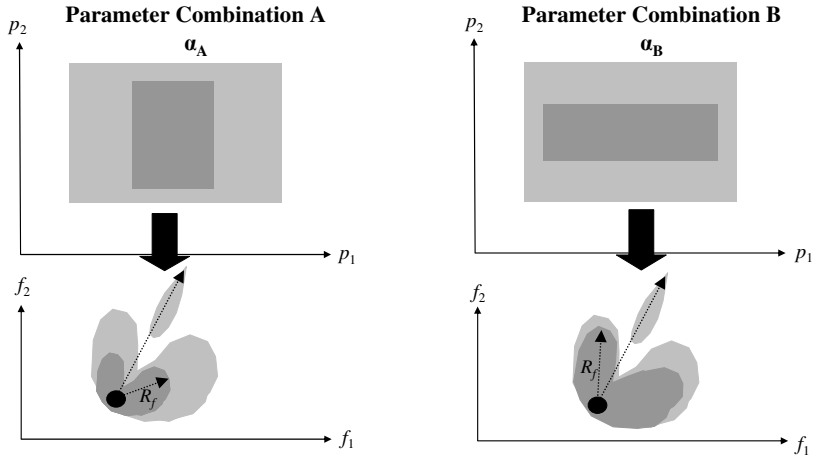


Figure 5.3 Varying parameter combinations

5.2.3 Shannon Entropy in the Objective Space

Claude Shannon [Shannon, 1948] established entropy as a measure of uncertainty associated with a random number or equivalently information content in a piece of data communicated across a transmission medium. Numerous applications and extensions have been developed for Shannon entropy in a variety of fields but entropy has rarely been applied as a sensitivity measure for system design [Liu et al., 2006]. In multi-objective design, entropy has been used to guide optimization [Gunawan et al., 2004], to assess the quality of Pareto solutions [Farhang-Mehr and Azarm, 2003] and to obtain the optimal weight coefficients in the weighted sum method [Barron and Schmidt, 1988]. In this section we extend the approach to measuring quality of a set of Pareto designs from Farhang-Mehr and Azarm [2003], to address parameter sensitivity for multiple objectives. Next we first define Shannon entropy and then show how this measure can be extended and used in a multi-objective space to quantify the level of uncertainty for Pareto designs.

Entropy in the objective space

For a discrete distribution of i=1,…, E possible events, the Shannon entropy measure H is defined in terms of the probability of the i-th event P_i, as shown in Eq. (5.5):

$$H = -\sum_{i=1}^{E} P_i \log_2(P_i) \text{ and } P = [P_1..., P_i,..., P_E], \ \sum_{i=1}^{E} P_i = 1, \ P_i \geq 0 \tag{5.5}$$

Events that we know will happen with complete certainty result in Shannon entropy of zero. The measure increases as the number of possible events becomes greater. Less obviously the measure is maximized for n possible events when the probability of each event P_i is equal to $1/E$.

We translate this notion of uncertainty based on events and event probabilities to varying design performance (objective space) realizations in the calculation of

uncertainty. That is when instances of parameters cause a design's performance to vary significantly, additional performance instances or events are generated which contribute to increased entropy. To find the entropy of a design(s) under uncertainty in a multi-objective space we must first define how great the objective function difference must be to be considered a change in performance. This has been called an indifference band or indifference region [Farhang-Mehr and Azarm, 2003]. Essentially, we define an indifference tolerance region in terms of the DM's indifference between objective realizations. The indifference band width $\boldsymbol{I} = (I_1,\ldots,I_M)$ for objective functions $\boldsymbol{f} = (f_1,\ldots,f_M)$ defines a hyper-cube, $\boldsymbol{D}=(D_1,\ldots,D_M)$, $\boldsymbol{D}\in \mathbb{Z}^M$. The upper limit $\boldsymbol{D^u}=(D_1^u,\ldots,D_M^u)$, $\boldsymbol{D^u} \in \mathbb{R}^M$ and lower limit $\boldsymbol{D^l}=(D_1^l,\ldots,D_M^l)$, $\boldsymbol{D^l} \in \mathbb{R}^M$ of the hyper-cube are defined through an entry-wise or via the inner product as shown in Eq. (5.6):

$$\begin{aligned} \boldsymbol{D^u} &= \boldsymbol{D}\cdot\boldsymbol{I} = [D_1 I_1,...,D_M I_M] \\ \boldsymbol{D^l} &= (\boldsymbol{D}-\boldsymbol{1})\cdot\boldsymbol{I} = [(D_1-1)I_1,...,(D_M-1)I_M] \end{aligned} \tag{5.6}$$

It is also useful to define the largest number of hyper-cubes that will be considered along each dimension. Let $\boldsymbol{f^{max}}$ be the largest value of the objective region and $\boldsymbol{f^{min}}$ be the minimum value. The total number of grids $\boldsymbol{G}$ along each dimension is shown in Eq. (5.7):

$$\boldsymbol{G} = \left[\frac{\left(f_1^{max}-f_1^{min}\right)}{I_1}, \frac{\left(f_2^{max}-f_2^{min}\right)}{I_2}, \ldots \frac{\left(f_M^{max}-f_M^{min}\right)}{I_M}\right] \tag{5.7}$$

As an example, consider a two objective problem of f_1 and f_2 with indifference bands I_1 and I_2, respectively. The indifference hyper-cube $\boldsymbol{D}_{a,b}$: a=1,2,…,G_1, b=1,2,…,G_2 with upper and lower limits $\boldsymbol{D}_{a,b}^{u}$, $\boldsymbol{D}_{a,b}^{l}$ are shown in Figure 5.4.

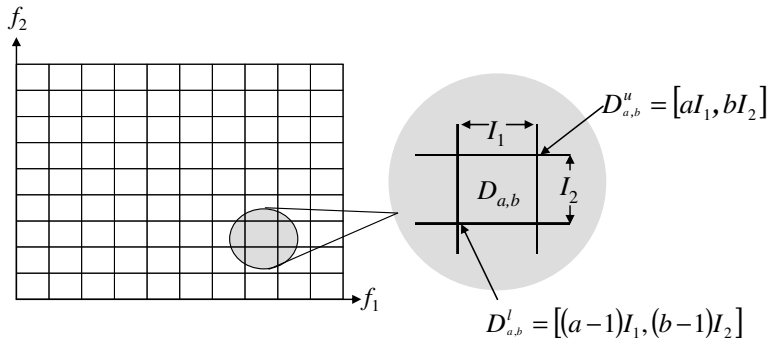


Figure 5.4 Indifference hyper-cube

The meaning of this indifference hyper-cube ***D*** is that any objective space realization that falls within this hyper-cube is essentially indistinguishable for the DM. As an example, consider a precision tuned stock race car that has a nominal peak horsepower of 550 bhp where race results can be affected by a small dip in horsepower (3 bhp indifference band) below the design level. If uncertainty in valve timing produces horsepower results ranging from 549.9 to 550.1 bhp it is unlikely that the DM will register significant concern but rather would be indifferent to the uncertainty surrounding the valve timing. A greater change in horsepower from the nominal (say as little as 5 bhp) may be enough of a difference to cause concern for the DM to investigate the uncertain parameter.

In terms of entropy, when an objective space realization (e.g., 544 bhp in the example) falls outside of the indifference band around the nominal objective realization, a new "event" or objective space realization *i* is possible in Eq. (5.5). Much like R_f, we calculate entropy in the objective space for the entire Pareto set of designs. Here we are concerned with what happens after the set of (usually the Pareto set) designs are known and wish to minimize entropy as a result of parameter variation which we call *Multi-objective Entropy Performance* (*MEP*).

In the calculation of *MEP* for the Pareto set of designs, first a retained tolerance region is dictated in parameter space by $\boldsymbol{\alpha}\cdot\Delta\boldsymbol{p}$ and realizations are sampled (see the next subsection) and mapped to objective function space as in Figure 5.1. Unlike the R_f approach presented previously we must sample throughout the tolerance region for *MEP* and consider realizations in objective space as they can contribute to entropy. *MEP* is found numerically by counting the number of sampled objective space realizations within any indifference hyper-cube $\boldsymbol{D}$. Consider two Pareto design points with tolerance region and parameter naïve sampling approach shown in Figure 5.5:

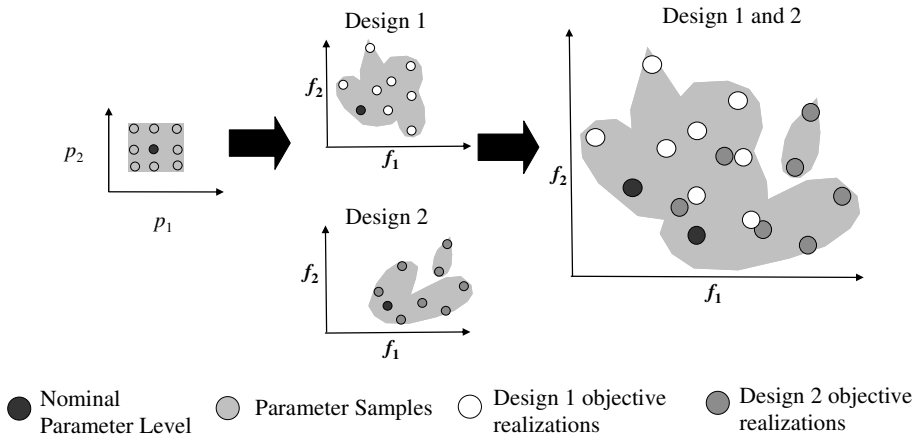


Figure 5.5 Parameter realizations in objective space for multiple designs

To calculate the entropy for these designs under the assumed level of uncertainty we overlay the indifference grid $\boldsymbol{D}$ in objective space and calculate the Shannon entropy by summing the number of design realizations $\tilde{\boldsymbol{f}} = \boldsymbol{f}(\boldsymbol{x}_n, \tilde{\boldsymbol{p}})$ in each grid $\boldsymbol{D}$, where $\boldsymbol{x}_n$ indicates any (Pareto optimal) design and $\tilde{\boldsymbol{p}}$ indicates a parameter sample. This can be described algorithmically in the following way:

(1) For any $\boldsymbol{D}$ count the number of $\tilde{\boldsymbol{f}}$ where $\boldsymbol{D}^l \le \tilde{\boldsymbol{f}} \le \boldsymbol{D}^u$ for all dimensions, *M*. This establishes the vector $\boldsymbol{C}$ which corresponds to each grid $\boldsymbol{D}$. $\boldsymbol{C} = \text{count}\left(\tilde{\boldsymbol{f}} \text{ in } \boldsymbol{D}\right)$

(2) Calculate the probability of event or realization $\boldsymbol{D}$ where Ns is the number of samples for each $\boldsymbol{x}_n$ of N_{np} Pareto design(s): $P(\boldsymbol{D}) = \dfrac{\boldsymbol{C}}{N_{np} \cdot Ns}$. The probability of a grid $\boldsymbol{D}$ realization is then the number of $\tilde{\boldsymbol{f}}$ that falls within the grid (i.e., $\boldsymbol{C}$) divided by the total number of sampled points: $N_{np} \cdot Ns$. If there is only one design is of interest, N_{np}=1.

(3) The Shannon entropy for all of the objective space realizations $\tilde{\boldsymbol{f}}$ is fairly simple to calculate once P is known. The entropy is calculated, as shown in Eq. (5.8), summed for each grid up to $\boldsymbol{G}$ and results in *MEP* as we are considering multiple objectives and design(s):

$$MEP = -\sum_{1}^{G_1}\sum_{1}^{G_2}\ldots\sum_{1}^{G_M} P(\boldsymbol{D})\log_2\left(P(\boldsymbol{D})\right) \tag{5.8}$$

The unit of *MEP* is the unit of Shannon entropy, the bit. By virtue of the fact that two or more $\tilde{\boldsymbol{f}}$ occupy any $\boldsymbol{D}$ the uncertainty is reduced as this provides greater information about possible design realizations.

Sampling in MEP

As mentioned previously, to compute *MEP* we must sample in the parameter space and map these samples to the objective space. An appropriate approach should capture the maximum amount of uncertainty possible in parameter space and require no more samples than necessary to estimate this maximum. Since no knowledge of objective response is assumed we chose an approach that maximizes the entropy in the input parameters and is augmentable so that samples can be added one by one until a maximum entropy is reached in the parameter space [Shewry and Wynn, 1987]. The approach

selected is Maximum Entropy Design and has been used to develop adaptive metamodels for computationally expensive simulations [Shewry and Wynn, 1987].

5.2.4 Uncertainty Reduction Optimization

Based on the two metrics in the objective space, R_f and *MEP*, and the *Investment* metric in parameter space for uncertainty, we form our sensitivity analysis approach as a two-objective optimization problem. Clearly one objective is to minimize *Investment* in Eq. (5.3), which represents the inverse of uncertainty in the parameter space (recall Section 3.1); another one is to minimize either the R_f in Eq. (5.4) (recall Section 5.2.2) or the *MEP* in Eq. (5.8) (recall Section 5.2.3), meaning to minimize the uncertainty in the objective space. The conflict between these two objectives reflects the situation that the DM always would like to save investment on uncertainty reduction but simultaneously reduce as much variation as possible in objective realizations. The variables of this optimization problem are the *PURI* vector $\boldsymbol{\alpha}$.

Given nominal (Pareto) designs of any design problem and the original tolerance region, we find the optimal *PURI* vector $\boldsymbol{\alpha}$ values that can minimize *Investment* and simultaneously minimize R_f or *MEP*, as shown in Eq. (5.9). Since there are trade-offs between these two objectives, it is expected that we will obtain a set of $\boldsymbol{\alpha}$ solutions in a Pareto sense. DM can then choose one of them according to his/her preferences. This problem is formulated as in Eq. (5.9):

$$
\begin{aligned}
&\min_{\boldsymbol{\alpha}} \; Investment \; = \theta_1 PRM \; + \theta_2 VRM \\
&\min_{\boldsymbol{\alpha}} \; R_f \text{ or } MEP \\
&\text{where} \quad PRM \; = \frac{K - \sum_{k=1}^{K} \alpha_k}{K}, VRM \; = 1 - \prod_{k=1}^{K} \alpha_k \\
&\qquad R_f(\boldsymbol{\alpha}) = \max_{\tilde{\boldsymbol{p}}} \frac{[\sum_{n=1}^{N_{np}} \left\| \Delta \boldsymbol{f}(\tilde{\boldsymbol{p}}) \right\|_2]}{N_{np}} \\
&\qquad MEP \; = -\sum_{1}^{G_1} \sum_{1}^{G_2} \ldots \sum_{1}^{G_M} P(\boldsymbol{D}) \log_2 \left(P(\boldsymbol{D}) \right), \; P(\boldsymbol{D}) = \frac{\text{count}\left(\boldsymbol{f}(\boldsymbol{x}_n, \tilde{\boldsymbol{p}}) \text{ in } \boldsymbol{D} \right)}{N_{np} \cdot Ns} \\
&\qquad \Delta \boldsymbol{f}(\tilde{\boldsymbol{p}}) = \frac{\boldsymbol{f}(\boldsymbol{x}_n, \tilde{\boldsymbol{p}}) - \boldsymbol{f}(\boldsymbol{x}_n, \boldsymbol{p}_0)}{\boldsymbol{f}(\boldsymbol{x}_n, \boldsymbol{p}_0)} \qquad (5.9) \\
&\qquad \boldsymbol{p}_0 - \boldsymbol{\alpha} \cdot \Delta \boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \boldsymbol{\alpha} \cdot \Delta \boldsymbol{p} \\
&\qquad \theta_1 + \theta_2 = 1 \\
&\qquad n = 1, \ldots, N_{np}, \; k = 1, \ldots, K, \; m = 1, \ldots, M \\
&\qquad N_{np} = \#\text{ of of Pareto designs}, \quad Ns = \#\text{ of samplings in } \tilde{\boldsymbol{p}} \\
&\qquad \boldsymbol{G} = \#\text{ of grids along each dimension} \\
&\qquad \boldsymbol{D} = \text{Indifference hypercube}
\end{aligned}
$$

In Eq. (5.9), the R_f value has been normalized by nominal objective values of optimal designs. The calculation of R_f only depends on the optimizer used to find its global maximum value, while the value of *MEP* depends on the sampling in the parameter space as well as the indifference hyper-cube used in the objective space. One should notice that according to their definitions, R_f and *MEP* focus on the different perspectives of the objective variation $\tilde{\boldsymbol{f}}$. R_f defines the distance from the nominal to the farthest $\tilde{\boldsymbol{f}}$ in the worst case scenario. It captures the worst-case distance. Meanwhile, *MEP* focuses on the diversity of the objective variation. If $\tilde{\boldsymbol{f}}$ has been realized evenly and diverse in the objective space, *MEP* will increase by its definition. That is *MEP* depends on the samplings of $\tilde{\boldsymbol{f}}$ realizations while R_f does not. For instance, two $\boldsymbol{\alpha}$ solutions are shown in Figure 5.6. Obviously in this case, $R_f(\boldsymbol{\alpha}_{\mathrm{I}}) < R_f(\boldsymbol{\alpha}_{\mathrm{II}})$; $MEP(\boldsymbol{\alpha}_{\mathrm{I}}) > MEP(\boldsymbol{\alpha}_{\mathrm{II}})$, though, given

the current $\tilde{f}$ realizations. The DM may choose the R_f metric if the worst-case scenario of the objective variation is of interest. On the contrary, one may consider the *MEP* an appropriate choice if the indifference hyper-cubes selected are credible and the diversity of the variation in objective functions is more important to the DM.

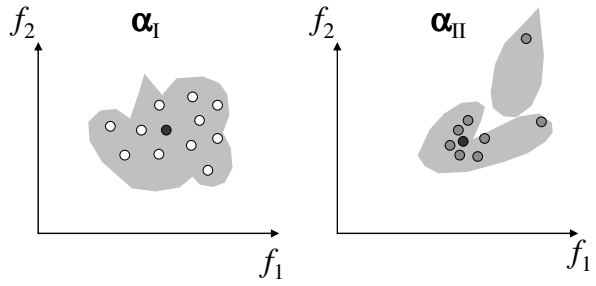


Figure 5.6 Preference for two α solutions α_I and α_{II}

Figure 5.7 is an example of the trade-off between investment and uncertainty reduction. If unlimited resources and uncertainty reduction potential is high (i.e., all uncertainty is reducible) the DM might set all α_k, $k = 1,\ldots,K$, to zero as shown. In this case the solution to the original problem is essentially deterministic. At the other extreme, all α_k, $k = 1,\ldots,K$, are set to one by the DM when uncertainty is irreducible due to investment restrictions or irreducible by virtue of the nature of the uncertain parameter. In reality, it is likely that some investment capability exists through additional investigations or effort and that not all uncertainty is reducible which yields any one of the *RTR*s (which fall inside the original tolerance region) shown in Figure 5.7 depending on the DM's preference.

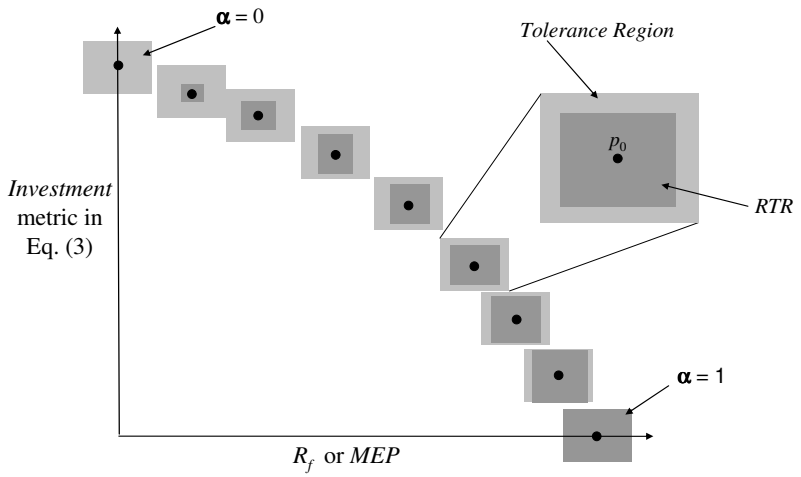


Figure 5.7 *Investment* vs. uncertainty metric

In the next section, we will use two engineering design examples to demonstrate the applicability of this sensitivity analysis and the meaning of obtained solutions.

5.3 EXAMPLES AND RESULTS

In this section, two engineering examples described in Chapter 3 are used to demonstrate the applicability of the proposed SA and uncertainty reduction approach. Two proposed uncertainty metrics in the objective space, R_f and *MEP*, are applied with the *Investment* metric of uncertainty in the parameter space.

5.3.1 Design of a Vibrating Platform

The first example is to design a vibrating platform [Narayanan and Azarm 1999], described in Section 3.6.2 and shown in Figure 3.10. The two-objective constrained optimization problem shown in Eq. (3.13) is shown here again as in Eq. (5.10) and the MOGA is used to obtain the nominal optimal solutions in the deterministic case.

$$
\begin{aligned}
\text{maximize} \quad & f_{\mathrm{n}} = \left(\frac{\pi}{2L^2}\right)\left(\frac{\mathrm{EI}}{\mu}\right)^{0.5} \\
\text{minimize} \quad & \text{cost} = 2wL\left[c_1 t_1 + c_2\left(t_2 - t_1\right) + c_3\left(t_3 - t_2\right)\right] \\
& \mathrm{EI} = \left(\frac{2b}{3}\right)\left[E_1 t_1^3 + E_2\left(t_2^3 - t_1^3\right) + E_3\left(t_3^3 - t_2^3\right)\right] \\
& \mu = 2w\left[\rho_1 t_1 + \rho_2\left(t_2 - t_1\right) + \rho_3\left(t_3 - t_2\right)\right] \\
\text{s. t.} \quad & g_1 \equiv \mu L\text{-}2800 \le 0 \\
& g_2 \equiv \left(t_1\text{-}t_2\right) \le 0 \\
& g_3 \equiv \left(t_2 - t_1\right) - 0.15 \le 0 \\
& g_4 \equiv \left(t_2 - t_3\right) \le 0 \\
& g_5 \equiv \left(t_3 - t_2\right) - 0.01 \le 0
\end{aligned}
\tag{5.10}
$$

The parameter variations are known to be $(\Delta\rho_{\mathrm{A},0}, \Delta c_{\mathrm{A},0})$ = (10 kg/m^3, 50 \$/m^3); 10% of the nominal ρ_{A} and c_{A} values. First, we obtain the nominal Pareto solutions of Eq. (5.10), as shown in Figure 5.8 (shown as a min-min plot by taking the negative of the frequency).

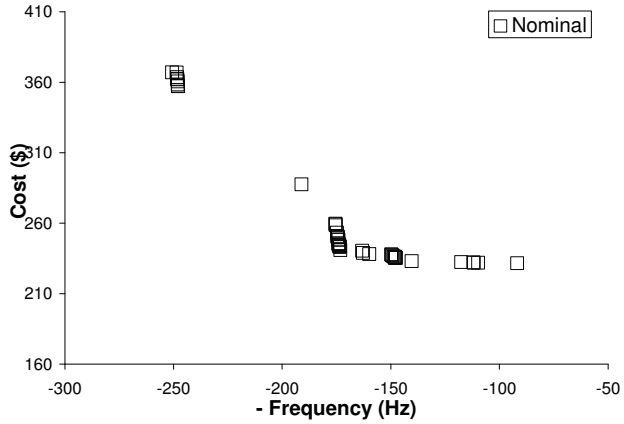
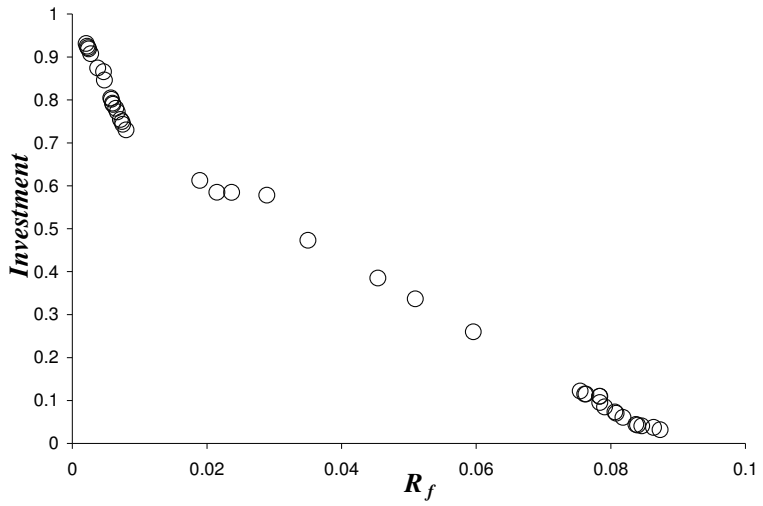


Figure 5.8 Nominal Pareto solutions of vibrating platforms for Eq. (5.10)

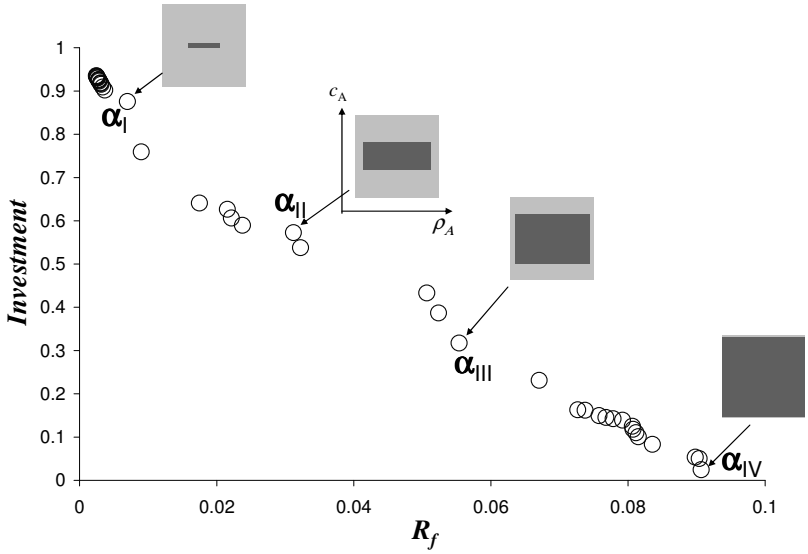
Then we use a MOGA to obtain the Pareto α solutions of Eq. (5.9) using *Investment* vs. R_f as two objective functions, as shown in Figure 5.9(a) and (b). In Figure 5.9(a), the α solutions of Eq. (5.9) using *Investment* vs. R_f, with respect to a single design are shown, i.e., $[t_1, t_2, t_3, w, L]$ = [0.20, 0.22, 0.22, 0.35, 3.02] with material types: {A, C, B}, starting

from the inner layer outward. With four typical α solutions shown in Figure 5.9(b), we minimize *Investment* and R_f simultaneously with respect to N_{np} =35 Pareto vibrating platform designs as shown in Figure 5.8. Since the solutions in Figure 5.9(a) and (b) are similar, we will focus on the solutions for the set of designs (e.g., solutions in Figure 5.9(b)) in the following discussion.

With increasing *Investment* (i.e. decreasing of the *RTR*), R_f value (i.e. the variation in objective values) becomes smaller. For instance, R_f value for α_{I} is near to the zero which means the problem is converging to a deterministic case and the investigations eliminate a significant amount of uncertainty in parameters at significant cost. On the contrary, uncertainty for the extreme case, α_{IV}, remains irreducible due to limited investment. Between the extreme points, α_{II} and α_{III} provide grayscale solutions for additional investment, depending on the DM's preference. With the combination of the reduction in both parameter ρ_{A} (82.7% of original retained) and c_{A} (33.2% of original retained), the variation in objective functions for α_{II} is reduced to approximate 3% of the nominal objective values of the Pareto set of designs, compare to variation for 9% of the objective values for almost full uncertainty case α_{IV}.



(a) For a single Pareto design



(b) For 35 Pareto designs

Figure 5.9 Obtained solutions for *Investment* vs. R_f with respect to (a) a single Pareto design and (b) 35 Pareto designs

It also can be noticed from these four typical α solutions that more uncertainty reduction is possible through parameter c_A than ρ_A. This indicates that c_A is the more important parameter but a more in depth analysis is warranted. In order to answer the classical SA question, identifying the relative importance of uncertainty parameters, we

use the correlation plot figures of α_1, α_2, *Investment* and R_f for all Pareto $\boldsymbol{\alpha}$ solutions in Figure 5.9(b), as shown in Figure 5.10. Figure 5.10(f) also corresponds to the Pareto of R_f and *Investment* levels shown in Figure 5.9(b). The near-to-linear shape indicates that increasing *Investment* in parameters space can linearly reduce the uncertainty in the objective values (measured by R_f). Figure 5.10(a) shows the relation between α_1 and α_2 which correspond to parameters ρ_A and c_A respectively. The figure includes all values of the Pareto $\boldsymbol{\alpha}$ solutions. While α_1 values are clustered into two sections [0.25, 0.4] and [0.8, 1], α_2 values scatter on the entire range from 0 to 1. In this regard, at least some ranges of α_1 did not make a contribution to the uncertainty reduction in objective values. On the contrary, all different levels of α_2 values made the contribution in the optimal $\boldsymbol{\alpha}$ solutions. This conclusion is strengthened by the fact that there is not a clear correlation relation between α_1 and R_f or between α_1 and *Investment,* as shown in Figure 5.10(b) or Figure 5.10(d). Figure 5.10(c) shows the strong correlation (near to linear) between α_2 and R_f which indicates that α_2 is relatively more important than α_1 with respect to the effect on the uncertainty reduction in R_f. That is, the reduction in uncertainty for the second parameter, in this example, c_A could bring more benefit in the variation reduction in the objective space for all optimal vibrating platform designs to the DM. Figure 5.10(e) confirmed the major contribution of α_2 in the uncertainty reduction in this example.

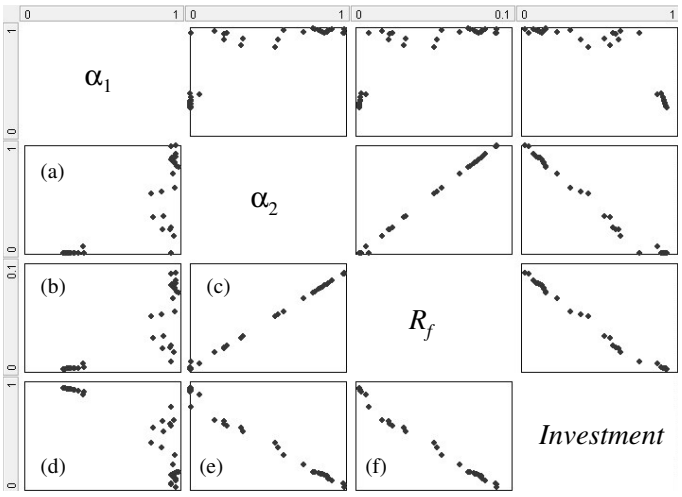
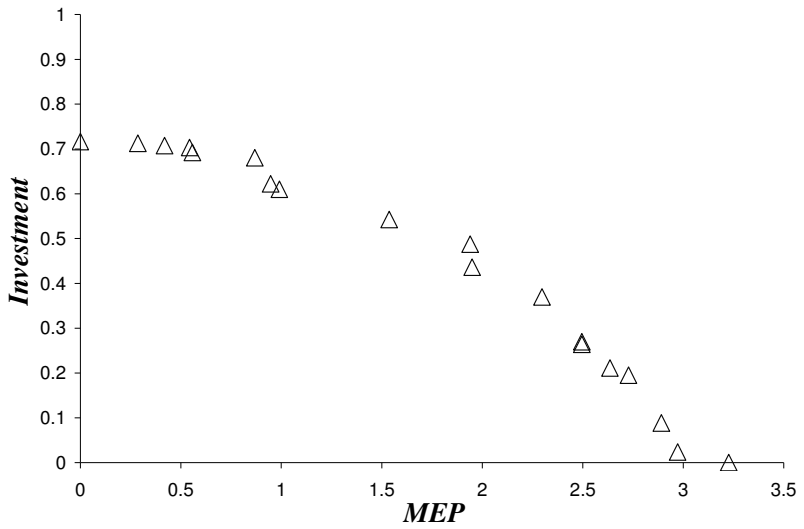
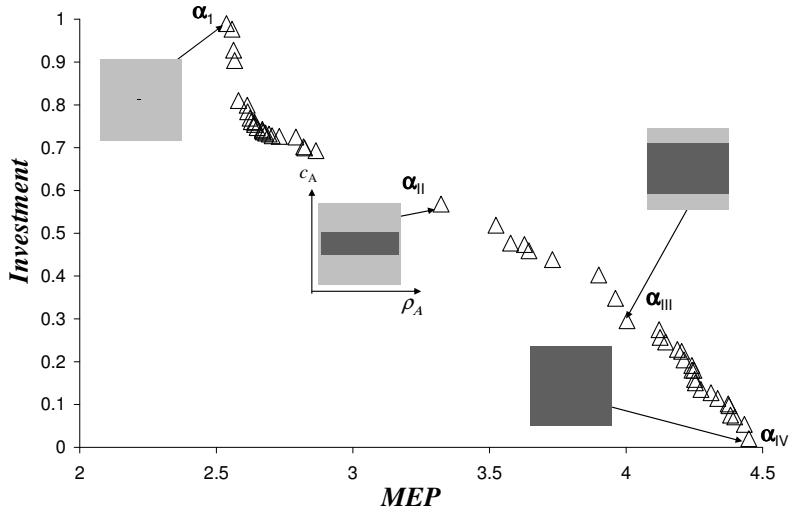


Figure 5.10 Plots of correlations among α, R_f and *Investment*

To verify the observation of the uncertainty reduction approach using the metric R_f, we use the same MOGA (all the settings of the optimizer are the same) to obtain the optimal α solutions of Eq. (5.9) using *Investment* vs. *MEP*, as shown in Figure 5.11(a), for the same single design and Figure 5.11(b) for the same 35 vibrating platform designs. The size of the indifference hyper-cube (in this two-dimensional example, indifference rectangle) is 10Hz for frequency and $5 for the cost. We use the sampling method discussed in Section 5.2.3 for *MED*. The solutions shown in Figure 5.11(a) and (b) are more different than those of using *Investment* and R_f. The possible reason behind might be more Pareto designs can provide more information in the objective function space, which helps determine the entropy more accurately.



(a) For a single Pareto design



(b) For 35 Pareto designs

Figure 5.11 Obtained solutions for *Investment* vs. *MEP* with respect to (a) a single Pareto design and (b) 35 Pareto designs

Again, four typical α solutions are selected in Figure 5.11(b). From α_{I} to α_{IV}, with more increasing in retained c_{A}, the *MEP* value is going up, which confirms the conclusion that reducing c_{A} will bring the largest gain in the uncertainty reduction in the objective functions. It can be observed that the Pareto frontier shape in Figure 5.11, *Investment* vs. *MEP*, is different with its counterpart in Figure 5.9, which demonstrates

the different properties of the two metrics, R_f and *MEP* in the objective space. That is, while *MEP* focuses on the diversity of the objective value variation, R_f is capturing the worst-case scenario in the variation of the objective value. This observation for two metrics in the objective space mutually verifies the applicability of the proposed SA approach.

As mentioned in the previous section, the value of *MEP* depends on the indifference hyper-cube determined according to the preference of the DM. Different indifference hyper-cubes can generate different results in *MEP*. To demonstrate this effect, the size of the indifference hyper-cube is changed to 10Hz for frequency and \$10 for the cost. The same N_{np} = 35 optimal vibrating platform designs in Figure 5.8 are used again. The obtained optimal $\boldsymbol{\alpha}$ solutions for this setting are shown in Figure 5.12.

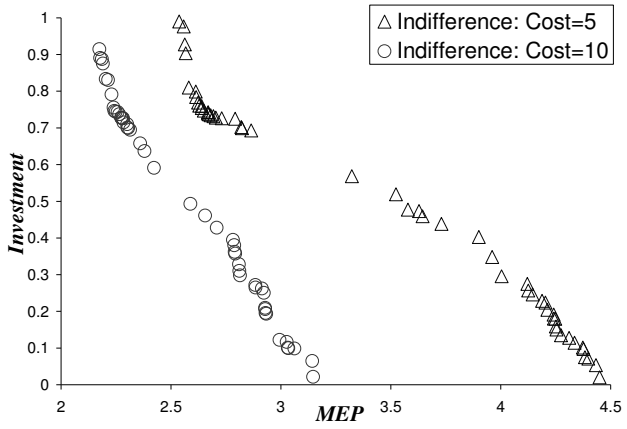


Figure 5.12 Solutions for different indifference hyper-cube, *Investment* vs. *MEP*

In Figure 5.12, the *MEP* values for indifferent \$10 are smaller than the results for indifferent \$5. This is because when indifferent cost is changed from \$5 to \$10, the indifference hyper-cube in the objective space becomes larger. Then for the same $\tilde{\boldsymbol{f}} = \boldsymbol{f}(\boldsymbol{x}_n, \tilde{\boldsymbol{p}})$ realizations for Pareto designs, the number of different events is reduced.

Thus, when the number of indifference hyper-cubes is reduced, the value of *MEP* becomes smaller.

To verify the correlation among all α solutions with *Investment* and *MEP*, we plot the correlation plot figures of α_1, α_2, *Investment* and *MEP* for all optimal α solutions, as shown in Figure 5.13. The same observation as from Figure 5.10 is concluded namely that α_2 is relatively more important than α_1 with respect to the effect on the uncertainty reduction in *MEP* (Figure 5.13(a), (c) and (e)). While α_2 values scatter on the entire range from 0 to 1, most α_1 values are clustered in [0.8, 1] (Figure 5.13(b)). Figure 5.13(c) shows the positive correlation between α_2 and *MEP*. It has been observed that the relative importance of α remain the same as the indifference cost values are changed.

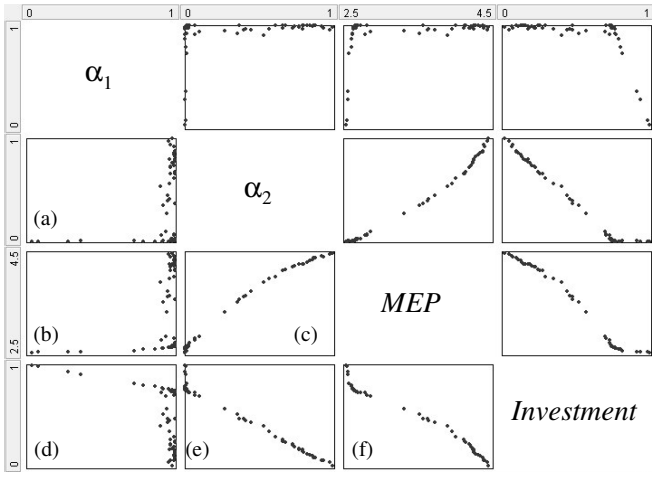


Figure 5.13 Plots of correlations among α, *MEP* and *Investment*

5.3.2 Design of a Grinder

The second example is the design problem of an angle grinder [Williams et al., 2006] described in Section 3.6.4. The design variable, objective and constraint functions are defined from Table 3.2 to Table 3.6.

As a demonstration of this case study, four design variables or parameters have uncontrollable uncertainty, represented by intervals. The standard alternating current voltage V , stator outer radius R_o, stack length L, and the coefficient in the cost function β_2, (ΔV, ΔR_o, ΔL, Δp) = (5v, 3%R_o, 3%L, 0.02); 3%R_o and 3%L mean 3% of the nominal R_o and L values.

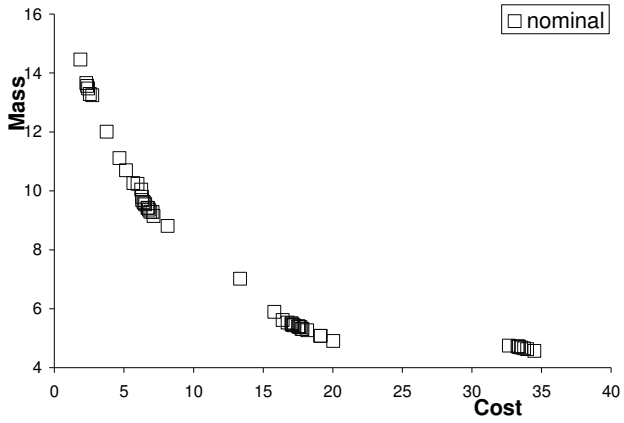


Figure 5.14 Nominal Pareto grinder designs, N_{np}= 40

First we obtain the nominal optimal grinder designs as shown in Figure 5.14, with N_{np}= 40. Then we obtain the optimal α solutions of Eq. (5.9) for *Investment* vs. R_f first, as shown in Figure 5.15 with three typical α solutions.

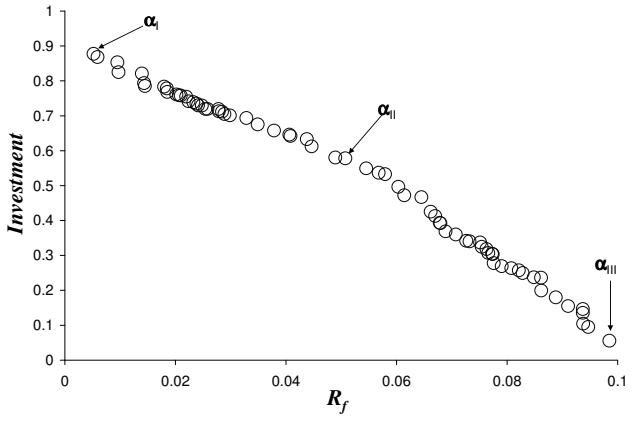


Figure 5.15 Obtained optimal α solutions, *Investment* vs. R_f

Similar to the vibrating platform design example, the R_f (i.e., the variation in the objective values) is decreasing with the increasing of *Investment*. As shown in Table 5.1, $\boldsymbol{\alpha}_{III}$ = [0.986 0.943 0.997 0.982] gives almost (98.6% of ΔV, 94.3% of ΔR_o, 99.7% of ΔL, and 98.2% of Δp) the original tolerance region, which represents the possible maximum uncertainty in the input parameters. $R_f(\boldsymbol{\alpha}_{III})$ is approximate 10% of the nominal objective values, which is almost the biggest variation in the objective space. When the original tolerance region is reduced to $\boldsymbol{\alpha}_{II}$ = [0.971 0.379 0.882 0.498] (which represents 97.1% of ΔV, 37.9% of ΔR_o, 88.2% of ΔL, and 49.8% of Δp), $R_f(\boldsymbol{\alpha}_{II})$ is reduced to 5.1% of the nominal objective values. If the *RTR* is further reduced to the point $\boldsymbol{\alpha}_{I}$ (= [0.750 0.009 0.171 0.051]), $R_f(\boldsymbol{\alpha}_{I})$ is only about 0.5% of the nominal objective values, which is near to the deterministic case. Clearly, as *Investment* values are increasing in this procedure, we eliminate the amount of uncertainty in input parameters and objectives, as shown in Figure 5.15.

Table 5.1 Typical $\boldsymbol{\alpha}$ solutions, *Investment* vs. R_f

	ΔV	ΔR_o	ΔL	Δp	R_f (% of nominal)	Investment
α_I	75.0%	0.9%	17.1%	5.1%	0.5%	87.7%
α_{II}	97.1%	37.9%	88.2%	49.8%	5.1%	57.8%
α_{III}	98.6%	94.3%	99.7%	98.2%	9.9%	5.6%

Different combinations of $\boldsymbol{\alpha}$ values in Figure 5.15 provide grayscale solutions for additional investment, depending on the DM's preference. Similar conclusions can be made according to the metric *Investment* vs. *MEP*, as the Pareto shown in Figure 5.16. Three typical $\boldsymbol{\alpha}$ values are shown in the Table 5.2.

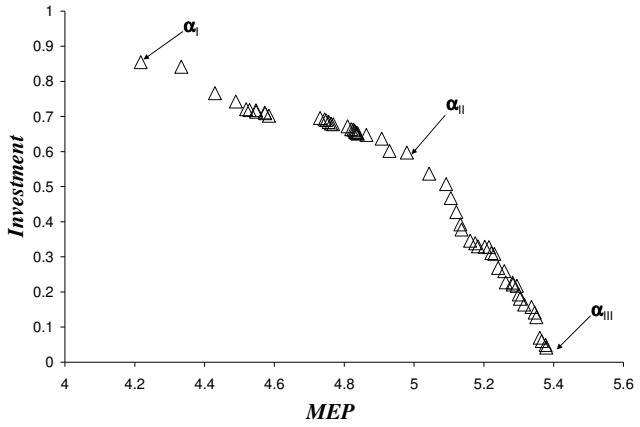


Figure 5.16 Obtained optimal α solutions, *Investment* vs. *MEP*

Table 5.2 Typical α solutions, *Investment* vs. *MEP*

	ΔV	ΔR_o	ΔL	Δp	*MEP* (bits)	Investment
α_I	99.0%	1.3%	14.4%	1.5%	4.2	85.5%
α_{II}	99.4%	31.5%	90.9%	47.0%	5.0	59.7%
α_{III}	99.7%	99.2%	97.1%	97.2%	5.4	4.2%

Again, to determine the relative importance of these four input parameters, we plot the correlation figure of α, R_f, and *Investment*, in Figure 5.17and α, *MEP*, and *Investment* in Figure 5.18. α_1, α_2, α_3, and α_4 are the elements of the *PURI* factor corresponding to ΔV, ΔR_o, ΔL, and Δp, respectively.

As shown in Figure 5.17, it is not necessary to reduce α_1 values (investment in ΔV) as most correlation values are clustered in [0.9, 1]. That is reducing ΔV will not benefit the variation reduction in the objective values. α_1 is not important in uncertainty reduction. α_2 has a relative strong correlation to R_f and *Investment*, which is the most important one for uncertainty reduction in this example, compared to α_3 and α_4. α_3 and α_4 also show similar correlations to the R_f and *Investment*, but not as strong as that of α_2.

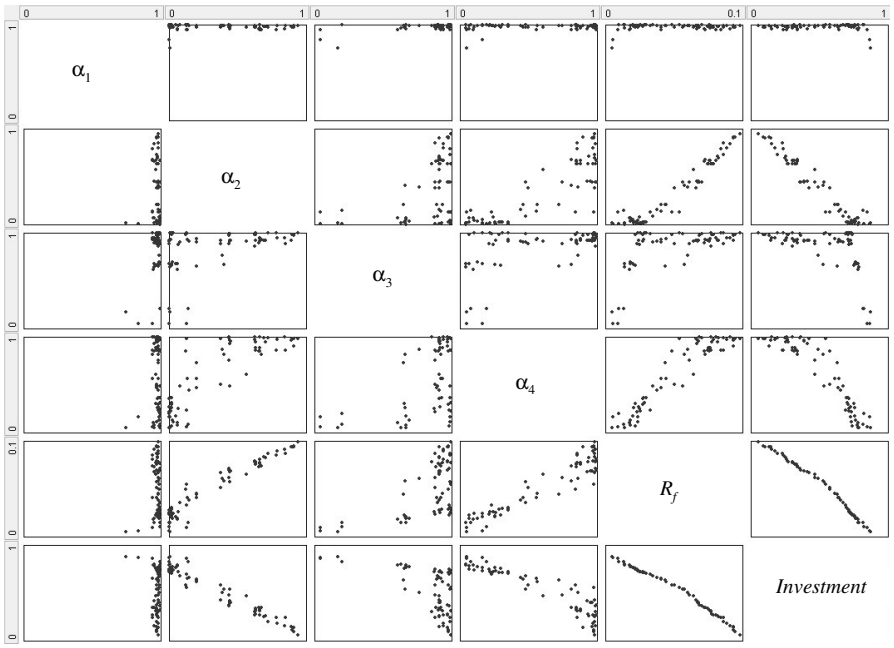


Figure 5.17 Plots of correlations among α, R_f, and *Investment*

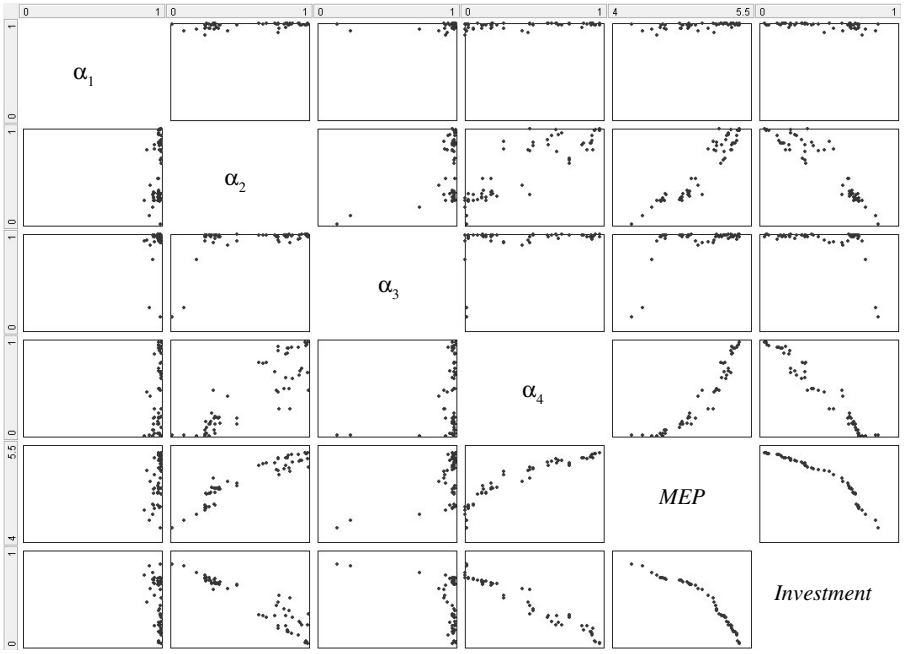


Figure 5.18 Plots of correlations among α, *MEP* and *Investment*

A similar observation can also be found in Figure 5.18. Similar to the R_f metric, α_1 is still the least important one in uncertainty reduction, all of which are in [0.87, 1]. However, α_2 and α_4 now have the comparable correlations to the *MEP* and *Investment*.

Both of them are comparably important in the uncertainty reduction. α_3 is in the middle rank as the importance order, similar to the R_f metric. This observation confirmed that both R_f and *MEP* are similar and can be used to measure the uncertainty in the objective space; though they focus on the different perspectives in the objective uncertainty and may generated similar but different results.

5.4 SUMMARY

A new global SA and uncertainty reduction method is proposed for the design problems with multiple objectives and intervals of uncertainty in this chapter. The proposed method in this chapter quantifies uncertainty as it affects the decision space in the earliest stages of decision making (e.g., for a set of Pareto designs under consideration for future development) but is equally amenable to the more typical analysis of one candidate design. A two objective optimization is formed to find the Pareto solutions that can provide the answer to the question: Which combination of input parameters should be selected for uncertainty reduction to gain the largest improvement in performance. As expected, different grayscales of the retained tolerance regions, instead of leave-one-out, can be selected according to the DM's preference to satisfy the desired variation reduction in the objective values.

The proposed method identifies the uncertainty parameters whose reduction or elimination will produce the largest payoffs in uncertainty reduction for any given investment. The method demonstrates correlations between the metric for retained uncertainty in input parameters and the metrics for the uncertainty in objective values. *Investment*, proposed as a metric for the uncertainty in input parameters, determines the retained tolerance regions of parameter by quantifying the grayscale factor *PURI* α and

appears to provide significant additional information about uncertainties relative to the traditional leave-on-out methods. Two metrics of uncertainty in objective values, R_f and *MEP*, developed from the robustness index from robust optimization approach and Shannon entropy respectively, are proposed and compared in this chapter and provide similar results with slightly difference emphasis. While they both measure uncertainty in objective values, by definition, R_f defines the worst-case deviation of the objective variation from the nominal and *MEP* more focuses on the diversity of possible resulting design performance. The conflict between the *Investment* and R_f or *MEP* reflects the situation that DM would likely face which is to use as little investment as possible or conduct as few investigations as possible to achieve maximum affect in uncertainty reduction of objectives.

In R_f, no probability distribution functions of input parameters are necessary and no sampling strategies are used. An inner optimization problem is used to find R_f. For *MEP* an appropriate sampling method that can capture the maximum amount of uncertainty possible in parameter space and require no more samples than necessary to estimate the maximum Entropy is applied. Additionally, for *MEP* a DM needs to specify the indifferent value for objective (or attribute) values, which affect the obtained optimal *MEP* values.

Two engineering examples of different difficulty have been applied to demonstrate the applicability of the proposed method. Similar uncertainty reduction results are observed. The most critical uncertainty resources are identified and relative importance of input parameters is concluded from the correlations among the *PURI* factors and metrics

in parameter/objective space. Different combinations of *PURI* factors are selected to meet the requirement on the objective variations.

In this chapter, the computational time used for the examples is similar to, but less than the computation time for robust optimization in Chapter 3, using the same computer platform discussed in Section 3.7. It usually takes about 1~2 or 10~15 hours for SA performed with respect to one design or multiple designs, respectively, for both R_f and *MEP* measures.

Next chapter will present a metamodel assisted MOGA approach.

CHAPTER 6: METAMODEL ASSISTED MULTI-OBJECTIVE GENETIC ALGORITHM

6.1 INTRODUCTION

High computational cost of population-based optimization approaches, such as Multi-Objective Genetic Algorithms (MOGAs) [Deb, 2001] or GAs, significantly limits applicability of these approaches for the solution of real-world engineering optimization problems. Researchers have been developing models and methods that improve the efficiency of GAs in terms of the number of simulation (or function) calls. A common strategy to reduce the computational effort of GAs and other optimization methods when integrated with a computationally intensive simulation is to use metamodeling. For such a strategy, there are several types of approaches. The first type is based on fitness approximations in which a metamodel can be constructed based on neural network [Farina, 2001 and 2002], response surface [Lian and Liou, 2004], kriging [Chung and Alonso, 2004], [Li et al., 2007b], or radial basis function [Fang et al., 2004] methods. A comprehensive review of fitness approximation can be found in [Jin, 2005]. The second type of approaches is fitness inheritance models [Chen et al., 2002], [Smith et al., 1995], in which the fitness of an offspring is inherited from its parents. The last type of approaches is based on using metamodeling to guide the search in the design space [Rasheed et al., 2005], [Shan and Wang, 2005], i.e., reproduction of individuals using metamodeling in addition to conventional GA operations. More detailed review of metamodeling approaches can be found in the literature [Simpson et al., 2001], [Simpson et al., 2004] and [Wang and Shan, 2007]. Unfortunately, all of the aforementioned methods heavily depend upon the accuracy of the metamodel used over the entire design

space or some specific neighborhood. Among the above mentioned approaches, the fitness approximation approach is reported to be among the most efficient [Jin, 2005].

The fitness approximation methods are of two types: off-line (non-adaptive) and on-line (adaptive). In off-line approaches, metamodels are developed separately and prior to the start of an optimization algorithm [Papadrakakis et al., 2001], [Wilson et al, 2001], [Koch et al., 2002], [Lian and Liou, 2004], [Chung and Alonso, 2004] and [Fang et al., 2004]. The shortcomings of the off-line methods is that they are difficult to obtain both a good fidelity metamodel over the entire design space and at the same time maintain a low number of simulation calls [Simpson et al., 2001], [Wilson et al, 2001]. The on-line approaches use a combination of metamodels with the simulation model during the optimization procedure while adaptively improving the metamodel [Nair and Keane, 1998], [Farina, 2001 and 2002], [Jin et al., 2001 and 2002], [Nain and Deb, 2003], [Li et al., 2007b]. Most of the on-line methods developed so far are focused on single-objective optimization. The research on how to embed metamodels within MOGAs remains on studied [Farina, 2001 and 2002], [Nain and Deb, 2003], [Li et al., 2007b].

In on-line approaches [Farina, 2001 and 2002], [Jin et al., 2001 and 2002], [Nain and Deb, 2003], rough metamodels are constructed in the initial stages of the GA. These metamodels are then gradually improved as more simulation data become available. Most of this type of approaches utilize neural network, which is known to require a large number of simulation calls [Simpson et al., 2001]. Another unresolved issue in the current adaptive methods is how to objectively decide when to switch to the metamodel instead of using the simulation during the optimization [Jin et al., 2001 and 2002], [Jin, 2005]. Usually the switching between the actual simulation model and the corresponding

metamodel is subjectively decided [Nain and Deb, 2003]. Moreover, the fidelity of the metamodel may vary significantly during the optimization process and this can cause oscillation [Jin, 2005]. Li et al., [2007b], developed a Kriging assisted MOGA (K-MOGA) approach in which adaptive metamodels are embedded within the genetic algorithm for fitness estimation and an objective criterion was used to determine whether the metamodel can be accepted as a substitute to a simulation model. However, the criterion used in [Li et al., 2007b] was devised for a worst case scenario as the distance between two sets, which still can be improved to increase the efficiency of the algorithm.

In this chapter a new and more effective criterion is used for deciding whether the simulation or its metamodel substitute should be used. This objective criterion is developed based on a measure of uncertainty in the prediction from the metamodels, which follows the idea of the worst-case estimation of the objective sensitivity region and the robustness index described in Chapter 3. The criterion can also be adopted to handle the constraint functions. The proposed approach provides further improvement on the saving of simulation calls. It is shown that the new approach, called Circled Kriging MOGA (CK-MOGA), reduces more simulation calls compared to a conventional MOGA or our previously developed K-MOGA [Li et al., 2007b].

The balance of this chapter is organized as follows. Review of K-MOGA and details of the CK-MOGA with the new criterion are presented in Section 6.2. Three examples are given in Section 6.3 to illustrate the applicability of these approaches. Concluding remarks are presented in Section 6.4.

The uncertainty prediction for constraint functions described in this chapter was also presented in Li et al., [2007b].

6.2 CIRCLED KRIGING MOGA (CK-MOGA)

As mentioned in Chapter 2, the response value from the kriging metamodel has uncertainties. With this uncertainty in prediction of simulation response (e.g., objective functions), however, as long as it is determined that the dominance status (recall Section 2.2) of design points in the current generation is not changed because of using the kriging metamodel, it is acceptable to use the kriging metamodel instead of the simulation. If the dominance status is changed, then the design points that are predicted to contribute to this change are observed (i.e., their objective function values are computed using the simulation); otherwise, the metamodel is used to obtain the response values.

Thus, the basic idea behind the kriging assisted MOGAs is to try to ensure that, in each generation, the dominance status does not change because of using the kriging metamodel. In this section, we first focus on a criterion used for objective functions. A similar criterion is developed for constraints as well at the end of this section. In this regard, the uncertainty in prediction represented by predicted RMSE, a byproduct of the kriging metamodel (recall Eq. (2.10)), is used as a main component in the criterion that determines whether the dominance status is changed. This criterion is used to decide when the predicted value from the kriging metamodel can be accepted as a substitute to that from a simulation model for an individual. We will first review the previously proposed criterion in K-MOGA. Then we develop a new criterion based on the idea of the worst-case sensitivity region which will provide a measure of uncertainty in the predicted responses from the kriging metamodel.

6.2.1 Review of the Criterion in K-MOGA

In the K-MOGA approach [Li et al., 2007b], we use a quantitative measure of domination as part of the criterion to determine whether the predicted value from the kriging metamodel should be accepted. This measure is called: Minimum of Minimum Distance (*MMD*). In any generation of GA, except the initial population where all individuals are observed, the kriging metamodel can be used to obtain the predicted objectives of individuals. Based on these predicted response values, the domination status of individuals can be determined. To do this, the current population is partitioned into two sets: dominated and non-dominated sets. Note that this partitioning is based on the kriging metamodel values, that is, no simulation calls are used at this stage.

MMD is defined, in the objective space, as the minimum distance between all pairs of non-dominated $\boldsymbol{x}_{nd}$ and dominated $\boldsymbol{x}_{d}$ points and calculated as follows. First, divide individuals in the current population into two sets: non-dominated and dominated. Then, compute *MMD* by Eq. (6.1)

$$\begin{aligned} MMD = \min\{\| f(\boldsymbol{x}_{nd}) - f(\boldsymbol{x}_{d}) \|_2\} \\ \boldsymbol{x}_{nd} \in \{\text{non - dominated set}\} \\ \boldsymbol{x}_{d} \in \{\text{dominated set}\} \end{aligned} \tag{6.1}$$

where the norm is defined in the $\boldsymbol{f}$-space. *MMD* is then projected along each objective function axis to obtain $MMDf_m$, $m = 1,..., M$, as shown in Figure 6.1.

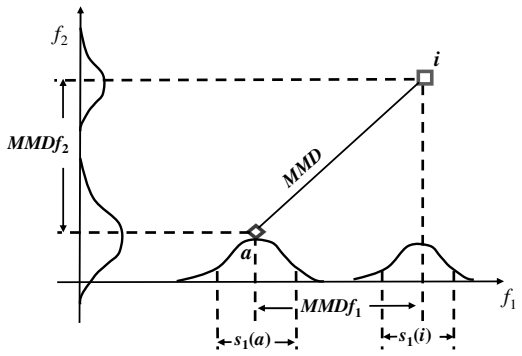


Figure 6.1 *MMD* in K-MOGA

Recall Eq. (2.10) in that the RMSE of an unobserved point $\boldsymbol{x}_0$ for the objective function f_m is $s_m(\boldsymbol{x}_0)$. For objective functions, the criterion for the K-MOGA is devised based on the worst-case scenario such that if $s_m(\boldsymbol{x}) \leq MMDf_m$ is true for all $m = 1,..., M$ and for design point $\boldsymbol{x}$, then the domination status of point $\boldsymbol{x}$ should not change in the current population.

In short, if for any design point $\boldsymbol{x}$ the following holds

$$s_m(\boldsymbol{x}) \leq MMDf_m \tag{6.2}$$

for all $m = 1,..., M$, then the predicted response values (as obtained by kriging metamodels) for $\boldsymbol{x}$ will be considered as "good" values. For those points for which the threshold imposed by Eq. (6.2) does not hold, the simulation will be used to calculate the actual responses. In this regard, the simulation values will help to improve the fidelity of the subsequent kriging metamodels.

Note that there are two main reasons why a point with "large uncertainty" must be observed. First, if the uncertainty in prediction is too large for a design point, then that point should be evaluated by the simulation so that its domination status would not change. Secondly, a point with large uncertainty in prediction implies that the kriging metamodel does not have enough sample points in its vicinity [Sacks et al., 1989]. In

other words, evaluating the point by the simulation and thus using it as a new sample point would improve the accuracy of the kriging metamodel. As a byproduct of our approach, the criterion provides the kriging metamodel with a self-improving mechanism. This is based on the fact that a point with large uncertainty is a potentially good choice for sampling.

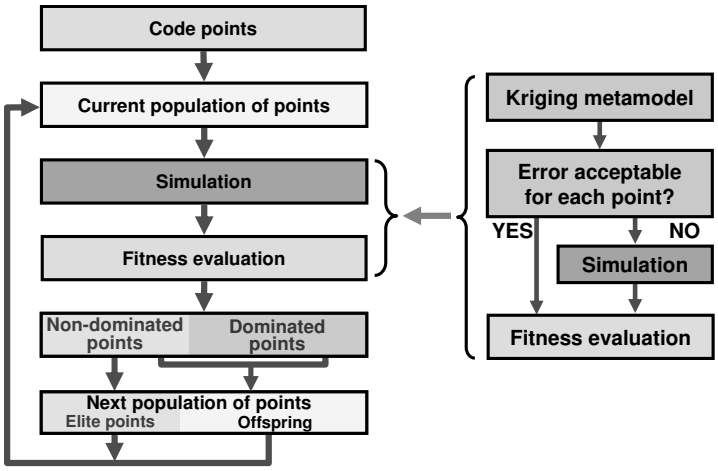


Figure 6.2 Conventional MOGA (left) with CK-MOGA addition (right)

In the conventional MOGA (Figure 6.2 left), the algorithm uses simulation to evaluate all points in the population. In K-MOGA, with a presumed confidence level, if the metamodel's uncertainty in the prediction is acceptable, then instead of the simulation, its metamodel (addition in Figure 6.2 right) is used for fitness evaluation; otherwise, and the simulation is used.

Note that *MMD* is defined based on the distance of two sets: dominated and non-dominated. It is possible that either of $\boldsymbol{x}_{nd}$ or $\boldsymbol{x}_{d}$ has already been evaluated by the simulation in a previous generation, or the distance between dominated set and non-dominated set is too large to show the pair-wise comparison in fitness evaluation, compared to the uncertainty in the prediction. In these cases, Eq. (6.2) is based on a worst-case scenario which did not reflect the non-dominated sorting situation in MOGAs.

Essentially, in fitness evaluation of MOGAs, non-dominated sorting is a pair-wise comparison method to determine the dominance status of each point in the current population or points under consideration. In this regard, it is possible to improve the criterion in MOGA so that the pair-wise comparison can be captured and the worst-case scenario can be avoided.

6.2.2 New criterion in CK-MOGA

In this section, we propose an improved criterion that helps with the decision as to when the switching between the simulation model and kriging metamodel should occur. In this new criterion, the distances of every pair of two points in the current generation are compared to a measure of the uncertainty in predictions from the kriging metamodels. The radius of the uncertainty, similar to the robustness index of uncertainty in the objective values, is used as the measure of uncertainty in this new criterion.

Radius of Uncertainty

Because the objective functions may have incommensurable units and scales, we normalize each objective in the objective space defined as the $\boldsymbol{f}$-space, as described before in Chapter 3. With a presumed confidential level (i.e., two standard deviations), the Uncertainty in Prediction (UP) is defined as in Eq. (6.3):

$$UP_m(\boldsymbol{x}) = 2 \times s_m(\boldsymbol{x}),\ m=1,\ldots,\ M \tag{6.3}$$

The radius of uncertainty R_u, similar to the radius of the objective sensitivity region in Chapter 5, for each individual $\boldsymbol{x}$ is defined as the distance metric $\|.\|_2$ of UP, shown in Eq. (6.4):

$$R_u(\boldsymbol{x}) = \left\| UP(\boldsymbol{x}) \right\|_2 \tag{6.4}$$

Essentially, $R_u(\boldsymbol{x})$ is the radius of a hyper-sphere centered at the predicted objective values of $\boldsymbol{x}$, determining the possible uncertainty in the predicted objective values of $\boldsymbol{x}$. With the presumed confidential level, the true objective values of $\boldsymbol{x}$ should be within this hyper-sphere, as shown in Figure 6.3. (Note that small hollow rectangles/squares shown in Figure 6.3 represent points obtained by the metamodel. Solid rectangles/squares are for points that are observed.) As mentioned in Section 2.6, the quantity $UP_m(\boldsymbol{x})$ is a byproduct of the kriging metamodel.

Given this absolute measure of the uncertainty in the prediction, we still require a quantitative measure of domination to determine whether or not the uncertainty in the prediction is large enough to change the dominance status. To do so, we introduce another measure, called Minimum Distance (*MD*), as described next.

Minimum Distance

MD is defined, in the objective space, as the minimum $\|.\|_2$ distance between any pair of individuals (e.g., designs) in the current generation.

The criterion for the kriging metamodel assisted approach is obtained as follows:

1) $R_u(\boldsymbol{x})$ estimates the maximum possible deviation (with the presumed confidence level) from the true value of objective functions. If the sum of $R_u(\boldsymbol{x}^i)$ and $R_u(\boldsymbol{x}^j)$ is less than the distance between points $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$, then the true value of objective functions for points $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$ should not change the dominance status as shown in Figure 6.3.
2) *MD* is the minimum distance between any pair of points in the current generation. So if $R_u(\boldsymbol{x}^i) + R_u(\boldsymbol{x}^j) \leq MD$, then points $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$ should not change the dominance status.

3) Mathematically, $R_u(\boldsymbol{x}^i) + R_u(\boldsymbol{x}^j) \le MD$ implies that: $\max(R_u(\boldsymbol{x}^i), R_u(\boldsymbol{x}^j)) \le MD/2$.

4) If $2\times R_u(\boldsymbol{x}) \le MD$ is true for any point $\boldsymbol{x}$, then the uncertainty in the prediction of $\boldsymbol{x}$ should not change the dominance status of $\boldsymbol{x}$ in the current population.

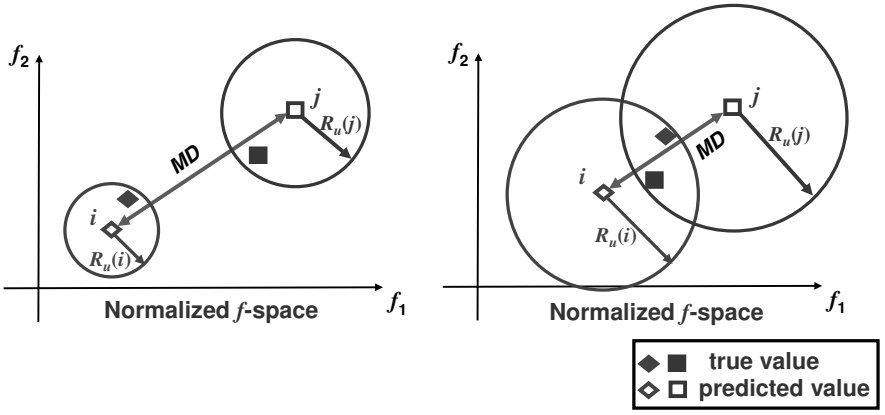


Figure 6.3 *MD* criterion for accepting the predicted value

In short, if for any design $\boldsymbol{x}$

$$2\times R_u(x) \le MD \tag{6.5}$$

then the predicted values of $\boldsymbol{x}$ will be considered as "good" values (i.e., it will not change the dominance status). For those points for which Eq. (6.5) do not hold, simulation will be used to calculate the true value of objective functions. Moreover, as more points are observed the simulation (or actual) values of objective functions will help to improve the fidelity of subsequent kriging metamodels.

Figure 6.2 shows the proposed CK-MOGA procedure in one generation. In the CK-MOGA approach, all points in the initial population are observed (i.e., their responses (or objective function values) are obtained by the simulation) to build the initial kriging metamodels. Since the initial points may be far away from the Pareto frontier or do not entirely fill the design space, the initial kriging metamodels may not be sufficiently accurate. However, these metamodels are adaptively improved as the algorithm evolves, generation by generation. During the early generations of the CK-MOGA, the percentage

of points for which predicted values from kriging metamodels are acceptable is small. However, as more observed points are added to the kriging metamodel the uncertainty in the kriging prediction for unobserved points is gradually reduced, and the percentage of the points for which the kriging metamodel is used increases as more generations are evolved. Note that, according to the criterion in Eq. (6.5), points with a large uncertainty are required to be observed. Thus the general concern in using a kriging metamodel that it should have a reasonable fidelity during the entire optimization algorithm and particularly during the initial stages can be avoided to some extend. Based on our observations, in the CK-MOGA, eventually the kriging metamodels can achieve high fidelity.

It should be noticed that there can be a situation in which the criterion used in this chapter (Eq. (6.5)) can still fail, i.e., a dominated point can be mistakenly considered as a non-dominated point. Suppose $R_u(\boldsymbol{x}^i)$ and $R_u(\boldsymbol{x}^j)$ have no overlap. If both of these points are non-dominated, as shown in Figure 6.4, then the dominance status of point $\boldsymbol{x}^j$ can be changed because of using the kriging metamodels.

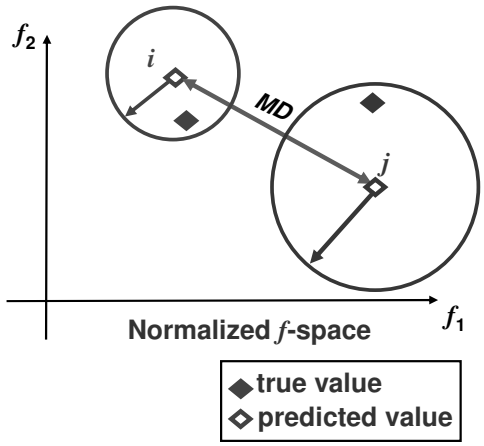


Figure 6.4 Failure of the criterion in Eq. (6.5)

However, the situation described in Figure 6.4 rarely happens, based on our test results, in the early and converging stages of CK-MOGA. In the early stages, it is

unlikely to happen since most of the points are observed. On the other hand, it is possible that in the middle stages of CK-MOGA, the situation in Figure 6.4 may occur and thus decrease the efficiency of the CK-MOGA approach. However, as CK-MOGA converges, more points are observed and the kriging metamodels' error becomes smaller and thus the likelihood of the situation described in Figure 6.4 decreases.

6.2.3 Uncertainty in Prediction for Constraint Functions

Each constraint function can be estimated by a kriging metamodel as well. Here, the criterion that is used as to whether the kriging metamodel or simulation should be used is even more critical than that for the objective functions. That is, the kriging metamodel can be used to determine whether or not a design point is feasible. More precisely, if by using the kriging metamodel it is determined that the design point is infeasible, and then the point is observed. On the other hand, if the point is determined to be feasible by the kriging metamodel, then the criterion in Eq. (6.6) has to be verified, as discussed in the next paragraph.

Similarly as in the objective function case, the RMSE of an unobserved point $\boldsymbol{x}_0$ for the constraint function g_l is $s_l(\boldsymbol{x}_0)$, whereby $s_l(\boldsymbol{x}_0)$ estimates the deviation of the constraint's value from a mean for a presumed normal distribution (recall Section 2.6). That is, with a presumed confidential level (i.e., two standard deviations), the distance from the true constraint value $g_l(\boldsymbol{x}_0)$ to the predicted constraint value from the kriging metamodel $\hat{g}_l(\boldsymbol{x}_0)$ is less than $2\times s_l(\boldsymbol{x}_0)$. On the other hand, the absolute value of $\hat{g}_l(\boldsymbol{x})$ provides a cushion to absorb the uncertainty in the prediction along g_l dimension, as shown in Figure 6.5. If $\hat{g}_l(\boldsymbol{x}_0)$ plus $2\times s_l(\boldsymbol{x}_0)$ is still less than zero, then the predicted

constraint value has a very little chance (i.e., less than 3%) to change feasibility of the design $\boldsymbol{x}_0$. Thus, if for any design point $\boldsymbol{x}$ the following criterion holds

$$2 \times s_l(\boldsymbol{x}) + \hat{g}_l(\boldsymbol{x}) \le 0 \tag{6.6}$$

for all $l = 1,..., L$, then the predicted constraint value $\hat{g}_l(\boldsymbol{x})\ \ l = 1,..., L$ of that design $\boldsymbol{x}$ will be considered to be acceptable. We only check Eq. (6.6) for the predicted feasible designs (i.e., with $\hat{g}_l(\boldsymbol{x}) \le 0\ \ l = 1,..., L$). If a design point is predicted to be infeasible, it is observed by the simulation.

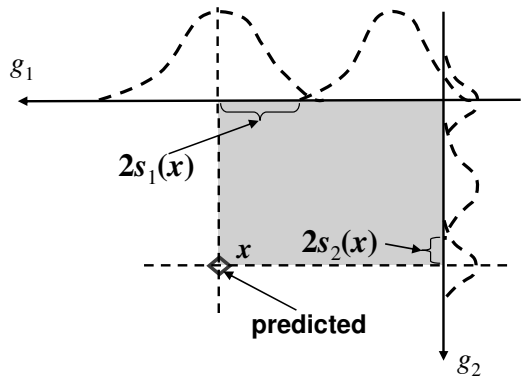


Figure 6.5 Criterion for constraint functions

6.2.4 Stopping Criteria

Since a comparison of the performance of the conventional MOGA and CK-MOGA is important, appropriate and consistent stopping criteria for the MOGA and CK-MOGA should be determined. The following two stopping criteria are used and both have to be satisfied.

1) When the number of non-dominated points is more than some pre-specified percentage of the population size (e.g., 80% for the examples in this chapter) and when it becomes steady (e.g., the number of non-dominated points is more than

"0.8×population size" for five generations for the examples in this chapter), it is concluded that the algorithm has converged to the Pareto frontier.

2) When the iteration history, i.e., the curve representing the number of simulation calls versus the number of generations becomes flat, it can be concluded that the algorithm has been converged since non-dominated points have been a major part in the population.

One may also employ other criteria or metrics for quality assessment of the Pareto frontier [Wu and Azarm, 2001] as additional stopping criteria. We applied two quality metrics proposed previously in the literature [Wu and Azarm, 2001] to compare the performance of the conventional MOGA, K-MOGA and CK-MOGA in terms of convergence and diversity of solutions, as discussed in Section 6.3.3.

6.2.5 CK-MOGA Steps

The steps for the CK-MOGA are as follows:

Step 1: Initialize. Start with generating an initial population. Simulation models are called to calculate the responses (i.e., objective/constraint functions) for individuals in the initial population and these are added to a sample set to build the initial kriging metamodels, one for each objective/constraint function. The non-dominated (or elite) points in the initial population are identified and migrated into the next generation. The remaining (or dominated) points for the next generation are generated by the GA operations. The same initial population is used for all 30 runs of each test example and the designs in the initial population are selected by a maximum entropy design (see, e.g., [Shewry and Wynn, 1987]) to fill out the entire design space.

Step 2: The algorithm evolves into the next generation [Deb, 2001].

Step 3: Apply the current kriging metamodels to predict response values for the current population. By the way of Eqs. (6.5) and (6.6), the individuals in the current generation are partitioned into two parts as follows: (i) Calculate the response values (from objective/constraint functions) and RMSE (recall Eq. (2.10)) for each design point in the current generation using the kriging metamodels. (ii) Calculate *MD*. (iii) Apply Eqs. (6.5) and (6.6) to each point. Individuals whose predicted response values satisfy Eqs. (6.5) and (6.6) will be considered as good: The simulation model for these points is not used but rather their metamodel is used. For the points that do not satisfy Eqs. (6.5) and (6.6), the simulation model is used to calculate their responses.

Step 4: Calculate the fitness value of each point. Non-dominated sorting algorithm [Deb, 2001] is used to calculate the fitness of each point.

Step 5: Identify non-dominated points and update the kriging metamodels. Non-dominated points in the current population are identified. Points (dominated or non-dominated) whose response values are calculated by the simulation are added to the sample set to update the kriging metamodels.

Step 6: Check the stopping criteria. Check the stopping criteria described in Section 6.2.4. If both stopping criteria are satisfied, stop the algorithm; otherwise, continue.

Step 7: Form the next population. The next population includes two parts: elite and offspring points. Go to Step 2.

6.3 EXAMPLES AND DISCUSSION

In this section, we use two numerical and one engineering examples with different degrees of difficulty to illustrate the applicability of the proposed CK-MOGA, compared to the MOGA and K-MOGA. All of these three examples are optimizations with

constraint functions. As a typical example of our results, we use the first example, OSY [Deb, 2001], to present a detailed comparison of the MOGA, K-MOGA and CK-MOGA including the verification via: (i) quality metrics, (ii) *MD*, and (iii) the uncertainty in prediction. We also present the results for the numerical example used in Section 3.6.1. Finally, an engineering example is presented. In order to compare the conventional MOGA, K-MOGA and our proposed CK-MOGA, the same initial population of design points is used for all experiments for each example. Also the MOGA, K-MOGA and CK-MOGA were run for 30 times each to account for randomness in these methods. The values of other genetic parameters are selected according to the description in Section 2.5. The same settings are used for all examples. For simplicity, the conventional MOGA is referred as "MOGA" hereafter.

6.3.1 OSY Example

We applied the MOGA which was described in Section 2.5 and K-MOGA [Li et al., 2007b] and CK-MOGA to this example that has two objective functions, six constraints, and four variables as shown in Eq. (6.7):

$$
\begin{aligned}
\text{minimize} \quad & f_1(x) = -[25(x_1 - 2)^2 + (x_2 - 2)^2 \\
& \qquad + (x_3 - 1)^2 + (x_4 - 4)^2 + (x_5 - 1)^2] \\
& f_2(x) = \sum_{i=1}^{n} x_i^2 \\
\text{s. t.} \quad & \\
& g_1(x) \equiv 2 - (x_1 + x_2) \le 0 \\
& g_2(x) \equiv (x_1 + x_2) - 6 \le 0 \\
& g_3(x) \equiv (x_2 - x_1) - 2 \le 0 \\
& g_4(x) \equiv (x_1 - 3x_2) - 2 \le 0 \\
& g_5(x) \equiv (x_3 - 3)^2 + x_4 - 4 \le 0 \\
& g_6(x) \equiv 4 - ((x_5 - 3)^2 + x_6) \le 0 \\
& 0 \le x_1, x_2, x_6 \le 10, \quad 1 \le x_3, x_5 \le 5 \quad 0 \le x_4 \le 6
\end{aligned}
\tag{6.7}
$$

The Pareto frontiers from the MOGA, K-MOGA and CK-MOGA, respectively, are non-convex as shown in Figure 6.6. For this example, two separate kriging metamodels

for the two objectives and six separate kriging metamodels for the six constraints are built in the K-MOGA and CK-MOGA; and all of them are adaptively improved to predict the response for the objective and constraint functions.

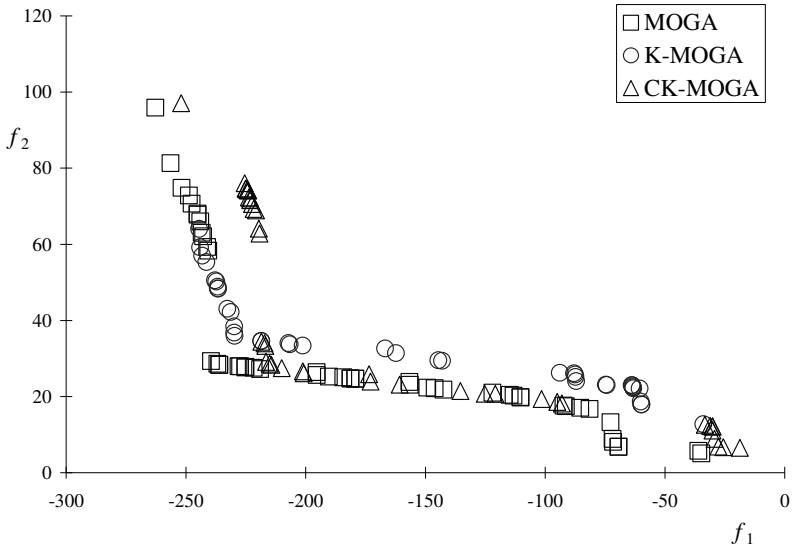


Figure 6.6 Pareto solutions for OSY

Figure 6.6 shows a typical set of Pareto optimal solutions as obtained from one of the 30 runs of the MOGA, K-MOGA and CK-MOGA. The results from CK-MOGA are in good agreement with the MOGA and K-MOGA. Figure 6.7 shows the *NumSimCall* (number of simulation calls) for 30 different runs. As shown in Figure 6.7, a MOGA run with the least number of simulation calls (i.e., 1455 in run 18) requires more simulation calls than a CK-MOGA run with the maximum number of simulation calls (i.e., 799). Compared to the *NumSimCall* for the K-MOGA (the mean value is 901 and the standard deviation (STD) is 126.1), the CK-MOGA's performance is better as the mean value is decreased to 673 and the STD is 83.2. The mean values and STDs for all 30 runs for the MOGA, K-MOGA and CK-MOGA are also shown in Table 6.2. The results show that for OSY example, the *NumSimCall* has been reduced by more than 60% using the

proposed CK-MOGA compared to the MOGA, and by 25% using the CK-MGOA compared to the K-MOGA.

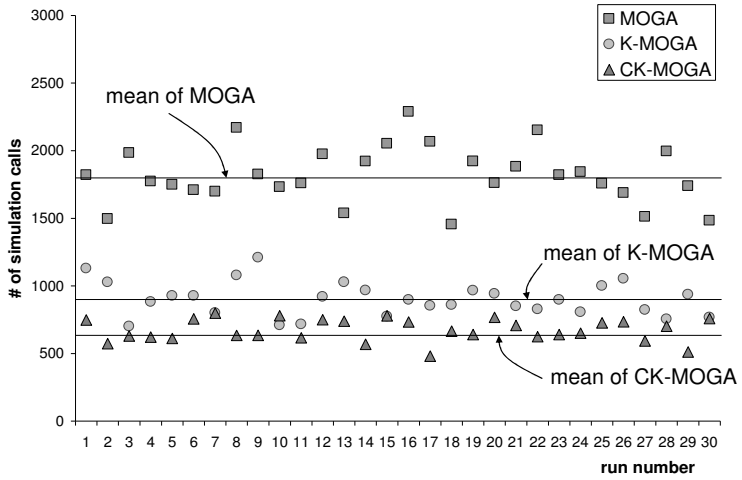


Figure 6.7 # of simulation calls (*NumSimCall*) vs. run number for OSY

6.3.2 Verification by Quality Metrics

In order to evaluate the quality of convergence and diversity of solutions for the proposed CK-MOGA and compare the results with the MOGA, two quality metrics proposed in the literature [Wu and Azarm, 2001], i.e., the Hyperarea Difference (*HD*) and Overall Spread (*OS*) metrics, are calculated for the OSY example.

Figure 6.8 shows the geometrical interpretation of these two metrics in a two-objective space. Let us assume $P = \{a, b, c, d\}$ be the current non-dominated set in the objective space and p_{bad} and p_{good} are the extreme "good" and "bad" points, respectively. The quantity *HD*, shown as the shaded area in Figure 6.8(a), is defined as the difference between the area (hyperarea (*HA*) or volume if there are three or more objectives) bounded by p_{bad} and p_{good} and the area bounded by p_{bad} and the set P:

$$HD(P) = HA(p_{bad}, p_{good}) - HA(p_{bad}, a, b, c, d) \tag{6.8}$$

The quantity *OS*, shown in Figure 6.8(b), is defined as the ratio between the area bounded by the two extreme points in *P* and the area bounded by p_{bad} and p_{good}:

$$OS = \frac{HA[extremes(P)]}{HA(p_{bad}, p_{good})}. \tag{6.9}$$

The quantities *HD* and *OS* serve as the quality metrics of convergence and diversity, respectively, for the obtained Pareto frontier.

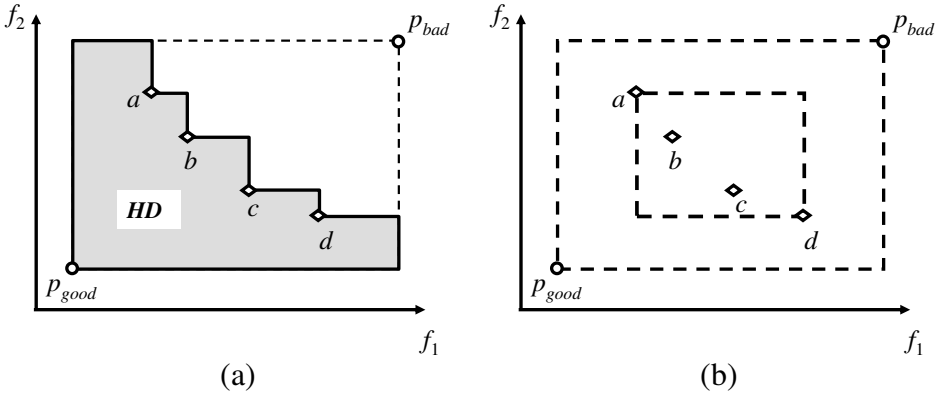


Figure 6.8 Quality metrics (a) *HD* and (b) *OS*

In the OSY example, we set p_{bad} = [0, 150] and p_{good} = [-300, 0] as their objective function values. We calculate the *HD* and *OS* values for the MOGA and CK-MOGA from the 30 runs as shown in Table 6.1. From Table 6.1 it can be seen that the convergence and diversity of the obtained Pareto frontiers from the MOGA and CK-MOGA are comparable. The convergence and diversity of the obtained Pareto solutions for the K-MGOA are also compared to the MOGA, as discussed in [Li et al., 2007b].

Table 6.1 Quality metrics for MOGA and CK-MOGA for OSY example

	MOGA			CK-MOGA		
	30 runs	Mean	STD	30 runs	Mean	STD
HD	[0.08-0.26]	0.148	0.048	[0.07-0.18]	0.128	0.023
OS	[0.11-0.52]	0.243	0.120	[0.09-0.41]	0.217	0.082

6.3.3 Verification by *MD*

As a further verification of the proposed approach, in Figure 6.9, we compare *MD* value calculated from simulation only with calculated from the kriging metamodels for the OSY example. Note that in our CK-MOGA approach, *MD* is calculated from the kriging metamodels, indicated as "kriging" in Figure 6.9. According to our experiments, the estimated *MD* as obtained in our approach is less than or equal to *MD* from the simulation for most generations (except in 37^{th} generation). From these results we can conclude that *MD* from kriging provides a good estimate of the actual *MD*. The same results were observed for other examples.

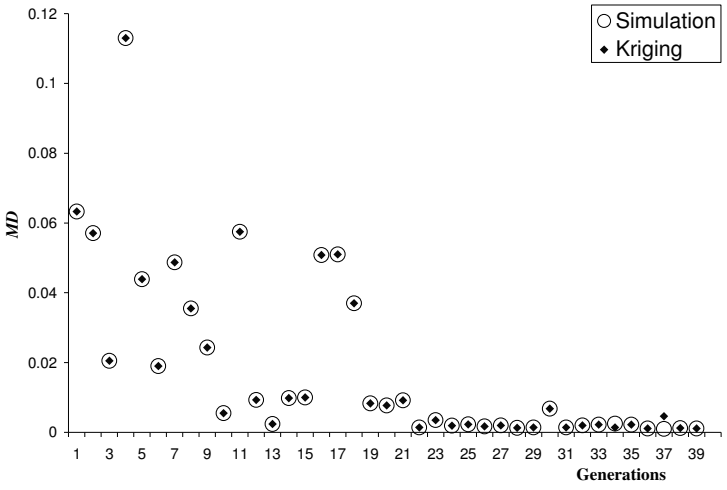
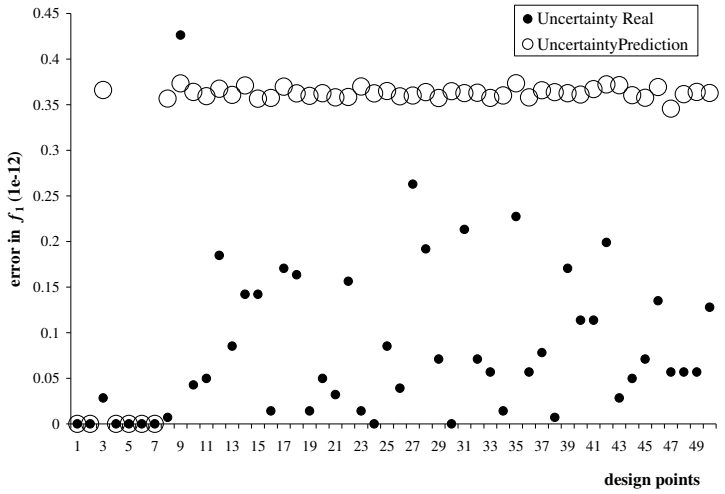


Figure 6.9 *MD* based on simulation and kriging metamodel

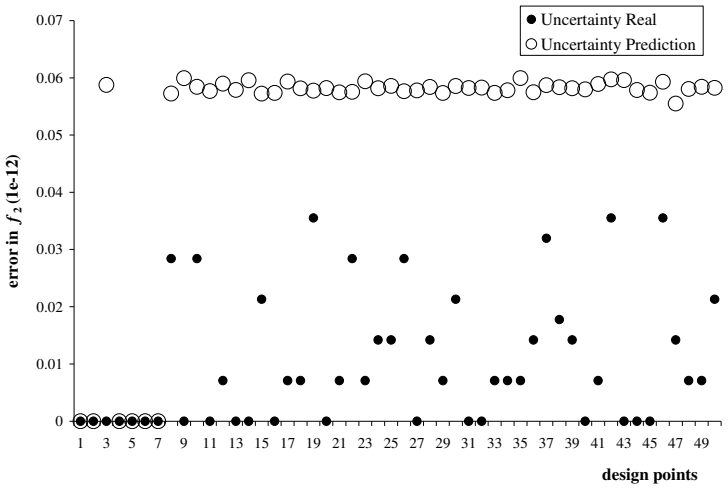
6.3.4 Verification by Uncertainty in Response Prediction

For verification as to whether the uncertainty in response prediction $s_m(\boldsymbol{x})$ is a valid

estimation of deviation in Eq. (6.3), we have obtained the uncertainty which is the absolute value of the difference between the actual and predicted values of $\boldsymbol{f}$ (for both f_1 and f_2 in this case) for each design point in a typical generation (e.g., the 10th generation in Figure 6.10 for the OSY problem). The term "Uncertainty Real", as shown in Figure 6.10, is for the deviation from the predicted value (from kriging metamodel) to the actual value (from simulation). The "Uncertainty Prediction" term is the $UP_m(\boldsymbol{x})$ (which is equal to $2\times s_m(\boldsymbol{x})$) and calculated from the kriging metamodel as in Eq. (6.3). As shown in Figure 6.10, for most design points (except the 9th point) in the 10th generation, the Uncertainty Real is less than the UP_m, which means that $UP_m(\boldsymbol{x})$ is a valid estimation of the standard deviation. Similar results were observed for the uncertainty in the constraint estimation, and for other generations and test problems.



(a)



(b)

Figure 6.10 Real and predicted uncertainty for (a) f_1, and (b) f_2 for OSY in the 10th generation

6.3.5 Additional Examples

In this section, two additional examples: the numerical example in Section 3.6.1 and a new cabinet design example are presented to demonstrate further applicability of the CK-MOGA. The comparison results of the number of simulation calls (*NumSimCall*) with STDs as obtained from the MOGA, K-MOGA, and CK-MOGA are shown in this section in Table 6.2.

Numerical Example

Here we show the formulation of the numerical example again. This is a bi-objective optimization problem with three design variables and two constraint functions, as shown in Eq. (6.10). The obtained Pareto solutions using the MOGA, K-MOGA and CK-MOGA are shown in Figure 6.11.

$$
\begin{aligned}
&\min_{x} \quad f_1 = x_2^{\,2} + x_3 + y_1 + e^{-y_2} \\
&\min_{x} \quad f_2 = x_1 + \sqrt{x_2} + (y_2^{\,2} - y_1^{\,3}) \big/ 10^{\,4} + 150 \\
&\text{s.t.} \quad g_1 = 8 - y_1 \le 0 \\
&\qquad g_2 = y_2 - 10 \le 0 \\
&\qquad -10 \le x_1 \le 10\,,\ 0 \le x_2, x_3 \le 10 \\
&\text{where} \quad y_1 = \boldsymbol{Y}_1(\boldsymbol{x},\, y_2) = x_1^{\,2} + x_2 - 0.2\, y_2 \\
&\qquad\quad y_2 = \boldsymbol{Y}_2(\boldsymbol{x},\, y_1) = x_1 + x_3 + \sqrt{y_1}
\end{aligned}
\tag{6.10}
$$

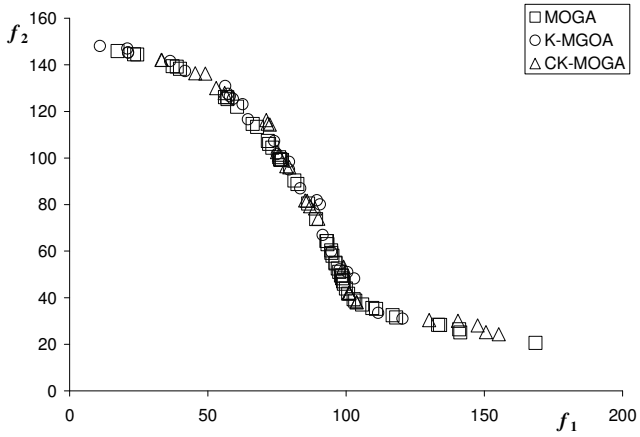


Figure 6.11 Obtained Pareto solutions using MOGA, K-MOGA and CK-MOGA for the numerical example

Cabinet Design

We apply the MOGA, K-MOGA and CK-MOGA to a more complex cabinet model application in a thermal cooling design for a PCM cabinet. The thermal analysis model (as shown in Figure 6.12) for this example was developed and updated by researchers at Georgia Tech (Dr. Yogendra Joshi) [Li et al., 2007a]. The optimization model is described in Figure 6.12. There are two design objectives for this problem. The first objective is to minimize air "inlet temperature" from the upper bay of the cabinet to the lower bay's cooling system (heat exchanger or HX). Another objective is to minimize the total power, including fan and water pumping power. The constraints are an upper bound on the air inlet temperature and lower and upper bounds on the three design variables:

"Air flow rate", "Water flow rate", and "number of HX units". Typical Pareto frontiers obtained from MOGA, K-MOGA and CK-MOGA, respectively, for this example are shown in Figure 6.13.

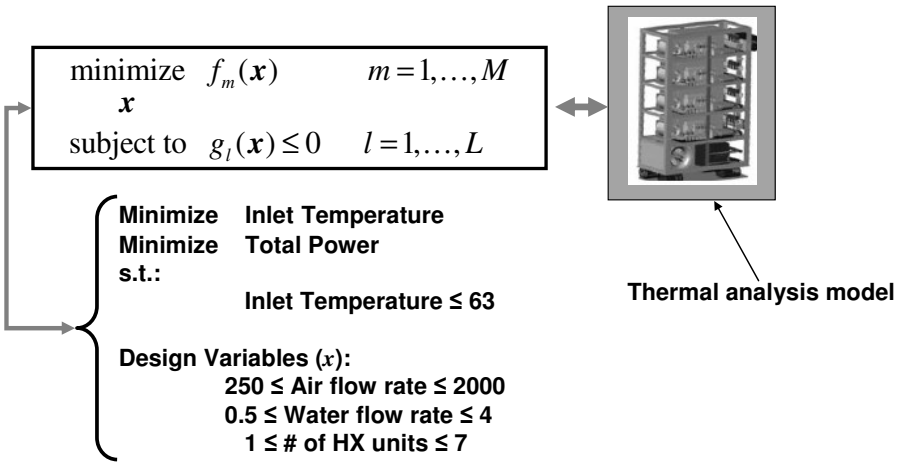


Figure 6.12 Cabinet design formulation

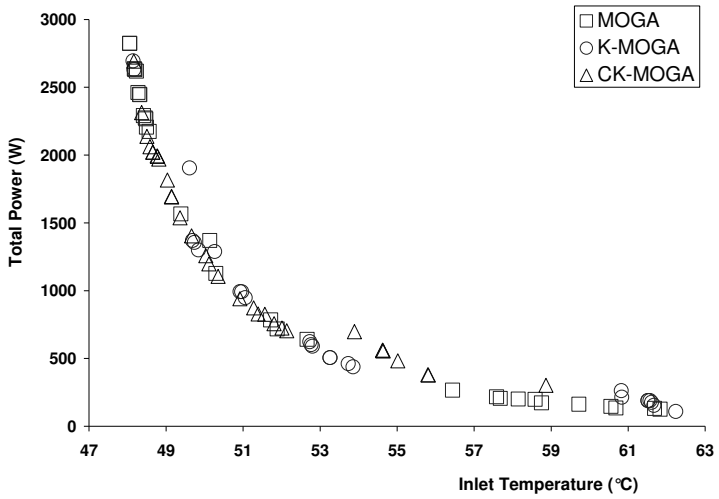


Figure 6.13 Obtained Pareto solutions using MOGA, K-MOGA and CK-MOGA for cabinet design

6.3.6 Comparison of MOGA, K-MOGA and CK-MOGA

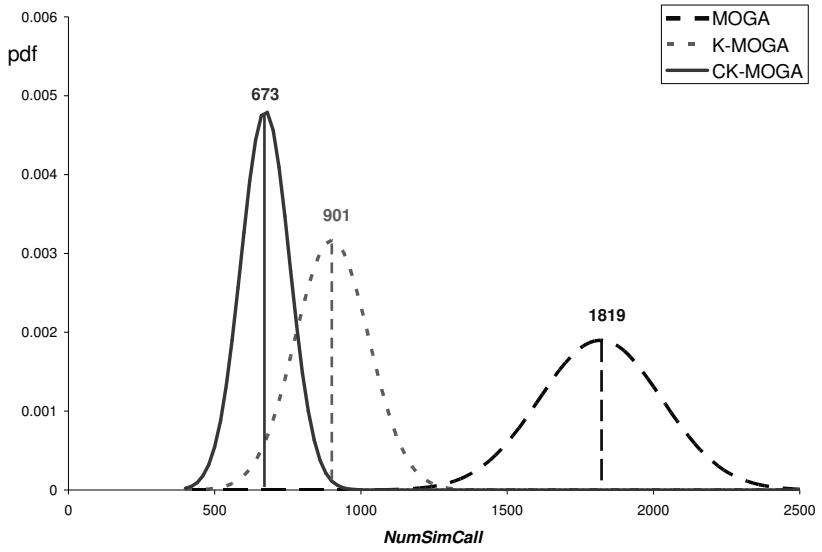
The obtained results for these three examples show that the number of simulation calls (*NumSimCall*) used in the CK-MOGA is significantly fewer than the MOGA, and also fewer than the K-MOGA, while the obtained Pareto solutions from these three

methods are comparable. Furthermore, as shown in Table 6.2, the CK-MOGA has smaller STD of the *NumSimCall* (based on 30 runs) than the MOGA and K-MOGA, which indicates that compared to the MOGA and K-MGOA, the CK-MOGA has a more stable performance on the reduction of the *NumSimCall*.

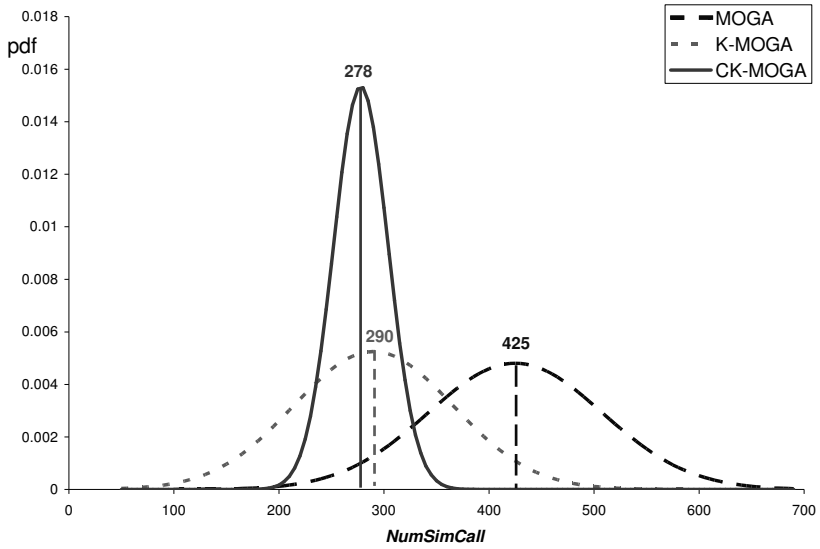
Table 6.2 Statistics for the *NumSimCall*

Example (popsize)	*NumSimCall*								
	MOGA			K-MOGA			CK-MOGA		
	30 runs	Mean	STD	30 runs	Mean	STD	30 runs	Mean	STD
OSY (50)	[1455-2288]	1819	210	[701-1210]	901	126	[480-799]	673	83
Numerical (40)	[269-578]	425	83	[169-399]	290	76	[198-327]	278	26
Cabinet (40)	[392-581]	472	39	[167-258]	210	18	[133-194]	159	14

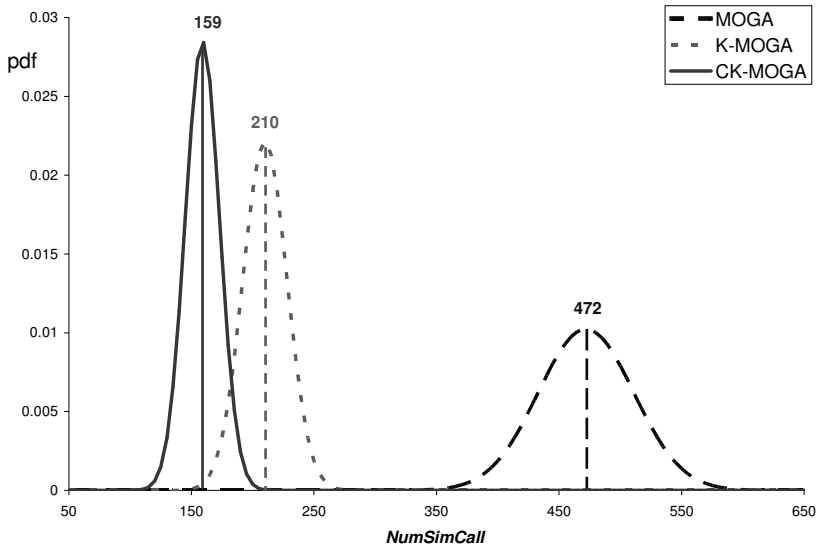
Based on the data in Table 6.2, the reduction of the *NumSimCall* for each example is calculated based on the mean and STD value. If we assume the *NumSimCall* follows the normal distribution, the statistics results of three test examples are shown in Figure 6.14.



(a) OSY



(b) Numerical



(c) Cabinet

Figure 6.14 Statistic results (or probability density functions) for (a) OSY (b) Numerical and (c) Cabinet

This calculation performing for the K-MOGA over the MOGA, the CK-MOGA over the MOGA, and the CK-MOGA over the K-MOGA, separately, is also shown in Table 6.3.

Table 6.3 Reduction in the *NumSimCall*

Example (*popsize*)	Reduction in the *NumSimCall*					
	$1-\left(\frac{\text{K-MOGA}}{\text{MOGA}}\right)$		$1-\left(\frac{\text{CK-MOGA}}{\text{MOGA}}\right)$		$1-\left(\frac{\text{CK-MOGA}}{\text{K-MOGA}}\right)$	
	Mean	STD	Mean	STD	Mean	STD
OSY (50)	50%	40%	63%	60%	25%	34%
Numerical (40)	32%	8%	35%	69%	4%	66%
Cabinet (40)	44%	54%	66%	64%	24%	22%
on the average	42%	34%	55%	64%	18%	41%

As shown in Figure 6.14 and Table 6.3, on the average, the proposed CK-MOGA can save about 55% in the *NumSimCall* over the MOGA, while the K-MOGA can save about 42% over the MOGA. In other words, the CK-MOGA uses about 18% fewer simulation calls over the K-MOGA. The STD in the CK-MOGA is also reduced 64% when compared to MOGA, and 41% compared to K-MOGA. It is observed that the CK-MOGA outperforms both the MOGA and the K-MOGA and are more stable than the MOGA and the K-MOGA, in terms of the number of simulation calls, for these three examples.

Since in the MOGA, K-MOGA and CK-MOGA, we used the same *popsize* and the *MaxNumGeneration* (the *NumGeneration* in these MOGAs is determined by the stop criteria and less than the *MaxNumGeneration*), the upper bound of the *NumSimCall* in the CK-MOGA will not be larger than $popsize \times MaxNumGeneration$. In the CK-MOGA, rep_{genc} (the number of new offspring in *genc*-th generation) is not changed as the dominance status of individuals is not changed. Also, it is observed that the *NumGeneration*'s are comparable for the MOGA, K-MOGA and CK-MOGA when stopping criteria are satisfied. Since some individuals in the *genc*-th generation are evaluated by kriging metamodels, thus the *NumSimCall* in the CK-MOGA is approximately equal to $\sum_{genc=1}^{NumGeneration}(MF_{genc} \times rep_{genc})$, in which MF_{genc} is the fraction of

unobserved individuals to the number of new offspring. According to our results, MF_{genc} in the CK-MOGA is approximately 0.55, on average, for these three examples.

6.4 SUMMARY

An improved multi-objective design optimization approach called CK-MOGA is presented in this chapter. In the proposed approach, the kriging metamodel is embedded within a conventional MOGA, similar to the previously proposed K-MOGA. However, compared to the MOGA and K-MOGA, CK-MOGA reduces the number of simulation calls by applying a new objective criterion based on the radius of uncertainty in the response prediction from the kriging metamodel and evaluating some individuals in the population by the kriging metamodel instead of the simulation. And the performance of the CK-MOGA in the reduction of simulation calls is more stable than K-MOGA. A new concept of the minimum distances (*MD*) and its relation with the uncertainty in the prediction that is easily obtained from kriging are derived. This criterion is developed based on the radius (or robustness index) of the OSR developed in Chapter 3, and then used to identify those individuals in the population that can be evaluated using kriging metamodels. The identified individuals are those that do not change the estimated dominance status in the objective space and do not change the estimated feasibility for the current generation. For other individuals in the generation, the responses are obtained from the simulation and used to adaptively update the next generation kriging metamodels so that more individuals can be evaluated by the updated kriging metamodels and thus an additional number of simulation calls can be saved in subsequent generations.

In the CK-MOGA, the general concern that the metamodel may be of low fidelity and that it may even produce false optima can be avoided to some extend. The proposed

criterion is objective rather than subjective and can be applied to other population-based optimization methods. The main advantage of using on-line kriging is that the uncertainty in the prediction of the response can be obtained without much extra computational effort in kriging. One of the advantages of adaptive approaches is that those points migrated from previous generations with incorrectly estimated kriging variance are most likely to be removed from the population by a more accurate kriging metamodel. Therefore, the side effect of such migrated points can be reduced when the kriging metamodels are updated adaptively in consecutive generations. In essence, the proposed CK-MOGA has a self-correcting mechanism in terms of identifying "good" points for kriging metamodeling.

Three examples of both numerical and engineering types are used to demonstrate the applicability of the proposed CK-MOGA. The results show that the CK-MOGA is able to achieve comparable convergence and diversity of the Pareto frontier as to that obtained from the MOGA and K-MOGA, while at the same time significantly reducing the number of simulation calls.

Next chapter will present the conclusions, main contributions and future research directions of this research work.

CHAPTER 7: CONCLUSIONS

In this dissertation, we presented the development of novel optimization methods for robust optimization and sensitivity analysis for multi-objective single- and multi-disciplinary design problems. After presenting our research motivation, objectives and terminologies in Chapters 1 and 2, we discussed the results from four research thrusts: 1) A new approach for all-at-once multi-objective and feasibility robust optimization (Chapter 3); 2) McRO approach for performance and collaborative robust optimization (Chapter 4); 3) Interval uncertainty reduction and sensitivity analysis approach for all-at-once multi-objective design problems (Chapter 5); 4) CK-MOGA approach as an efficient multi-objective optimization algorithm (Chapter 6).

This chapter is organized as follows. First in Section 7.1, the concluding remarks for the four research thrusts are summarized. The main contributions of this dissertation are discussed in Section 7.2. Finally, some future research directions are outlined in Section 7.3.

7.1 CONCLUDING REMARKS

In this section, we present the concluding marks for each research thrust separately.

7.1.1 Performance Robust Optimization in Single-Disciplinary Optimization

In Chapter 3, we presented a deterministic non-gradient based approach that uses robustness indices in robust multi-objective single-disciplinary (all-at-once) optimization problems where parameter uncertainty causes variation in the objective and constraint values. This so called Performance Robust Optimization approach is applicable for optimization problems that have discontinuous objective and constraint functions with

respect to uncontrollable parameters. This approach can be used for objective or feasibility robust optimization, or both together.

In our approach, the known parameter tolerance region maps forward into the sensitivity regions in the objective and constraint spaces. The sensitivity regions can be oddly shaped, or be disconnected. In this regard, we define a worst-case estimate, the radius of the worst-case sensitivity regions. Rather than calculating the sensitivity regions directly, we use an optimizer to solve for these worst-case estimates for the OSR and the CSR. The overall robust optimization problem thus becomes an outer-inner optimization problem. We use a MOGA for the outer multi-objective optimization problem and a GA for the inner single-objective problems. The robustness measures are indices calculated from the sizes of the acceptable objective and constraint variation regions and from worst-case estimates of the sensitivity regions. Five numerical and engineering examples are used to demonstrate the applicability of the proposed approach.

The main shortcoming of this approach is the computational efficiency. The outer-inner structure can make this approach computationally intensive. Approximation methods should be used to alleviate the difficulty in the future.

Another concern is how to decide the appropriate acceptable objective variation range. Note that knowing upfront whether robust solutions exist for a presumed AOVR is difficult, if not impossible. This is because we have not assumed any mathematical form for the objective functions (e.g., they can be discontinuous with respect to uncertain parameters). In Chapter 3 we assume that robust solutions exist with a presumed AOVR. If they do not and if it is permissible, the AVOR is iteratively enlarged until such solutions do exist. A possible approach to tackle this problem is for the DM to start with

an initial sensitivity analysis study as proposed in Chapter 5, which can help provide a reasonable estimate of the sizes of OSRs.

7.1.2 Performance and Collaborative Robust Optimization in Decentralized MDO

As presented in Chapter 4, we proposed a new robust optimization approach to handle the decentralized MDO problems with uncertainty, which is called Multi-objective collaborative Robust Optimization (McRO). Other than the all-at-once optimization approach in Chapter 3, real-world engineering design optimization problems often involve systems that have coupled subsystems or disciplines with uncontrollable variations in their parameters at system and subsystem levels. No approach has yet been reported in the engineering design for the solution of these design optimization problems when there are multiple objectives in each discipline, mixed continuous-discrete variables, and when there is a need to account for uncertainty and also uncertainty propagation across disciplines. We present the McRO approach for this class of problems that have interval uncertainty in their parameters. McRO obtains decentralized Multi-disciplinary Design Optimization (MDO) solutions which are as best as possible in a multiobjective and multidisciplinary sense.

For McRO solutions, the sensitivity of objective and/or constraint functions is kept within an acceptable range. McRO involves a technique for the interdisciplinary uncertainty propagation, which is called Collaborative Robustness. To satisfy this Collaborative Robustness, the variation of couplings must be enclosed within the acceptable variation ranges of targets, which provide the cushion to absorb the uncertainty propagated across disciplines. The approach can be used for robust optimization of MDO problems with multiple objectives, or constraints, or both together

at system and subsystem levels. Results from an application of the approach to a numerical and two engineering examples are presented. It is observed that the McRO approach solved fully coupled MDO test problems with interval uncertainty and can obtain solutions that are comparable to an all-at-once robust optimization approach.

Since McRO is developed based on the Performance Robust Optimization in Chapter 3, the main shortcomings of McRO are still the computational efficiency and the determination of existence of robust solutions as discussed in the Section 7.1.1.

7.1.3 Interval Uncertainty Reduction and Sensitivity Analysis in Single-Disciplinary Optimization

As presented in Chapter 5, we presented an uncertainty reduction and sensitivity analysis approach for multi-objective problems with interval uncertainty in an all-at-once format.

Uncertainty analysis and sensitivity analysis has received significant attention in engineering design. While sensitivity analysis methods can be global, taking into account all variations, or local, taking into account small variations, they generally identify which uncertain parameters are most important and to what extent their effect might be. The extant methods do not, in general, tackle the question of which ranges of parameter uncertainty are most important or how to best allocate investments to partial uncertainty reduction in parameters under a limited budget. The methods that can address these questions with multiple objectives still remain sparse.

The proposed method in this dissertation is not only applicable to the sensitivity analysis of a single design with interval uncertainty, but also can quantify uncertainty as it affects a set of designs. Two new global uncertainty metrics, radius of objective

sensitivity region and multi-objective entropy performance, are presented. With these metrics a new optimization method is developed that finds investments and fractional levels of parameter uncertainty reduction that provide the greatest payoff for system performance. Two case studies of varying difficulty are presented to demonstrate the metrics and resulting investment information gleaned from the proposed approach.

7.1.4 Metamodel Assisted MOGA

As presented in Chapter 6, we discussed an improved metamodel assisted MOGA approach to improve the computational efficiency of a population-based multi-objective optimization approach. We present an improved MOGA, called Circled Kriging MOGA (CK-MOGA) in which the kriging-based metamodel is embedded within a MOGA.

In the CK-MOGA, some of the design points are evaluated on-line using kriging metamodels instead of the actual simulation model. The decision as to whether the simulation or its kriging metamodel should be used for evaluating a design point is based on objective criterion which is developed from a worst-case distance measure of uncertainty from the predicted responses. The criterion is applied for the objective and constraint functions. It is determined whether by using the objective/constraint functions' kriging metamodels for a design point, its dominance status in the current generation can be changed. The results show that, on the average, CK-MOGA outperforms both a conventional MOGA and our recently developed Kriging MOGA and has higher stability in terms of the number of simulation calls used in the optimization.

The main shortcoming of this CK-MOGA is the number of simulation calls used in the approach cannot be determined *a priori*. That is, in the CK-MOGA, the positions of new offspring are controlled by GA operations and whether an offspring should be

observed or not is controlled by the proposed objective criterion. Thus, since the kriging metamodel depends on the shape of the actual response from the simulation and on the observation of points used to build up and update the kriging metamodels, estimating a lower bound of on the number of simulation calls prior to the start of the algorithm is very problematic, if not impossible.

7.2 MAIN CONTRIBUTIONS

In this dissertation, we have introduced and discussed several new approaches for robust optimization and sensitivity analysis for multi-objective multi-disciplinary design optimization problems with efficient optimization methods.

The proposed Performance Robust Optimization approach is a new deterministic non-gradient based robust optimization approach that uses robustness measures in multi-objective optimization problems where uncontrollable parameters variations cause variation in the objective and constraint values.

- The approach is applicable for cases that have discontinuous objective and constraint functions with respect to uncontrollable parameters
- Only intervals, instead of probability distributions, are necessary for the problem
- Variations in the objectives and constraints are quantified by a single measure

The proposed McRO approach is a new non-gradient based robust optimization approach that can solve the decentralized MDO problems where there are multiple objectives in each discipline, mixed continuous-discrete variables, and when there is a need to account for uncertainty and also uncertainty propagation across disciplines.

- McRO preserves disciplinary autonomy in MDO problems with multiple disciplines that are fully coupled
- McRO solves MDO problems with multiple objectives/constraints in each discipline
- McRO handles uncertainty within and across disciplines by extending enclosure criterion

The proposed Sensitivity Analysis approach is the first approach applicable for the problem with multiple objective (or multiple outputs) with respect to a family of designs. This approach identifys the most important ranges of uncertainty in all parameters, and more importantly provides the decision maker with multiple optimal solutions for investing in uncertainty reduction as measured by performance variation. The approach is also capable for suggesting uncertainty reduction investments with respect to a family of designs which are typically under consideration by the DM in the early stages of system development.

- The proposed SA approach solves the problem with multiple objectives, with respect to multiple optimal designs
- Compared to currently reported SA methods, the optimal solutions from the SA method can provide the DM multiple choices on diffferent levels of investments in order to achieve acceptable (or desired) variations on the output side
- As a result of the proposed approach, the relative importance of input parameters can be identified

The proposed CK-MOGA approach is a new approach applicable for the problems with multiple objectives in which an objective criterion is developed and used to determine whether the simulation model or the metamodel should be used to obtain the response values. The approach is shown to be applicable to both the objective and constraint space.

- The CK-MOGA has an objective criterion to use simulation models or metamodels for both objective and constraint functions
- Uncertainty in the predicted response from the CK-MOGA is quantified by a single distance measure
- The CK-MOGA can significantly reduce the number of simulation calls compared to a conventional MOGA and the previously proposed K-MOGA
- The CK-MOGA provides more stable performance on the reduction of the number of simulation calls, compared to the K-MOGA

7.3 FUTURE RESEARCH DIRECTIONS

In this section, several possible directions for the future research are discussed. Based on the approaches described in the previous chapters, these directions can be applied to overcome the shortcomings of proposed approaches or extend the applicability of these approaches.

7.3.1 Representing Uncertainty with Additional Statistical Information

In this dissertation, only interval uncertainty has been considered. The interval uncertainty considered assumes that a range (or interval) of uncertainty exists rather than a probability distribution. Frequently, probability distributions are difficult to obtain for sparse data making these approaches applicable in early design stages. It is common for

different uncertainty factors to be considered from quite different sources and, on that basis, to take very different forms. Some of the types of uncertainty sources that occur in modeling and simulation of physical systems include:

- Strong statistical information: Sometimes, large quantities of experimental data are available, sufficient to use a particular statistical model.
- Sparse statistical information: More commonly, only a limited amount of experimental data is available and collection of further data might be very expensive or impossible. Further, the available experimental data may provide only indirect or inferential information on the parameters actually used in a particular analysis. In these cases, attempts to fit particular statistical models will leave a substantial residue of epistemic uncertainty. A significant research gap here exists that needs to be investigated.
- Intervals: Upper and lower bounds or levels of belief on parametric values can be provided, typically from expert opinion.

Real-world problems typically present a mixture of the above mentioned uncertainty sources. Fully using all of the available data sets, in addition to the intervals, should be useful in design under uncertainty. Accordingly, integrating probabilistic and Bayesian theory, in addition to the interval analysis, is also of interest to improve and combine different types of information. However, how to efficiently combine different representations for different types of uncertainties and utilize them during the robust design optimization problems remains to be studied. Some initial insights and study along this thread have been reported in the literature [Gunawan and Papalambros, 2006], [Du et al., 2007].

It should be noted, however, that even when the probability distribution functions might be available for some uncertain parameters, typical stochastic robust optimization approaches may not be applicable for the situation considered in this dissertation. This is because there are mixed continuous-discrete variables in the optimization and the simulation under consideration is a black box, which is a more general situation in engineering design. The application of mixed integer linear or nonlinear programming methods, such as branch-and-bound type methods [Floudas, 1995] might be of interest if the explicit formulation of objective and constraint functions is available. However, in these cases when convexity assumptions are relaxed, typical linear or nonlinear programming methods for robust optimization might fail because of the limitation of these methods for obtaining global solutions. For the future search, it will be of great interest to compare the robust optimization approach proposed in this dissertation to other stochastic approaches, in terms of computational performance and capability of convergence, given the probability distribution functions of uncertain parameters and the assumptions for objective/constraint functions (e.g., convexity, continuity, linearity).

7.3.2 Uncertainty Reduction and Sensitivity Analysis for Decentralized MDO Problems with Multiple Objectives

As discussed in Chapter 4, uncertainty propagation in MDO problems implies that the outputs (or performance) from one discipline will be affected not only by the uncertainty from that discipline's inputs but also by the uncertainty from other disciplines due to interdisciplinary couplings. This issue of uncertainty propagation is important to any design with multiple subsystems.

In Chapter 5, we developed a Sensitivity Analysis (SA) approach to determine the importance of uncertainties by performing sensitivity analysis of the performance of all disciplines in a centralized all-at-once formulation. That SA method identifies the input component(s) whose input uncertainty reduction will produce the largest payoffs in the overall system performance.

However, the proposed all-at-once SA formulation had two main disadvantages. First, that formulation was developed to encompass the entire system and, as a result, did not have the subsystem autonomy that is necessary to maintain in the development of most complex systems. Moreover, for such a formulation, the effect of uncertainty propagated from one subsystem over the other's performance could not be determined. In short, only system-level information was available and used in the all-at-once SA formulation.

A deeper analysis of the effects of subsystem's parameters is warranted to determine if any opportunities exist for further reduction in uncertainty given a variety of possible investment levels in the overall system or subsystem uncertainties. The all-at-once SA approach for multi-output problems can be extended to provide designers and program managers in multi-disciplinary design an environment that allocates investments in uncertainty reductions of subsystem(s) that provide the greatest impact for future weapon development decisions.

With the extended SA for decentralized MDO, we will be able to first find in each subsystem: which uncertain parameter(s), if its uncertainty interval is reduced through appropriate design changes, will produce maximum reduction in the subsystem's objective variation and system's objective variation. Grouping uncontrolled (or uncertain) parameters by subsystem and treating them as candidates for possible improvement is

clearly reasonable given that different disciplines are controlled by different design groups. Obviously, input parameters from one subsystem also can affect the performance of other subsystems, which is the meaning of uncertainty propagation. In this regard, SA must be performed across disciplines. That is, the SA approach can be applied to each connected subsystem to determine the extent of uncertainty propogation to each. This can even include the system level problem which in effect identifies the relative importance of each subsystem to the system's performance.

7.3.3 Improved Approximation Approach

As mentioned in Chapter 6, a common strategy to reduce the computational effort of optimization methods such as MOGAs when integrated with expensive simulation models is to use metamodeling. In Chapter 6, we developed and implemented a CK-MOGA approach, which used adaptive (online) metamodels in the genetic algorithm's fitness estimation. However, not all information from the response side was been used in CK-MOGA to improve the accuracy of the metamodels.

For future work, we anticipate that the simulation of some subsystems will be very expensive (it takes several hours or even days for one simulation run) or the entire system can be very complex, and may have numerous subsystems. As a result, only a very limited number of simulations may be available. The Design of Experiments (DOE) and metamodeling methods could be suitable for the decomposed (or decentralized) multi-disciplinary design environment. Thus an efficient decomposition based approximation approach with a decomposed formulation becomes critical.

One possible approach is to first devise an off-line metamodeling method with efficient DOE methods such as Latin Hype-cube [Helton and Davis, 2003] or Maximum

Entropy Design [Shewry and Wynn, 1987]. However, since each subsystem can have multiple responses (objectives and/or constraints), treating each of these responses as a separate "model" and separately constructing one metamodel for each response is inefficient. This so called "independent metamodeling" approach does not exploit information such as correlations among responses. The research here should focus on a decomposed approximation approach, including DOE and metamodeling, and should account for efficient couplings among subsystems' approximations. Based on these off-line metamodels, an adaptive method could be devised to improve the accuracy of metamodels when necessary. Finally, a coordinated approach for integrating the approximations from all subsystems in a decentralized formulation needs to be addressed.

Based on these investigations, it can be expected that using the proposed metamodel assisted MOGA as the optimizer with improved objective criteria can improve the efficiency of robust optimization and MDO problems where the accuracy of involved metamodels becomes more important.

REFERENCES

1. Albada, S. J., and P. A. Robinson, 2007, "Transformation of arbitrary distributions to the normal distribution with application to EEG test–retest reliability," *Journal of Neuroscience Methods*, 161(2), pp. 205-211.

2. Alexandrov, N. M., and R. M. Lewis, 2002, "Analytical and Computational Aspects of Collaborative Optimization for Multidisciplinary Design," *AIAA journal*, 40(2), pp. 301-309.

3. Arora, J. S., 2004, *Introduction to Optimum Design*, 2nd eds., Elsevier, New York, USA.

4. Aughenbaugh, J. M., and C. J. Paredis, 2006, "The Value of Using Imprecise Probabilities in Engineering Design," *Journal of Mechanical Design*, 128(4), pp. 969-979.

5. Aute, V., and S. Azarm, 2006, "A Genetic Algorithms Based Approach for Multidisciplinary Multiobjective Collaborative Optimization," AIAA-2006-6953, *Proceedings of the 11th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization Conference*, Portsmouth, Virginia, Sep. 6-8.

6. Avila, S. L., A. C. Lisboa, L. Krahenbuhl, W. P. Carpes, J. A. Vasconcelos, R. R. Saldanha, and R. H. C. Takahashi, 2006, "Sensitivity Analysis Applied to Decision Making in Multiobjective Evolutionary Optimization," *IEEE Transactions on Magnetics*, 42(4), pp. 1103-1106.

7. Balbas, A., E. Galperin, and P. J. Guerra, 2005, "Sensitivity of Pareto Solutions in Multiobjective Optimization," *Journal of Optimization Theory and Applications*, 126(2), pp. 247-264.

8. Balbas, A., M. Ballvé, and P. J. Guerra, 1999, "Sensitivity in Multi-Objective Programming under Homogeneity Assumptions," *Journal of Multi-Criteria Decision Analysis*, 8(3), pp. 133-138.

9. Balling, R. J., J. C. Free, and A. R. Parkinson, 1986, "Consideration of Worst-Case Manufacturing Tolerances in Design Optimization," *Journal of Mechanical Design*, 108, pp. 438-441.

10. Barron, H., and C. P. Schmidt, 1988, "Sensitivity Analysis of Additive Multiattribute Value Models," *Operations Research*, 36(1), pp. 122-127.

11. Bauer, K. W., G. S. Parnell, and D. A. Meyers, 1999, "Response Surface Methodology as A Sensitivity Analysis Tool in Decision Analysis," *Journal of Multi-Criteria Decision Analysis*, 8(3), pp. 162 – 180.

12. Bazaraa, M. S., H. D. Sherali, and C. M. Shetty, 1993, *Nonlinear Programming Theory and Algorithms,* John Wiley & Sons, Inc., New York, USA.

13. Belegundu, A. D., and T. R. Chandrupatla, 1999, *Optimization Concepts and Applications in Engineering*, Prentice Hall, New Jersey, USA.

14. Benders, J. F., 1962, "Partitioning Procedures for Solving Mixed-Variables Programming Problems," *Numerische Mathematik*, 4, pp. 238-252.

15. Birge, J.R., and F. Louveaux, 1997, *Introduction to Stochastic Programming*, Springer Series in Operations Research, New York, USA.

16. Braun, R. D., 1996, "Collaborative Optimization: An Architecture for Large Scale Distributed Design," Ph.D. Dissertation, Stanford Univ., Stanford, CA, May.

17. Burdic, W. S., 2003, *Underwater Acoustic System Analysis*, Peninsula Pub., Los Altos, CA, USA.

18. Chen, J.-H., D. E. Goldberg, S.-Y. Ho, and K. Sastry, 2002, "Fitness Inheritance in Multi-objective Optimization," *Proceedings of the Genetic and Evolutionary Computation Conference*, NY, Morgan Kaufmann.

19. Chen, W. and K. Lewis, 1999, "A Robust Design Approach for Achieving Flexibility in Multidisciplinary Design", *AIAA Journal*, 7(8), pp. 982-989.

20. Chen, W., M. M. Wiecek, and J. Zhang, 1999, "Quality Utility –A Compromise Programming Approach to Robust Design." *Journal of Mechanical Design*, 121, pp. 179–187.

21. Chen, W., R. Jin, and A. Sudjianto, 2005, "Analytical Variance-Based Global Sensitivity Analysis in Simulation-Based Design under Uncertainty," *Journal of Mechanical Design*, 127(5), pp. 875-886.

22. Choi, K. K., J. Tu, and Y. H. Park, 2001, "Extensions of Design Potential Concept for Reliability-Based Design Optimization to Nonsmooth and Extreme Cases," *Structural and Multidisciplinary Optimization*, 22, pp. 335-350.

23. Chung, H., and J. Alonso, 2004, "Multiobjective Optimization Using Approximation Model-Based Genetic Algorithms," AIAA-2004-432, *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, Aug. 30- Sep.1.

24. Coello Coello, C. A., D. A. Van Veldhuizen, and G. B. Lamont, 2002, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, Boston, USA.

25. Conejo, A. J., E. Castillo, R. Mínguez, and R. García-Bertrand, 2006, *Decomposition Techniques in Mathematical Programming*. *Engineering and Science Applications*, Springer, Berlin.

26. Cover, T. M., and J. A. Thomas, 1991, *Elements of Information Theory*, John Wiley & Sons, New York, NY, USA.

27. Dantzig, G. B., and P. Wolfe, 1961, "The Decomposition Algorithm for Linear Programs," *Econometrica*, 29(4), pp. 767-778.

28. Deb, K., 2001, *Multiobjective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, New York, NY, USA.

29. Du L., K. K. Choi, and I. Lee, 2007, "Robust Design Concept in Possibility Theory and Optimization for System with both Random and Fuzzy Input Variables," *Proceedings of IDETC/CIE 2007, ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Las Vegas, Nevada, September 4-7.

30. Du, X., and W. Chen, 2000(a), "A Methodology for Uncertainty Propagation and Management in Simulation-Based Systems Design", *AIAA Journal*, 38(8), pp. 1471-1478.

31. Du, X., and W. Chen, 2000(b), "Towards a Better Understanding of Modeling Feasibility Robustness in Engineering Design," *Journal of Mechanical Design*, 122, pp. 385-394.

32. Du, X., and W. Chen, 2002, "Efficient Uncertainty Analysis Methods for Multidisciplinary Robust Design," *AIAA Journal*, 40(3), pp. 545-552.

33. Du, X., and W. Chen, 2004, "Sequential Optimization and Reliability Assessment Method for Efficient Probabilistic Design," *Journal of Mechanical Design*, 126(2), pp. 225-233.

34. Du, X., and W. Chen, 2005, "Collaborative Reliability Analysis under the Framework of Multidisciplinary Systems Design," *Journal of Optimization & Engineering*, 6(1), pp.63-84.

35. Fang, H., M. Rais-Rohani, and M. Horstemeyer, 2004, "Multiobjective Crashworthiness Optimization with Radial Basis Functions," AIAA-2004-4487, *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, Aug. 30- Sep. 1.

36. Farhang-Mehr, A., and S. Azarm, 2003, "An Information-Theoretic Performance Metric for Quality Assessment of Multi-Objective Optimization Solution Sets", *Journal of Mechanical Design*, 125(4), pp. 655-663.

37. Farina, M., 2001, "A Minimal Cost Hybrid Strategy for Pareto Optimal Front Approximation," *Evolutionary Optimization*, 3(1), pp. 41-52.

38. Farina, M., 2002, "A Neural Network Based Generalized Response Surface Multiobjective Evolutionary Algorithms," *Proceedings of Congress on Evolutionary Computation, IEEE Press*, pp. 956-961.

39. Ferson, S., and L. R. Ginzburg, 1996, "Different Methods Are Needed to Propagate Ignorance and Variability," *Reliability Engineering & System Safety*, 54(2-3), pp. 133-144.

40. Ferson, S., R. B. Nelsen, J. Hajagos, D. J. Berleant, J. Zhang, W. T. Tucker, L. R. Ginzburg and W. L. Oberkampf, 2004, "Dependence in Probabilistic Modeling,

Dempster-Shafer Theory, and Probability Bounds Analysis," Sandia National Laboratories, SAND2004-3072, Albuquerque, NM.

41. Floudas, C. A., 1995, *Nonlinear and Mixed-Integer Optimization*, Oxford Univ. Press.

42. Frey, H. C., and S. R. Patil, 2002, "Identification and Review of Sensitivity Analysis Methods," *Risk Analysis*, 22(3), pp. 553-578.

43. Frits, A. P., 2004, "Formulation of an Integrated Robust Design and Tactics Optimization Process for Undersea Weapon Systems," Ph. D. dissertation, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA.

44. Gano, S. E., J. E. Renaud, J. D. Martin, and T. W. Simpson, 2006, "Update Strategies for Kriging Models for Using in Variable Fidelity Optimization," *Structural and Multidisciplinary Optimization*, 32(4), pp. 287-298.

45. Goldberg, D. E., 1989, *Genetic Algorithms in Search*, *Optimization*, *and Machine Learning*, Addison-Wesley, Reading, Mass., USA.

46. Greenland, S., 2001, "Sensitivity Analysis, Monte Carlo Risk Analysis, and Bayesian Uncertainty Assessment," *Risk Analysis*, 21(4), pp. 579-584.

47. Gu, X., and J. E. Renaud, 2002, "Implementation Study of Implicit Uncertainty Propagation (IUP) in Decomposition-Based Optimization," AIAA-2002-5416, *Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization Conference*, Atlanta, Georgia, Sep. 4-6.

48. Gu, X., J. E. Renaud, and C. L. Penninger, 2006, "Implicit Uncertainty Propagation for Robust Collaborative Optimization," *Journal of Mechanical Design*, 128(4), pp. 1001-1013.

49. Gu, X., J. E. Renaud, and S. M. Batill, 1998, "Investigation of Multidisciplinary Design Subject to Uncertainties," AIAA Paper 98-4747, Sept.

50. Gu, X., J. E. Renaud, S. M. Batill, R. M. Brach, and A. S. Budhiraja, 2000, "Worst Case Propagated Uncertainty of Multidisciplinary Systems in Robust Design Optimization," *Structure Multidisciplinary Optimization*, 20, pp. 190-213.

51. Gunawan, S., A. Farhang-Mehr, and S. Azarm, 2004, "On Maximizing Solution Diversity in Multiobjective Multidisciplinary Genetic Algorithm for Design Optimization," *Mechanics Based Design of Structures and Machines,* 32(2), pp. 491-514.

52. Gunawan, S., and P. Papalambros, 2006, "A Bayesian Approach to Reliability-Based Optimization with Incomplete Information," *Journal of Mechanical Design*, 128(4), pp. 909-918.

53. Gunawan, S., and P. Papalambros, 2007, "Reliability Optimization with Mixed Continuous-Discrete Random Variables and Parameters," *Journal of Mechanical Design*, 129(2), pp. 158-165.

54. Gunawan, S., and S. Azarm, 2004, "Non-Gradient Based Parameter Sensitivity Estimation for Single Objective Robust Design Optimization," *Journal of Mechanical Design*, 126(3), pp. 395-402.

55. Gunawan, S., and S. Azarm, 2005(a), "Multi-Objective Robust Optimization Using a Sensitivity Region Concept," *Structural and Multidisciplinary Optimization*, 29(1), pp. 50-60.

56. Gunawan, S., and S. Azarm, 2005(b), "A Feasibility Robust Optimization Method Using a Sensitivity Region Concept," *Journal of Mechanical Design*, 127(5), pp. 858-868.

57. Haimes, Y. Y., K. Tarvainen, T. Shima, and J. Thadathil, 1990, *Hierarchical Multiobjective Analysis of Large-scale Systems*, Hemisphere Pub., New York, USA.

58. Hamby, D. M., 1994, "Review of Techniques for Parameter Sensitivity Analysis of Environmental Models," *Environmental Monitoring and Assessment*, 32(2), pp. 135-154.

59. Helton, J. C., 1993, "Uncertainty and Sensitivity Analysis Techniques for Use in Performance Assessment for Radioactive Waste Disposal," *Reliability Engineering & Systems Safety*, 42(2-3), pp. 327-367.

60. Helton, J. C., and F. J. Davis, 2003, "Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems," *Reliability Engineering & Systems Safety*, 81(1), pp. 23-69.

61. Holland, J., 1975, *Adaptation in Natural and Artificial Systems*, the University of Michigan Press, Michigan, USA.

62. Homma, T., and A. Saltelli, 1996, "Importance Measures in Global Sensitivity Analysis of Nonlinear Models," *Reliability Engineering & System Safety*, 52(1), pp. 1-17.

63. Hurricks, P. L., 1994, *Electromechanical Product Design*, John Wiley & Sons Inc, New York, USA.

64. Iman, R. L., and J. C. Helton, 1988, "An Investigation of Uncertainty and Sensitivity Analysis Techniques for Computer Models," *Risk Analysis*, 8(1), pp. 71-90.

65. Jin, Y., 2005, "A Comprehensive Survey of Fitness Approximation in Evolutionary Computation," *Soft Computing*, 9(1), pp. 3-12.

66. Jin, Y., M. Olhofer, and B. Sendhoff, 2001, "Managing Approximate Models in Evolutionary Aerodynamic Design Optimization," *Proceedings of IEEE Congress on Evolutionary Computation*, 1, pp. 592-599.

67. Jin, Y., M. Olhofer, and B. Sendhoff, 2002, "A Framework for Evolutionary Optimization with Approximate Fitness Functions," *IEEE Transactions on Evolutionary Computation*, 6(5), pp. 48-494.

68. Jung, D. H., and B. C. Lee, 2002, "Development of a Simple and Efficient Method for Robust Optimization," *Int. Journal for Numerical Methods in Engineering*, 23, 2201-2215.

69. Kalsi, M., K. Hacker, and K. Lewis, 2001, "A Comprehensive Robust Design Approach for Decision Trade-Offs in Complex Systems Design," *Journal of Mechanical Design*, 123(1), pp. 1-10.

70. Kasperskia, A., and P. Zieli´nskib, 2006, "The Robust Shortest Path Problem in Series–Parallel Multidigraphs with Interval Data," *Operations Research Letters*, 34, pp. 69 -76.

71. Kern, D., X. Du., and A. Sudjianto, 2003, "Forecasting Manufacturing Quality During Design Using Process Capability Data," in *Proceedings of the IMECE' 03, ASME 2003 International Mechanical Engineering Congress and RD&D Expo*, Washington, DC, Nov. 15-21,

72. Kim, H. M., 2001, "Target Cascading in Optimal System Design," Ph.D. Dissertation, University of Michigan, Ann Arbor, MI.

73. Koch, P. N., B. A. Wujek, O. Golovidov, and T.W. Simpson, 2002, "Facilitating Probabilistic Multidisciplinary Design Optimization Using Kriging Approximation Models," AIAA 2002-5415, *Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, Sep. 4-6.

74. Koch, P. N., T. W. Simpson, J. K. Allen, and F. Mistree, 1999, "Statistical Approximations for Multidisciplinary Design Optimization: The Problem of Size," *Journal of Aircraft*, 36(1), pp. 275-286.

75. Kodiyalam, S., and J. Sobieszczanski-Sobieski, 2000, "Bilevel Integrated System Synthesis with Response Surfaces," *AIAA Journal*, 38(8), pp. 1479-1485

76. Kodiyalam, S. and J. Sobieszczanski-Sobieski, 2001, "Multidisciplinary Design Optimization – Some Formal Methods, Framework Requirements, and Application to Vehicle Design," *Int. J. Vehicle Design* (Special Issue), pp. 3-22.

77. Kouvelis, P., and G. Yu, 1997, *Robust Discrete Optimization and Its Applications*, Kluwer Academic Publishers, Boston, USA.

78. Kurapati, A., S. Azarm, and J. Wu, 2002, "Constraint Handling Improvements for Multi-Objective Genetic Algorithms," *Structural and Multidisciplinary Optimization*, 23(3), pp. 204-213.

79. Lee, K., and G. Park, 2001, "Robust Optimization Considering Tolerances of Design Variables," *Computers and Structures,* 79, pp. 77-86.

80. Li, M., and S. Azarm, 2007, "Multiobjective collaborative Robust Optimization (McRO) with Interval Uncertainty and Interdisciplinary Uncertainty Propagation," *Journal of Mechanical Design* (under review) (Also in *Proceedings of IDETC/CIE 2007, ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DETC2007-34818, Las Vegas, Nevada, September 4-7, 2007).

81. Li, G., M. Li, S. Azarm, J. Rambo, and Y. Joshi, 2007a, "Optimizing Thermal Design of Data Center Cabinets with a New Multi-Objective Genetic Algorithm," *Distributed and Parallel Databases*, 21(2-3), pp. 167-192.

82. Li, M., G. Li and S. Azarm, 2007b, "A Kriging Metamodel Assisted Multi-objective Genetic Algorithm for Design Optimization," *Journal of Mechanical Design* (forthcoming) (Also in *Proceedings of IDETC/CIE 2006, ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Philadelphia, Pennsylvania, September 10-13, 2006).

83. Li, M., N. Williams and S. Azarm, 2007c, "Uncertainty Reduction and Sensitivity Analysis for Multi-Objective Optimization Design," *Reliability Engineering & System Safety* (under review).

84. Li, M., S. Azarm, and A. Boyars, 2006, "A New Deterministic Approach using Sensitivity Region Measures for Multi-Objective and Feasibility Robust Design Optimization," *Journal of Mechanical Design*, 128(4), pp. 874-883.

85. Li, M., S. Azarm, and V. Aute, 2005, "A Multi-Objective Genetic Algorithm for Robust Design Optimization," *Genetic and Evolutionary Computation Conference (GECCO) '05*, Washington, DC, June 25-29.

86. Lian, Y., and M. Liou, 2004, "Multiobjective Optimization Using Coupled Response Surface Model and Evolutionary Algorithm," AIAA-2004-4323, *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, Aug. 30- Sep. 1.

87. Liu, H., W. Chen, and A. Sudjianto, 2006, "Relative Entropy Based Method for Global and Regional Sensitivity Analysis in Probabilistic Design," *Journal of Mechanical Design*, 128(2), pp. 1-11.

88. Liu, H., W. Chen, M. Kokkolaras, P. Papalambros and H. Kim, 2006, "Probabilistic Analytical Target Cascading - A Moment Matching Formulation for Multilevel Optimization under Uncertainty," *Journal of Mechanical Design*, 128(4), pp.991-1000.

89. Luo, Z. Q., J. S. Pang, and D. Ralph, 1996, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge, United Kingdom.

90. Martin, J. D. 2007, "Using Maximum Likelihood Estimation to Estimate Kriging Model Parameters," *Proceedings of IDETC/CIE 2007, ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DETC2007-34662, Las Vegas, Nevada, September 4-7.

91. Mavris, D. V., O. Bandte, and D. A. DeLaurentis, 1999, "Robust Design Simulation: A Probabilistic Approach to Multidisciplinary Design," *Journal of Aircraft*, 36(1), pp. 298-397.

92. McAllister, C. D., and T. W. Simpson, 2003, "Multidisciplinary Robust Design Optimization of An Internal Combustion Engine," *Journal of Mechanical Design*, 125(1), pp. 124-130.

93. McAllister, C. D., T. W. Simpson, and M. Yukish, 2000, "Goal Programming Applications in Multidisciplinary Design Optimization,'' AIAA-00-4717, *Proceedings of the 8th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, September 6–8.

94. Messac, A., and A. I. Yahaya, 2002, "Multiobjective Robust Design Using Physical Programming," *Structural and Multidisciplinary Optimization*, 23(5), pp. 357-371.

95. Miettinen K. M., 1999, *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers, Boston, USA.

96. Mourelatos, Z. P., and J. Zhou, 2006, "A Design Optimization Method Using Evidence Theory," *Journal of Mechanical Design*, 128(4), pp.901-908.

97. Nain, P. K. S., and K. Deb, 2003, "Computationally Effective Search and Optimization Procedure Using Coarse to Fine Approximations," *Proceedings of the Congress on Evolutionary Computation (CEC-2003)*, Canberra, Australia, pp. 2081-2088.

98. Nair, P. B., and A. J. Keane, 1998, "Combining Approximation Concepts with Genetic Algorithm-based Structural Optimization Procedures," AIAA-1998-1912, *AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, 39th, and AIAA/ASME/AHS Adaptive Structures Forum*, Long Beach, CA, Apr. 20-23, Collection of Technical Dissertations. Pt. 2 (A98-25092 06-39).

99. Narayanan, S. and S. Azarm, 1999, "On Improving Multiobjective Genetic Algorithms for Design Optimization," *Structural and Multidisciplinary Optimization*, 18, pp. 146-155.

100. Oakley, J. E., and A. O'Hagan, 2004, "Probabilistic Sensitivity Analysis of Complex Models: A Bayesian Approach," *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, 66(3), pp.751-769.

101. Papadrakakis, M., N. Lagaros, and Y. Tsompanakis, 1999, "Optimization of Large-Scale 3D Trusses Using Evolution Strategies and Neural Networks," *International Journal of Space Structures*, 14(3), pp. 211-223.

102. Papalambros, P. Y., and D. J. Wilde, 2000, *Principles of Optimal Design*: *Modeling and Computation*, 2nd eds., Cambridge University Press, New York, USA.

103. Parkinson, A., C. Sorensen, and N. Pourhassan, 1993, "A General Approach to Robust Optimal Design," *Journal of Mechanical Design*, 115, pp. 74-80.

104. Rasheed, K., X. Ni, and S. Vattam, 2005, "Comparison of Methods for Developing Dynamic Reduced Models for Design Optimization," *Soft Computing*, 9, pp. 29-37.

105. Ray, T., 2002, "Constrained Robust Optimal Design using a Multiobjective Evolutionary Algorithm," in *Congress on Evolutionary Computation* (*CEC'2002*), 1, pp. 419-424, IEEE Service Center, Piscataway, New Jersey, May.

106. Renaud, J., and G. Gabriele, 1993, "Improved Coordination in Non-Hierarchic System Optimization," *AIAA Journal*, 31(12), pp. 2367-2373.

107. Ruszczynski, A., and A. Shapiro, 2003, *Stochastic Programming*, Handbooks in Operation Research and Management Science, Volume 10, Elsevier, Boston, USA.

108. Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn, 1989, "Design and Analysis

of Computer Experiments," *Statistical Science*, 4(4), pp. 409-435.

109. Saltelli, A., K. Chan, and E. M. Scott, 2000, *Sensitivity analysis*, John Wiley & Sons, New York, USA.

110. Saltelli, A., S. Tarantola, and K. Chan, 1999(a), "A Quantitative Model-Independent Method for Global Sensitivity Analysis of Model Output," *Technometrics*, 41(1), pp. 39-56.

111. Saltelli, A., S. Tarantola, and K. Chan, 1999(b), "A Role for Sensitivity Analysis in Presenting the Results from MCDA Studies to DMs," *Journal of Multi-Criteria Decision Analysis*, 8(3), pp. 139-145.

112. Sasena, M. J., M. Parkinson, M. P. Reed, P. Y. Papalambros, and P. Goovaerts, 2005, "Improving an Ergonomics Testing Procedure via Approximation-based Adaptive Experimental Design," *Journal of Mechanical Design*, 127(5), pp. 1006-1013.

113. Seller, R., S. Batill, and J. Renaud, 1996, "Response Surface Based, Concurrent Subspace Optimization for Multidisciplinary System Design," AIAA Dissertation 96-0714, January.

114. Shan, S., and G. G. Wang, 2005, "An Efficient Pareto Set Identification Approach for Multiobjective Optimization on Black-box Functions," *Journal of Mechanical Design*, 127(5), pp. 866-874.

115. Shannon, C. E., 1948, ''A Mathematical Theory of Communication,'' *Bell System Technical Journal*, 27, pp. 379-423 and 623-656 (July and October).

116. Shewry, M. C., and H. P. Wynn, 1987, "Maximum Entropy Sampling", *Journal of Applied Statistics*, 14, pp. 165-170.

117. Simpson, T. W., 1998, "A Concept Exploration Method for Product Family Design", Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA.

118. Simpson, T. W., A. J. Booker, D. Ghosh, A. A. Giunta, P. N. Koch, and R.-J. Yang, 2004, "Approximation Methods in Multidisciplinary Analysis and Optimization: A Panel Discussion," *Structural and Multidisciplinary Optimization*, 27, pp.302-313.

119. Simpson, T. W., J. Peplinski, P. N. Koch, and J. K. Allen, 2001, "Metamodels for Computer-based Engineering Design: Survey and Recommendations," *Engineering with Computers*, 17(2), pp. 129-150.

120. Smith, R., B. Dike, and S. Stegmann, 1995, "Fitness Inheritance in Genetic Algorithms," *Proceedings of the ACM Symposiums on Applied Computing*, ACM, pp. 345-350.

121. Sobieski, I. P., and I. M. Kroo, 2000, "Collaborative Optimization Using Response Surface Estimation," *AIAA Journal*, 38(10), pp. 1931-1938.

122. Sobieski-Sobieszczanksi, J., 1988, "Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems," NASA CP-3031, Septempter.

123. Sobieszczanski-Sobieski, J., and R. J. Balling, 1996, "Optimization of Coupled Systems: A Critical Overview of Approaches," *AIAA Journal*, 34(1), pp. 6-17.

124. Sobieszczanski-Sobieski, J., and R. T. Haftka, 1997, "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," *Structure Optimization*, 14(1), pp. 1-13.

125. Sobieszczanski-Sobieski, J., J. Agte, and R. Sandusky, Jr., 1998, "Bi-Level Integrated System Synthesis (BLISS)," AIAA-1998-4916, *Proceedings of the 7th*

AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization Conference, St. Louis, Missouri, Sep.

126. Sobol, I. M., 1993, ‘‘Sensitivity Analysis for Non-Linear Mathematical Models,’’ *Mathematical Modeling and Computational Experiment*, 1, pp. 407-414 [Translation of Sobol, I. M., 1990, ‘‘Sensitivity Estimates for Nonlinear Mathematical Models,” *Matematicheskoe Modelirovanie*, 2, pp. 112-118 (in Russian)].

127. Sobol, I. M., 2001, “Global Sensitivity Indices for Nonlinear Mathematical Models and Their Monte Carlo Estimates,” *Mathematics and Computers in Simulation*, 55(1-3), pp. 271-280.

128. Su, J., and J. E. Renaud, 1997, “Automatic Differentiation in Robust Optimization,” *AIAA Journal*, 35(6), pp. 1072-1079.

129. Sues, R. H., M. A. Cesare, S. S. Pageau, and J. Y.-T. Wu, 2001, “Reliability-Based Optimization Considering Manufacturing and Operational Uncertainties,” *Journal of Aerospace Engineering*, 14, pp. 166-174.

130. Sundaresan, S., K. Ishii., and D. Houser, 1992, “Design Optimization for Robustness Using Performance Simulation Programs,” *Engineering Optimization*, 20, pp. 163-178.

131. Taguchi, G., 1978, “Performance Analysis Design,” *International Journal of Production Research*, 16, pp. 521-530.

132. Tapetta, R. V., and J. E. Renaud, 1997, “Multiobjective Collaborative Optimization,” *Journal of Mechanical Design*, 119(3), pp. 403-411.

133. Tu, J., K. K. Choi, and Y. H. Park, 1999, "A New Study on Reliability-Based Design Optimization," *Journal of Mechanical Design*, 121, pp. 557-564.

134. Vincent, L., 1993, "Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms," *Trans. IEEE, Image Processing*, 2(2), pp. 176-201.

135. Wang, G. G., and S. Shan, 2007, "Review of Metamodeling Techniques in Support of Engineering Design Optimization," *Journal of Mechanical Design*, 129(4), pp. 370-380.

136. Williams, N., S. Azarm, and P. K. Kannan, 2006, "Engineering Product Design Optimization for Retail Channel Acceptance," DETC2006-99118, *Proceedings of IDETC/CIE 2006, ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Philadelphia, Pennsylvania, September 10-13.

137. Wilson, B., D. Cappelleri, M. Frecker, and T. W. Simpson, 2001, "Efficient Pareto Frontier Exploration Using Surrogate Approximations," *Optimization and Engineering*, 2, pp. 31-50.

138. Wu, J., and S. Azarm, 2001, "Metrics for Quality Assessment of a Multiobjective Design Optimization Solution Set," *Journal of Mechanical Design*, 123(1), pp. 18-25.

139. Wu, W. D., and S. S. Rao, 2007, "Uncertainty Analysis and Allocation of Joint Tolerances in Robot Manipulators Based on Interval Analysis," *Reliability Engineering & System Safety*, 92(1), pp. 54-64.

140. Wu, Y-T., 1987, "Demonstration of a new, fast probability integration method for reliability analysis," *Journal of Engineering for Industry*, Serial B, 109(1), pp. 8-24.

141. Youn, B. D., K. K. Choi, and Y. H. Park, 2003, "Hybrid Analysis Method for Reliability-Based Design Optimization," *Journal of Mechanical Design*, 125(2), pp. 221-232.

142. Yu, J. C., and K. Ishii, 1998, "Design for Robustness Based on Manufacturing Variation Patterns," *Journal of Mechanical Design*, 120, pp. 196-202.

143. Zadeh, L. A., 1965, "Fuzzy Sets," *Information and Control*, 8, pp. 338-353.

144. Zhang W. H., 2003, "On the Pareto Optimum Sensitivity Analysis in Multicriteria Optimization," *International Journal for Numerical Methods in Engineering*, 58(6), pp. 955-977.

145. Zhu, J., and K. L. Ting, 2001, "Performance Distribution Analysis and Robust Design," *Journal of Mechanical Design*, 123(1), pp. 11-17.

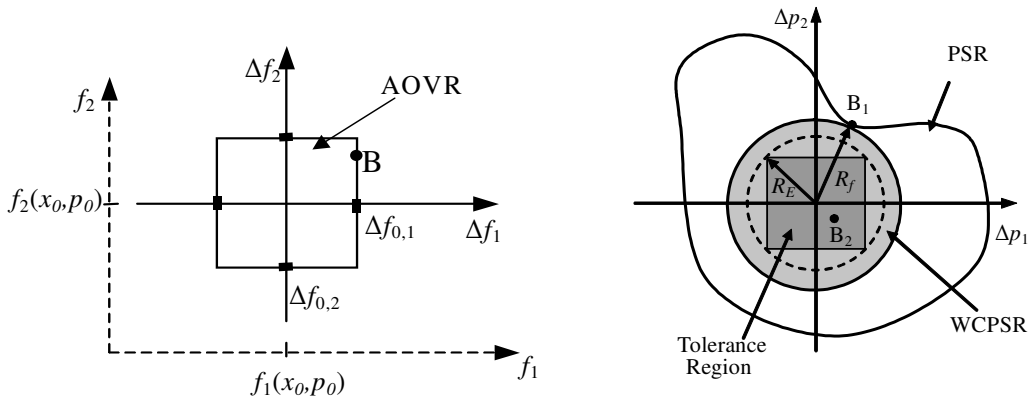
APPENDIX: FORWARD MAPPING VS. BACKWARD MAPPING

In this section, we compare our approach (forward mapping) with Gunawan and Azarm's approach (backward mapping) [Gunawan and Azarm, 2004, 2005(a), 2005(b)]. We examine only objective robust optimization; the comparisons for feasibility optimization and for performance optimization are similar.

We first describe Gunawan and Azarm's approach. We then explain why their approach is not applicable in general with discontinuous functions (i.e., discontinuity with respect to uncertain parameters), and why the proposed approach in this dissertation does not have this limitation. Note that because both methods are deterministic, neither method requires any information about probability distributions of the parameter variations. Also, neither method assumes linearity of the objective or constraint functions, and, hence, both are applicable even when the variations of parameter are large. However, since both methods use a similar outer-inner structure, the computational cost (i.e., number of simulation calls) is comparable.

A. 1 DESCRIPTION OF GUNAWAN'S APPROACH

Gunawan's approach is based on the concept that design $\boldsymbol{x}_0$'s AOVR in $\Delta\boldsymbol{f}$-space maps backward into a Parameter Sensitivity Region (PSR) in normalized $\Delta\boldsymbol{p}$-space; see Figure A.1. This PSR represents the amount of parameter variation that $\boldsymbol{x}_0$ can absorb without $\boldsymbol{x}_0$'s objective function values being outside the AOVR. Design $\boldsymbol{x}_0$ is robust if its PSR totally contains $\boldsymbol{x}_0$'s tolerance region.



(a) **(b)**

Figure A.1 (a) AOVR, and (b) corresponding PSR in Gunawan's approach

Gunawan's approach uses a Worst-Case Parameter Sensitivity Region (WCPSR) estimate of $\boldsymbol{x}_0$'s PSR. Graphically, the WCPSR is the hyper-sphere inside the PSR that touches the PSR's boundary at the closest point to the origin, as shown in Figure A.1 for a two-parameter case. R_f is the radius of the WCPSR, and R_E is the radius of the exterior hyper-sphere of the normalized tolerance region. This method defines the objective robustness index $\eta_f = \frac{R_f}{R_E}$. Design $\boldsymbol{x}_0$ is robust if $\eta_f \geq 1$ (i.e., $R_f \geq R_E$; tolerance region contained in WCPSR). They use formulations Eqs. (A.1) and (A.2) to obtain the robust alternatives. R_f in Eq. (A.1) is calculated by Eq. (A.2).

$$\begin{aligned} \min_{x} \quad & f_m(\boldsymbol{x}, \boldsymbol{p}_0) && m = 1, \ldots, M \\ \text{s.t.} \quad & g_l(\boldsymbol{x}, \boldsymbol{p}_0) \leq 0 && l = 1, \ldots, L \\ & \eta_{0,f} - \eta_f \leq 0 \\ & \boldsymbol{x}^{lower} \leq \boldsymbol{x} \leq \boldsymbol{x}^{upper} \end{aligned} \tag{A.1}$$

$$
\begin{aligned}
&\min_{\tilde{\boldsymbol{p}}} \quad R_f(\tilde{\boldsymbol{p}}) = \left[\sum_{k=1}^{K} \left| p_{0,k} - \tilde{p}_k \right|^2\right]^{\frac{1}{2}} \\
&\text{s.t.} \quad \max_{m=1,\ldots,M} \left(\frac{|\Delta f_m|}{\Delta f_{0,m}}\right) - 1 = 0 \\
&\text{where} \quad \Delta f_m = f_m(\boldsymbol{x}_0, \tilde{\boldsymbol{p}}) - f_m(\boldsymbol{x}_0, \boldsymbol{p}_0) \\
&\boldsymbol{p}_0 - \Delta \boldsymbol{p} \le \tilde{\boldsymbol{p}} \le \boldsymbol{p}_0 + \Delta \boldsymbol{p}
\end{aligned}
\tag{A.2}
$$

A. 2 CONSEQUENCES OF WORST-CASE ESTIMATES

It should be noted that our approach and Gunawan's approach are conservative because directly calculating the sensitivity region is intractable. Both approaches estimate the sensitivity region by a worst case method, instead of calculating it directly. The benefit of the worst case method is that the robustness of obtained designs is guaranteed.

However, Gunawan's approaches may reject as non-robust some designs that are, in fact, robust. Figure A.2 illustrates the condition in Gunawan's method that leads to rejection of robust designs. The tolerance region is contained in the actual PSR, but is not contained in the WCPSR.

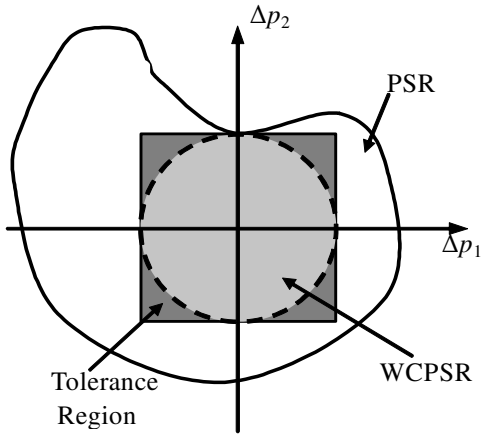


Figure A.2 Condition causing rejection of robust designs in Gunawan's approach

A. 3 APPLICABILITY OF THE TWO APPROACHES

For two reasons, Gunawan's approach is not applicable in general when the objective functions f_m are not continuous with respect to the parameter variations. First, his robustness criterion, the tolerance region contained within the PSR, depends on the requirement that the points on the boundary of the AOVR mapping to points on the boundary of the PSR. (Refer to [Gunawan and Azarm, 2004] for a detailed proof showing that the continuity assumption is required for Eq. (A.1) and Eq. (A.2) to give the correct WCPSR.) For instance, an AOVR boundary point could map to an interior point of the PSR. Figure A.1 illustrates the case of concern: point B could map to point B_1 and to point B_2. In such a case the correct radius of the WCPSR is from the origin to B_1, but the optimization solution in Eq. (A.2) will give the distance to B_2. Thus, the WCPSR will be erroneously small, with an effect that can be understood two ways: solutions that are, in fact, robust within the worst-case limit will be rejected; the robustness threshold is erroneously over-stringent. The result would be the solutions' objective values being worse than should have been obtained.

Second, if the functions are not continuous, then the PSR might be disconnected or have holes, as shown in Figure A.3. If the origin of $\Delta\boldsymbol{p}$-space is at O_1 in Figure A.3, then Eq. (A.2) will give the distance from O_1 to point A for the radius of the WCPSR, indicating that design $\boldsymbol{x}_0$ is robust (tolerance region contained within WCPSR). If the origin is at O_2, Eq. (A.2) will give the distance from O_2 to point B as the radius of the WCPSR, again indicating that $\boldsymbol{x}_0$ is robust. However, in both cases $\boldsymbol{x}_0$ is, in fact, not robust: the PSR lies entirely outside the tolerance region.

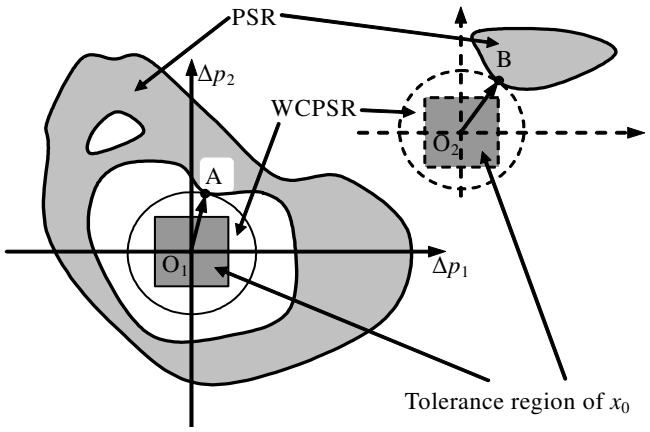


Figure A.3 Cases where Gunawan's method fails

Thus, Gunawan's approach has two shortcomings if used with discontinuous functions. Either the robust solutions will have degraded objective values, or the solutions might, in fact, not be robust. Non-robustness of a solution could be discovered by simulating instances of the design. Degradation of objective values is intractable.

In contrast, our approach does not require that the objective functions be continuous. Referring to Figure 3.1, tolerance region points, whether on the boundary or interior, can map to the boundary or the interior of the OSR, and can even be isolated points. However, for the WCOSR we find the largest $\|\cdot\|_\infty$ distance norm from the origin to the point of the OSR. Thus, the calculated WCOSR always includes all points of the OSR. If a design's WCOSR is contained within the AOVR, then its actual OSR will be contained in the AOVR. The solutions will not incorrectly include non-robust designs, and the solutions will always have the best possible objective values obtainable designs within the worst-case limit.